

Industrial Internship Report on Banking Information System

Prepared by

MANASI PANT

Executive Summary

This report provides details of the Industrial Internship provided by upskill Campus and The IoT Academy in collaboration with Industrial Partner UniConverge Technologies Pvt Ltd (UCT).

This internship was focused on a project/problem statement provided by UCT. We had to finish the project including the report in 4 weeks' time.

My project was 'Banking Information System' which involved creating a prototype of a banking system using Core Java.

This internship gave me a very good opportunity to get exposure to Industrial problems and design/implement solution for that. It was an overall great experience to have this internship.

Contents

MANASI PANT	1
Executive Summary	1
1. Preface	3
2. Introduction	4
(2.1) About UniConverge Technologies Pvt Ltd.....	4
(2.2) About upskill Campus (USC)	8
(2.3) The IoT Academy	9
(2.4) Objectives of this Internship program	10
(2.5) Glossary	10
(2.6) References	11
3. Problem Statement.....	12
4. Existing and Proposed solution.....	13
5. Proposed Design/ Model.....	14
(5.1) High Level Diagram:	21
(5.2) Low Level Diagram	23
6. Performance Test.....	24
(6.1) Test Plan / Test Cases.....	25
(6.2) Test Procedure.....	25
(6.3) Performance Outcome.....	26
7. My learnings.....	26
8. Future work scope.....	27

1. Preface

This report summarizes the 6 weeks of internship experience as part of the Core Java project 'Banking Information System'.

During this period, I strengthened my foundational knowledge of Java, learned core object-oriented concepts, and built a console-based prototype of a banking system. The project involved implementing real-world banking operations like user registration, deposit, withdrawal, fund transfer, transaction history, and secure login.

Week by week, I progressed from learning Java basics to building functional modules and finally ensuring data persistence with file handling. These 6 weeks provided deep insight into how software systems are planned, developed, tested, and delivered.

I extend heartfelt thanks to my mentors at upskill Campus, The IoT Academy, and UniConverge Technologies Pvt. Ltd. for their guidance and support. I also appreciate all the online learning communities and resources that enriched my experience.

To my juniors: Stay curious and persistent. Real learning comes from doing—this internship proves that.

2. Introduction

(2.1) About UniConverge Technologies Pvt Ltd

A company established in 2013 and working in Digital Transformation domain and providing Industrial solutions with prime focus on sustainability and RoI.

For developing its products and solutions it is leveraging various **Cutting Edge Technologies e.g. Internet of Things (IoT), Cyber Security, Cloud computing (AWS, Azure), Machine Learning, Communication Technologies (4G/5G/LoRaWAN), Java Full Stack, Python, Front end** etc.



UCT IoT Platform

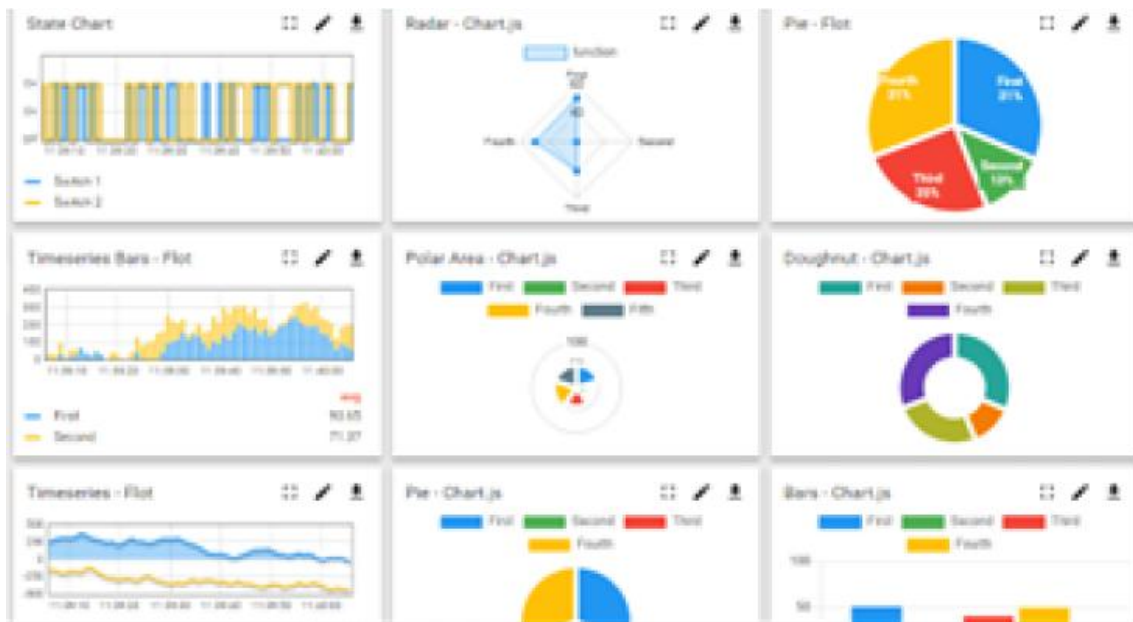
UCT Insight is an IOT platform designed for quick deployment of IOT applications on the same time providing valuable “insight” for your process/business. It has been built in Java for backend and ReactJS for Front end. It has support for MySQL and various NoSql Databases.

