Industrial Internship Report on

"Bank Management System using Python"

Prepared by

Mahesh Dipak Kekan

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 6 weeks' time.

My project was (Tell about your Project)

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.

TABLE OF CONTENTS

1 Preface

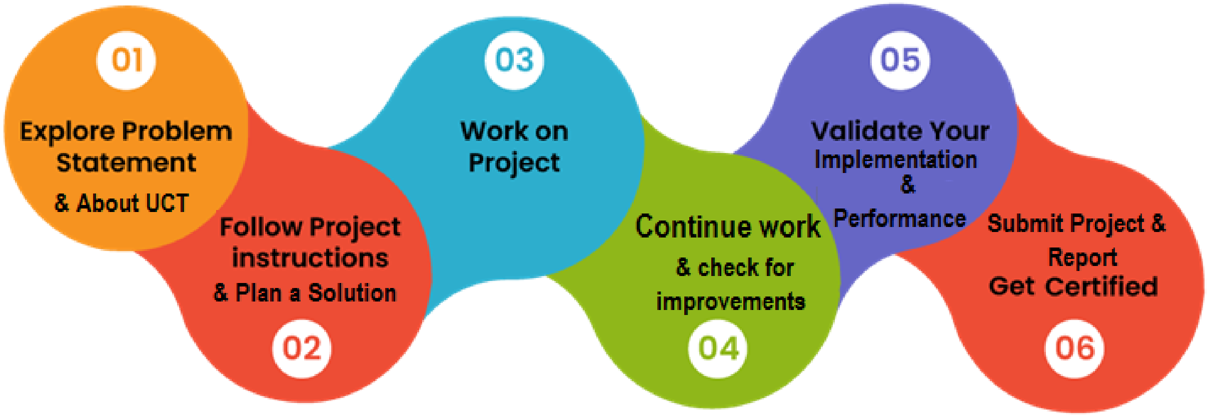
Summary of the whole 6 weeks' work.

About need of relevant Internship in career development.

Brief about Your project/problem statement.

Opportunity given by USC/UCT.

How Program was planned



Your Learnings and overall experience.

Thank to all (with names), who have helped you directly or indirectly.

Your message to your juniors and peers.

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

Uniconverge Technologies

IIOT Products

We offer product ranging from Remote IOs, Wireless IOs, LoRaWAN Sensor Nodes/ Gateways, Signal converter and IoT gateways

IIOT Solutions

We offer solutions like OEE, Predictive Maintenance, LoRaWAN based Remote Monitoring, IoT Platform, Business Intelligence…

OEM Services

We offer solutions ranging from product design to final production we handle everything for you..

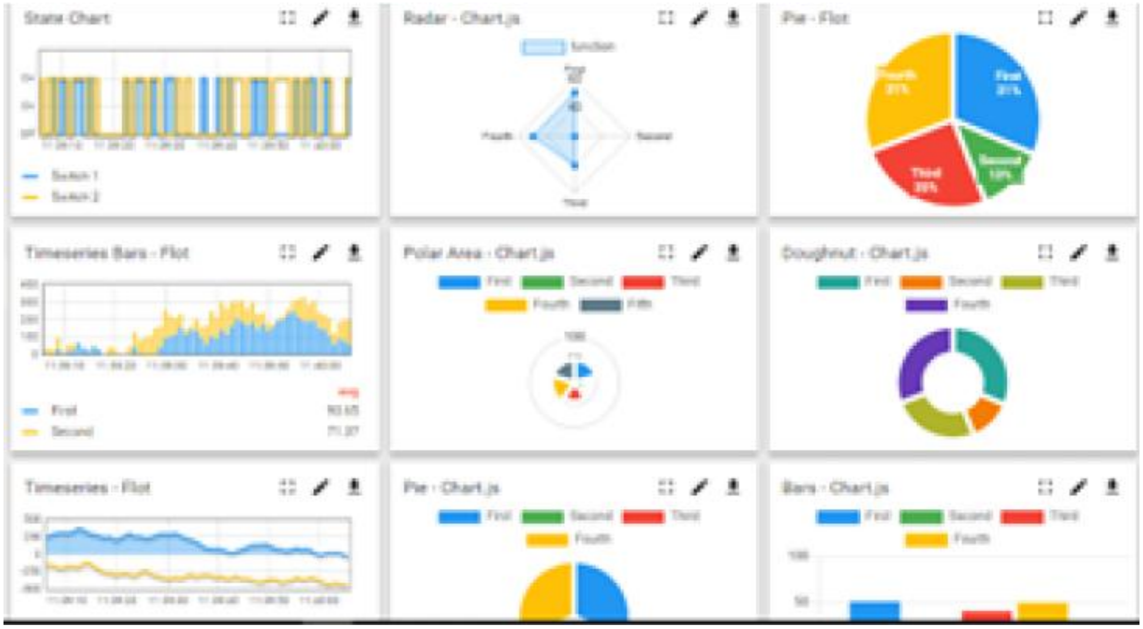
i. UCT IoT Platform (uct Insight)

