Documentation

BY AMNA MUSTAFA ALWANI
TO : SIR MANSOOR

Overview

Project Name	Virtual Personal Finance Manager
Project Manager	Amna Mustafa Alwani University Name: Salim Habib University Batch Fall, Computer Science, 1st Semester Course: PF Lab
Project Dates	Start Date: Jan 8, 2025 End Date: Jan 10, 2025
Background	The purpose of this project is to simplify personal financial management by providing an easy-to-use application. It also serves as a practical exercise in applying fundamental programming concepts.
Objectives	• To apply programming fundamentals such as loops, conditions, and functions. • To develop a user-friendly application for tracking and managing finances. • To enhance skills in le handling and console-based user interfaces.
Target Audience	This project is ideal for anyone looking for an introductory nancial management tool or a practical demonstration of basic programming concepts.

Project Overview

The **Virtual Personal Finance Manager** is a **console-based application** specifically designed to assist users in effectively managing their personal finances. With the increasing complexity of financial decisions in our daily lives, this tool aims to simplify the process of tracking income, expenses, and savings, thereby empowering users to take control of their financial health.

At its core, the application serves a dual purpose: it not only provides a practical solution to financial management but also integrates fundamental programming concepts. The design of the Virtual Personal Finance Manager emphasizes user-friendliness while incorporating essential programming elements such as data structures, algorithms, and input/output operations. By engaging with this application, users will not only enhance their financial literacy but also gain insight into the underlying coding principles that drive its functionality.

The primary functions of the Virtual Personal Finance Manager include recording various sources of income, categorizing expenses, and calculating savings to provide users with a comprehensive overview of their financial situation. This enables users to make informed decisions regarding their spending habits and savings strategies. The console interface ensures accessibility and ease of use, allowing users to navigate through the application without unnecessary complications.

Moreover, the development of this application reflects a growing need for effective financial tools in a fast-paced world. As individuals strive to achieve their financial goals, the Virtual Personal Finance Manager stands out as an essential resource that not only meets practical needs but also enhances programming skills, making it an invaluable project for both personal and educational growth.

Objective

- **Expense Tracking**: Users can record daily expenses and categorize them for insightful analysis.
- **Budget Creation**: Users can set monthly or yearly budgets for various categories and monitor their spending against these budgets.
- **Financial Reporting**: The application generates detailed reports on spending habits, income, and savings, enabling informed financial decisions.
- **Data Security**: User data is stored securely and managed with confidentiality.

