1. Passport Management System

Aim: To design a passport Management System using UML.

Algorithm:

* Applicant submit application and documents
* Passport office forwards details to central computer
* Central computer verify identity and record
* If valid → Approve application
* If not valid → Reject application
* If approved → Issue passport

Class Diagram

Result:

The system successfully verifies the applicant detail

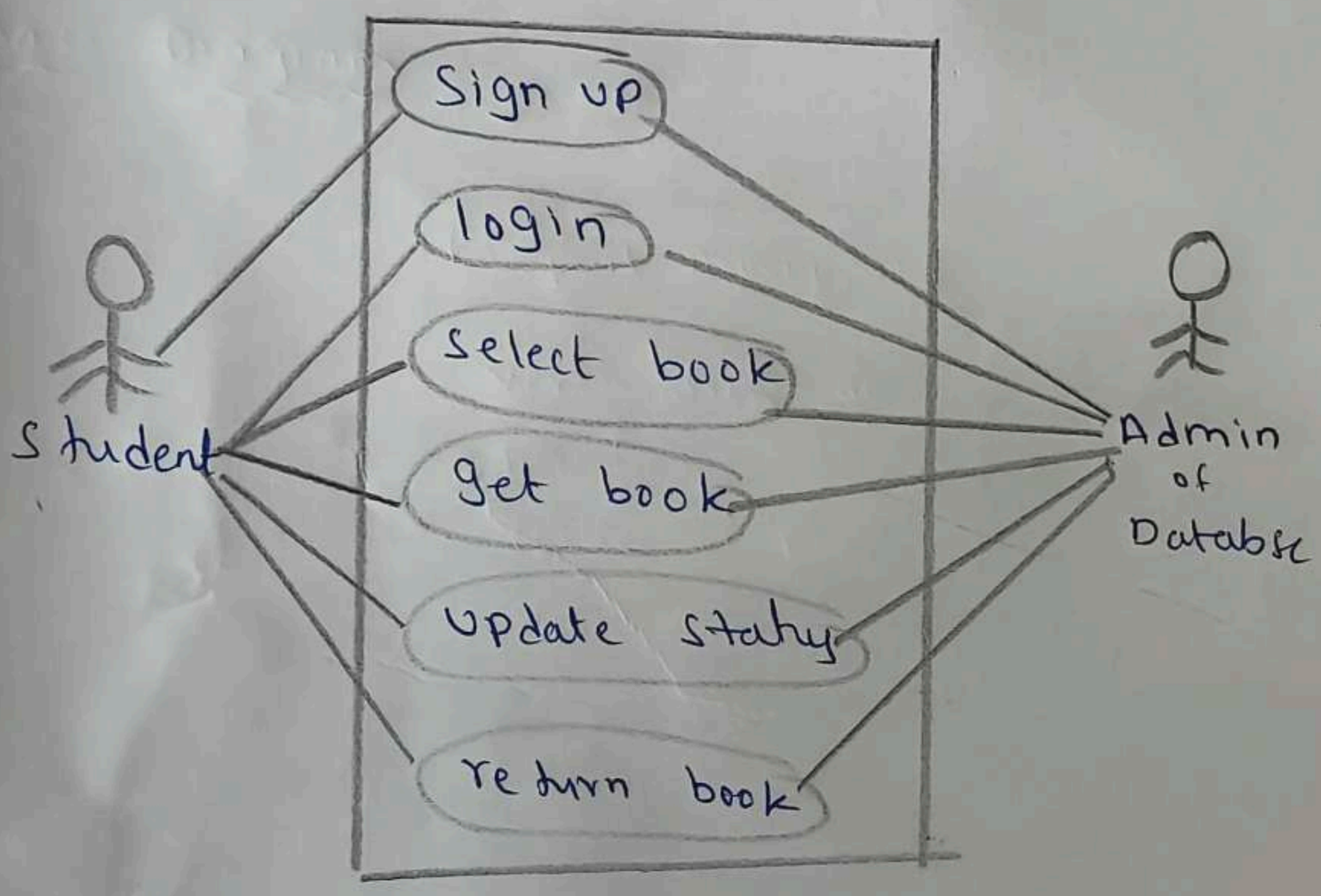
2. Book Bank Process

Aim: To Design a Book Bank Process using UML that verify student details

Algorithm:

* student submit detail to Administrator
* Admin forward detail to Computer
* It verify student record
* If valid- send Approval
* If not valid- Reject
* update book availability status.