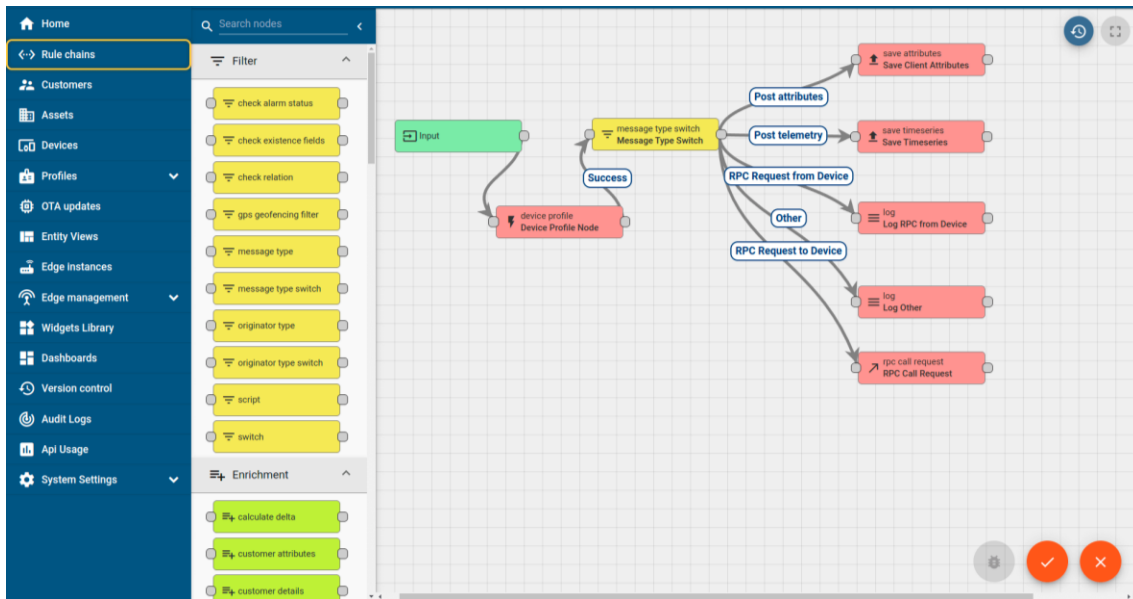
It enables device connectivity via industry standard IoT protocols - MQTT, CoAP, HTTP, Modbus TCP, OPC UA

It supports both cloud and on-premises deployments.

It has features to

- Build Your own dashboard
- Analytics and Reporting
- Alert and Notification
- Integration with third party application(Power BI, SAP, ERP)
- Rule Engine





FACTORY WATCH

Smart Factory Platform

Factory watch is a platform for smart factory needs.

It provides Users/ Factory

- with a scalable solution for their Production and asset monitoring
- OEE and predictive maintenance solution scaling up to digital twin for your assets.
- to unleashed the true potential of the data that their machines are generating and helps to identify the KPIs and also improve them.
- A modular architecture that allows users to choose the service that they what to start and then can scale to more complex solutions as per their demands.

