

CSE

18/09/2025

Year 2 Semester 1

Object Oriented Analysis and Design with Java

Cse24

Cse24-153@thuto.bac.ac.bw

Theo Katso Mompati

Cse23-153

Cse23-153

Cse23-153

A screenshot of a computer

AI-generated content may be incorrect.

17/09/2025



**REQUIREMENTS**

What is requirements Engineering?

It’s the process of finding, analyzing and documenting the needs of software.

**Importance of requirements engineering**

**Customer Satisfaction:** The involvement of stakeholders during the development of software ensures that stakeholders’ needs are satisfied.

**Saves time:** Errors are identified and fixed at early stages of development.

**Ensures quality work:** As the project is modelized before development, developers know exactly what to do and reduces confusion.

**Continuous management:** Engagement of stakeholders during the project life cycle ensures project objectives are achieved.

**Stages of Requirement Engineering**

**Requirement elicitation:** This is gathering information from stakeholders on how they want the banking system to be like, as well as the functionalities, as per conducted in an interview yesterday, I was able to identify the customer’s needs.

**Requirement Elaboration:** it’s using gathered requirements from elicitation to expand, refine to get a cleaner detailed structure of how the banking system will be.

**Requirement Validation**: It’s the regularly checking of stakeholders to confirm if the requirements met their desired expectations of the bank system.

**Requirement Management:** Checking tracking and modification of the requirements throughout the lifecycle development.

**FUNCTIONAL REQUIREMENTS**

Describing what the bank system should do, defining its behaviors, according to the interview that was conducted yesterday, our stakeholder mentioned that he wants following requirements in his bank system.

**Why are we doing this banking system?**

Our stakeholder mentioned that he identified a need to make a banking system was to reduce costs of traveling between customers since customers were to perform transactions only at ATMs, he mentioned that instead of waiting in long banking ques to perform transactions, our banking system would be fast and reliable enough to perform transactions and lastly to keep records of transactions in the banking system.

The following requirements are needed in the system

* The user must login to the system
* The user must be able to register for an account
* The user should be able to open Savings, Cheque, or Investment accounts with an assistant of the bank clerk
* The user should be able to deposit money into any account
* The user should be able to withdraw money from accounts depending on which account it is
* The system must keep track of all transactions made by the user
* The user should be able to view their transaction history
* The system should calculate monthly interest for Savings and Investment accounts
* The user should be able to check their account balance
* The user should be able to close their account
* The admin should be able to verify employment details for Cheque accounts
* The user should be able to contact the bank clerk if there are any problems regarding their accounts.

