



INSURANCE MANAGEMENT SYSTEM

Object Oriented Design and Programming Assignment



***By: Arpita Muleva (RA1911026010046)
and Shravya Sharan (RA1911026010055)***

INDEX

1.	OVERVIEW	4
2.	SECTION 1: SCENARIO	5
3.	INTRODUCTION	6
4.	ABOUT THE SYSTEM	7
	a) User Section	
	b) Policy Section	
	c) Payment Section	
	d) Actors	
5.	PROCEDURE	14
	a) Mechanism: Company Associate	
	b) Mechanism: Customer	
6.	SECTION 2: DIAGRAMS	19

7. UML CLASS DIAGRAM 20

- a) **Diagram**
- b) **Definition**

8. USE CASE DIAGRAM 22

- a) **Diagram**
- b) **Definition**

9. SEQUENCE DIAGRAM 24

- a) **Diagram**
- b) **Definition**

10. COLLABORATION DIAGRAM 26

- a) **Diagram**
- b) **Definition**

11. STATE CHART DIAGRAM 28

- a) **Diagram**
- b) **Definition**

12. ACTIVITY DIAGRAM 30

- a) Diagram**
- b) Definition**

13. PACKAGE DIAGRAM 32

- a) Diagram**
- b) Definition**

14. COMPONENT DIAGRAM 34

- a) Diagram**
- b) Definition**

15. DEPLOYMENT DIAGRAM 36

- a) Diagram**
- b) Definition**

16. CONCLUSION 39

OVERVIEW

- ❖ Insurance Management System is the complete solution for organizations or individuals who seek to effectively manage the data associated with insurance policies.
- ❖ Insurance can be explained as when an insurer indemnifies another against losses from specific contingencies and/or perils.
- ❖ From registration to purchase of any policy, all processes are executed in online mode which drastically reduces the scope for manual error and makes it easy to keep track of and access data when required. This system undertakes a variety of policies and assists in improvement of productivity since all data is available in one place.
- ❖ The construction and mechanism of the system is described using different types of UML diagrams.
- ❖ Insurance Management System as a whole seeks to improve accessibility and management of data associated with policies and promote optimum utilization of time and effort by allowing most of the requisite processes to be executed online.

SECTION 1:

SCENARIO

In this Section we will be explaining:

- What is Insurance?
- What is Insurance Management System?
- The Components of the System
- The Users and their role.
- Users' interaction with the system

INTRODUCTION

- ❖ Insurance is a contract, represented by a policy, in which an individual or entity receives financial protection or reimbursement against losses from an insurance company. The company pools clients' risks to make payments more affordable for the insured.
- ❖ There many types of insurance policies. Life, health, homeowners, and auto are the most common forms of insurance.
- ❖ Insurance policies are used to hedge against the risk of financial losses, both big and small, that may result from damage to the insured or her property, or from liability for damage or injury caused to a third party.
- ❖ The concept of insurance is very simple to understand. You pay a monthly or yearly fee to the insurance company to insure your life, health, vehicle, property etc. for a certain period. In return, the insurer pays for the financial damages in case of any harm to the insured person or object.
- ❖ So you are transferring the risk of a financial loss you might experience due to life's uncertainties to an insurance company for a small fee.

ABOUT THE SYSTEM

- ❖ An insurance management system deals with managing and retaining the details of the insurance owned by an individual.
- ❖ It is developed for tracking the details of the insurance policy, customer details and company details.
- ❖ It is an online insurance analysis and information management system that provides easy access of information regarding the customers and resources of insurance.
- ❖ The system has complete access to the crud operations that are to create, read, update and delete the database entries containing details of customer, policies or payment methods.
- ❖ For ease of understanding, it is broadly divided into three parts:
 - User Section
 - Policy Section
 - Payment Section
- ❖ These three sections can be observed in the UML Class Diagram.

User Section

The User Section is comprised of four classes namely:

❖ *Role Class-*

This class is mainly used for defining roles within the company associated with the system. It distinguishes between the employee and the administrator and directs the user according to the role assigned to them within the system.