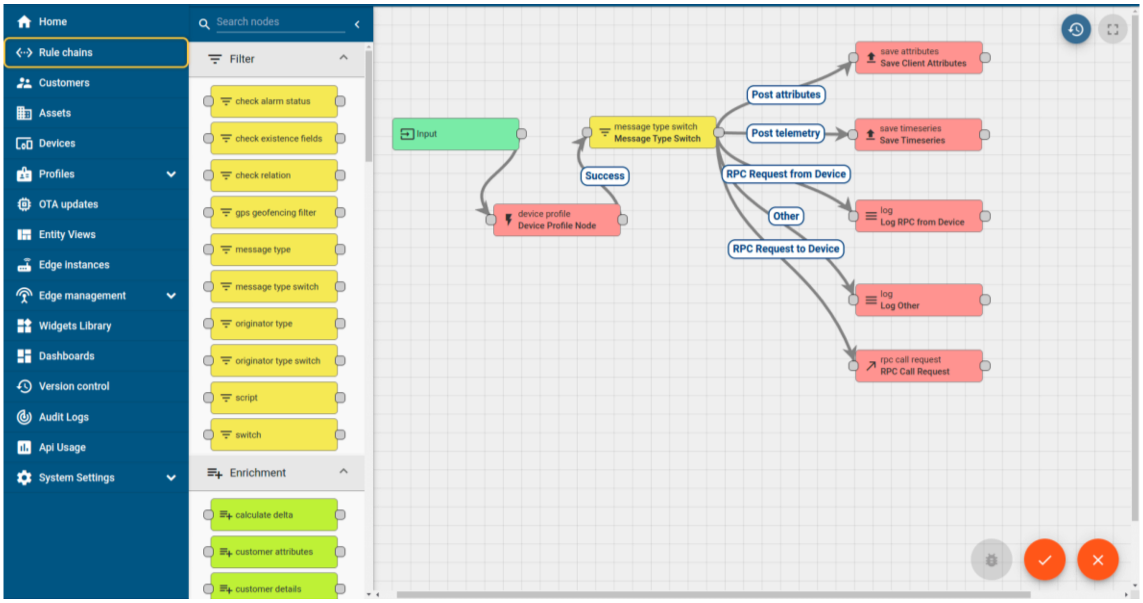
UCT Insight is an IOT platform designed for quick deployment of IOT applications on the same time providing valuable “insight” for your process/business. It has been built in Java for backend and ReactJS for Front end. It has support for MySQL and various NoSql Databases.