Use-case diagram



Result:

Thus the Book Bank process is successfully designed using UML that verifies student detail through computer before issuing book.

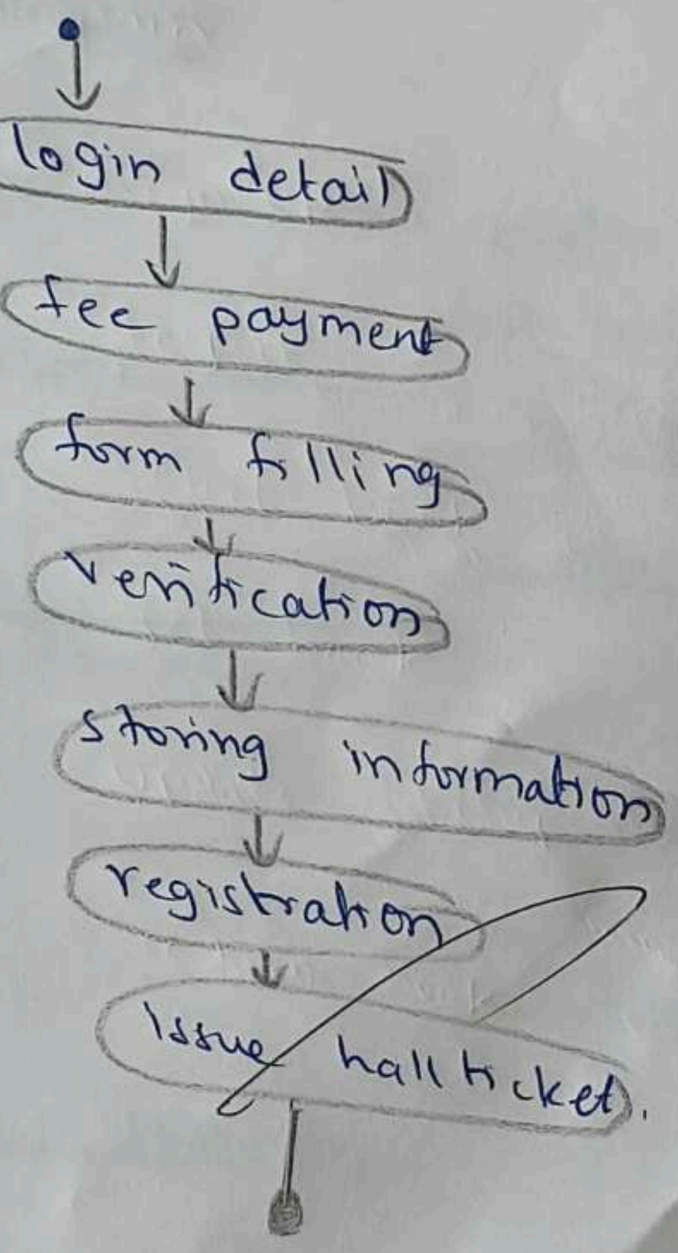
3. Exam Registration Portal

Aim: Develop an Exam Registration portal wing UML that verify candidate detail to issue hall ticket.

Algorithm:

* Input the candidate detail
* Verify details (eligibility)
* If approved: Generate hall ticket
* If rejected: Notify the reason

Activity diagram



Result:

Approved candidate receive hall ticket.

