A Report

on

Creational Activity Learning

"Personal Finance Management System"

Submitted By

Machave Sahil Sandeep

Roll No: 97

PRN : UCS22M1085

(TY Computer)

Guided By

Prof. P.M. Dhanrao

Department of Computer Engineering

SANJIVANI COLLEGE OF ENGINEERING

(An Autonomous Institution)

Kopargaon– 423 603, Maharashtra

In the academic year 2024-25

Sanjivani College of Engineering, Kopargaon

Department of Computer Engineering

CERTIFICATE

This is to certify that

Machave Sahil Sandeep

(TY Computer)

Has successfully completed his CAL report on

" Personal Finance Management System"

Towards the partial fulfillment of Bachelor's degree In Computer Engineering During the academic year 2024-25.

Prof. P.M. Dhanrao
[Subject Teacher)]

Dr. M.A.Jawale
[H.O.D. Comp Engg]

ACKNOWLEDGEMENT

I would like to take this opportunity to express my sincere gratitude to all those who helped me directly or indirectly in the successful completion of this project.

First and foremost, I would like to thank my project supervisor, **Prof. P.M. Dhanrao** for their invaluable guidance, timely suggestions, and continuous support throughout the development of the Personal Finance Management System project. Their insights and encouragement inspired me to explore new dimensions of learning and enhanced the quality of my work significantly.

I would also like to extend my heartfelt thanks to the **Dr. M.A.Jawale** of Head of the Computer Department and all the faculty members of the Computer Department at Sanjivani College of Engineering for providing the necessary facilities and resources that enabled me to work efficiently on this project.

Special thanks to my family and friends for their constant motivation and emotional support throughout the course of this project. Their encouragement and belief in my capabilities helped me stay focused and confident.

Finally, I would like to thank all those who contributed directly or indirectly in making this project a success. This experience has enriched my knowledge and given me the confidence to take up future technical challenges with greater assurance.

Thank you all.

Mr. Machave Sahil
T.Y. Computer
Roll No.97

INDEX

Abstract

The **Personal Finance Management System (PFMS)** is a Java-based desktop application developed using the Eclipse IDE, aimed at helping users track and manage their personal finances efficiently. The system allows users to input and monitor their income, expenses, savings, and investments, while also providing tools for budget planning and financial goal tracking. PFMS categorizes transactions, generates detailed reports, and offers visual representations through charts and graphs for better understanding of financial trends. The application ensures data persistence through database integration and emphasizes security and usability for a smooth user experience. This project not only promotes financial discipline but also introduces users to essential budgeting practices through a simple and interactive interface.

Technologies Used

- **Frontend/UI:**
  - Java Swing (for GUI development)
  - AWT (Abstract Window Toolkit) for basic window management
- **Backend:**
  - Java SE (Standard Edition)
- **Database:**
  - MySQL or SQLite (for storing financial data)
  - JDBC (Java Database Connectivity) for database integration
- **Development Environment:**
  - Eclipse IDE

Introduction

In today's dynamic economic environment, personal financial management has become a vital aspect of everyday life. With the growing complexity of income sources, expenses, savings, investments, and debts, individuals often face challenges in keeping their finances organized. Manual tracking methods, such as spreadsheets or written records, can be inefficient, error-prone, and difficult to maintain. To address these issues, the Personal Finance Management System (PFMS) is developed as a comprehensive software solution to help individuals gain control over their financial lives.

The PFMS is a Java-based desktop application designed and implemented using the Eclipse IDE. It provides users with a centralized platform to monitor their income and expenses, plan budgets, manage savings goals, and visualize their financial data. The application uses Java Swing for a user-friendly graphical interface, and JDBC for establishing database connectivity with MySQL or SQLite, ensuring efficient and secure data management. Features like transaction categorization, real-time balance tracking, monthly summaries, and graphical reports (integrated via JFreeChart) help users understand their spending patterns and make informed financial decisions.

The system emphasizes simplicity, accuracy, and data privacy while promoting financial literacy. Users can set budgets, receive alerts for overspending, and track financial goals over time. The ability to generate detailed reports and visualize data through charts makes PFMS a powerful tool for both short-term budgeting and long-term financial planning.

This project is especially relevant for individuals seeking an organized approach to managing personal finances without relying on third-party online services. The PFMS promotes responsible financial behaviour and enables users to take charge of their economic well-being through a simple, efficient, and secure system. It demonstrates the practical application of core programming concepts and database management in real-world financial scenarios.

Problem Statement

In today's fast-paced digital era, managing personal finances has become increasingly complex. Individuals must track various financial elements such as income from multiple sources, recurring expenses, loans, savings, and investments. Despite the importance of financial planning, many people either rely on manual methods like notebooks and spreadsheets or fail to track their finances altogether. This leads to common problems such as overspending, missed bill payments, inadequate savings, and poor budgeting.

Manual tracking methods are time-consuming, prone to human error, and lack automation or intelligent insights. Moreover, these methods do not provide real-time visibility into a user's financial status and often fail to offer predictive analysis or trend visualizations. On the other hand, many available online finance management tools come with limitations such as subscription fees, complex interfaces, or security/privacy concerns—especially when dealing with sensitive financial data.

