DEPARTMENT OF NETWORKING AND COMMUNICATIONS

FACULTY OF ENGINEERING & TECHNOLOGY

MINI PROJECT

SUBJECT CODE: 18CSC202J

SUBJECT TITLE: OBJECT ORIENTED DESIGN AND PROGRAMMING

PROJECT TITLE : FUNCTIONING OF ATM

TEAM MEMBER'S (NAME AND REG.NO)

C KARTHIK REDDY [RA2111030010081]
K HEMANTH REDDY [RA2111030010087]

SRM University, SRM Nagar, Kattankulathur-603203
Kancheepuram District, Tamil Nadu

Rubrics

Experiment Component	Max. Marks	Grading Rubrics		
Documentation/ Procedure	10	UML Diagrams are well documented. The documentation supporting all functional requirement and non-functional requirement(10 Marks)	Missing two or more required functional requirement .The documentation work not up to the mark. (5 Mark)	
Concept	5	Completeness of concept, consistent variable naming and relationship in static view. (5 Marks)	Completeness of concept, inconsistent variable naming and relationship in static view. (3 Marks)	Incomplete static view. (1 Mark)
Usage of Symbols	3	Precise usage of symbols in dynamic view. (3 Marks)	Improper usage of Symbol's. (2 Marks)	Symbol's misplaced in diagram. (1 Mark)
Diagrams	4	Completion of all 8 UML Diagrams using Visual Paradigm Tool. (4 Marks)	Construction of UML Diagrams using other tools. (2 Marks)	Construction of few diagrams'. (1 Mark)
Viva and Innovative Idea	3	Oral Viva and Innovative approach. (3 Marks)	Oral Viva and partial idea. (2 Marks)	Oral Viva not fulfilled. (1 Mark)
TOTAL	25			

INDEX

List of Figures

1. Problem Description

With the evolution of technology and fast-moving lives of people, no one has the time or patience to go the bank and stand in long queues just to get done their money withdraw or money deposit. In addition to this, these days the banks fill up fast, especially during holidays and on the weekends, which means that many people might find themselves without a long queue
To rectify this situation, we have built an ATM MANAGEMENT SYSTEM.
By using ATM machines, they can deposit ther money and withdraw money from the ATM without visiting the bank and wait there for a long time.

Note: To be able to use the ATM you need to be a registered bank customer and you need to apply for the ATM card.

2. Software Requirements Specification

2.1 Introduction

2.1.1 Purpose

The purpose of this application is to provide user the opportunity to do their financial transactions. They can easily withdraw or deposit the money by using the ATM machine using ATM card.

2.1.2 Document Conventions

N/A

2.1.3 Indented Audience and Reading Suggestion

This document should be read by developers, users, project managers, and teasers. The developers should read every section to ensure that there is an Understanding of the project. The main sections for the customers to review are section 2.1.4 Project Scope, 2.2.7 Assumptions, and section 2.4 Features.

2.2 Project Scope

The ATM management system that can be accessed throughout the ATM card and can be accessed by anyone who has a the registered atm card. This application will help us to do financial transactions easily. It provides complete information regarding current balance on the screens with details of registerd user.

2.3 Reference:

The references for the above software are as follows:-

1. www.google.co.in
2. www.wikipedia.com

2.3 Overall Description

2.3.1 Product Perspective

The proposed ATM management system will give the user a wide range transaction across the country/world to choose from further saving them the hassle of long-standing queues.

2.3.2 Product Functions

The Online Movie Ticket Booking System provides online real time information currently running movies on all the screens with details of the user, available balance. The main purpose of this project is to reduce the hassle of long-standing queues. This software is capable of:

- USER DETAILS
- AVAILABLE BALANCE
- CURRENCY DENOMINATIONS
- MINI STATEMENT
- PRINT RECEIPT
- PIN CHANGE

The administrator will control users and manage and update the database.

2.3.3 User Classes and Characteristics

We have 3 levels of users:

User module: In the user module, the user will check the currently displayed movies :

- Login/Logout
- view balance
- phone number change
- otp verification
- change pin
- Make Payment

Administrator Module:

- Add succesful payment history
- Update succesful payment history
- Delete unsuccesful payment Records

2.3.4 Operating Environment

Manual use only

2.3.5 Design and Implementation Constraints

Any update regarding the ATM database is to be recorded to have updated & correct information, and any cancellation or changes should be made known as soon as possible.

2.3.6 User Documentation

There will be a basic tutorial document.

2.3.7 Assumptions and Dependencies

The requirements stated in the SRS could be affected by the following factors:

- One major dependency that the project might face is the changes that a huge number of users using the app at the same time can lead to its slow-down.
- Another constraint relating to the operating environment is that we are specific to Oracle Database.
- At this stage, no quantitative measures are imposed on the software in terms of speed and memory although it is implied that all functions will be optimized with respect to speed and memory.

It is furthermore assumed that the scope of the package will increase considerably in the future.

2.4 External Interface Requirements

2.4.1 User Interface

The software provides good graphical interface for the user any administrator can operate on the system should prompt User Interface performing the required task such as view their available balance, transaction history, Mini statement, print receipt, etc..

2.4.2 Hardware Interface

Operating system: window

Hard Disk: 40GB

RAM:256 MB

Processor: Pentium (R)

2.4.3 Software Interface

- Java Language
- NetBeans IDE 7.0.1
- MS SQL server 2005

2.4.4 Communication Interface

Window

2.5 System Features

2.5.1 Portability

The system is developed for secured purposes, so it can't be portable.

2.6 Other Non-Functional Requirements

2.6.1 Functional Requirements

- Customer: In this we can store the login details of the customer
- Administrator: In this we can store the details and functions of admin.

2.6.2 Safety Requirements

Checking the fact that all the clients must be attachable to one server, so there would be appropriate control of the test statistics and information. Also in case of a potential loss of connection between the client and the server, the client's current transaction progress is lost. When the client finishes its booking progress (by pressing the confirm button) then its progress is sent to the server and logged in.

2.6.3 Security Requirements

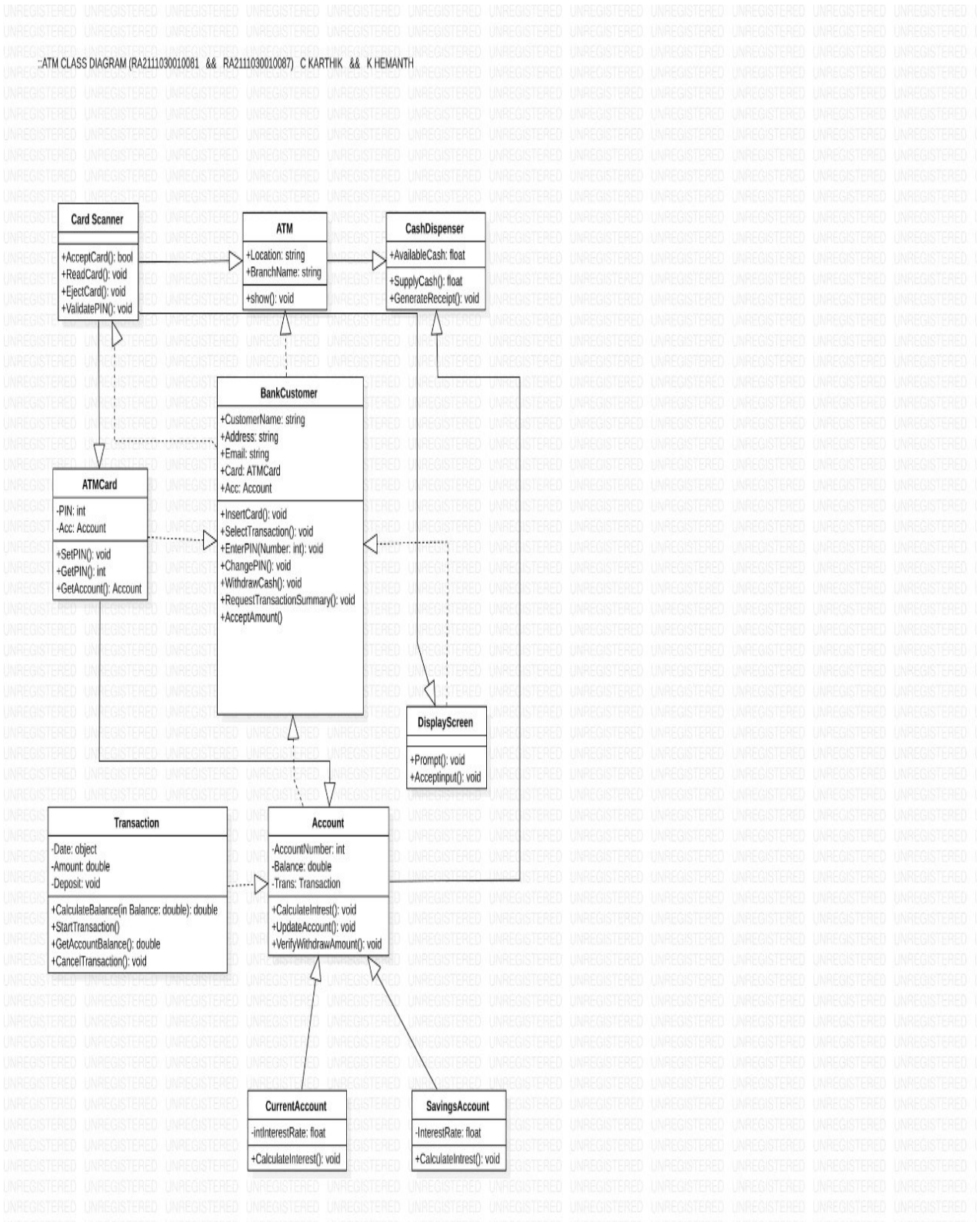
The application must reflect online ethics, policies, and legal obligations concerning user privacy and confidentiality.