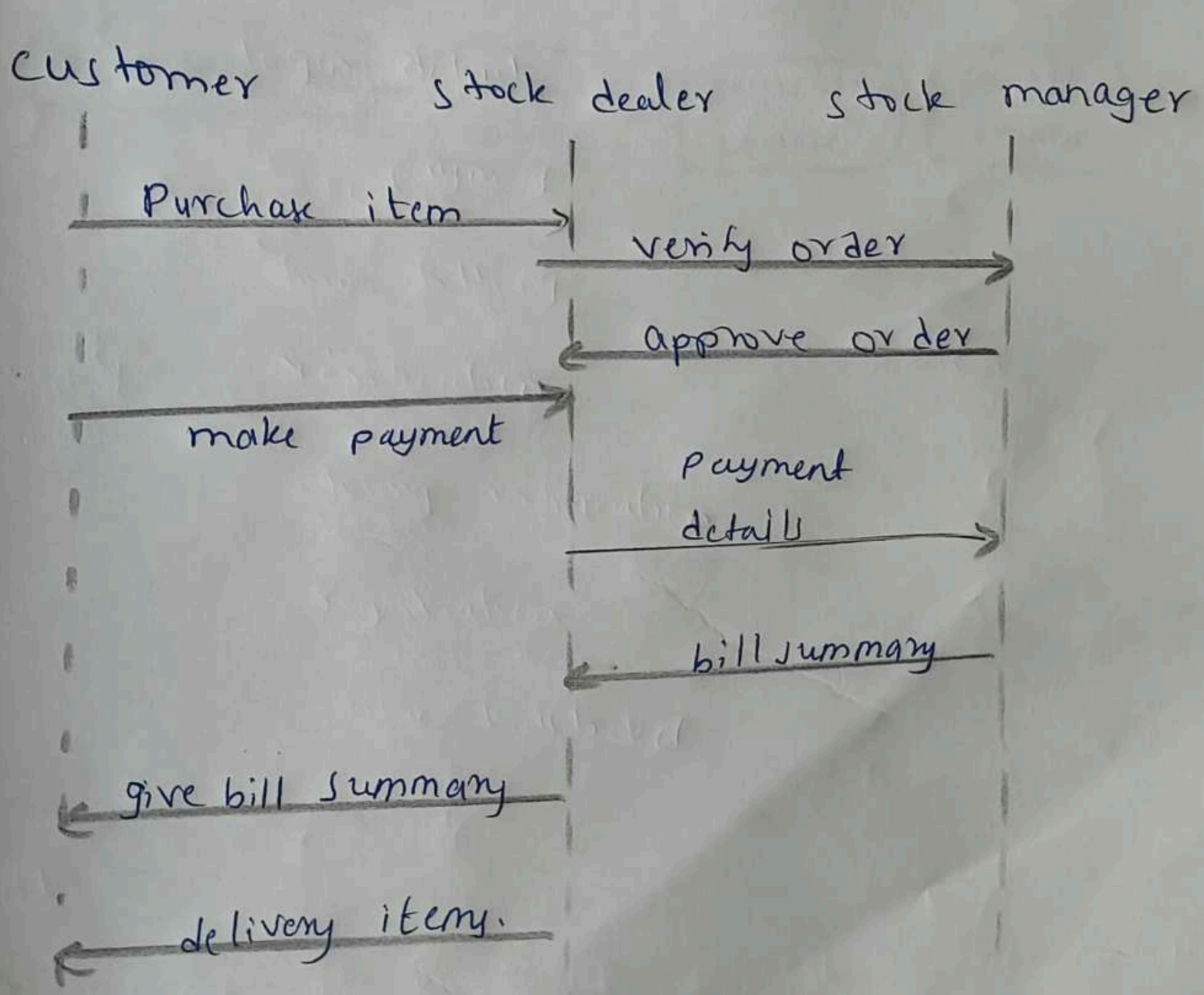
4. Stock Maintaince System

Aim: Develop a system for managing stock order and update through stock system.

Algorithm:

* Input order details
* verify stock availability
* If available
  - process order
  - update stock
* Deliver items.

Sequence diagram



Result:

Order processed and stock updated, customer receives item.