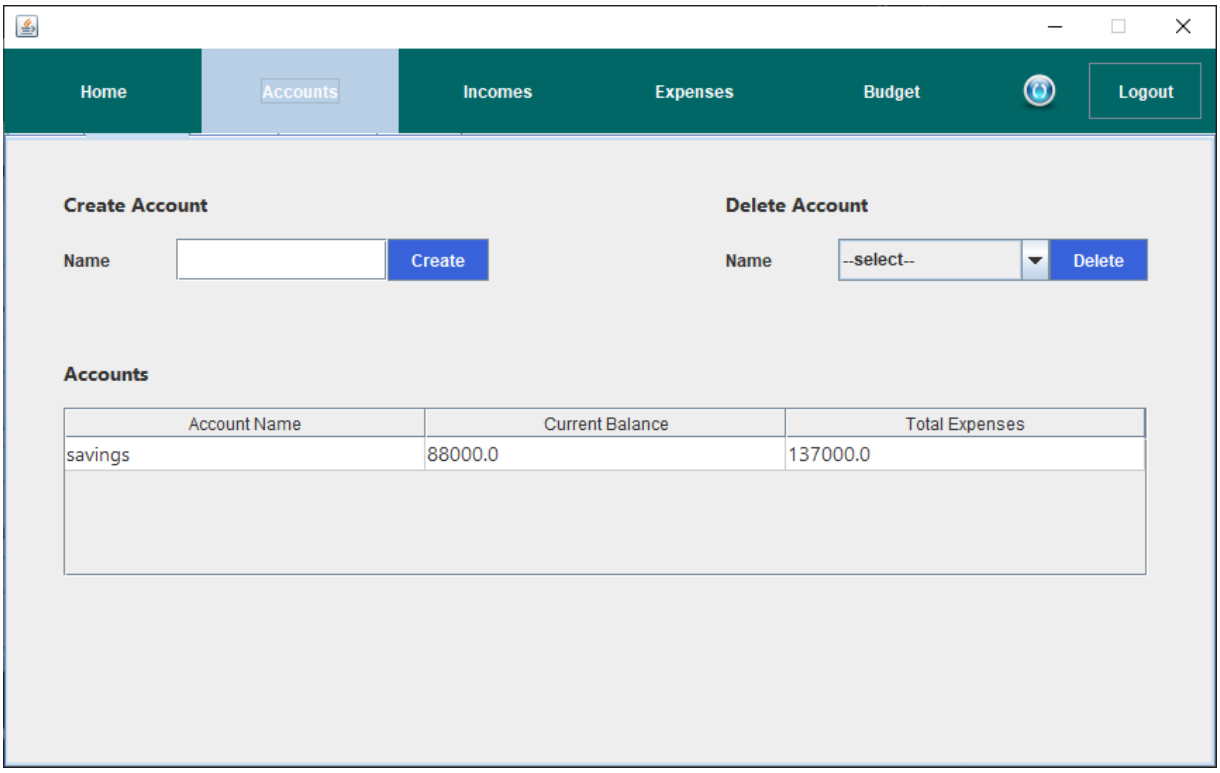
There is a clear need for a user-friendly, secure, and offline desktop application that allows users to efficiently manage their personal finances in one place. Such a system should provide features like categorizing expenses, setting budgets, tracking income, monitoring savings goals, and generating visual reports to better understand spending behaviour.

The Personal Finance Management System aims to address these challenges by offering a robust, Java-based desktop application that is accessible, easy to use, and tailored for individual needs. Developed using Eclipse IDE, the system leverages technologies like Java Swing, JDBC, and MySQL/SQLite to provide a smooth and interactive experience. It not only helps users track financial data but also promotes financial discipline and planning.

In summary, the PFMS solves the problem of scattered and inefficient personal finance tracking by offering a centralized, secure, and intuitive solution for everyday financial management.