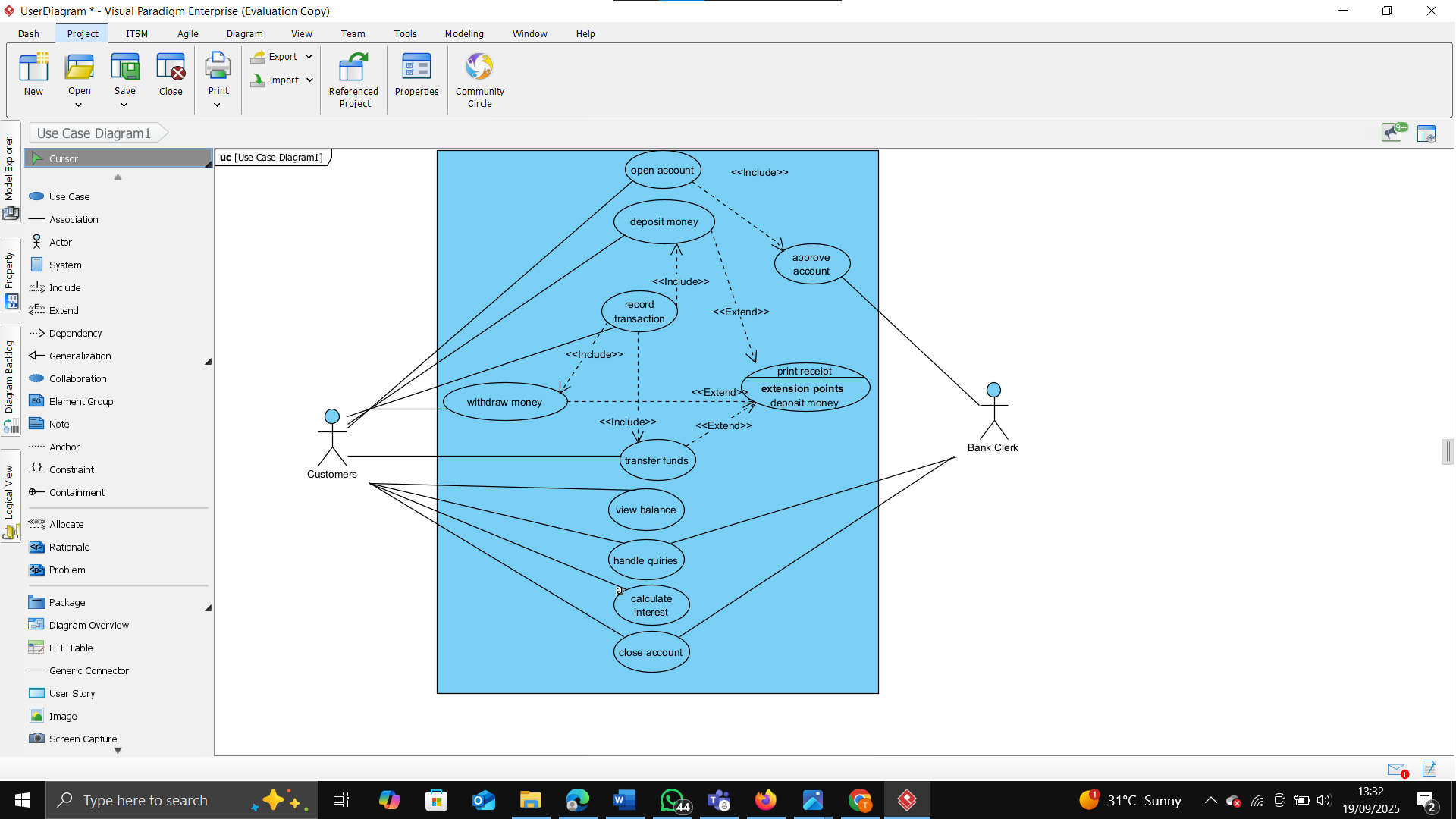
**NON-FUNCTIONAL REQUIREMENTS**

Describes what the banking system must have.

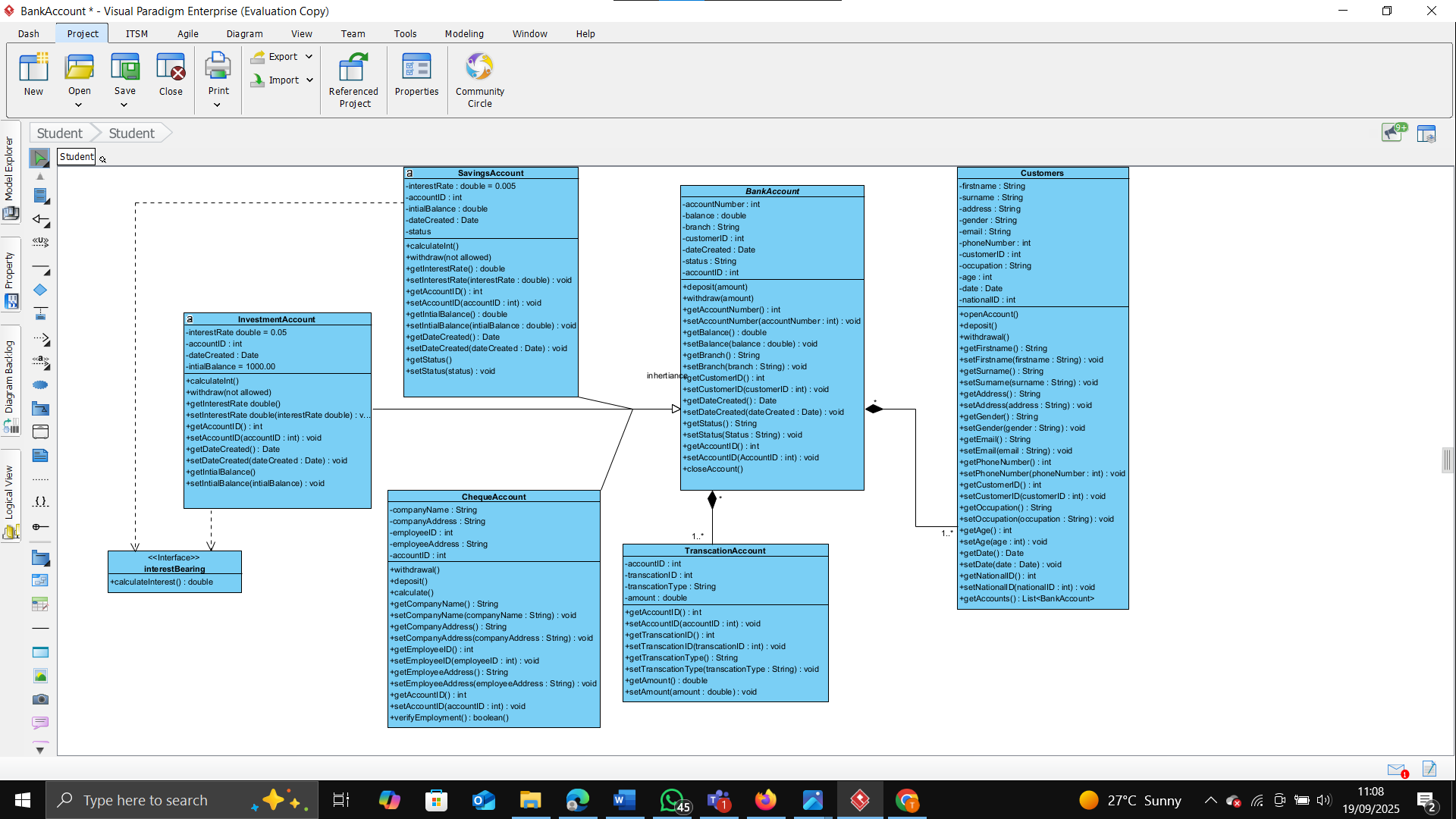
The following are non-functional requirements of the banking system according to our client.