❖ *Admin Class-*

This class deals with the data and function associated with the administrator. The functions and data members are defined in a manner that allow access to different modules of the system based on the clearance level.

❖ *Employee Class-*

This class is responsible for managing the data of the users accessing the system as employees and directing them to the required part of the system depending on the function they perform.

❖ *Customer class-*

This class is independent of the role class as the user entering the system as customer are restricted to a pre-defined path. This class deals with the data accumulated from the customer and managing of the information provided by them.

Policy Section

The Policy Section is comprised of three classes namely:

❖ *Policy Class-*

This class is responsible for the policies within the system. It collects, stores and manages the data associated with the policies and directs the user based on the type of policy they wish to peruse.

❖ *Life Insurance Class-*

This class manages and collects the data associated with policies that come under the umbrella term: Life Insurance.

❖ *General Insurance Class-*

This class is responsible for the data associated with the policies that do not come under life insurance such as motor insurance, home insurance etc.

Payment Section

The Payment Section is comprised of three classes namely:

❖ *Payment Class-*

This class is responsible for handling the transactions within the system. It deals with the data required for smooth progression of transaction and the data members are only partially accessible to protect the user's payment information.

❖ *Premium Class-*

This class deals with the processing of premium associated with the policies chosen by a user. It possesses requisite functions for accessing and managing the premium.

❖ *Receipt Class-*

This class is responsible for keeping track of transactions and all the details associated with it. This minimizes the load of paperwork on the user as they can easily access past transactions using this class.

Actors

To understand how the system works let us assume there exists a user named X. The user X can enter the system as:

❖ *Company Associates-*

- This includes those who are part of the company that is utilizing and managing the system.
- They can enter as employee or admin; X gains the access to view and modify certain data related to policies and may oversee the transaction process being executed.
- They must undergo the registration process to be provided with the ID with which the system can identify them as company associate and then login with the same credentials.
- While the admin may oversee all the components within the system, the employee has restricted access based on their clearance level.

❖ *Customer-*

- As customer, user X can either simply view policies or register to avail the benefits provided to a registered user.

- A registered user must login with the correct credentials to enter the system. Once the user has successfully completed the login procedure they can:

- **Purchase new policy-**

The user X will be directed to a new page where they have to select the desired policy and enter the required details. After collecting the data and addition of nominees, requisite documents are uploaded by the user which are sent for verification. Once the verification process is successful, the user X can view the policy and its details.

- **View purchased policy-**

The user X will be directed to a new page which will display their policy status-whether the insurance policy is active or inactive and other associated details. The premium details will be shown as well to ensure that the user stays up to date with their payments.

- If any transaction is required, then the user may proceed to payment from where they will be directed to gateway to select mode of payment and complete the transaction.
- In case of any query, the user can click on Help and post their query. Concerned employee or admin will address it accordingly.

❖ **Bank-**

- In this case the user has access to only the transaction module and cannot access any other part of the system.
- They are linked to the system by the Customer and can only view and assist in the transaction process.

PROCEDURE

- ❖ As discussed earlier, Insurance Management System has three main actors, i.e. Admin, Employee and Customer.
- ❖ Bank is not considered as a main actor due to its restricted access to the system. Bank can only access the system when the Customer requests it.
- ❖ Users Admin and Employee have nearly identical functions with the only difference being level of access. The rest of the mechanism is similar hence while elucidating the mechanism of the system for such users, they are referred under the term of Company Associates.
- ❖ The mechanism for Company Associates has been described using the Sequence Diagram and discussed in other forthcoming diagrams.
- ❖ The mechanism for Customers has been described using Communication diagram and also elucidated in other forthcoming diagrams.
- ❖ The mechanism for these two groups have been discussed below and will be further elaborated through the diagrams.