2.6.4 Software Quality Attributes

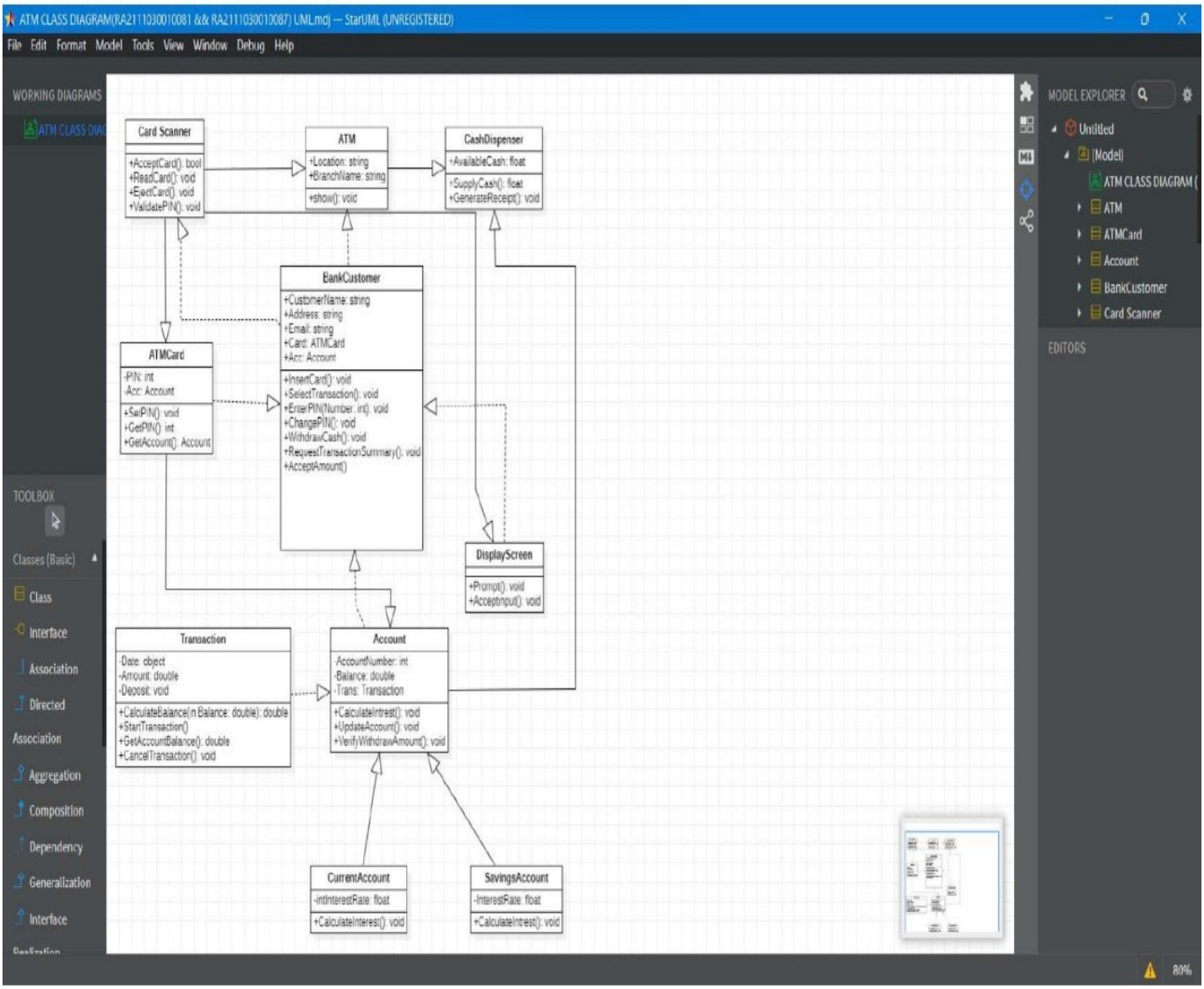
- The data communication protocol shall be such that it ensures the reliability and quality of data and voice transmission in a mobile environment. For example, CDMA.
- The memory system shall be of a non-volatile type.

1. **Class Diagram**:

Class Diagram is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations, and the relationships among objects. Here the system is the functions of the ATM machine.