- 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





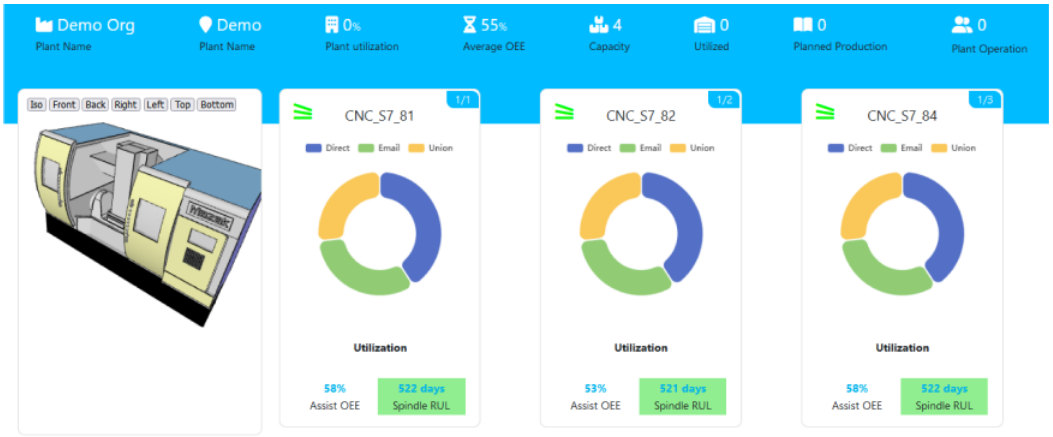
ii. Smart Factory Platform (FACTORY WATCH)

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 unleased 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.



Demo Org
Plant Name

Demo
Plant Name

0%
Plant utilization

55%
Average OEE

4
Capacity

0
Utilized

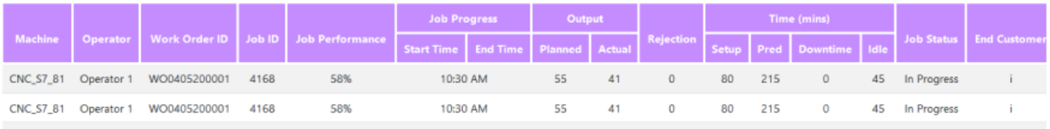
0
Planned Production

0
Plant Operation

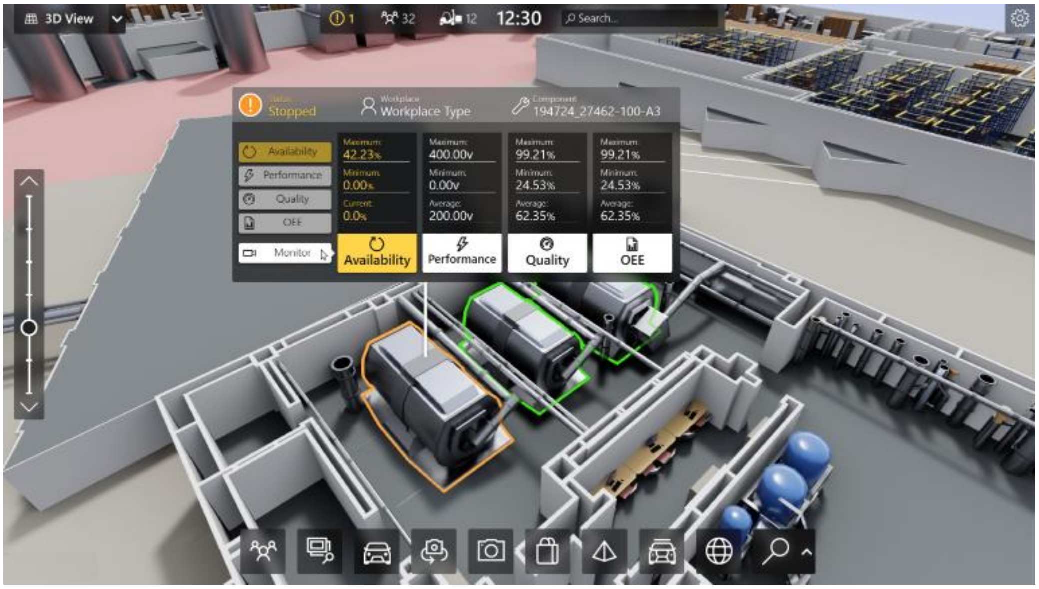
CNC_S7_81 1/1
Direct Email Union
Utilization
58% Assist OEE
522 days Spindle RUL