Mechanism: Company Associate

- ❖ **Register:** The Company Associate has to undergo the registration process and enter the requisite details after which they shall receive their login credentials using which they can access the system.
- ❖ **Login:** The Company Associate enters his login credentials. Their login credentials reflect their designation within the system. Based on whether they are an Employee or an Admin, the system gives them access to selective parts of the database.
- ❖ **Forgot Password:** If the Company Associate is unable to recall their password, they may proceed to Forgot Password. From here they will be asked certain pre-mediated security questions to confirm their identity. If the answers entered by them matches with the one stored in the database, they receive a reply allowing them to reset their password.
- ❖ **Login Verification:** The login credentials entered by the Company Associate are checked by the system. If they are incorrect then the Company Associate receives a message stating 'Invalid Credentials'. In case of correct

credentials, the Company Associate is given access to the system's database.

- ❖ **Database:** From the database, the Company Associate can choose to access or perform operations related to policy data or customer data. Depending on their clearance level and designation, they are given access to parts of the system. Employees can access only the policy database while the Admin can access the Customer database as well.
- ❖ **Policy Verification:** The Company Associate can perform operations related to policy such as addition, upgradation, deletion etc.
- ❖ **Customer Data Verification:** The Company Associate can verify the data entered by the customer and add, remove or update as per his discretion. This function is only available to the Admin.
- ❖ **Help:** The queries sent by customers is received by the Company Associate who is responsible for managing customer queries that is the Admin.

Mechanism: Customer

- ❖ **Register/Login:** If the Customer has not registered, he can only view policies offered within the system. If the Customer registers with the system, they can purchase policies and store keep track of their policies. When the Customer registers, they shall receive their login credentials using which they can access the system.
- ❖ **Forgot Password:** If the Customer is unable to recall their password, they may proceed to Forgot Password. From here they will be asked certain pre-mediated security questions to confirm their identity. If the answers entered by them matches with the one stored in the **Database**, they receive a reply allowing them to reset their password.
- ❖ **Login Verification:** The login credentials entered by the Customer are checked by the system. If they are incorrect then the Customer receives a message stating 'Invalid Details-Try Again'. In case of correct credentials, the Customer can proceed to **policy** section.
- ❖ **Policy:** The customer can view the policies offered within the system and decide whether they want to view **Policy Details** of previously purchased policies or purchase a new one.

- ❖ **Policy Details:** In this section the Customer can view the terms, conditions and premium details of any policy. If they don't want to make any payments, they are directed to the **Logout** page so that they can exit the system,
- ❖ **Payment:** If the Customer wishes to make any monetary transactions, they are directed to the payment section. In this section they are required to choose the mode of payment and enter the requisite details after which they are directed to **payment status** where they can receive the receipt and check if the transaction was completed or not.
- ❖ **Payment Status:** In case of successful transaction, the Customer is directed to the **Logout** page and a message is displayed stating 'Payment Successful'. In case of failed transaction due to any error, the Customer is directed back to payment page with a message stating 'Payment Failed-Try Again'.
- ❖ **Help:** The queries sent by the Customer is received by the Admin who is responsible for managing customer queries. The Customer receives the solution to their queries in this section.