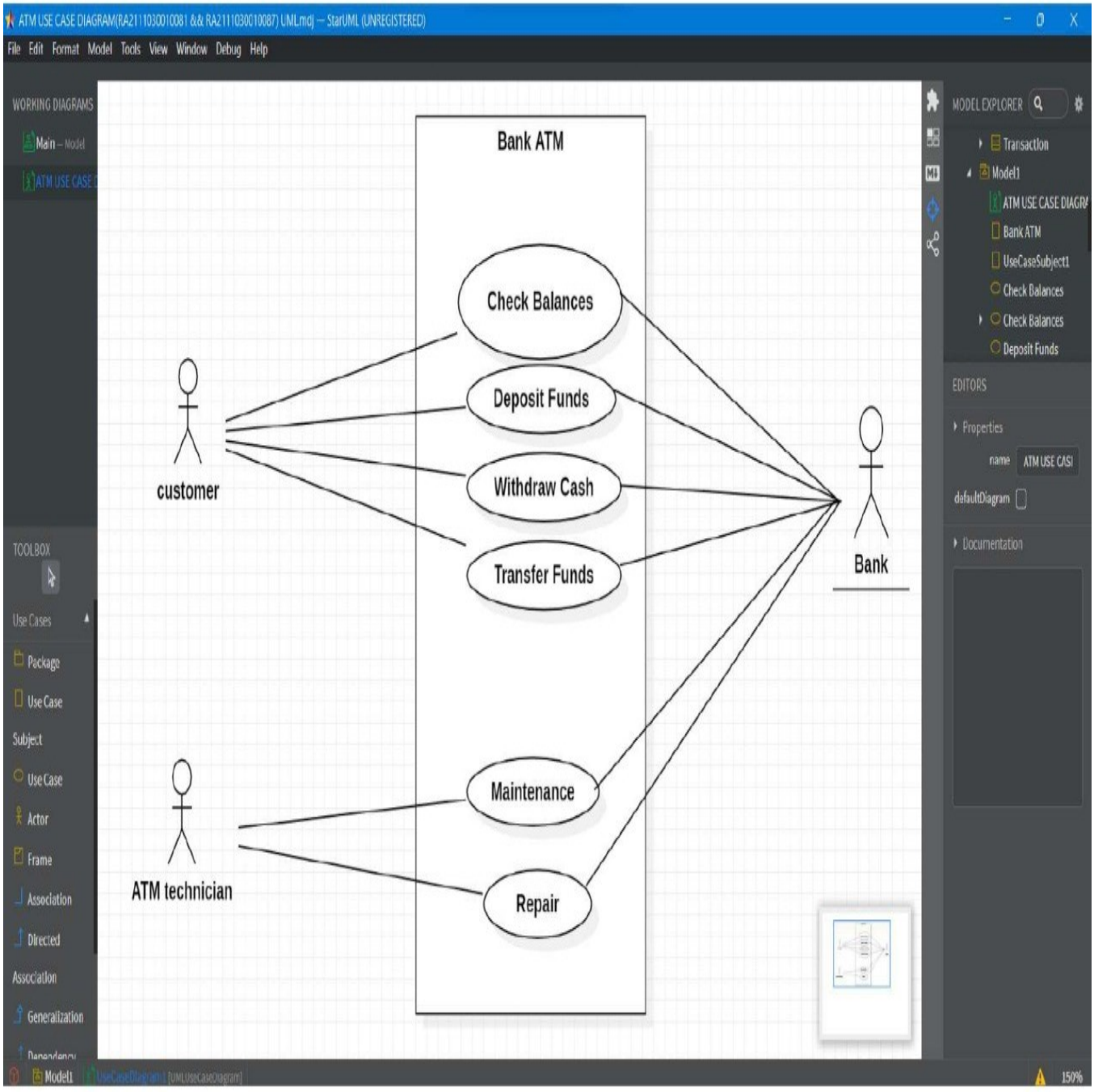
ATM CLASS DIAGRAM IN STAR UML



2. **Use Case Diagram:**

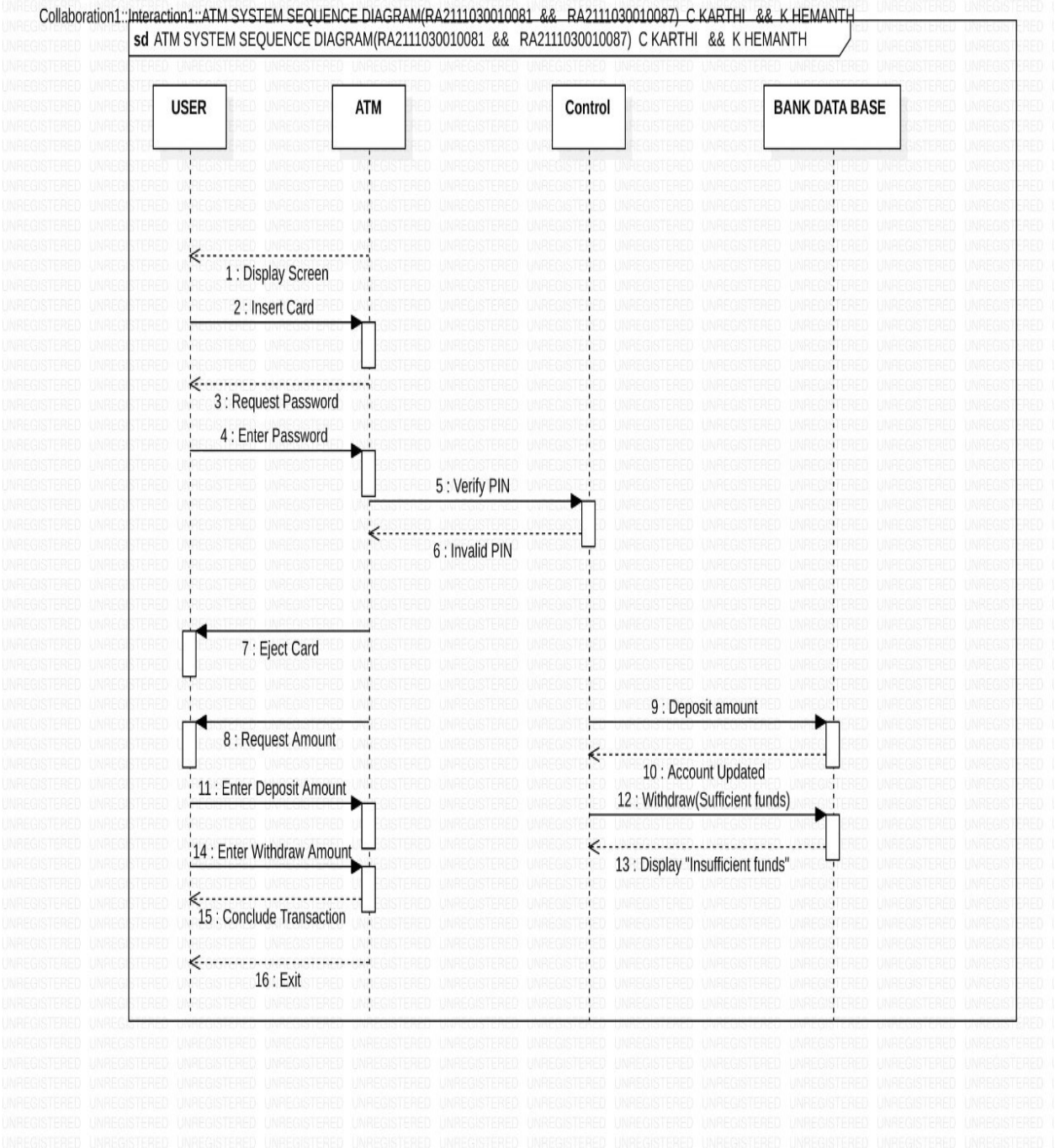
The objective of a UML use case diagram is to show the interactions of numerous items called actors with the use case and to capture fundamental functionalities of the ATM functions.

ATM USE CASE DIAGRAM IN STAR UML

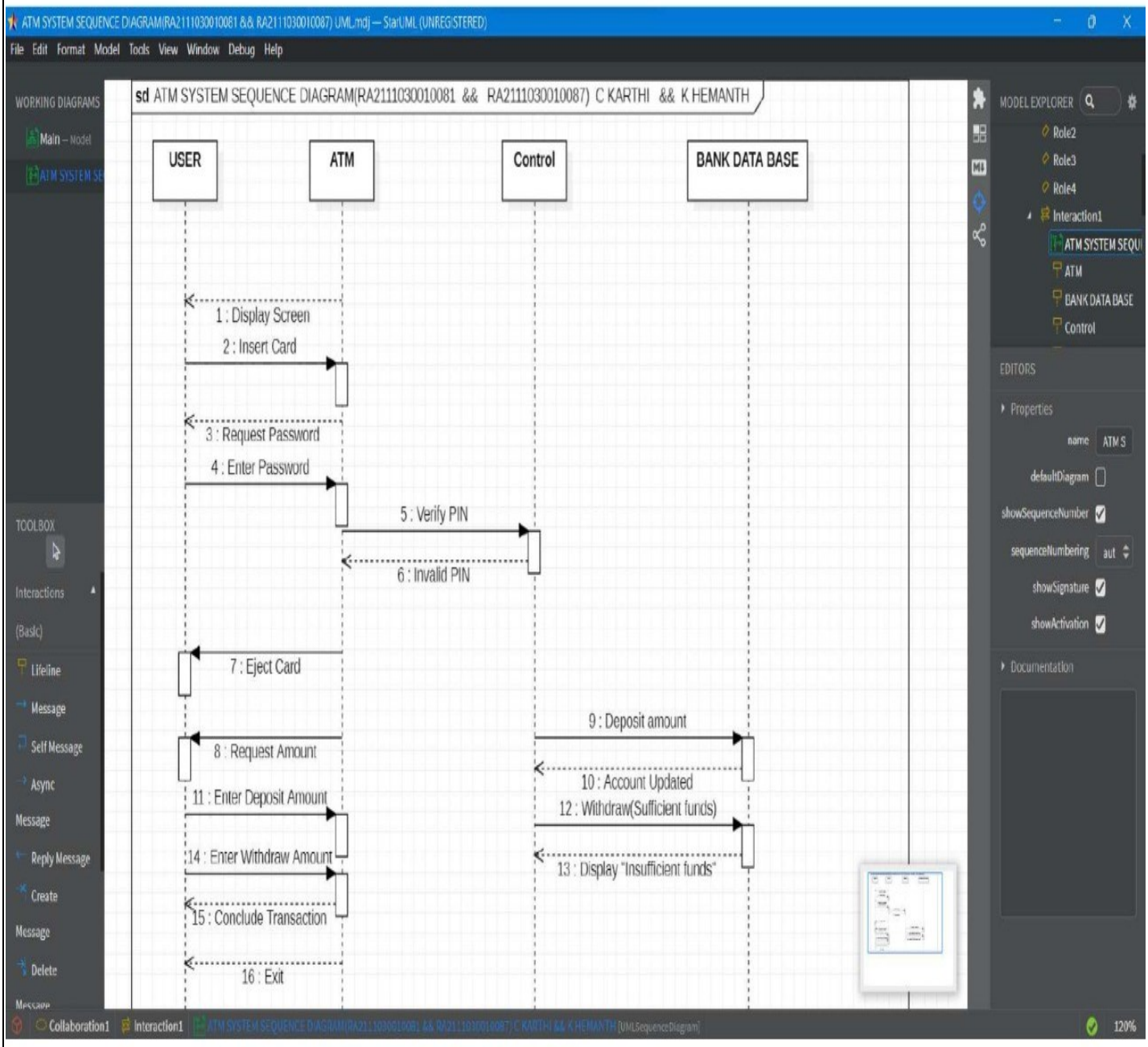


3. **Sequence Diagram:**

Sequence diagram s object interactions arranged in time sequence. It depicts the objects involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of ATM management functions.