Screenshots

Home Page ➔



Project Code

```
import java.time.LocalDate;

public class Transaction {
    private LocalDate date;
    private String description;
    private double amount;
    private String type; // "income" or "expense"

    public Transaction(LocalDate date, String description, double amount, String type) {
        this.date = date;
        this.description = description;
        this.amount = amount;
        this.type = type;
    }

    public LocalDate getDate() { return date; }
    public String getDescription() { return description; }
    public double getAmount() { return amount; }
    public String getType() { return type; }

    @Override
    public String toString() {
        return date + "," + description + "," + amount + "," + type;
    }

    public static Transaction fromString(String line) {
        String[] parts = line.split(",");
        return new Transaction(LocalDate.parse(parts[0]), parts[1],
Double.parseDouble(parts[2]), parts[3]);
    }
}
```

```
import java.io.*;
import java.util.*;
import java.time.LocalDate;

public class FinanceManager {
    private List<Transaction> transactions = new ArrayList<>();
    private final String fileName = "transactions.txt";

    public FinanceManager() {
        loadTransactions();
    }
```

```
    public void addTransaction(Transaction t) {
        transactions.add(t);
        saveTransaction(t);
    }

    public double getBalance() {
        double balance = 0;
        for (Transaction t : transactions) {
            if (t.getType().equalsIgnoreCase("income")) {
                balance += t.getAmount();
            } else if (t.getType().equalsIgnoreCase("expense")) {
                balance -= t.getAmount();
            }
        }
        return balance;
    }

    public void showTransactions() {
        System.out.println("Date\t\tDescription\tAmount\tType");
        for (Transaction t : transactions) {
            System.out.printf("%s\t%s\t%.2f\t%s\n", t.getDate(), t.getDescription(),
t.getAmount(), t.getType());
        }
    }

    private void saveTransaction(Transaction t) {
        try (BufferedWriter writer = new BufferedWriter(new FileWriter(fileName, true))) {
            writer.write(t.toString());
            writer.newLine();
        } catch (IOException e) {
            System.out.println("Error saving transaction: " + e.getMessage());
        }
    }

    private void loadTransactions() {
        try (BufferedReader reader = new BufferedReader(new FileReader(fileName))) {
            String line;
            while ((line = reader.readLine()) != null) {
                transactions.add(Transaction.fromString(line));
            }
        } catch (FileNotFoundException e) {
            // Ignore if file does not exist
        } catch (IOException e) {
            System.out.println("Error loading transactions: " + e.getMessage());
        }
    }
}
```

```
import java.util.Scanner;
import java.time.LocalDate;

public class Main {
   public static void main(String[] args) {
      Scanner scanner = new Scanner(System.in);
      FinanceManager manager = new FinanceManager();

      while (true) {
         System.out.println("\n--- Personal Finance Manager ---");
         System.out.println("1. Add Income");
         System.out.println("2. Add Expense");
         System.out.println("3. View Transactions");
         System.out.println("4. View Balance");
         System.out.println("5. Exit");
         System.out.print("Choose an option: ");
         int option = scanner.nextInt();
         scanner.nextLine(); // clear buffer

         switch (option) {
            case 1:
            case 2:
               System.out.print("Enter description: ");
               String desc = scanner.nextLine();
               System.out.print("Enter amount: ");
               double amt = scanner.nextDouble();
               scanner.nextLine(); // clear buffer
               String type = (option == 1) ? "income" : "expense";
               Transaction t = new Transaction(LocalDate.now(), desc, amt, type);
               manager.addTransaction(t);
               System.out.println(type + " recorded.");
               break;

            case 3:
               manager.showTransactions();
               break;

            case 4:
               System.out.printf("Current Balance: %.2f\n", manager.getBalance());
               break;

            case 5:
               System.out.println("Goodbye!");
               scanner.close();
               System.exit(0);
               break;

            default:
               System.out.println("Invalid option!");
```

```
            }
        }
    }
}
```

Conclusion

The Personal Finance Management System developed in Java provides a simple yet effective solution for tracking income and expenses. Designed as a console-based application, it allows users to record transactions, view a detailed list of their financial activities, and monitor their current balance. The system emphasizes ease of use through a clear menu-driven interface, making it accessible even for users with minimal technical experience.

By implementing object-oriented programming principles, the system is structured and modular, separating concerns such as transaction data, financial logic, and user interaction. File handling is used to store and retrieve data persistently through a local text file (transactions.txt), ensuring that financial records are maintained across sessions without requiring a database.

This project demonstrates the practical application of Java concepts like classes, lists, file I/O, and date handling, while also introducing basic financial management principles. Although simple, it can be extended further with features such as data categorization, reporting, budget setting, or graphical user interfaces using JavaFX or Swing.

In conclusion, the Personal Finance Management System serves as a solid foundation for both educational purposes and basic financial tracking needs. It highlights the importance of managing personal finances efficiently and how software tools can support that goal.

References

- **Oracle Documentation.** (n.d.). *The Java™ Tutorials.* Retrieved from https://docs.oracle.com/javase/tutorial/

    - Official guide for learning Java programming, including object-oriented principles and file handling.

- **Schildt, H.** (2018). *Java: The Complete Reference* (11th Edition). McGraw-Hill Education.

    - A comprehensive reference book on Java programming, covering basics to advanced features.

- **Eck, D. J.** (2021). *Introduction to Programming Using Java* (Version 9.0). Retrieved from https://math.hws.edu/javanotes/

    - A free and well-structured book for understanding Java and building projects.

- **Baeldung.** (n.d.). *Java File IO (Reading and Writing Files).* Retrieved from https://www.baeldung.com/java-read-write-file

    - Practical examples and explanations on how to handle file input and output in Java.

- **GeeksforGeeks.** (n.d.). *Java Projects – Beginner to Advanced.* Retrieved from https://www.geeksforgeeks.org/java-projects/

    - Reference for simple and advanced Java projects for students and developers.

- **Investopedia.** (n.d.). *Personal Finance Basics.* Retrieved from https://www.investopedia.com/personal-finance-4427765

    - Basic financial concepts useful for structuring a personal finance application.