Its unique SaaS model helps users to save time, cost and money.



Machine	Operator	Work Order ID	Job ID	Job Performance	Job Progress		Output		Rejection	Time (mins)				Job Status	End Customer
					Start Time	End Time	Planned	Actual		Setup	Pred	Downtime	Idle		
CNC_S7_81	Operator 1	WO0405200001	4168	58%	10:30 AM		55	41	0	80	215	0	45	In Progress	i
CNC_S7_81	Operator 1	WO0405200001	4168	58%	10:30 AM		55	41	0	80	215	0	45	In Progress	i



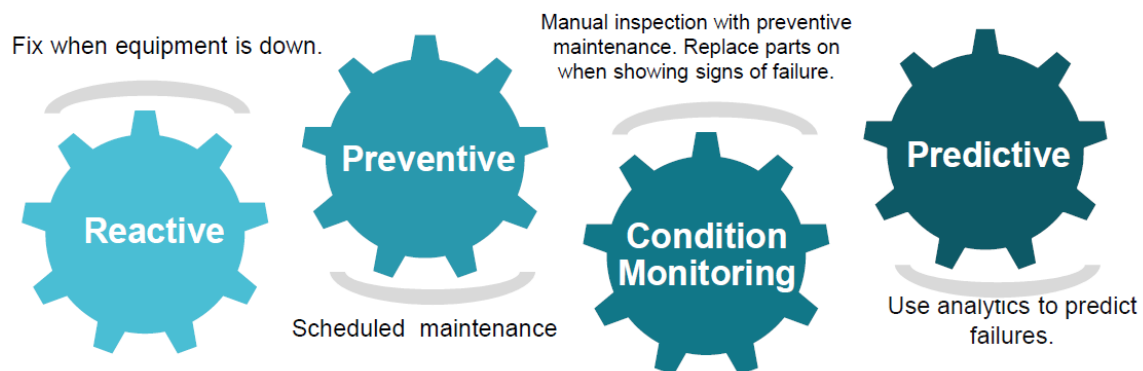


based Solution

UCT is one of the early adopters of LoRAWAN teschnology and providing solution in Agritech, Smart cities, Industrial Monitoring, Smart Street Light, Smart Water/ Gas/ Electricity metering solutions etc.

Predictive Maintenance

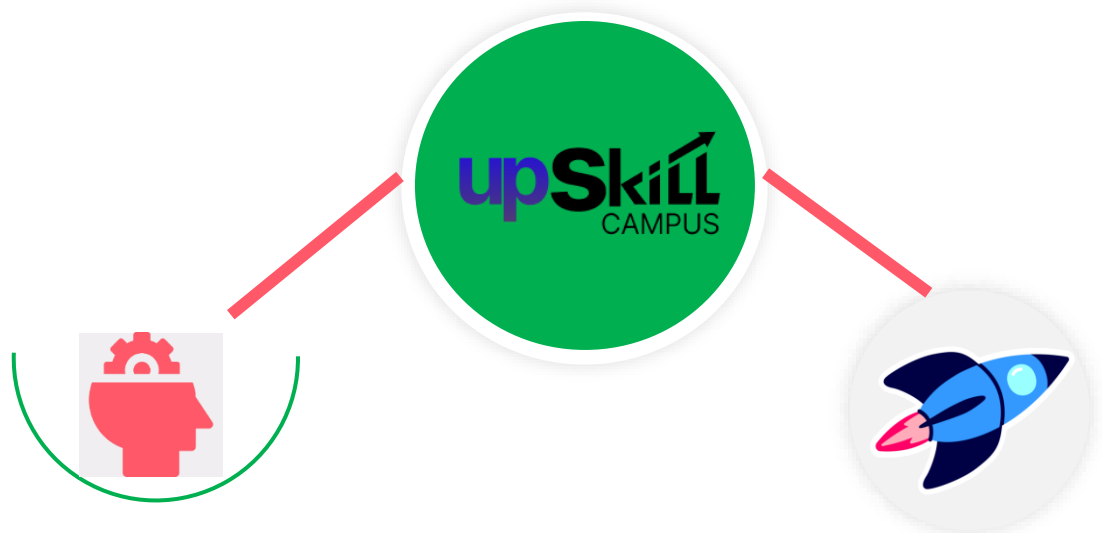
UCT is providing Industrial Machine health monitoring and Predictive maintenance solution leveraging Embedded system, Industrial IoT and Machine Learning Technologies by finding Remaining useful life time of various Machines used in production process.