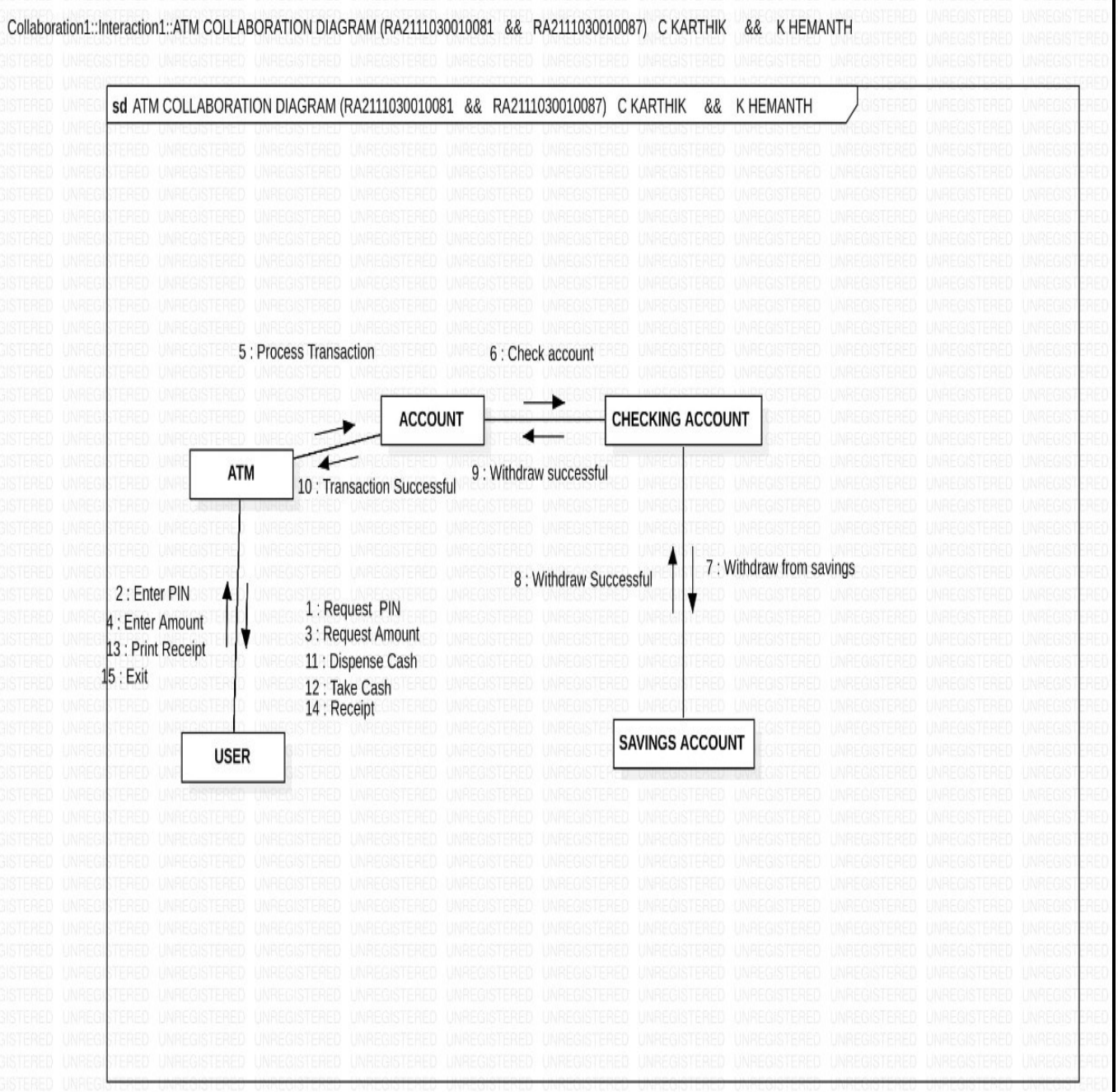


ATM SYSTEM SEQUENCE DIAGRAM IN STAR UML

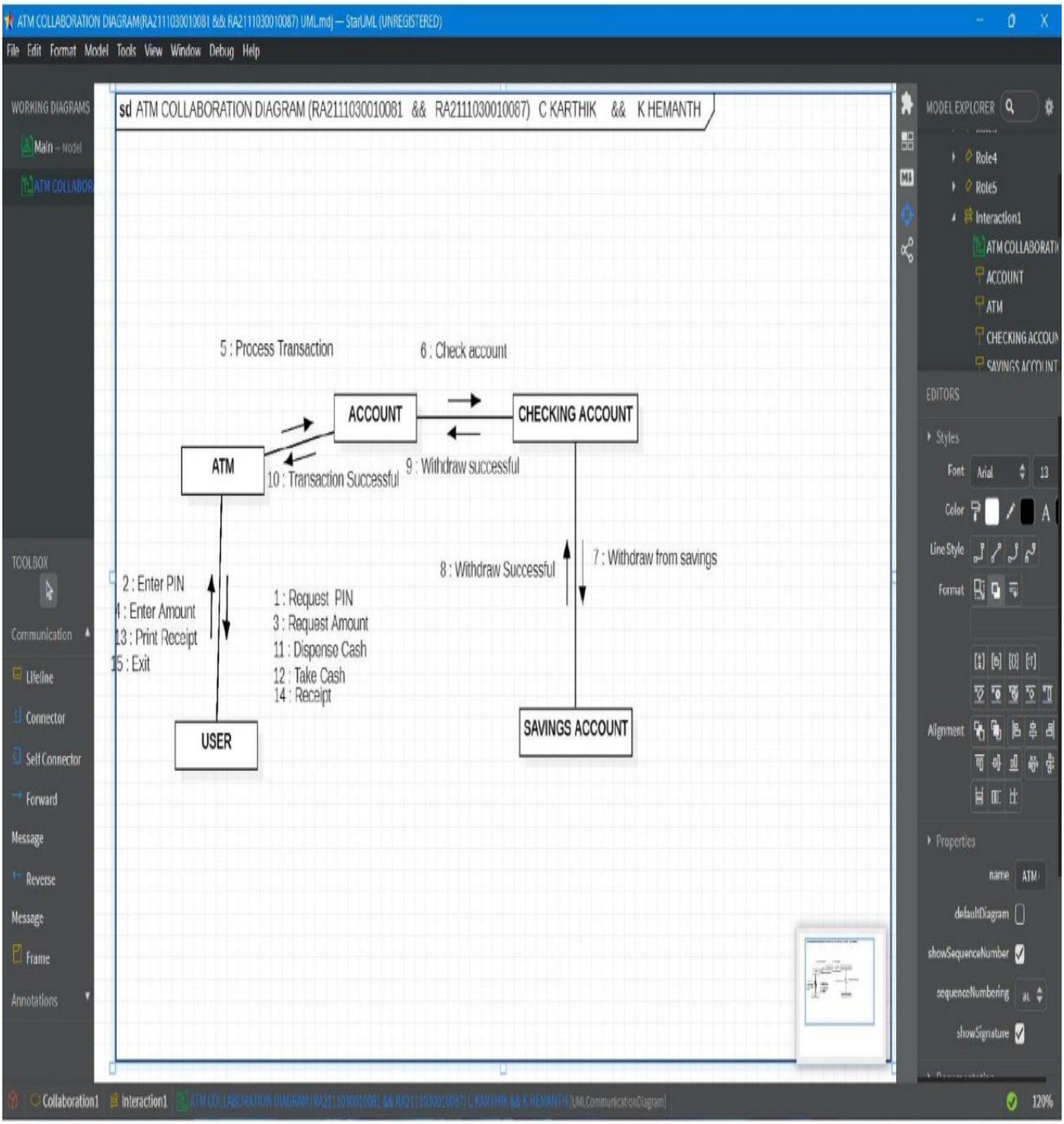


4. **collaboration diagram:**

Collaboration diagrams (known as communication diagram in uml 2.x) are used to show how objects interact to perform the behaviour of a particular use case, or a part of a use case. Along with sequence diagrams, collaboration is used by designers to define and clarify the roles of the objects that perform a particular flow of events of a use case. They are the primary source of information used to determining class responsibilities and interfaces.