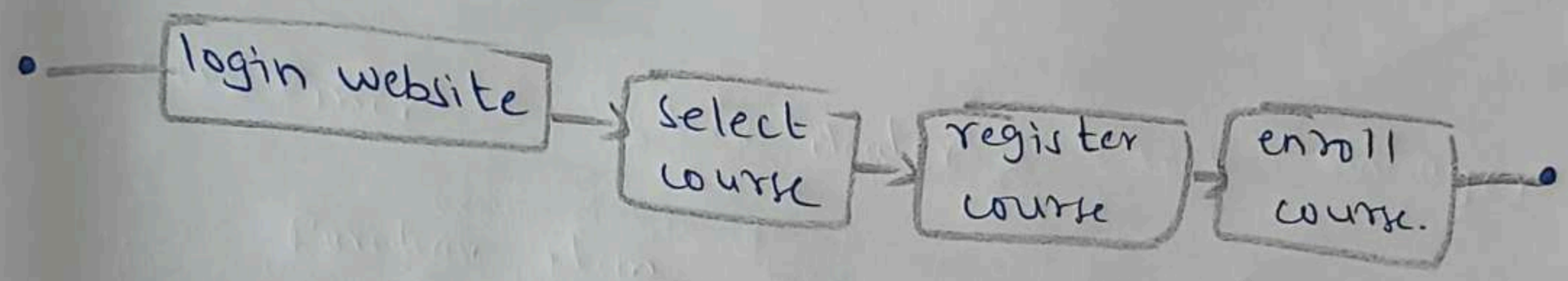
5. Online Course Registration

Aim: Develop a system for students to register for online course using UML.

Algorithm:

* Goto the online website
* Select course
* After Register the course
* If eligible enroll course.

State chart Diagram



Result:

Student register for course.

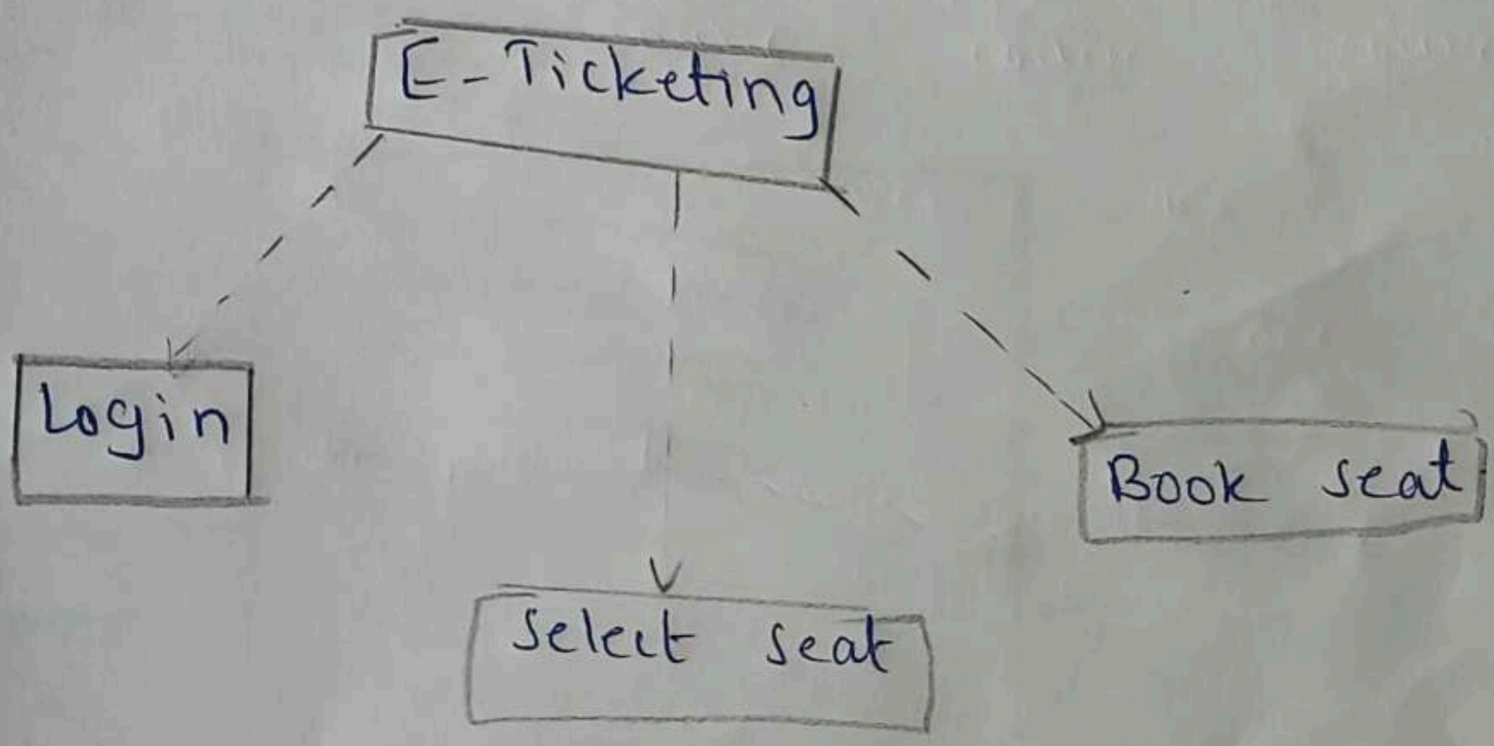
6. E-Ticketing System

Aim :- Design a E-Ticketing system for online seat booking using UML.