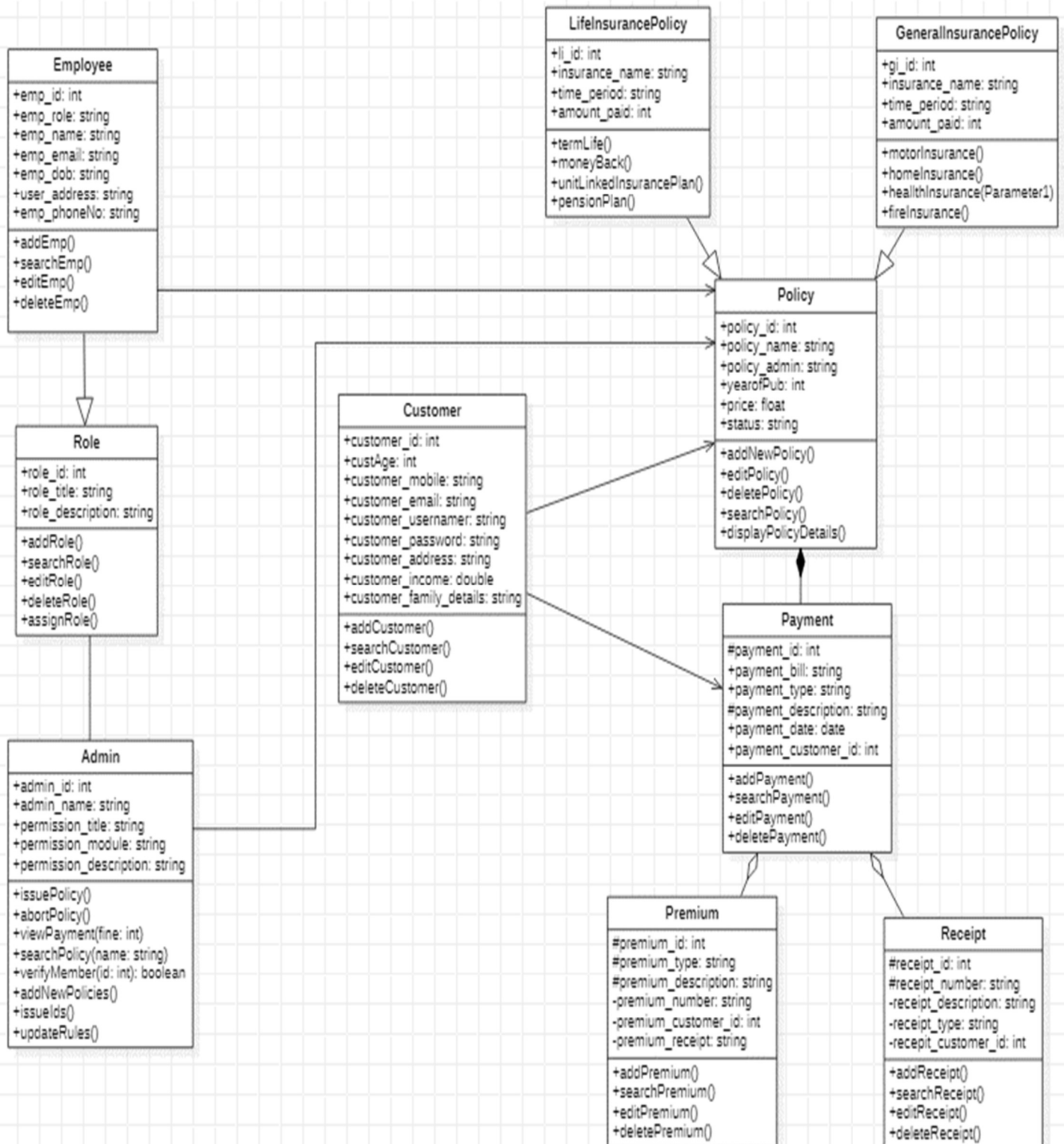
SECTION 2:

DIAGRAMS

In this Section, using UML diagrams we will be discussing:

- Construction of the system.
- Structural nature of the system.
- Behavioral nature of the system.
- Linking between different sections.