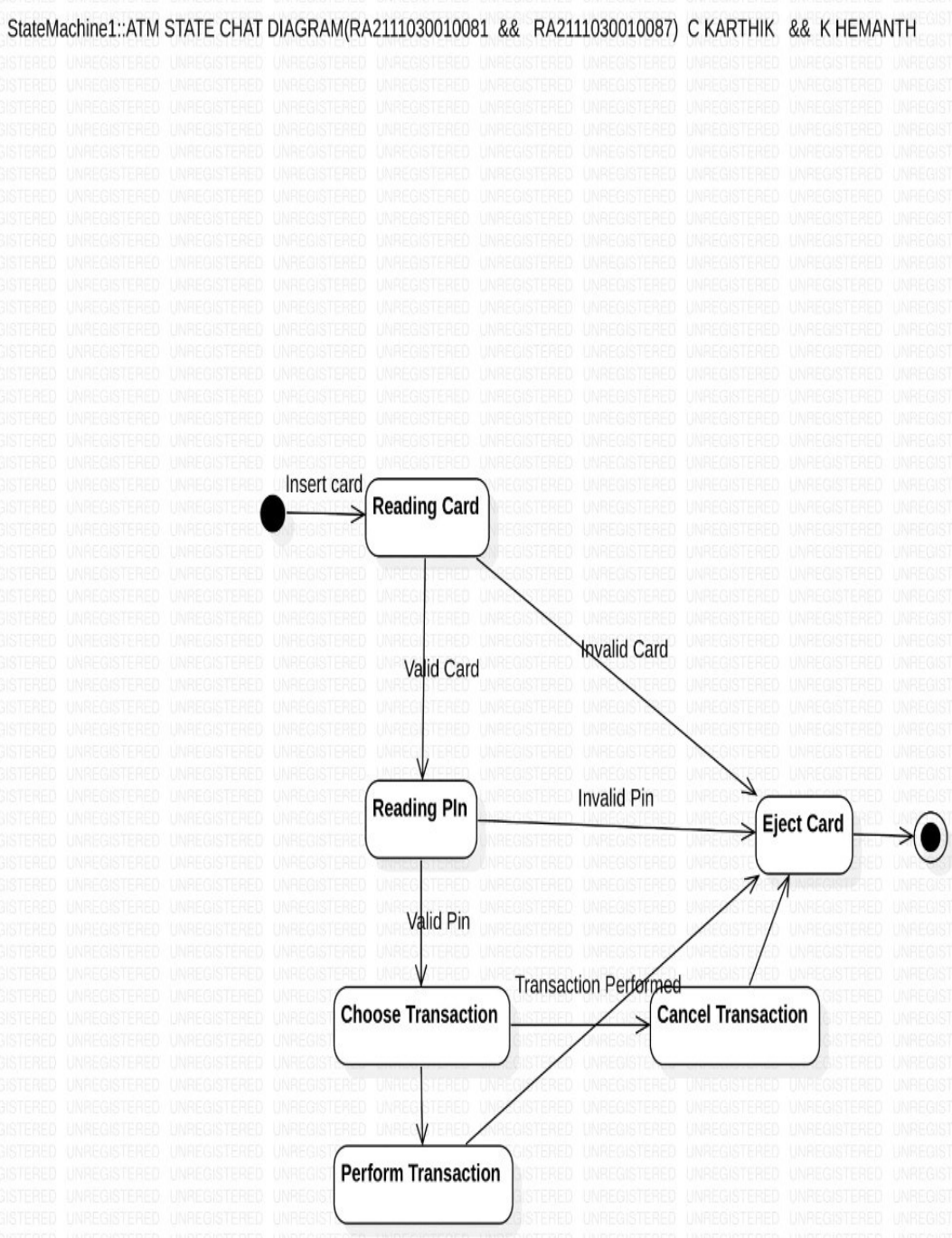


ATM COLLABORATION DIAGRAM IN STAR UML

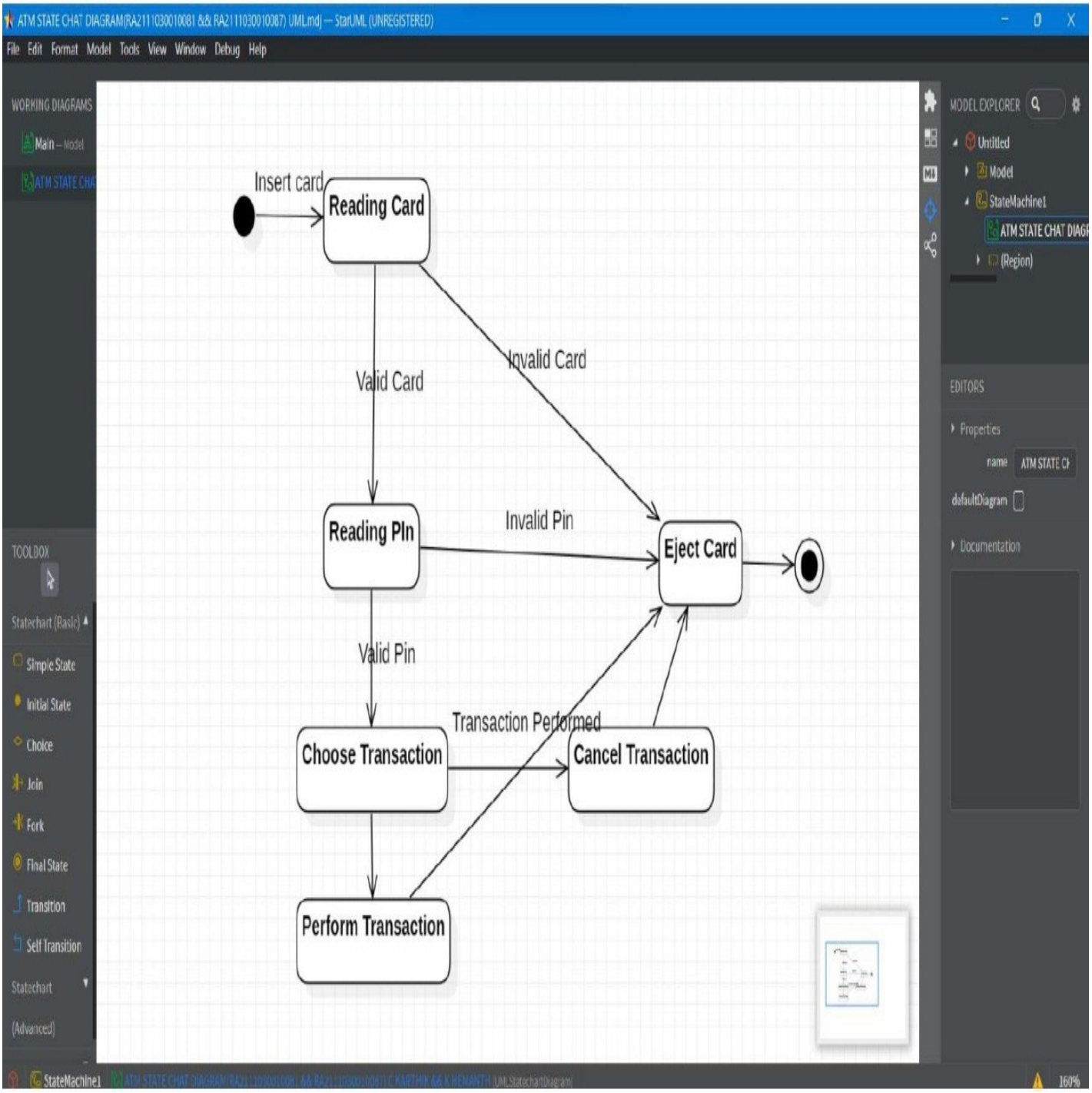


5. **State Chart Diagram:**

State transition diagrams provide a way to model the various states in which an object can exist. While the class diagram shows a static picture of the classes and their relationships, state transition diagrams model the dynamic behaviour of a system in response to external events (stimuli).

