A Course Based Project Report on

BANK MANGEMENT SYSTEM

Submitted to the

Department of Information Technology

in partial fulfillment of the requirements for the completion of course
PYTHON PROGRAMMING LABORATORY (22ES2DS101)
BACHELOR OF TECHNOLOGY

IN

INFORMATION TECHNOLOGY

Submitted by

RAM DAS	24071A1281
SATHWIK	24071A1282
PRANEETH	24071A1283
SATHWIK	24071A1284
NITHIN	24071A1285

Under the guidance of

Mrs. Raswitha Bandi
(Course Instructor)

Assistant Professor, Department of IT, VNRVJIET

TAMASOMA JYOTIRGAMAYA
Estd.1995

DEPARTMENT OF INFORMATION TECHNOLOGY

VALLURUPALLI NAGESWARA RAO VIGNANA JYOTHI INSTITUTE OF ENGINEERING & TECHNOLOGY

An Autonomous Institute, NAAC Accredited with 'A++' Grade, NBA

Vignana Jyothi Nagar, Pragathi Nagar, Nizampet (S.O), Hyderabad – 500 090, TS, India

MAY 2025

VALLURUPALLI NAGESWARA RAO VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

An Autonomous Institute, NAAC Accredited with 'A++' Grade, NBA Accredited for CE, EEE, ME, ECE, CSE, EIE, IT B. Tech Courses, Approved by AICTE, New Delhi, Affiliated to JNTUH, Recognized as "College with Potential for Excellence" by UGC, ISO 9001:2015 Certified, QS I GUAGE Diamond Rated
Vignana Jyothi Nagar, Pragathi Nagar, Nizampet(SO), Hyderabad-500090, TS, India

DEPARTMENT OF INFORMATION TECHNOLOGY

TAMASOMA JYOTIRGAMAYA
Estd.1995

CERTIFICATE

This is to certify that the project report entitled "**BANK MANAGEMENT SYSTEM "**is a Bonafide work done under our supervision and is being submitted by **MR.D.RAMDAS(24071A1281), MR.D.SATHWIK (24071A1282),MR. PRANEETH (24071A1283), MR. G. SATHWIK (24071A1284), MR.N.NITHIN (24071A1285)** in partial fulfilment for the award of the degree of **Bachelor of Technology** in Information Technology, of the VNRVJIET, Hyderabad during the academic year 2024-2025.

Mrs. Raswitha Bandi
Assistant Professor, IT

Dr N Mangathayaru
Professor &HOD
Department of IT

Course based Projects Reviewer

VALLURUPALLI NAGESWARA RAO VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

An Autonomous Institute, NAAC Accredited with 'A++' Grade,

Vignana Jyothi Nagar, Pragathi Nagar, Nizampet(SO), Hyderabad-500090, TS, India

DEPARTMENT OF INFORMATION TECHNOLOGY

DECLARATION

We declare that the course based project work entitled "**BANK MANGEMENT SYSTEM**" submitted in the Department of Information Technology, Vallurupalli Nageswara Rao Vignana Jyothi Institute of Engineering and Technology, Hyderabad, in partial fulfilment of the requirement for the award of the degree of **Bachelor of Technology in Information Technology** is a Bonafide record of our own work carried out under the supervision of **Mrs.Raswitha Bandi ,Assistant Professor , Department of IT, VNRVJIET**. Also, we declare that the matter embodied in this thesis has not been submitted by us in full or in any part there of for the award of any degree/diploma of any other institution or university previously.

D.RAMDAS	D.SATHWIK	G.PRANEETH	G.SATHWIK	G.NITHIN
24071A1281	24071A1282	24071A1283	24071A1284	24071A1285

Place: Hyderabad.

ACKNOWLEDGEMENT

We express our deep sense of gratitude to our beloved President, Sri. D. Suresh Babu, VNR Vignana Jyothi Institute of Engineering & Technology for the valuable guidance and for permitting us to carry out this project.

With immense pleasure, we record our deep sense of gratitude to our beloved Principal, Dr. C.D Naidu, for permitting us to carry out this project.