CNC_S7_82 1/2
Direct Email Union
Utilization
53% Assist OEE
521 days Spindle RUL

CNC_S7_84 1/3
Direct Email Union
Utilization
58% Assist OEE
522 days Spindle RUL



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



Status: Stopped
Workplace: Workplace Type
Component: 194724_27462-100-A3

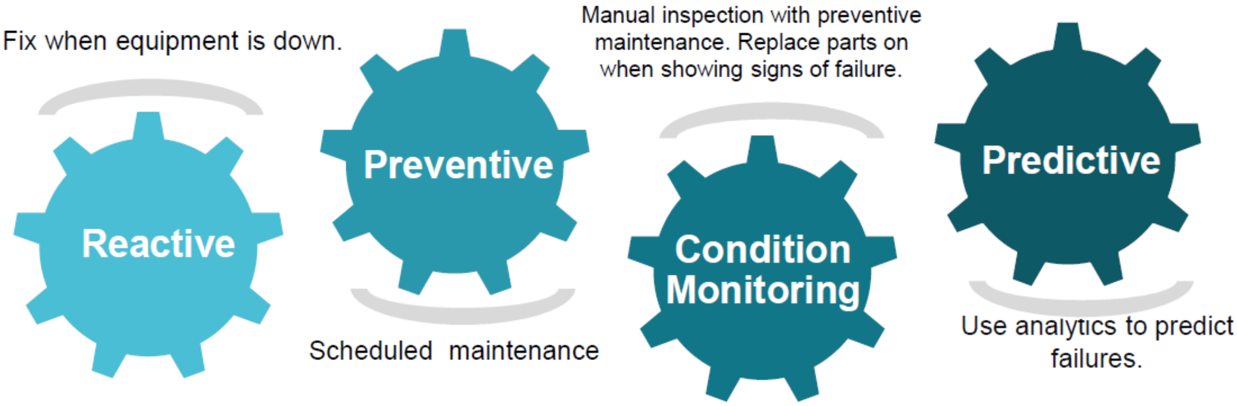
Availability	Performance	Quality	OEE
Maximum: 42.23%	Maximum: 400.00v	Maximum: 99.21%	Maximum: 99.21%
Minimum: 0.00%	Minimum: 0.00v	Minimum: 24.53%	Minimum: 24.53%
Current: 0.0%	Average: 200.00v	Average: 62.35%	Average: 62.35%

iii. LoRaWAN 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.

iv. Predictive Maintenance

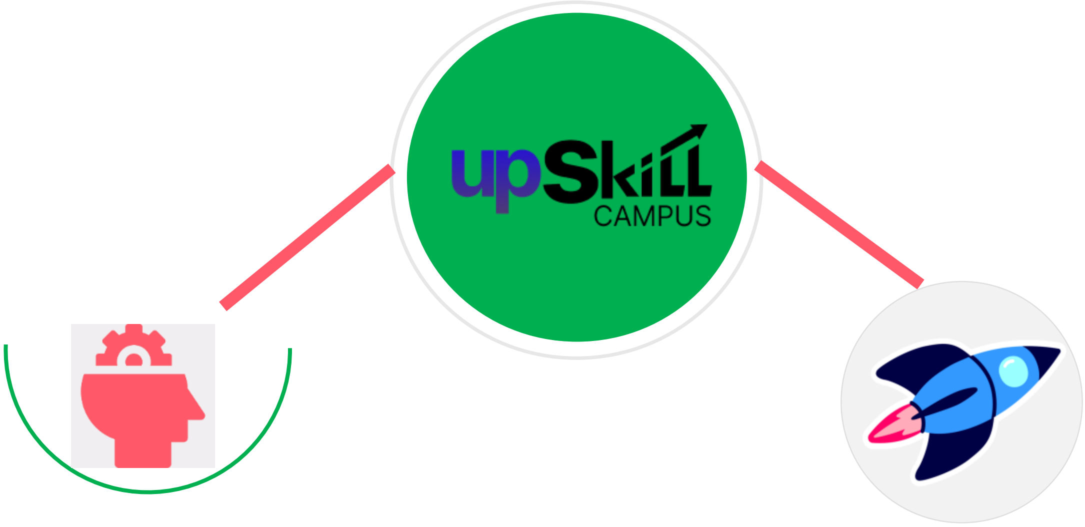
UCT is providing Industrial Machine health monitoring and Predictive maintenance solution leveraging Embedded system, Industrial IoT and Machine Learning Technologies by finding Remaining useful life time of various Machines used in production process.