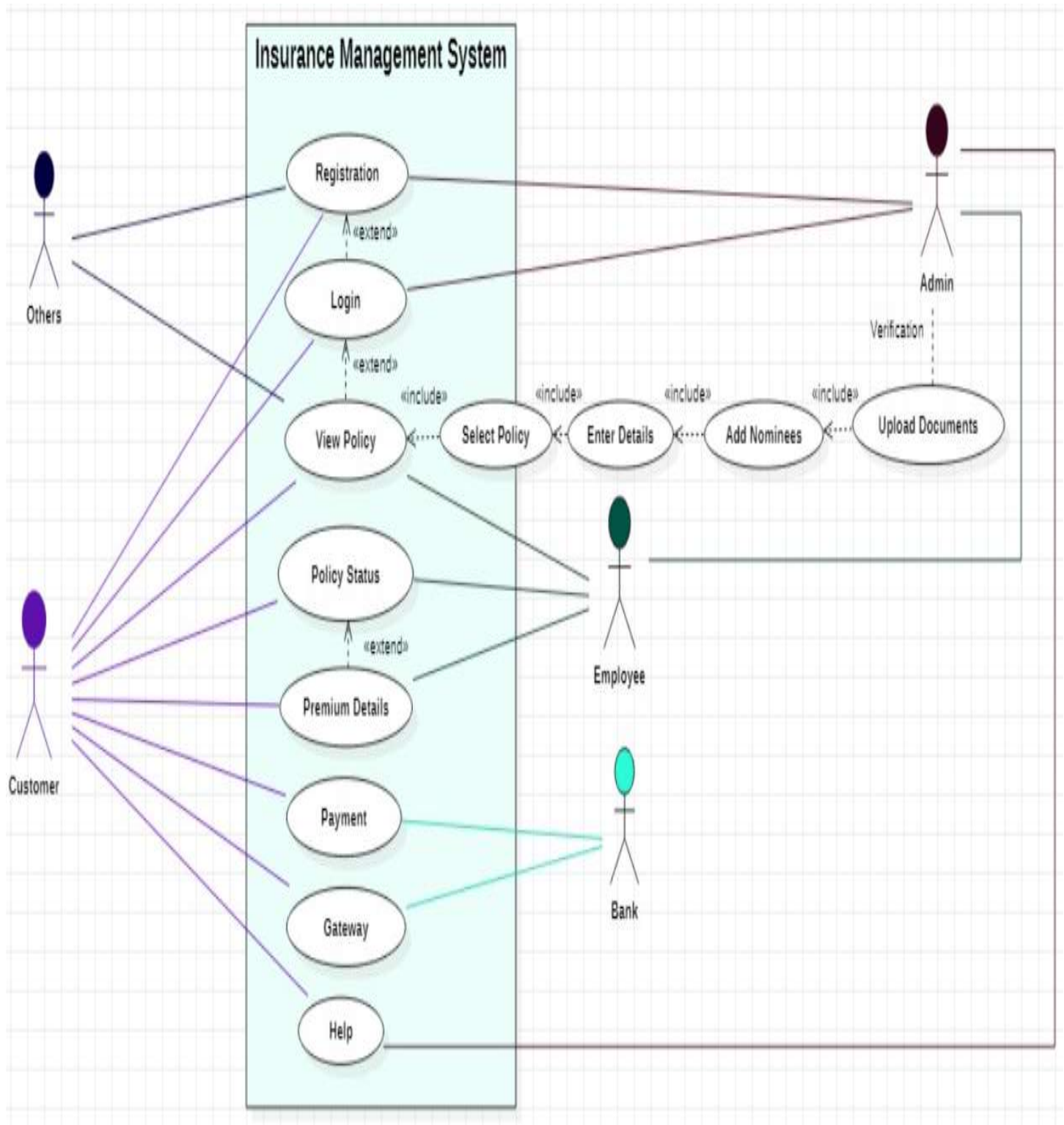
UML CLASS DIAGRAM



Definition: UML Class Diagram

- ❖ A UML class diagram is a diagram based on the UML (Unified Modeling Language) with the purpose of visually representing a system along with its main actors, roles, actions, artifacts or classes, in order to better understand, alter, maintain, or document information about the system.
- ❖ They provide both a more standardized way of modeling workflows as well as a wider range of features to improve readability and efficacy.
- ❖ A UML Class diagram is made up of a set of classes and the relationship between those classes.
- ❖ The purpose of UML Class diagrams is:
 - Shows static structure of classifiers in a system
 - Diagram provides a basic notation for other structure diagrams prescribed by UML
 - Helpful for developers and other team members too
 - Business Analysts can use class diagrams to model systems from a business perspective