(2.2) About upskill Campus (USC)

upskill Campus along with The IoT Academy and in association with Uniconverge technologies has facilitated the smooth execution of the complete internship process.

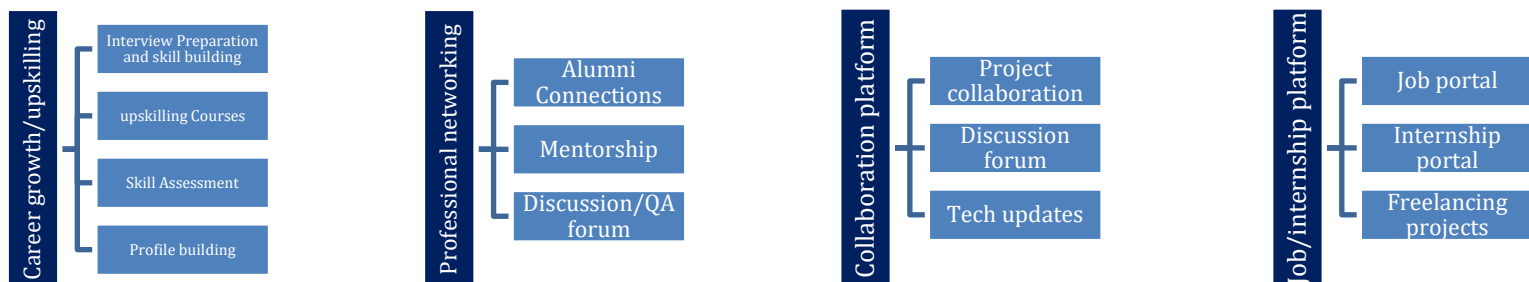
USC is a career development platform that delivers personalized executive coaching in a more affordable, scalable and measurable way.



Seeing need of upskilling in self paced manner along-with additional support services e.g. Internship, projects, interaction with Industry experts, Career growth Services

upSkill Campus aiming to upskill 1 million learners in next 5 year

<https://www.upskillcampus.com/>



(2.3) The IoT Academy

The IoT academy is EdTech Division of UCT that is running long executive certification programs in collaboration with EICT Academy, IITK, IITR and IITG in multiple domains.

(2.4) Objectives of this Internship program

The objective for this internship program was to

- get practical experience of working in the industry.
- to solve real world problems.
- to have improved job prospects.
- to have Improved understanding of our field and its applications.
- to have Personal growth like better communication and problem solving.

(2.5) Glossary

Term	Acronym	Description
Object-Oriented Programming	OOP	A programming paradigm based on the concept of objects, which contain data and methods.
Input/Output	I/O	Refers to the communication between a computer and the outside world (e.g., keyboard input, file output).
Command Line Interface	CLI	A text-based user interface used to interact with software and operating systems.
Integrated Development Environment	IDE	A software suite that provides tools for programming, such as code editor, debugger, and compiler (e.g., Eclipse, IntelliJ).
Java Development Kit	JDK	A software development environment used for developing Java applications.

(2.6) References

1. GitHub Repository (Code):
<https://github.com/mansipant17/upskillcampus>
2. GitHub Report Link:
[BankingInformationSystem ManasiPant USC UCT.pdf](#)
3. Java Tutorials – Oracle Documentation:
<https://docs.oracle.com/javase/tutorial/>
4. Java OOP Concepts – W3Schools:
https://www.w3schools.com/java/java_oop.asp
5. File Handling in Java – GeeksforGeeks:
<https://www.geeksforgeeks.org/file-handling-in-java/>

3. Problem Statement

Develop a prototype of a Banking Information System in Core Java that provides a working preview of the key functionalities of a real banking system. The prototype should demonstrate the core features and flow of the system, showcasing its functionality and usability.