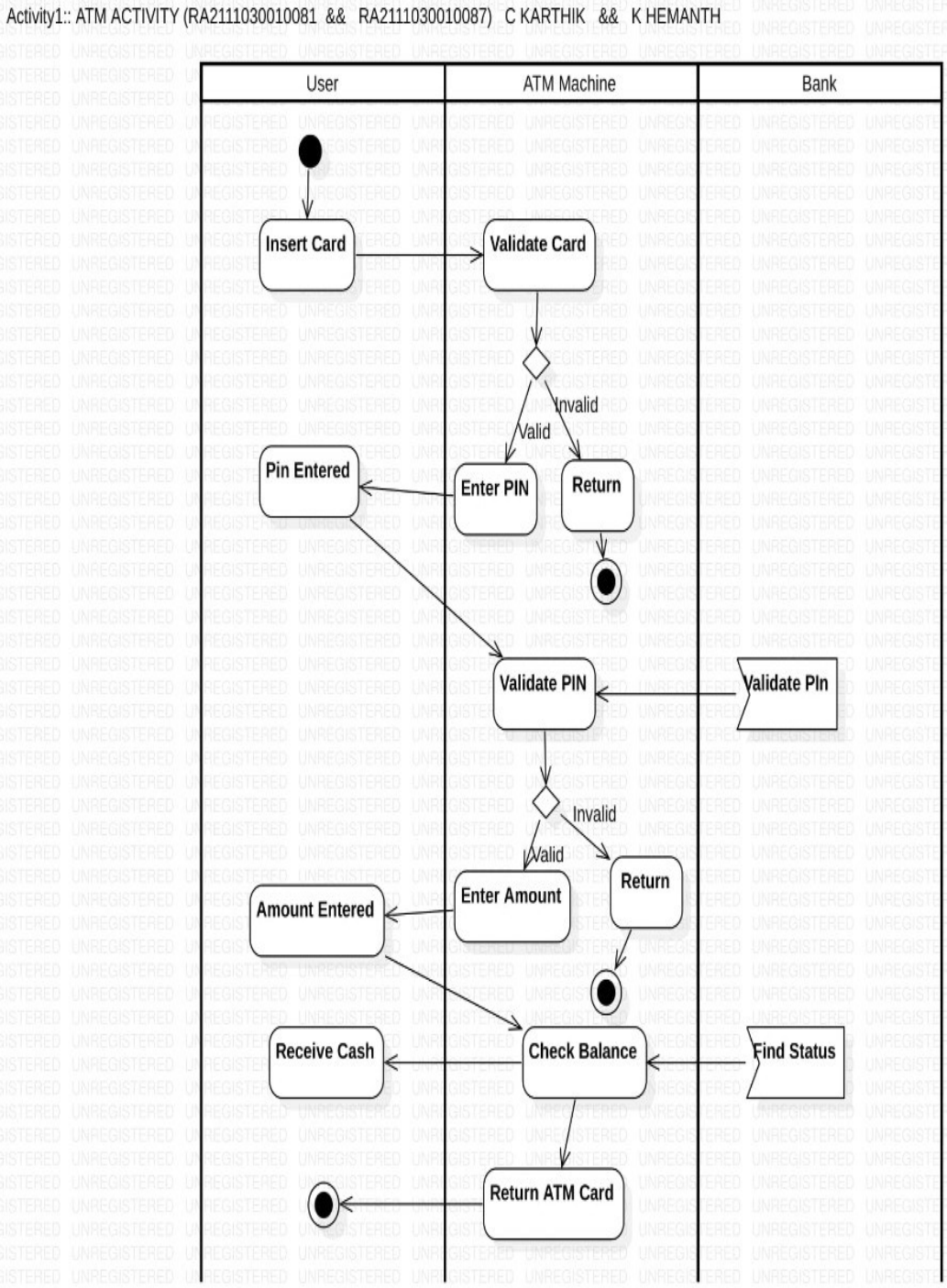


ATM STATE CHAT DIAGRAM IN STAR UML

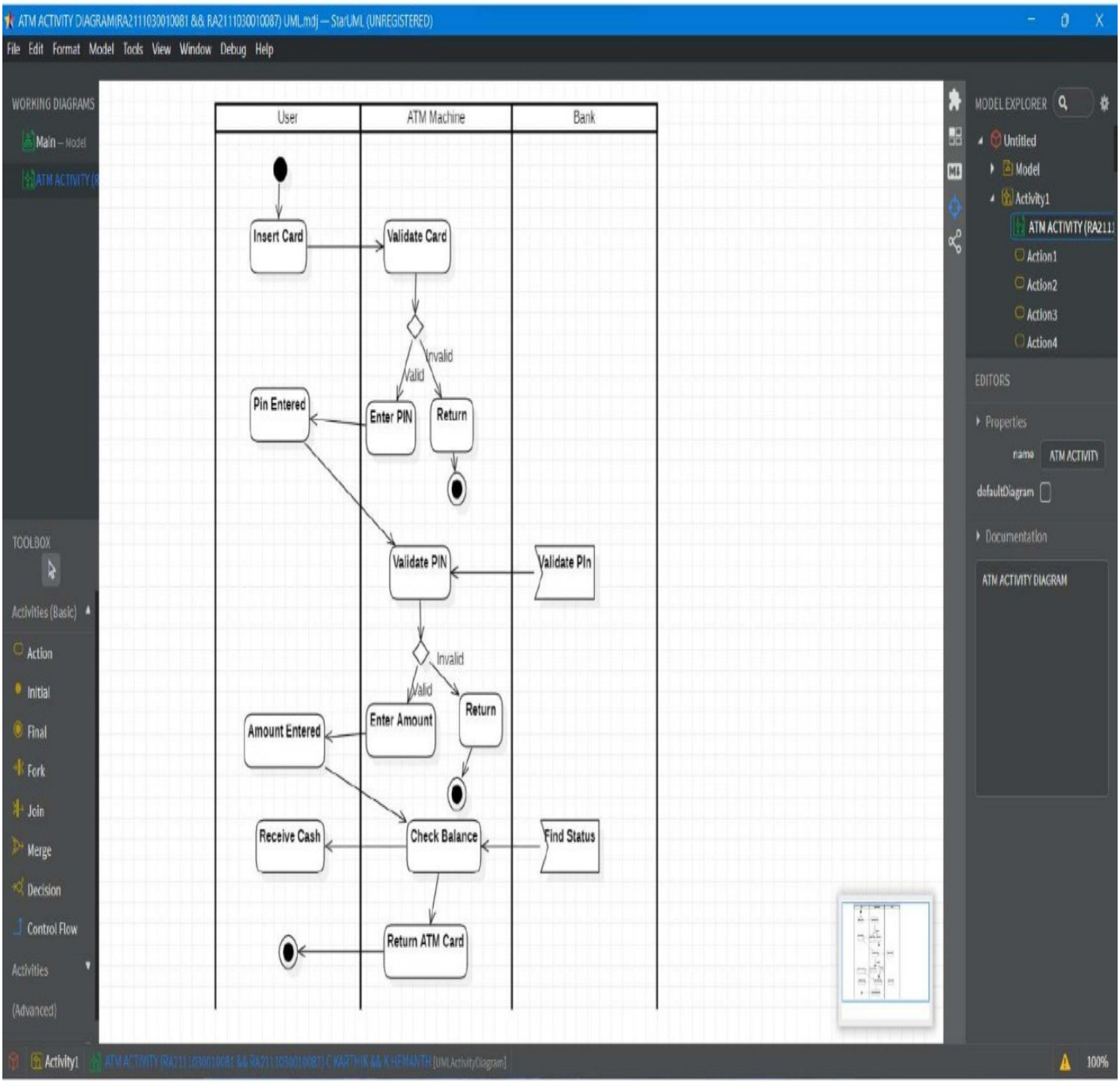


6. **Activity Diagram:**

Activity Diagram: - Activity diagrams describe the activities of a class. They are similar to state transition diagrams and use similar conventions, but activity diagrams describe the behaviour/states of a class in response to internal processing rather than external events.