Fix when equipment is down.
Reactive
Preventive
Scheduled maintenance
Manual inspection with preventive maintenance. Replace parts on when showing signs of failure.
Condition Monitoring
Predictive
Use analytics to predict failures.

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.

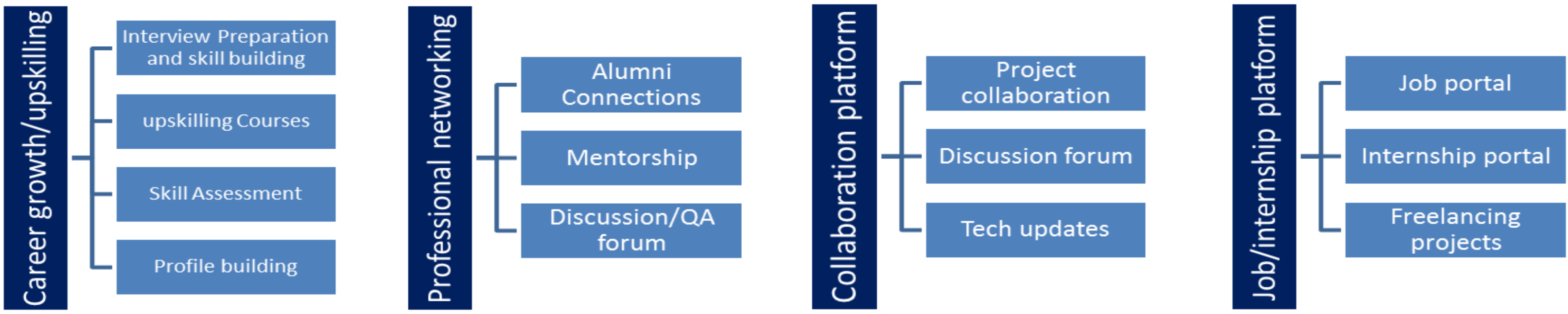
USC is a career development platform that delivers **personalized executive coaching** in a more affordable, scalable and measurable way.



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/



Career growth/upskilling
- Interview Preparation and skill building
- upskilling Courses
- Skill Assessment
- Profile building

Professional networking
- Alumni Connections
- Mentorship
- Discussion/QA forum

Collaboration platform
- Project collaboration
- Discussion forum
- Tech updates

Job/internship platform
- Job portal
- Internship portal
- Freelancing projects

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 Reference

[1] Python Software Foundation. *Python 3.12 Documentation*. Retrieved from: https://docs.python.org/3/

[2] GeeksforGeeks. *Bank Management System Project in Python*. Retrieved from: https://www.geeksforgeeks.org/python-bank-management-system-project/

[3] UniConverge Technologies Pvt. Ltd. *Industrial Internship Problem Statements and Guidance Documents*, 2025.