Key functionalities include:

1. User Registration
2. Account Management
3. Deposit and Withdrawal
4. Fund Transfer
5. Account Statements
6. Password Protection
7. Error Handling
8. User Interface
9. Persistence

4. Existing and Proposed solution

Existing solutions often involve complex interfaces and backend setups. Many fail to provide a beginner-friendly, secure, and extendable framework.

My proposed solution simplifies the banking system in a modular way, using Core Java and file-based storage, allowing easy modifications, better error handling, and a clean interface for demonstration.

GitHub Link (Code): <https://github.com/mansipant17/upskillcampus>

GitHub Link (Report):

https://github.com/mansipant17/upskillcampus/blob/main/BankingInformationSystem_ManasiPant_USC_UCT.pdf

5. Proposed Design/ Model

The application follows a **modular design** approach using Object-Oriented Programming in Java. Each class in the system is designed to handle a specific responsibility to maintain separation of concerns. The modules include:

- **User Class:** Handles user details like name, address, contact, and password.
- **Account Class:** Manages account number, balance, deposit, withdrawal, and transactions.
- **BankSystem Class:** Controls the overall flow of registration, login, fund transfer, and other operations.
- **Transaction Class:** Keeps logs of each transaction with date, type, amount, and resulting balance.
- **FileHandler Class (Optional):** Manages file read/write operations for data persistence.

```
--- Welcome to the Banking System ---  
1. User Registration  
2. Login  
3. Exit  
Choose an option: 1
```

Initial Screen (Main Menu)

```
--- Welcome to the Banking System ---  
1. User Registration  
2. Login  
3. Exit  
Choose an option: 1  
Enter name: manasi pant  
Enter address: haldwani  
Enter contact: 8791344254  
Enter initial deposit (must be > 0): 2000  
Set a password (min 4 characters): 1234  
Registration successful! Your account number is: 1000
```

User Registration (First User)

```
--- Welcome to the Banking System ---  
1. User Registration  
2. Login  
3. Exit  
Choose an option: 1  
Enter name: suresh gariya  
Enter address: dehradun  
Enter contact: 9557771126  
Enter initial deposit (must be > 0): 3000  
Set a password (min 4 characters): 1234  
Registration successful! Your account number is: 1001
```

User Registration (Second User)


```
--- Welcome to the Banking System ---  
1. User Registration  
2. Login  
3. Exit  
Choose an option: 2  
Enter account number: 1000  
Enter password: 1234  
Login successful! Welcome, manasi pant
```

Login (First User)

```
--- Account Menu ---  
1. Deposit  
2. Withdraw  
3. Fund Transfer  
4. View Account Statement  
5. Update Account Information  
6. Change Password  
7. Logout
```

Account Menu (Options for logged-in user)

```
--- Account Menu ---  
1. Deposit  
2. Withdraw  
3. Fund Transfer  
4. View Account Statement  
5. Update Account Information  
6. Change Password  
7. Logout  
Choose an option: 1  
Enter deposit amount: 500  
Deposit successful! New balance: 2500.0
```

Deposit Operation (User deposits 500)

```
--- Account Menu ---  
1. Deposit  
2. Withdraw  
3. Fund Transfer  
4. View Account Statement  
5. Update Account Information  
6. Change Password  
7. Logout  
Choose an option: 2  
Enter withdrawal amount: 100  
Withdrawal successful! New balance: 2400.0
```

Withdraw Operation (User withdraws 100)

```
--- Account Menu ---  
1. Deposit  
2. Withdraw  
3. Fund Transfer  
4. View Account Statement  
5. Update Account Information  
6. Change Password  
7. Logout  
Choose an option: 3  
Enter recipient account number: 1001  
Enter transfer amount: 500  
Transfer successful!  
Your new balance: 1900.0
```