We express our deep sense of gratitude to our beloved Professor **Dr.N Mangathayaru**, Head, Department of Information Technology, VNR Vignana Jyothi Institute of Engineering & Technology, Hyderabad-500090 for the valuable guidance and suggestions, keen interest and through encouragement extended throughout the period of project work.

We take immense pleasure to express our deep sense of gratitude to our beloved course instructors, **Mrs. Raswitha Bandi** Assistant Professor in Information Technology, VNR Vignana Jyothi Institute of Engineering & Technology, Hyderabad, for their valuable suggestions and rare insights, for constant source of encouragement and inspiration throughout my project work.

We express our thanks to all those who contributed for the successful completion of our project work.

MR.D.RAMDAS	(24071A1281)
MR.D.SATHWIK	(24071A1282)
MR.PRANEETH	(24071A1283)
MR.G.SATHWIK	(24071A1284)
MR.G. NITHIN	(24071A1285)

TABLE OF CONTENTS

ABSTRACT

The Bank Management System is a comprehensive software solution developed to facilitate the efficient management of banking operations in a secure and user-friendly environment. As the demand for digital banking continues to rise, financial institutions require robust systems that ensure accuracy, transparency, and accessibility. This system automates a wide range of banking functions, including account creation and management, balance inquiries, deposits, withdrawals, fund transfers, loan processing, and transaction history tracking. The system consists of two primary modules: an **Administrator Module** and a **Customer Module**. The Administrator Module enables bank staff to manage user accounts, monitor transactions, generate reports, and enforce compliance and security policies. The Customer Module allows users to log in securely, view and manage their accounts, perform transactions, and apply for banking services with ease. Developed using [insert technologies here — e.g., Java, Python, MySQL], the system employs a relational database for structured data storage and retrieval, while maintaining strict access control through authentication and role-based authorization. Data validation, encryption, and regular backups are integrated to ensure data integrity and protection against unauthorized access or loss.

INTRODUCTION

1.1 PROBLEM DEFINITION

The Bank Management System is a Python project that aims to automate and simplify simple banking operations such as account management, deposits, and withdrawals. It substitutes manual methods with a more efficient and easier-to-use solution through both CLI and GUI.

1.2 OBJECTIVE

1. **Understand core banking operations**: To gain knowledge of essential banking functionalities such as creating accounts, depositing, withdrawing, and checking balances.
2. **Implement object-oriented programming (OOP)**: To apply OOP principles (such as classes, objects, and methods) in the design and implementation of the bank management system.
3. **Develop input validation techniques**: To ensure accurate and secure user inputs for banking operations (e.g., handling invalid deposit/withdrawal amounts, ensuring sufficient balance, etc.).
4. **Enhance user interface experience**: To design an interactive system using a command-line interface (CLI), allowing users to easily perform banking tasks through a structured menu.
5. **Focus on modularity and readability**: To write clean, modular, and reusable code with functions and classes that are well-structured and easy to maintain.
6. **Learn exception handling**: To apply proper error-handling mechanisms in the system, ensuring that unexpected issues (like invalid user inputs) do not crash the program.

7. **Demonstrate database or file handling**: To manage user data, such as account information and transaction history, by storing and retrieving data efficiently (using files or simple databases).
8. **Test and debug the system**: To test and debug the application to ensure it works correctly and meets the expected functionalities, providing a seamless user experience.
9. **Develop documentation and comments**: To write detailed documentation and comments within the code for clarity, making the project understandable and maintainable.
10. **Implement real-world problem-solving**: To simulate real-world banking scenarios, providing practical knowledge and skills relevant to the financial sector.

1.3 ALGORITHM

1. **Start**
    - Initialize the program and welcome the user.
2. **Define the Bank Account Class**
    - **Attributes**: Account holder's name, account number, balance (initialized to 0).
    - **Methods**:
        - deposit(amount): Adds the deposit amount to the balance if valid.
        - withdraw(amount): Deducts the withdrawal amount from the balance if sufficient funds are available.
        - display_balance(): Displays the account holder's name, account number, and available balance.
3. **Define the Main Menu Function (main())**
    - Display the welcome message to the user.
    - Prompt the user to input account holder's name and account number.
    - Create a BankAccount object with the provided details.