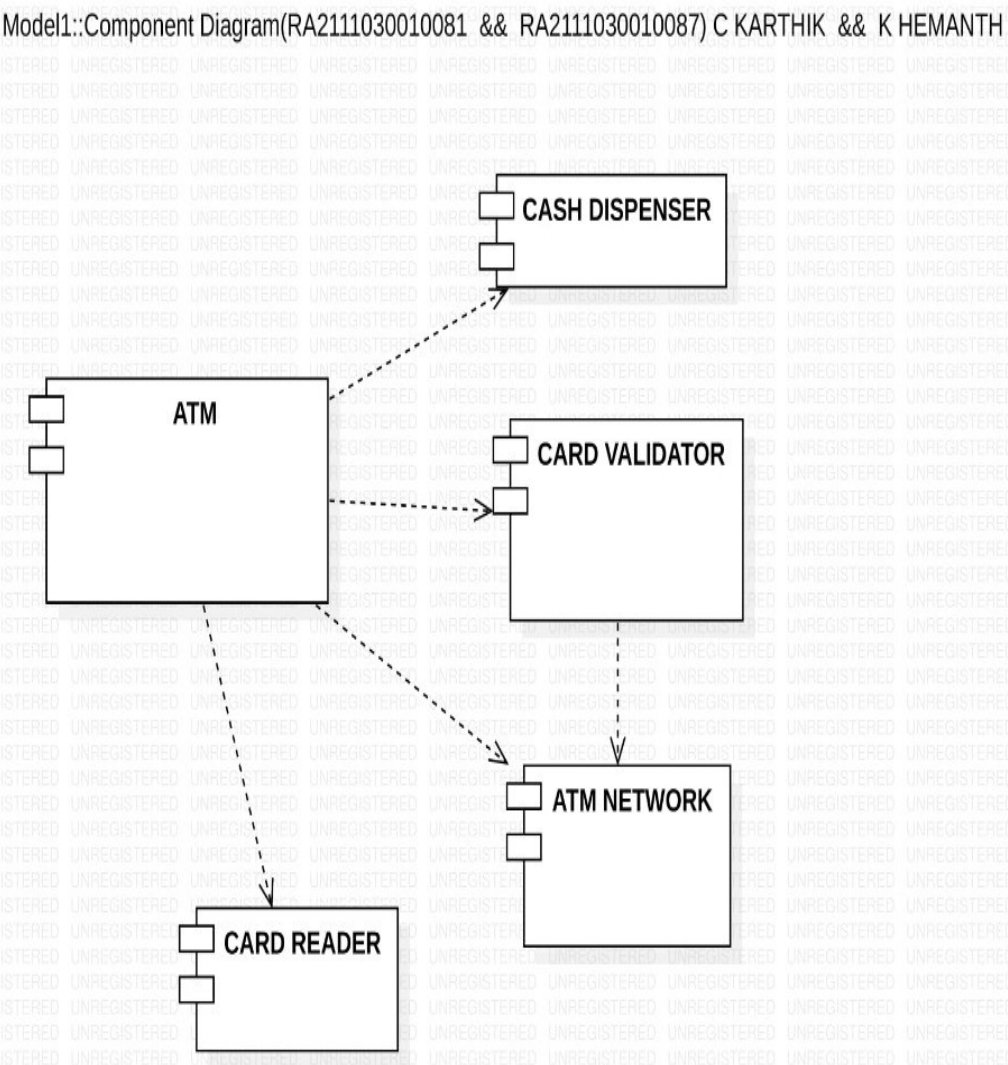


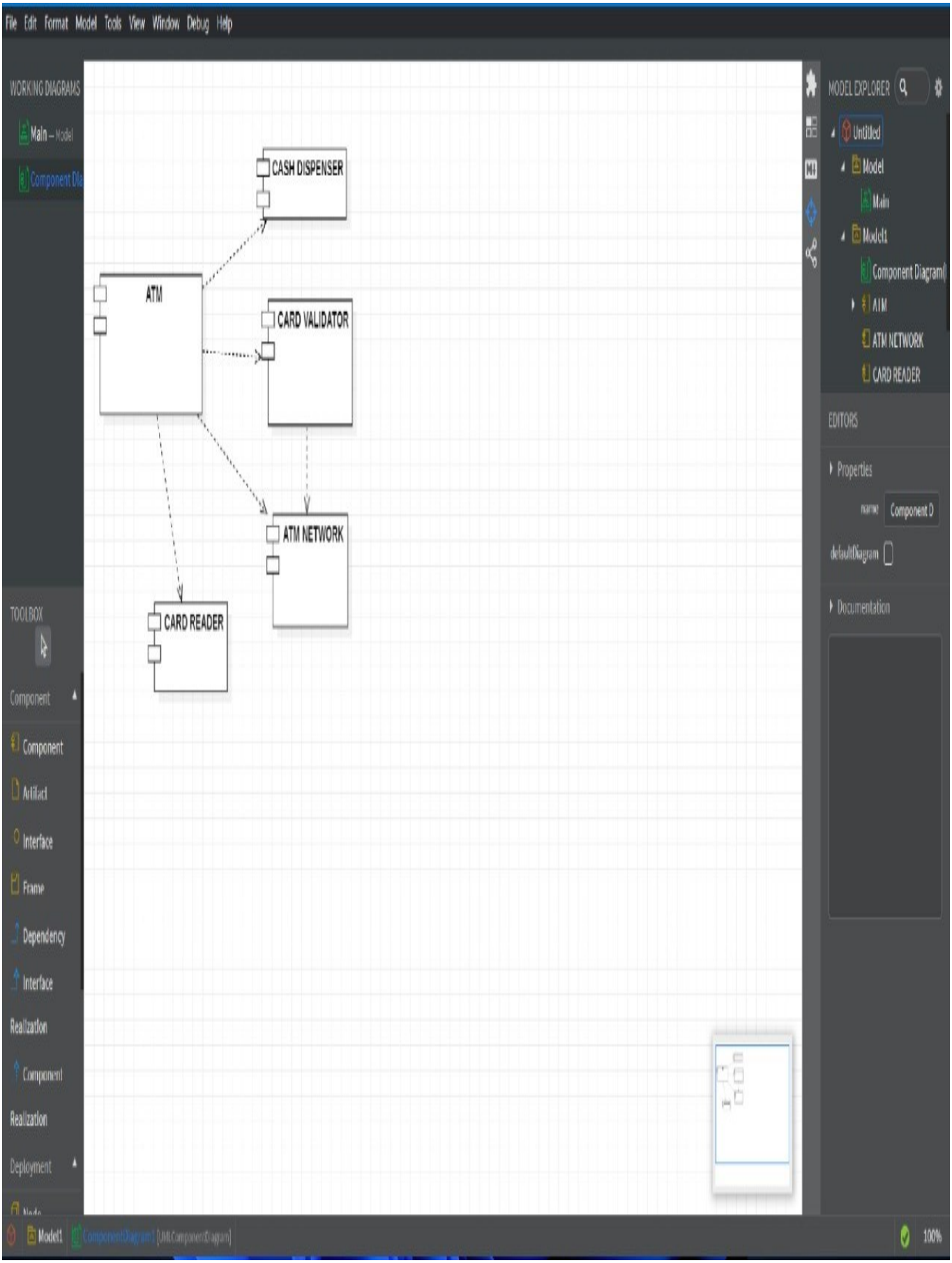
ATM ACTIVITY DIAGRAM IN STAR UML



7. **Component Diagram:**

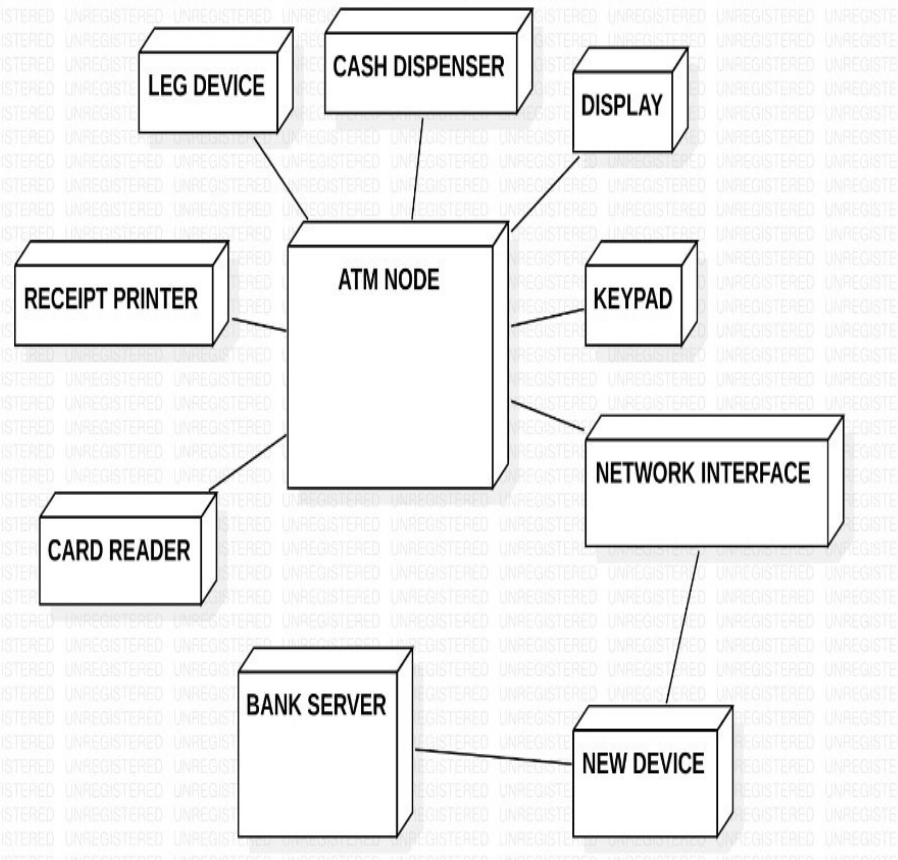
A component diagram in the Unified Modeling Language depicts how components are wired together to form larger components and or software systems. Components diagrams can be used to illustrate the structure of arbitrarily complex systems.

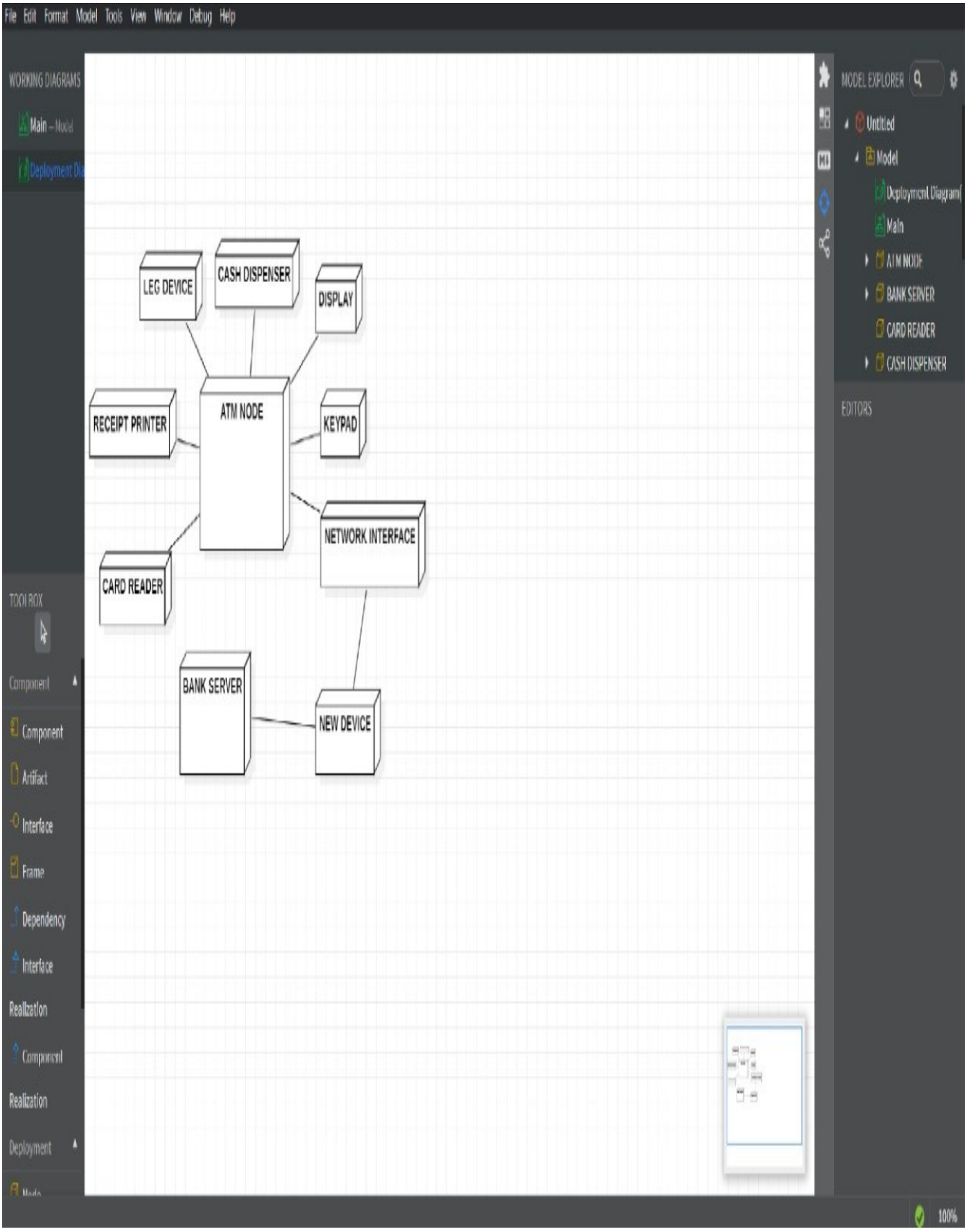



File Edit Format Model Tools View Window Debug Help
WORKING DIAGRAMS
Main – Model
Component Dia
CASH DISPENSER
ATM
CARD VALIDATOR
ATM NETWORK
CARD READER
TOOLBOX
Component
Component
Artifact
Interface
Frame
Dependency
Interface
Realization
Component
Realization
Deployment
MODEL EXPLORER
Untitled
Model
Main
Model1
Component Diagram
ATM
ATM NETWORK
CARD READER
EDITORS
Properties
name
Component D
defaultDiagram
Documentation
Model1
ComponentDiagram [UMLComponentDiagram]
100%