Algorithm:

* User login to E-Ticket system
* System enable select the seat
* choose the seat and proceed to Book ticket
* System processes booking and generate ticket.

Component diagram



Result:

User successfully book a seat and receive an e-ticket.

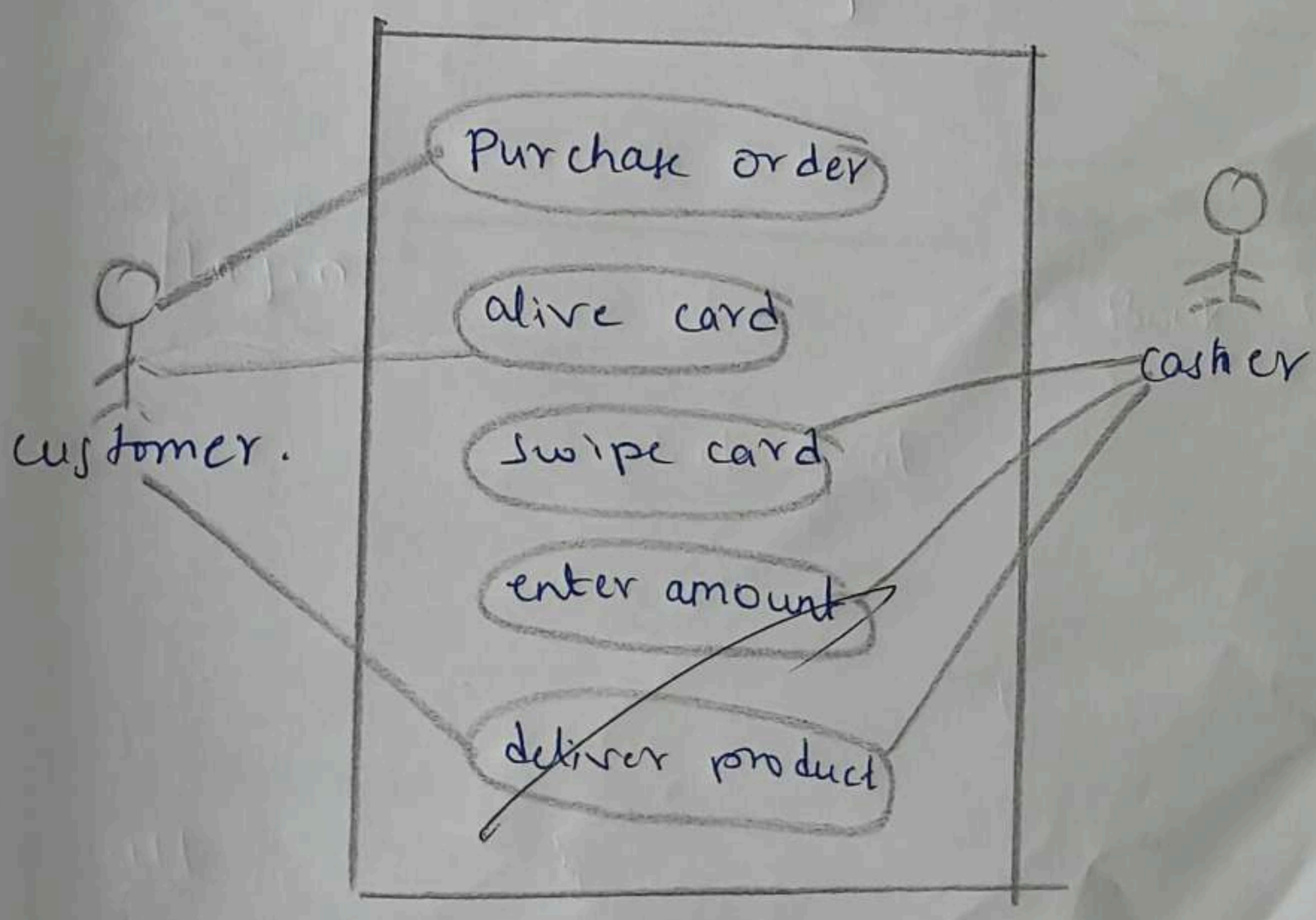
7. Credit Card Processing

Aim: To design a system for credit card processing using UML that verify whether the transaction done.