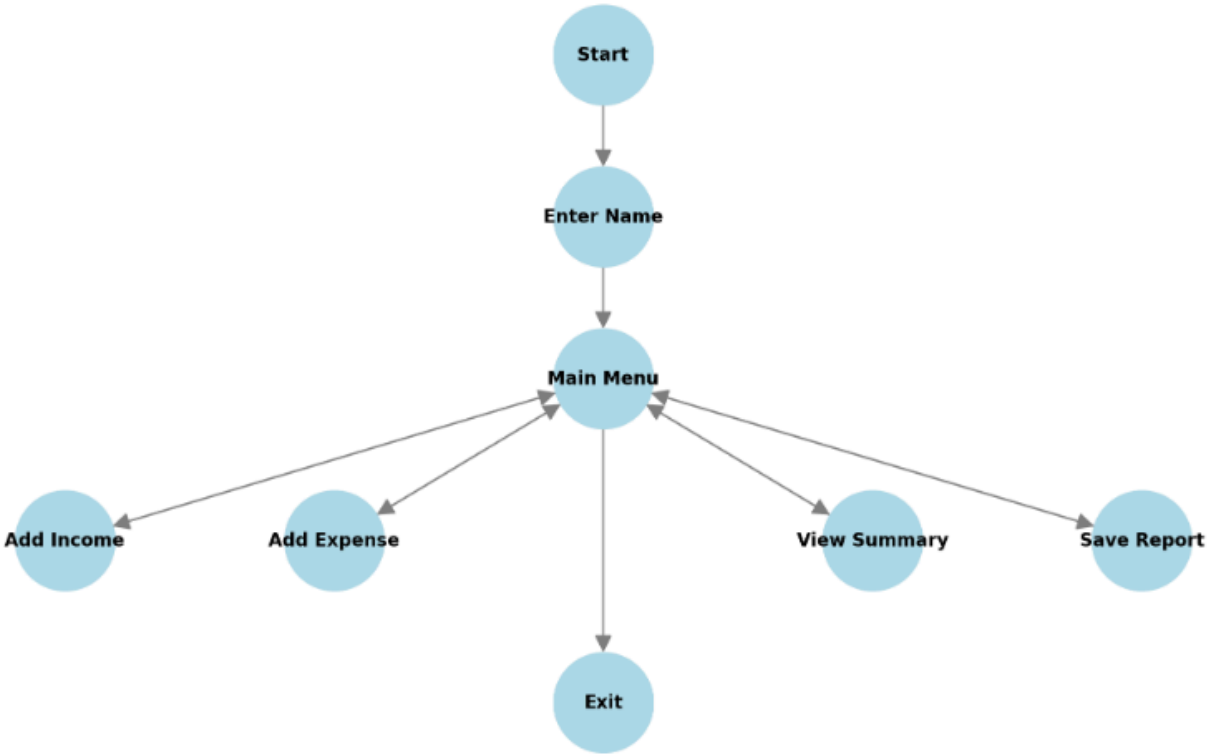
Purpose

The **"Virtual Personal Finance Manager"** project was chosen to address a real-life problem of managing personal finances effectively. It provides an opportunity to apply and reinforce programming fundamentals such as object-oriented programming, file handling, loops, and conditional statements in a practical and meaningful way. The project's relevance lies in its ability to integrate these core concepts into a functional, user-friendly application, demonstrating how programming can solve everyday challenges while enhancing coding skills.

Tools and Technologies

- **Programming Language:** C++
- **Development Environment:** Visual Studio Code, or any C++ IDE

Project Design



Implementation

The implementation of the Virtual Personal Finance Manager revolves around a well-structured class named ***FinanceManager*** , which encapsulates all functionalities related to tracking income and expenses, viewing financial summaries, and generating reports. This class serves as the backbone of the application, utilizing core programming concepts such as object-oriented programming, loops, conditions, and file handling to provide a robust user experience.

The FinanceManager Class

Is the heart of the application, the ***FinanceManager*** class is responsible for managing financial data. It includes methods for adding income and expenses, generating summaries, and saving reports.

Code

```
#include <iostream>
#include <fstream>
#include <iomanip>
#include <string>
using namespace std;

class FinanceManager {
private:
    double income;
    double expenses;
    double savings;
    string username;

public:
    FinanceManager() : income(0), expenses(0), savings(0) {}

    void setUsername() {
        cout << "Enter your name: ";
        getline(cin, username);
    }

    void addIncome() {
        double amount;
        cout << "Enter income amount: $";
        cin >> amount;
        income += amount;
        updateSavings();
        cout << "Income added successfully!\n";
    }
```

```
    void viewSummary() const {
        cout << "\n--- Financial Summary for " << username << " ---\n";
        cout << fixed << setprecision(2);
        cout << "Total Income: $" << income << endl;
        cout << "Total Expenses: $" << expenses << endl;
        cout << "Savings: $" << savings << endl;
    }

    void saveToFile() const {
        ofstream file("FinanceReport.txt");
        if (!file) {
            cout << "Error creating report file.\n";
            return;
        }

        file << "--- Financial Report ---\n";
        file << "Name: " << username << "\n";
        file << fixed << setprecision(2);
        file << "Total Income: $" << income << endl;
        file << "Total Expenses: $" << expenses << endl;
        file << "Savings: $" << savings << endl;

        file.close();
        cout << "Report saved to 'FinanceReport.txt'\n";
    }

private:
    void updateSavings() {
        savings = income - expenses;
    }
};

int main() {
    FinanceManager manager;
    int choice;

    cout << "=== Virtual Personal Finance Manager ===\n";
    manager.setUsername();

    do {
        cout << "\n1. Add Income\n2. Add Expense\n3. View Summary\n4. Save Report\n5. Exit\n";
        cout << "Choose an option: ";
        cin >> choice;
        cin.ignore(); // To clear the input buffer

        switch (choice) {
        case 1:
            manager.addIncome();
            break;
        case 2:
            manager.addExpense();
            break;
        case 3:
            manager.viewSummary();
            break;
        case 4:
            manager.saveToFile();
            break;
        case 5:
            cout << "Exiting... Have a great day!\n";
            break;
        default:
            cout << "Invalid choice! Please try again.\n";
        }
    } while (choice != 5);

    return 0;
}
```

Output

```
=== Virtual Personal Finance Manager ===
Enter your name: Amna

1. Add Income
2. Add Expense
3. View Summary
4. Save Report
5. Exit
Choose an option: 1
Enter income amount: $24567
Income added successfully!

1. Add Income
2. Add Expense
3. View Summary
4. Save Report
5. Exit
Choose an option: 2
Enter expense amount: $1980
Expense added successfully!

1. Add Income
2. Add Expense
3. View Summary
4. Save Report
5. Exit
Choose an option: 3

--- Financial Summary for Amna ---
Total Income: $24567.00
Total Expenses: $1980.00
Savings: $22587.00

1. Add Income
2. Add Expense
3. View Summary
4. Save Report
5. Exit
Choose an option: 4
Report saved to 'FinanceReport.txt'
```

```
1. Add Income
2. Add Expense
3. View Summary
4. Save Report
5. Exit
Choose an option: 5
Exiting... Have a great day!

--------------------------------
Process exited after 36.6 seconds with return value 0
Press any key to continue . . .
```

Testing

Result

The application successfully tracks income, expenses, and savings. It generates financial reports, ensuring accurate financial management.

Conclusion

This project provided hands-on experience with programming fundamentals, file handling, and user interaction. It reinforced the importance of modular coding and debugging in creating efficient software solutions.

THANKYOU!!