2.6 Glossary

Terms	Acronym
OOP	Object-Oriented Programming — a programming paradigm that organizes code into objects containing data and behavior.
JSON	JavaScript Object Notation — a lightweight data-interchange format used for storing

	structured data.
Exception Handling	A programming mechanism to handle runtime errors gracefully without terminating the program.
File I/O	File Input/Output — reading and writing data to external files for storage and retrieval.
Account Class	A Python class that defines attributes and methods related to individual bank accounts (deposit, withdraw, etc.).
Bank Class	A Python class that manages all accounts collectively and handles inter-account transactions.
Transaction History	Record of all deposits, withdrawals, and transfers for an account.
Persistence	The ability of data to remain stored and accessible even after the program is closed (implemented here using JSON).
InsufficientFundsError	A custom-defined exception triggered when withdrawal or transfer exceeds available balance.
AccountNotFoundError	A custom-defined exception raised when the specified account number does not exist in the system.
Data Structure	A way to organize and store data efficiently; here a dictionary {account_number: AccountObject} is used.

3 Problem Statement

In the assigned problem statement

In the modern financial world, banks must manage millions of customer accounts efficiently while maintaining security and reliability. Manual record-keeping often leads to delays, data redundancy, and errors. Therefore, the challenge was to design and implement a **Bank Management System** that can perform essential banking operations such as account creation, deposits, withdrawals, fund transfers, and transaction tracking using Python programming, while ensuring data security and persistence.

4 Existing and Proposed solution

Existing Solutions:
Many small-scale banking applications or spreadsheet-based systems lack automation, transaction history, or proper data validation. They often don't support exception handling or persistent data storage.

Proposed Solution:
The proposed **Bank Management System** automates all essential banking activities using Object-Oriented Programming (OOP) principles in Python.
Key features include:

- Creation of new accounts with unique account numbers
- Deposit, withdrawal, and transfer functionalities
- Real-time transaction logging
- Persistent data storage using JSON
- Custom exception handling for invalid operations

This solution offers a **realistic and scalable foundation** for future integration with databases or APIs for real-world banking systems.

4.1 Code submission (Github link):

upskill-campus/main.py at main · kekanmahesh18-oss/upskill-campus

4.2 Report submission (Github link):

5 Proposed Design/ Model

- **High Level Design:**

The system is divided into two main classes:

1. **Account Class:** Handles operations related to a single user (deposit, withdraw, transaction history).
2. **Bank Class:** Manages all accounts collectively and coordinates transactions, fund transfers, and data persistence.

- **Low Level Design:**
- **Data Structure Used:** Python Dictionary ({account_number: AccountObject})
- **Persistence:** Data saved in bank_data.json using the json module
- **Exception Handling:** Custom exceptions such as InsufficientFundsError and AccountNotFoundError ensure smooth program flow
- **File Handling:** The system loads existing account data when started and saves updates when exiting

6 Performance Test

Constraints Identified:

- Data persistence must remain consistent after multiple operations
- System should handle invalid inputs without crashing
- Transfers should not allow overdrafts

6.1 Test Plan/ Test Cases

Test Case	**Input**	**Expected Output**
Create Account	Valid details	Account created successfully
Deposit	Positive amount	Balance updated
Withdraw	More than balance	Raises InsufficientFundsError
Transfer	Valid accounts and amount	Transfer successful
Invalid Account	Wrong account number	Raises AccountNotFoundError

6.2 Performance Outcome

All test cases executed successfully. The program handled exceptions gracefully, provided correct outputs, and stored data persistently across sessions.

7 My learnings

- Gained strong practical understanding of **Object-Oriented Programming** in Python.
- Learned how to design modular and reusable classes.
- Implemented **file handling and JSON-based storage** for data persistence.
- Improved skills in **debugging, exception handling, and user interface design**.
- Understood how real-world banking systems structure their core logic.

8 Future work scope

- Integrate with a **SQLite database** for more reliable data management.
- Add **login authentication and user roles** (Admin / Customer).
- Include **graphical user interface (GUI)** using Tkinter or Flask web version.
- Enhance the system with **interest calculation and monthly statements**.
- Deploy on the web to simulate a real banking portal.