Algorithm:

* Customer select item.
* Casier generate bill
* Customer provide credit card
* Send transaction detail to system
* verify card validity and balance
* If valid. Approve
* If not valid. Reject

Use-case diagram:



Result:

The credit card processing sucessfully designed using UML.

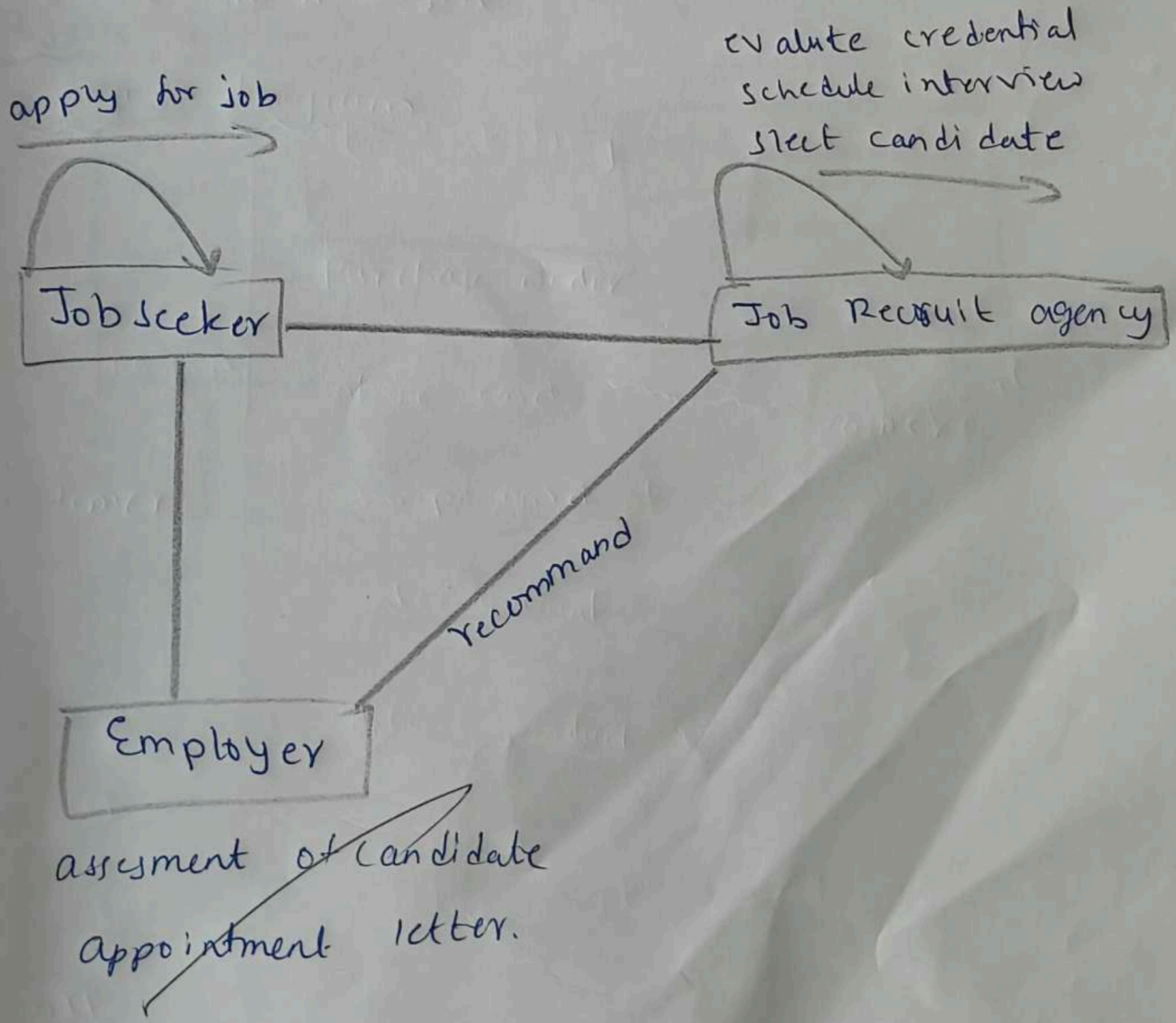
8. Recruitment System

Aim: To design a system for Recruitment using UML that verify eligible.