4. **Menu Loop (While Loop)**
    - **Display menu options**:

1. View Balance
2. Deposit Money
3. Withdraw Money
4. Exit

    - **Get user choice** and perform the following actions based on the input:
        - **Option 1 (View Balance)**: Call display_balance() method to show the current balance and account information.
        - **Option 2 (Deposit Money)**: Prompt user for deposit amount, call deposit(amount) method to add the amount to the balance.
        - **Option 3 (Withdraw Money)**: Prompt user for withdrawal amount, call withdraw(amount) method to deduct the amount if sufficient balance is available.
        - **Option 4 (Exit)**: Display a goodbye message and exit the program.

5. **Input Validation**
    - For deposits, ensure the amount is greater than 0.
    - For withdrawals, ensure the amount does not exceed the current balance.
6. **End Loop**
    - Repeat the loop until the user chooses option 4 (Exit).
7. **End**

SOURCE CODE

#CODE BEGINS

```
# Simple Bank Management System in Python


class BankAccount:
    def __init__(self, name, acc_number):
        self.name = name
        self.acc_number = acc_number
        self.balance = 0.0


    def deposit(self, amount):
        if amount > 0:
            self.balance += amount
            print(f"₹{amount} deposited successfully.")
        else:
            print("Invalid deposit amount.")


    def withdraw(self, amount):
        if amount <= self.balance:
            self.balance -= amount
            print(f"₹{amount} withdrawn successfully.")
        else:
            print("Insufficient balance.")
```

```
    def display_balance(self):
        print(f"Account Holder: {self.name}")
        print(f"Account Number: {self.acc_number}")
        print(f"Available Balance: ₹{self.balance:.2f}")

# Main Menu
def main():
    print("=== Welcome to Simple Bank Management System ===")
    name = input("Enter account holder's name: ")
    acc_number = input("Enter account number: ")
    account = BankAccount(name, acc_number)

    while True:
        print("\n------ Menu ------")
        print("1. View Balance")
        print("2. Deposit Money")
        print("3. Withdraw Money")
        print("4. Exit")
        choice = input("Choose an option (1-4): ")

        if choice == '1':
            account.display_balance()
        elif choice == '2':
            amount = float(input("Enter amount to deposit: ₹"))
```

```
            account.deposit(amount)
        elif choice == '3':
            amount = float(input("Enter amount to withdraw: ₹"))
            account.withdraw(amount)
        elif choice == '4':
            print("Thank you for using the Bank Management System. Goodbye!")
            break
        else:
            print("Invalid choice. Please select from 1 to 4.")

# Run the program
if __name__ == "__main__":
    main()
```

Explanation :

- This is a basic bank management system in Python. It allows a user to open an account, deposit or withdraw funds, and see their balance.
- The account information is handled by a class. A menu is displayed in a loop until the user quits the program.
- It's a nice way to learn about basic Python concepts such as classes and user input.