Fund Transfer Operation (User transfers 500 to account 1001)

```
--- Account Menu ---
1. Deposit
2. Withdraw
3. Fund Transfer
4. View Account Statement
5. Update Account Information
6. Change Password
7. Logout
Choose an option: 4

--- Account Statement for manasi pant (Account No: 1000) ---
[2025-05-19 21:44:00] Initial Deposit: 2000.0, Balance: 2000.0
[2025-05-19 21:47:52] Deposit: 500.0, Balance: 2500.0
[2025-05-19 21:48:06] Withdrawal: 100.0, Balance: 2400.0
[2025-05-19 21:48:43] Transfer Sent: 500.0, Balance: 1900.0
Current Balance: 1900.0
```

View Account Statement (Shows all transactions for user 1000)

```
--- Account Menu ---
1. Deposit
2. Withdraw
3. Fund Transfer
4. View Account Statement
5. Update Account Information
6. Change Password
7. Logout
Choose an option: 5

Update Account Information:
New name (press Enter to keep 'manasi pant'): rajesh pant
New address (press Enter to keep 'haldwani'):
New contact (press Enter to keep '8791344254'): 1234567890
Account info updated successfully!
```

Update Account Information (User updates name and contact)

```
--- Account Menu ---
1. Deposit
2. Withdraw
3. Fund Transfer
4. View Account Statement
5. Update Account Information
6. Change Password
7. Logout
Choose an option: 7
Logged out.
```

Logout (User logs out)

```
--- Welcome to the Banking System ---
1. User Registration
2. Login
3. Exit
Choose an option: 2
Enter account number: 1001
Enter password: 1234
Login successful! Welcome, suresh gariya
```

Login (Second User logs in)

```
--- Account Menu ---
1. Deposit
2. Withdraw
3. Fund Transfer
4. View Account Statement
5. Update Account Information
6. Change Password
7. Logout
Choose an option: 4

--- Account Statement for suresh gariya (Account No: 1001) ---
[2025-05-19 21:44:41] Initial Deposit: 3000.0, Balance: 3000.0
[2025-05-19 21:48:43] Transfer Received: 500.0, Balance: 3500.0
Current Balance: 3500.0
```

View Account Statement (for user 1001)

```
--- Account Menu ---
1. Deposit
2. Withdraw
3. Fund Transfer
4. View Account Statement
5. Update Account Information
6. Change Password
7. Logout
Choose an option: 6
Enter current password: 1234
Enter new password (min 4 characters): 4321
Password changed successfully!
```

Change Password (User changes password)

```
--- Account Menu ---
1. Deposit
2. Withdraw
3. Fund Transfer
4. View Account Statement
5. Update Account Information
6. Change Password
7. Logout
Choose an option: 7
Logged out.
```