Algorithm:

* Applicant submit Application
* System store application
* HR review application
* Eligible candidate short list
* Interview scheduled
* Final decision made.

Collaboration diagram:



Result:

Recruitment system is sucessfully designed using

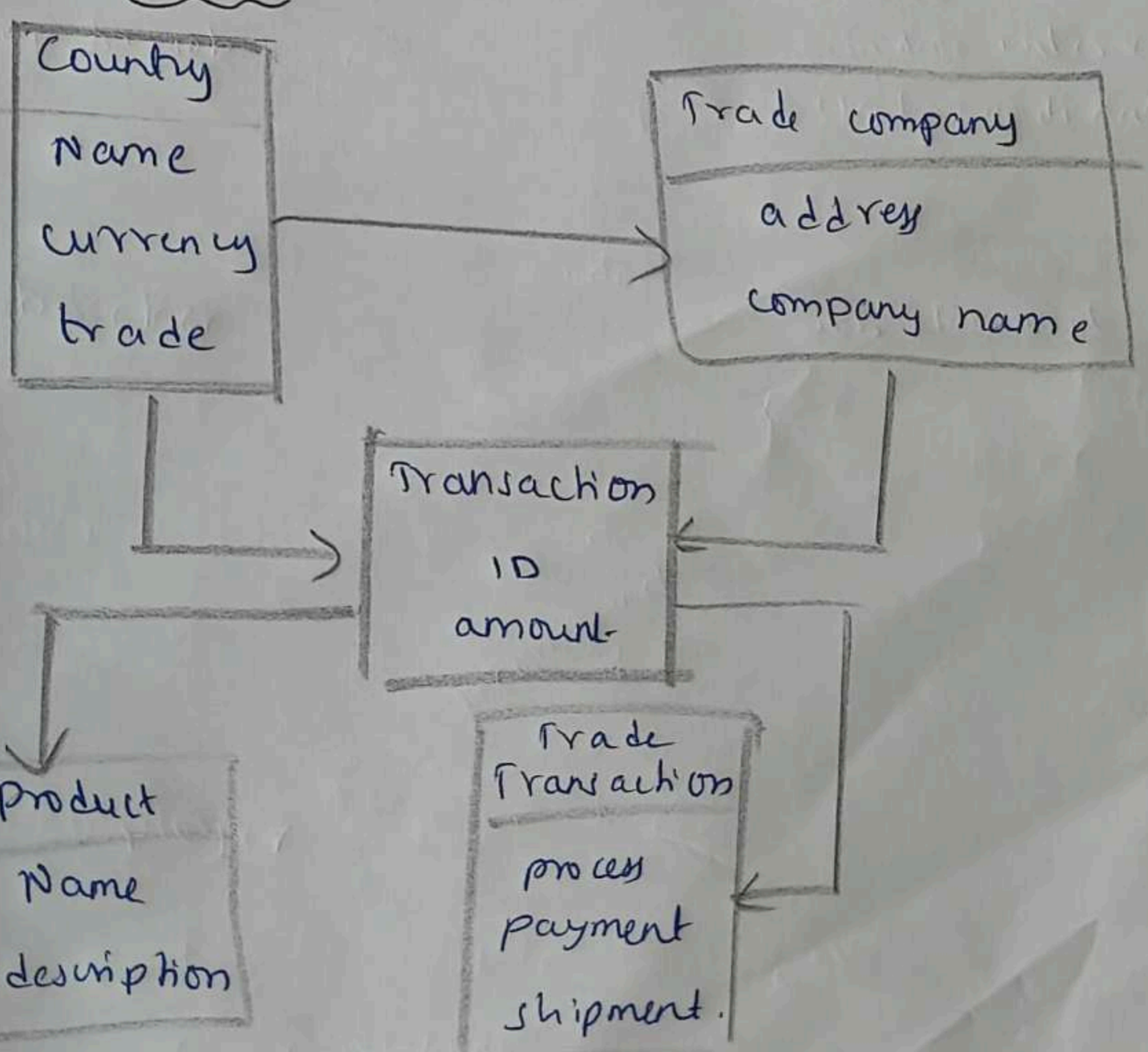
9. Foreign Trading System

Aim: To design a foreign Trading system using UML

Algorithm:

* Start
* Exporter create trade transaction
* Importer confirm order
* Bank verify and approve agent
* Customer authority check document
* If payment and clearence approved
* Ship goody

Object diagram



Result:

Thus, the foreign Trading System successfully represented using UML

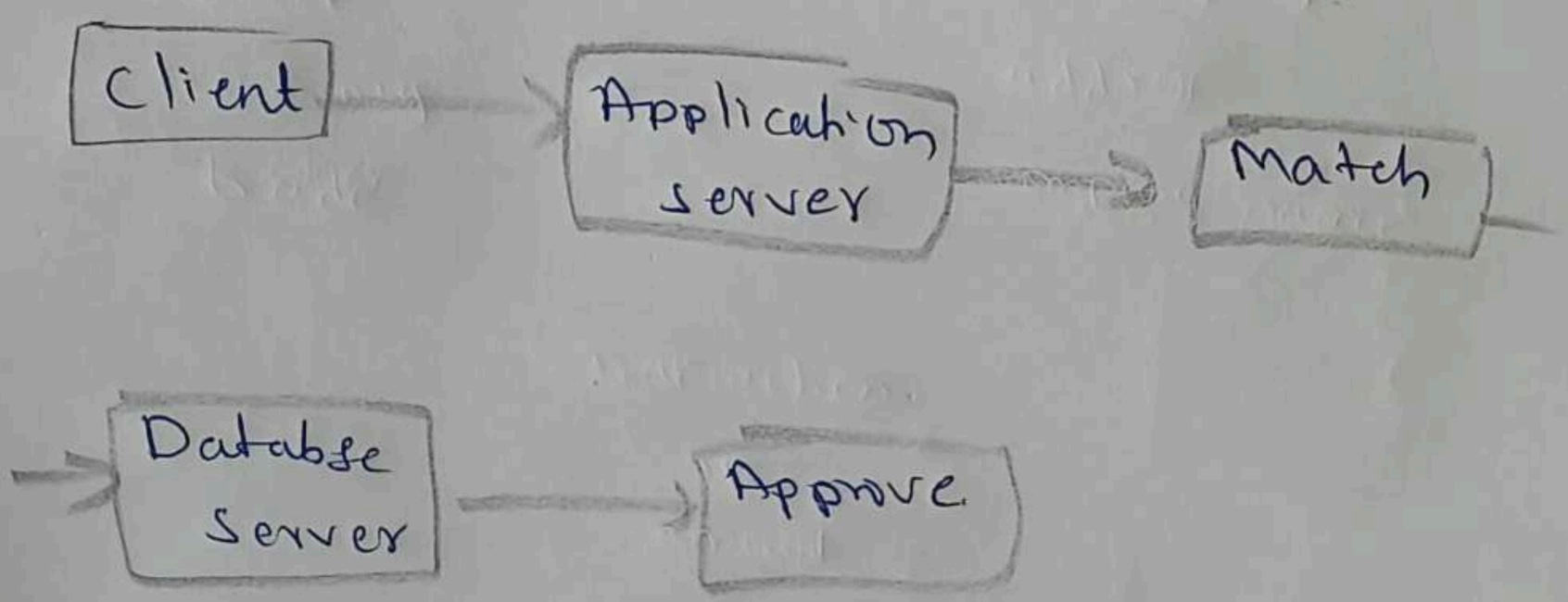
10. Blood Bank System

Aim: To design a Blood Bank system using UML for blood donation.

Algorithm

* Donor register through Client server
* Reception verify donar detail
* Blood donation detail stred in database
* Hospital send blood request
* If available - Approve
* If not - Notify horpital.

Deployment diagram.



Result:

Thus the blood bank system sucessfully represent using UML