8. **Deployment Diagram:**

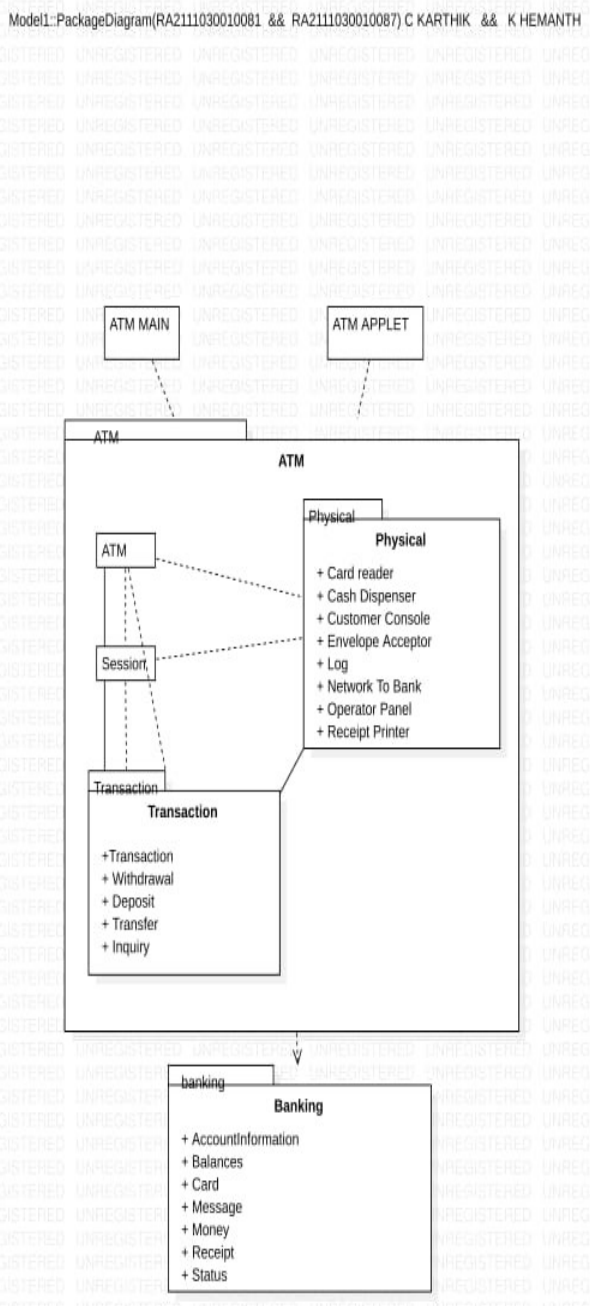
A deployment diagram in the Unified Modeling Language serves to model the physical deployment of artifacts on deployment targets. Deployment diagrams show the allocation of Artifacts to Nodes according to the Deployments defined between them. Deployment of an artifact to a node is indicated by placing the artifact inside the node.

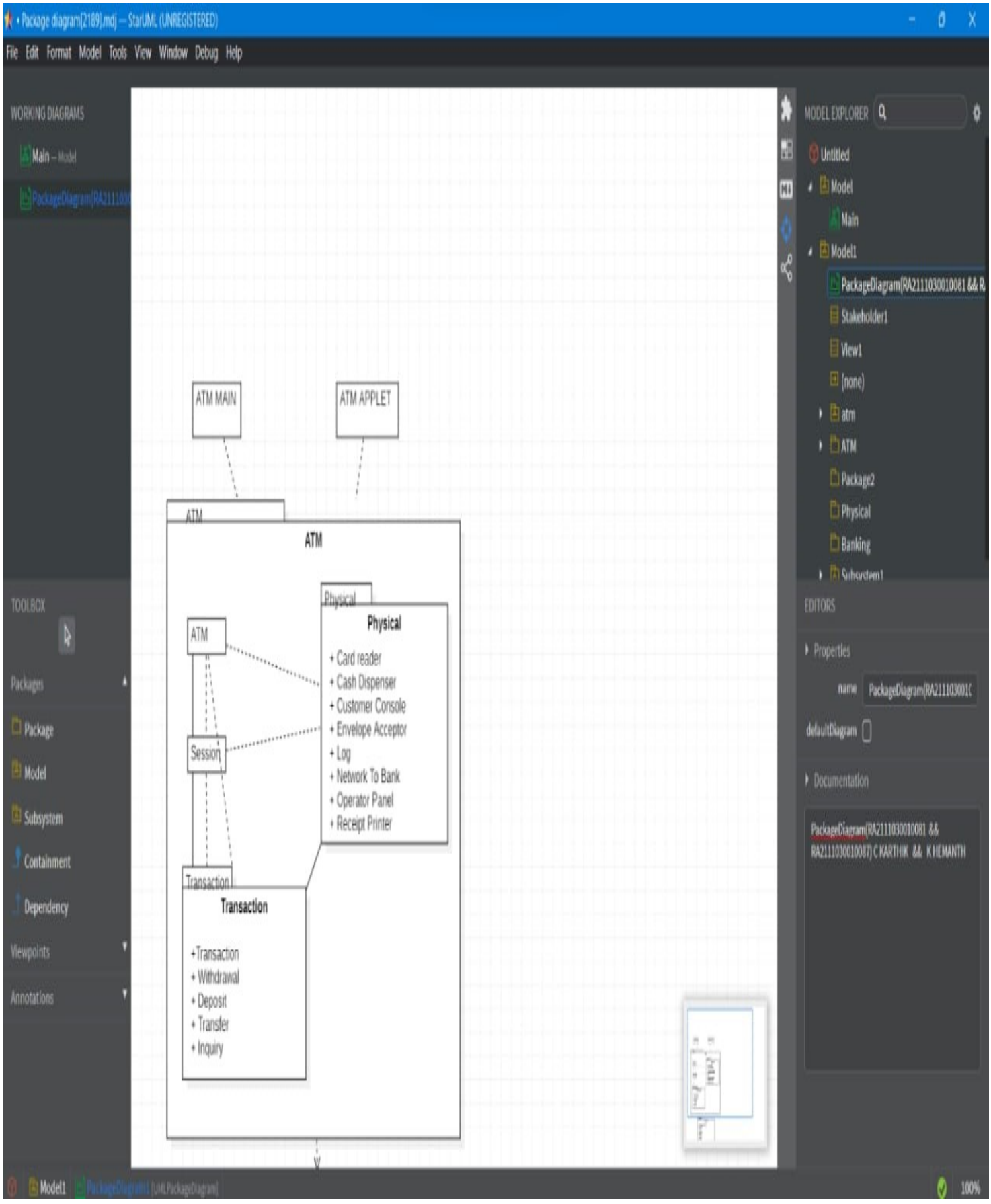



File Edit Format Model Tools View Window Debug Help
WORKING DIAGRAMS
Main – Model
Deployment Dia
LEG DEVICE
CASH DISPENSER
DISPLAY
RECEIPT PRINTER
ATM NODE
KEYPAD
NETWORK INTERFACE
CARD READER
BANK SERVER
NEW DEVICE
TOOLBOX
Component
Component
Artifact
Interface
Frame
Dependency
Interface
Realization
Component
Realization
Deployment
MODEL EXPLORER
Untitled
Model
Deployment Diagram
Main
ATM NODE
BANK SERVER
CARD READER
CASH DISPENSER
EDITORS
100%

9. **Pacakage diagram:**

package diagram is a pattern for grouping elements and defining their interdependencies (packages). The main goal of package diagrams is to simplify the complex class diagrams that can be used to group classes into packages. These groups help define the hierarchy.




• Package diagram(2189).mdj — StarUML (UNREGISTERED)
File Edit Format Model Tools View Window Debug Help
WORKING DIAGRAMS
Main – Model
PackageDiagram(RA2111030
MODEL EXPLORER
Untitled
Model
Main
Model1
PackageDiagram(RA2111030010081 && R
Stakeholder1
View1
(none)
atm
ATM
Package2
Physical
Banking
Subsystem1
TOOLBOX
Packages
Package
Model
Subsystem
Containment
Dependency
Viewpoints
Annotations
EDITORS
Properties
name
PackageDiagram(RA211103001
defaultDiagram
Documentation
PackageDiagram(RA2111030010081 &&
RA2111030010087) C KARTHIK && K HEMANTH
ATM MAIN
ATM APPLET
ATM
ATM
Physical
Physical
+ Card reader
+ Cash Dispenser
+ Customer Console
+ Envelope Acceptor
+ Log
+ Network To Bank
+ Operator Panel
+ Receipt Printer
ATM
Session
Transaction
Transaction
+Transaction
+ Withdrawal
+ Deposit
+ Transfer
+ Inquiry
Model1
100%

CONCLUSION

- Even in the case of a specialist subject such as ATM system modelling,
- it is possible to see the wide range of applications for UML diagrams and object-oriented modelling methods.

.

REFERENCES

https://youtu.be/WnMQ8HlmeXc

www.conceptdraw.com/examples/package-diagram-for-atm

https://creately.com/diagram/example/ilo6lr0c1/atm-package-diagram-classic