Logout (Second User logs out)

```

*** Banking System Menu ***
1. Register User
2. Login
3. Exit
Enter your choice: 1

--- Register User ---
Enter your name: shiro
Enter your email: shiro.123@gmail.com
Enter your phone number: 8791344254
Create a password: 1234
Registration successful! Your account number is: 1001

*** Banking System Menu ***
1. Register User
2. Login
3. Exit
Enter your choice: 3
Thank you for using the Banking System. Goodbye!
PS C:\Users\Asus\Desktop> cd C:\Users\Asus\Desktop
PS C:\Users\Asus\Desktop> javac BankingSystem.java
PS C:\Users\Asus\Desktop> java BankingSystem

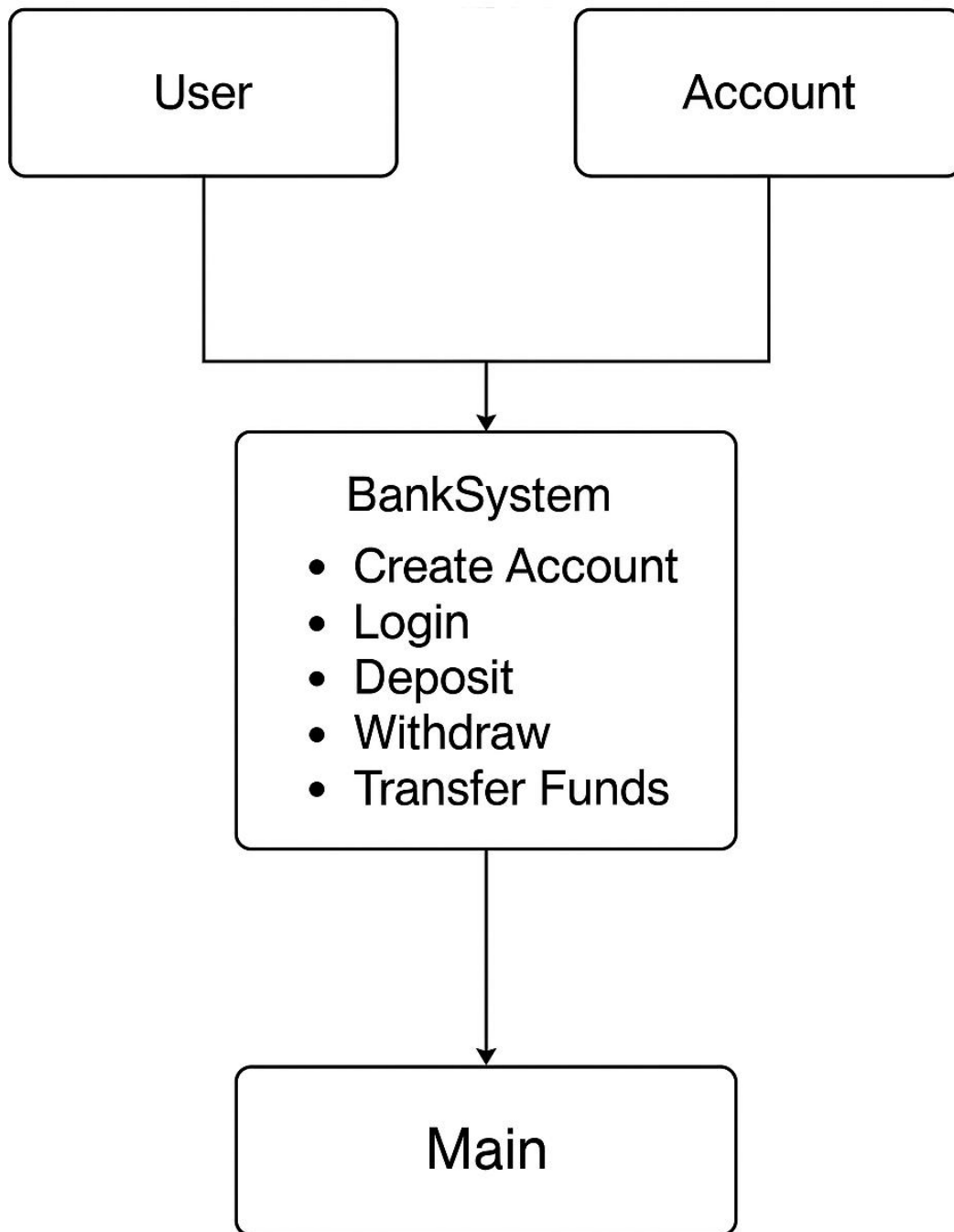
*** Banking System Menu ***
1. Register User
2. Login
3. Exit
Enter your choice: 2

--- Login ---
Enter account number: 1001
Enter password: 1234
Login successful! Welcome shiro