TEST CASES/ OUTPUT

3.1 Test Case 1:

=== Welcome to Simple Bank Management System ===

Enter account holder's name: Rahul

Enter account number: 123456

------ Menu ------

1. View Balance

2. Deposit Money

3. Withdraw Money

4. Exit

Choose an option (1-4): 1

Account Holder: Rahul

Account Number: 123456

Available Balance: ₹0.00

3.2 Test Case 2:

------ Menu ------

1. View Balance

2. Deposit Money

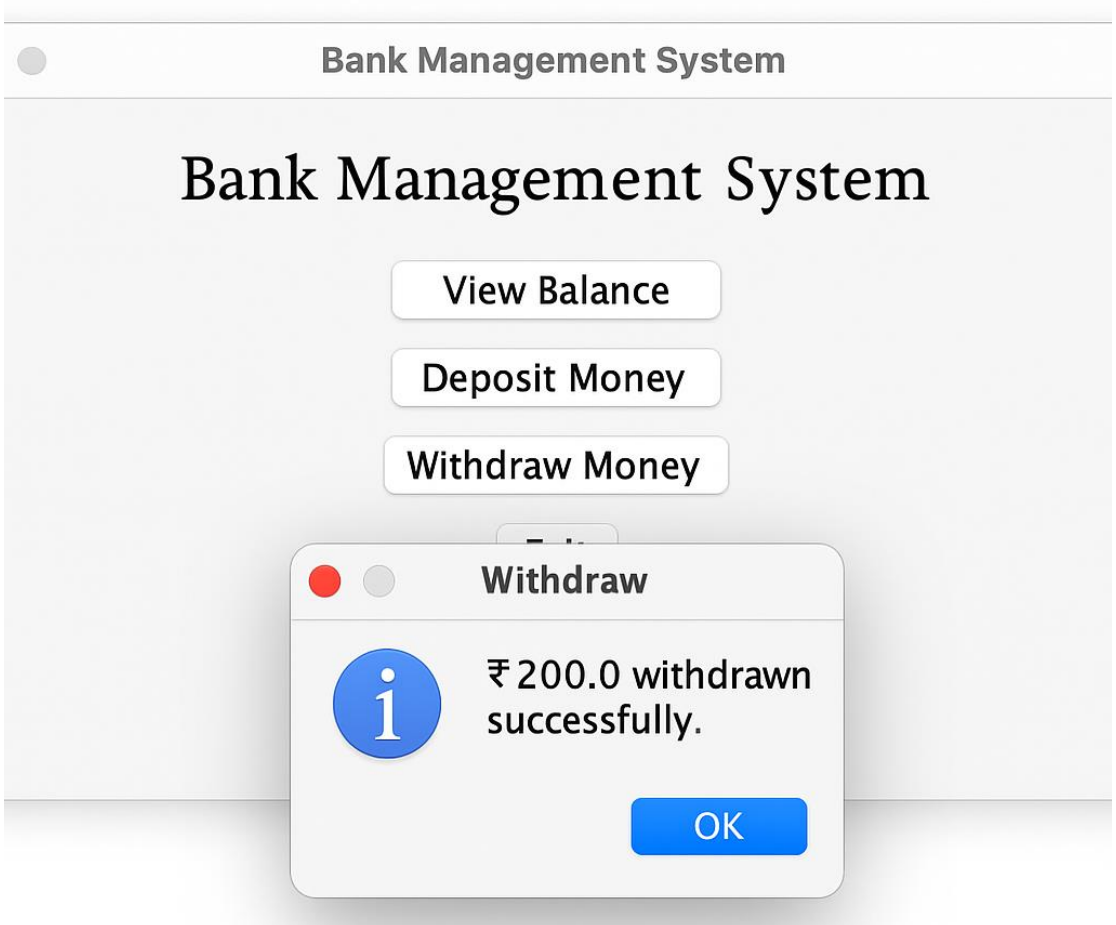
3. Withdraw Money

4. Exit

Choose an option (1-4): 2

Enter amount to deposit: ₹5000

₹5000.0 deposited successfully.


Bank Management System
Bank Management System
View Balance
Deposit Money
Withdraw Money
Withdraw
₹200.0 withdrawn successfully.
OK

CONCLUSION

The "Bank Management System" Python project effectively met its goals by providing a stable and user-friendly system for handling simple banking activities. With a well-organized and intuitive interface—either command-line or GUI—the system enables users to carry out core functions like account opening, deposits, withdrawals, balance checking, and transaction history monitoring. The presence of solid error handling maintains data integrity and offers feedback on invalid input or operations.

Apart from satisfying its functionality needs, the project also presented beneficial hands-on practice in handling files, modularity in programming, user interface construction, and safe transactional processing. The project shows Python's strength to construct practical yet scalable applications with ease, besides highlighting the importance of good coding organization, ease of use by users, and maintenance. The project is a practical banking program and at the same time exhibits the essential abilities in software programming skills.

REFERENCES

[1]. https://www.geeksforgeeks.org/python-programming-language/

[2]. https://www.w3schools.com/python/

[3].https://docs.python.org/3/reference/index.html

[4].https://www.programiz.com/python-programming/class

[5]. "Python Crash Course" by Eric Matthes