# ‘REQUIREMENTS ELICITATION

**Appendicle of the Interview Record:**

**Interviewer:** Systems Analyst  
**Interviewee:** Bank Manager (Client)  
**Date:** September 18 2025

**Q: How do you register a customer?**  
**A:** We don’t register a customer, for someone to be a customer he/she must have an account. So we open an account for a person, their account number can be associated with their captured details.

**Q:** **What are the core services customers need?**  
**A:** Open accounts (savings, investment, cheque), deposit, withdraw (except savings account), view statements, and interest needs to be paid monthly.

**Q:** **Any business rules for account opening?**

**A:** The currency doesn’t matter. Investment needs 500 starting deposit. Cheque only for employed customers, we must capture company name and address. Savings should not allow withdrawals.

**Q: Can one person have more than one account?**  
**A:** Yes. A customer can have a Savings account, a Cheque account, and an Investment account. All types of accounts but not more than one of the same accounts.

**Q:** **What interest rates should we apply?**

**A:** Investment: 5% monthly. Savings: 0.025% monthly for individual customers and 0.075% for company customers . Cheque: 0%.

**Q: Should it give an error after many attempts for login?**

**A**: What matters most is the OOP principles. Don’t overload with a lot of functionality.

**Q: For a Cheque account, what information is mandatory?**  
**A:** We need proof of employment. So, we must record where the customer works the company name and its physical address.

**Q: How should the system handle interest?**  
**A:** It must be automatic. At the end of each month, the system should add the interest to the respective accounts without any teller having to trigger it. The rates are fixed: 5% for Investment, a small 0.05% for Savings, and nothing for Cheque accounts.

**Q: Are there any security concerns?**  
**A:** Yes, data integrity is paramount. The system must be reliable and ensure that customer balances are always accurate and never corrupted.

**Q: Should it require human approvement?**

**A:** It should be fully automated

NON-FUNCTIONAL REQUIREMENTS

1. **SECURITY**

The system shall ensure data integrity. All financial transactions (deposits, withdrawals, interest calculations) must be accurate and permanent once committed. Customer data (e.g., address, employment info) shall be stored securely.

1. **RELIABILITY**

The system shall be available during all bank operating hours.

1. **USABILITY**

The user interface shall be intuitive and easy to use for bank employees, requiring minimal training.

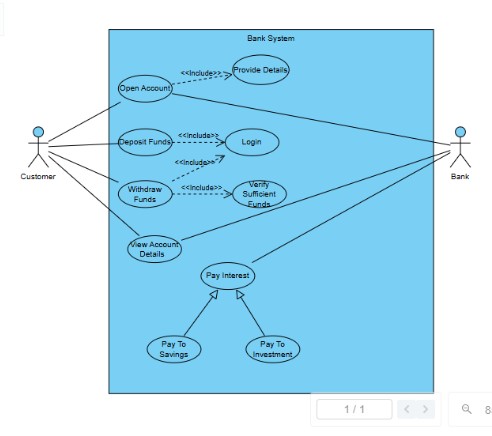
1. **PERFORMANCE**

Response times for key operations (e.g., account balance inquiry, deposit processing) shall be under 5 seconds. It will also allow for multiple users to access at one time.

1. **MAINTAINABILITY**

The system shall be designed using modular Object-Oriented Principles to allow for easy future expansion (e.g., changing interest rates) with minimal changes to existing code.

## STRUCTURAL UML MODELLING

A screenshot of a computer

AI-generated content may be incorrect.