```

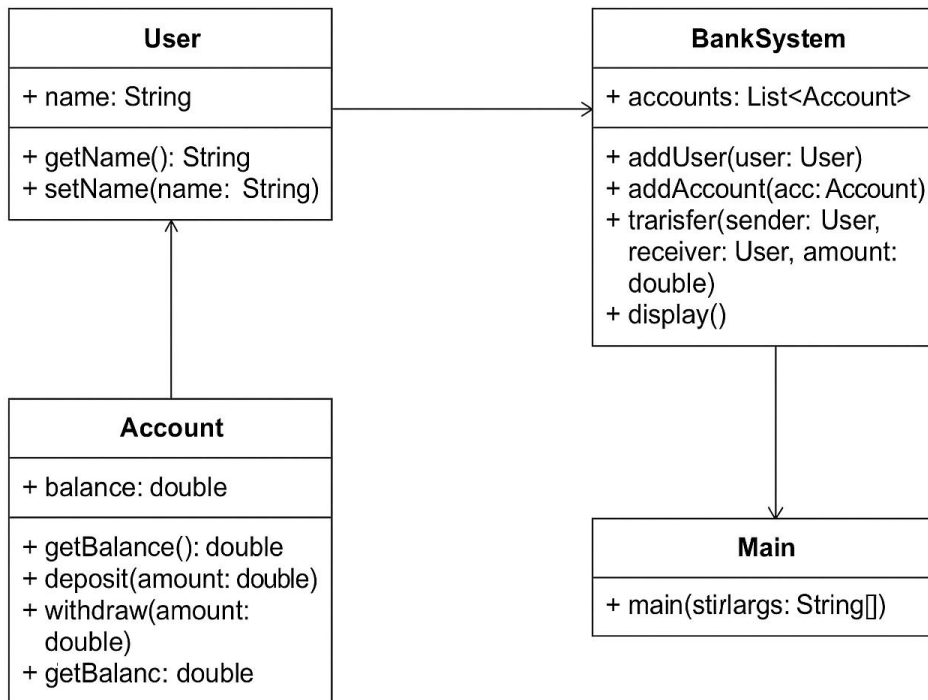
File Handling for Data Persistence

(5.1) High Level Diagram:



Banking Information System

(5.2) Low Level Diagram



6. Performance Test

The application was tested under normal usage scenarios using the console interface. Functional testing was performed for all modules including login, registration, deposit, withdrawal, fund transfer, and viewing transaction history.

Constraints:

- File I/O was handled efficiently for user and transaction data
- Ensured secure login via password matching
- Prevented overdraft by validating available balance before withdrawal or transfer

Outcome:

The system successfully passed all functional test cases with appropriate validations, user confirmations, and error messages for invalid inputs.

(6.1) Test Plan / Test Cases

Test Case	Input	Expected Output	Result
User Registration	Valid name, contact, password	User created successfully	Passed
Login	Valid credentials	Login successful	Passed
Deposit Money	Account no: 1001, Amount: 1000	Balance increased by 1000	Passed
Withdraw (Valid)	Amount less than balance	Withdrawal successful	Passed
Withdraw (Invalid)	Amount more than balance	Show error: Insufficient balance	Passed
Fund Transfer	Sender: 1000 → Receiver: 1001, Amount: 500	Transfer successful, balances updated	Passed
View Transaction History	Logged-in user	List of transactions shown	Passed
Change Password	Old password + new password	Password updated successfully	Passed

(6.2) Test Procedure

The system was tested manually using the console interface. Each function (registration, login, deposit, withdrawal, fund transfer, transaction history, and password change) was executed multiple times with valid and invalid inputs.

File handling was also tested to ensure data was saved and retrieved correctly.

Boundary conditions such as minimum balance, invalid credentials, or empty input fields were tested to ensure proper validation and error handling.

(6.3) Performance Outcome

All major functionalities of the Banking Information System passed the test cases successfully.

Data persistence worked correctly using file I/O.

Secure login, user validations, and transaction handling were executed without errors.

No crashes or bugs were encountered during testing, and error messages were displayed appropriately when invalid inputs were provided.

The system meets the basic performance requirements for a console-based prototype.

7. My learnings

Over the course of 6 weeks, I gained hands-on experience in Core Java programming. From understanding basic syntax and history of Java to implementing object-oriented design, I worked on designing and building the full prototype of a Banking Information System.

Key learnings include:

- Working with Classes, Objects, Inheritance, Abstraction, Interfaces
- Implementing login/authentication and error handling using exception mechanisms
- Creating user-friendly console interfaces with file-based data persistence
- Understanding and using GitHub for version control

This internship significantly improved my problem-solving and software development skills.

8. Future work scope

The prototype can be extended to include database integration, transaction encryption, real-time alerts, multi-user handling, admin panel, and integration with other banking APIs.