* The system must process transactions quickly (performance)
* The system must provide secure login and data encryption (security)
* The system must be available 24/7 with high uptime (availability)
* The system must be reliable and prevent data loss (reliability)
* The system must provide a simple and user-friendly interface (usability)
* The system must support more customers as it grows (scalability)
* The system must be easy to maintain and update (maintainability)
* The system must comply with banking regulations (compliance)

**Use case diagram**



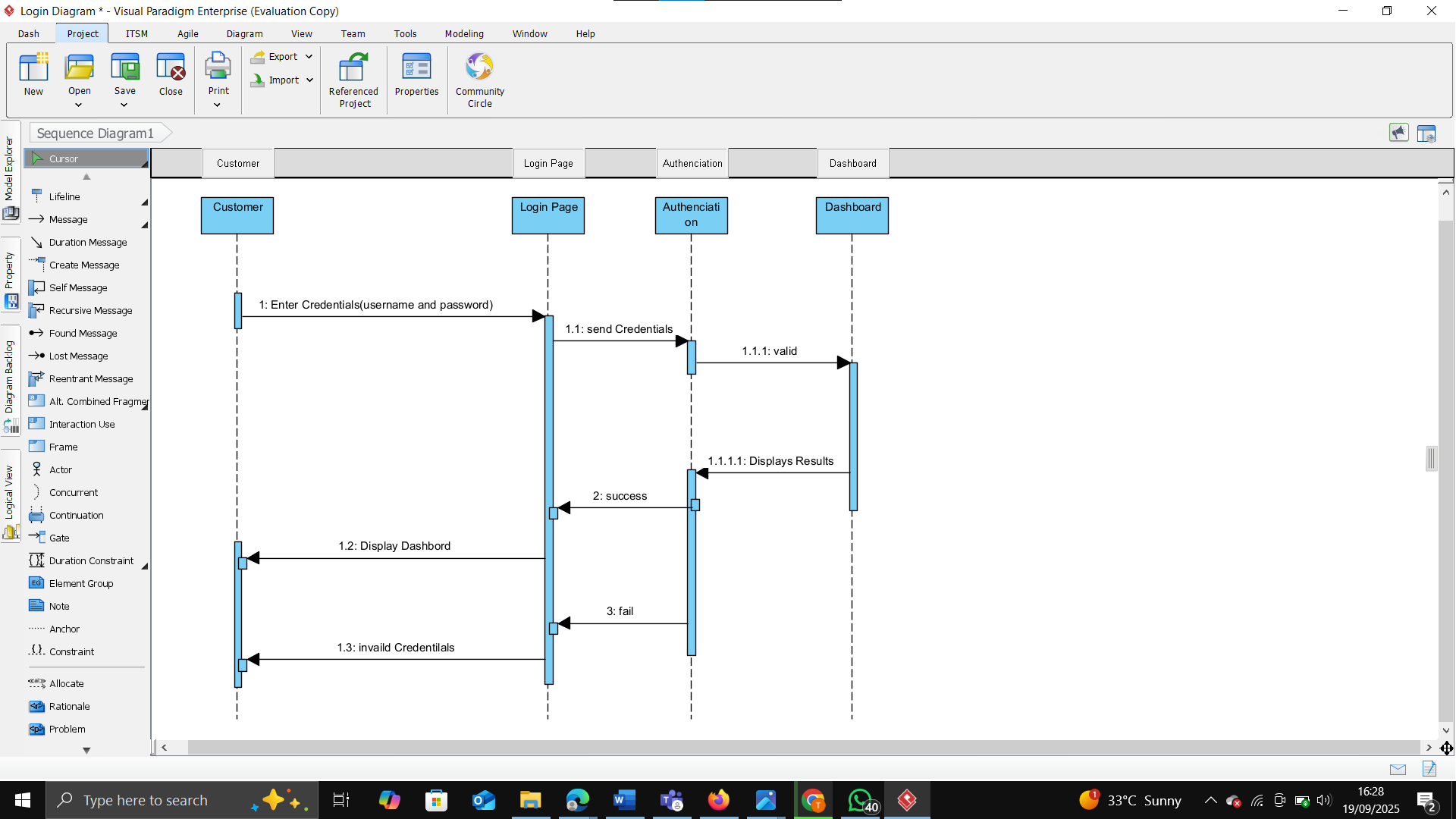
**Class Diagram**

****

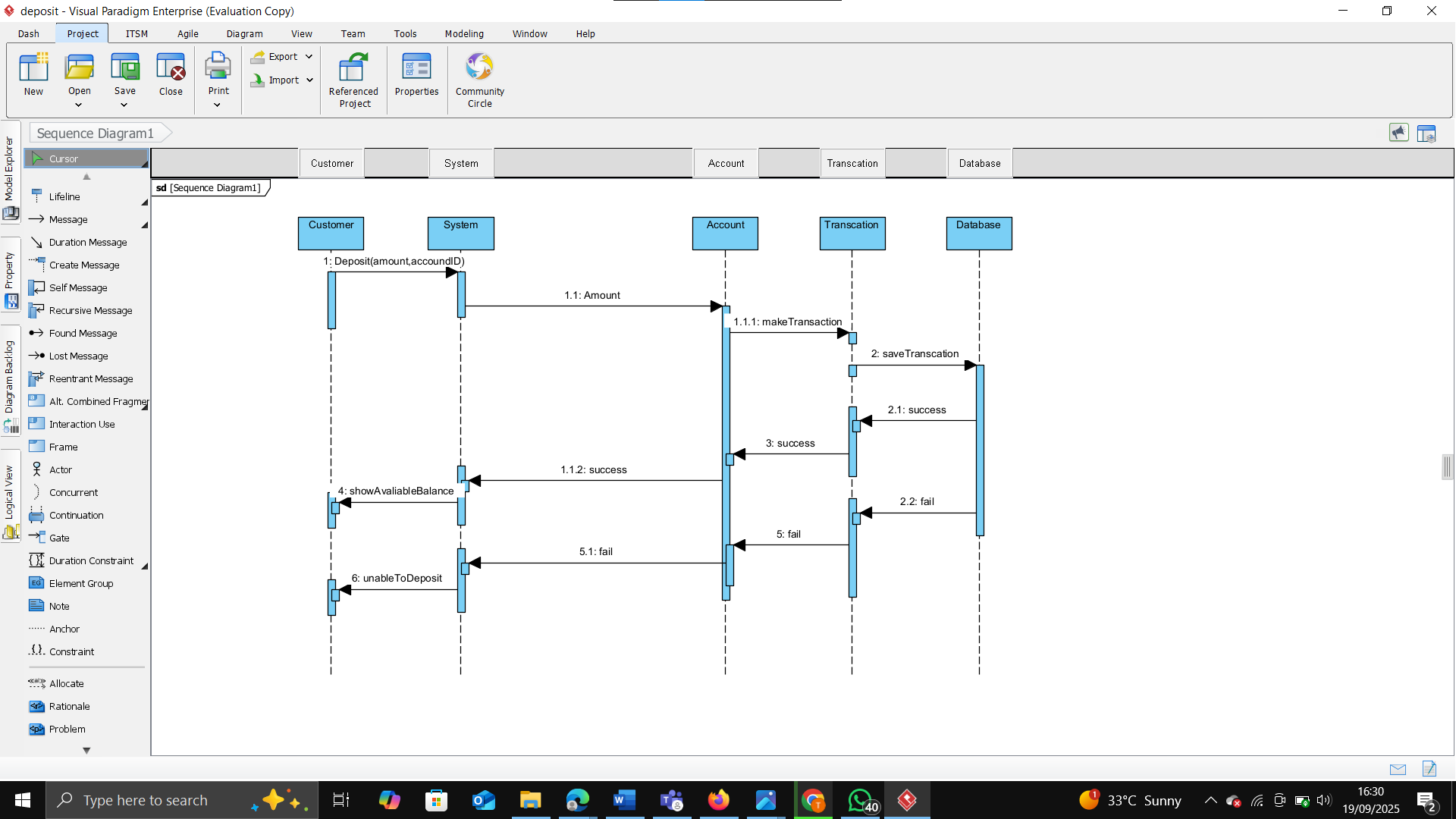
**Behavioral UML Modelling**

**Sequential diagrams**

**Login**

****

**Deposit**



**State Diagram**

Customer state Diagram