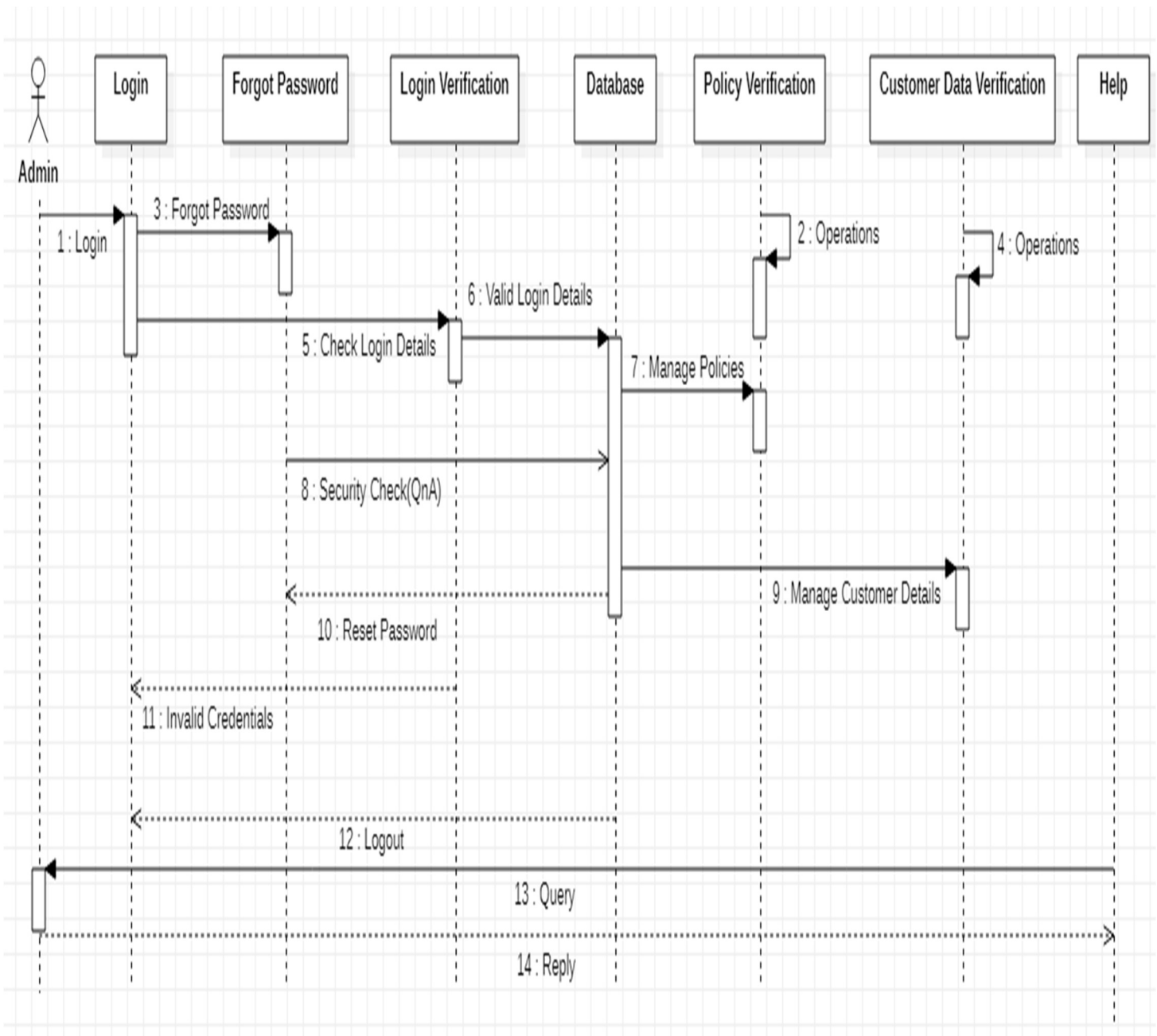
USE CASE DIAGRAM



Definition: Use Case Diagram

- ❖ A use case diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved.
- ❖ Use case diagrams are used to gather the requirements of a system including internal and external influences. These requirements are mostly design requirements. Hence, when a system is analyzed to gather its functionalities, use cases are prepared and actors are identified
- ❖ A key concept of use case modeling is that it helps us design a system from the end user's perspective. It is an effective technique for communicating system behavior in the user's terms by specifying all externally visible system behavior.
- ❖ A use case diagram is usually simple. It does not show the detail of the use cases. It only summarizes some of the relationships between use cases, actors, and systems. It does not show the order in which steps are performed to achieve the goals of each use case.

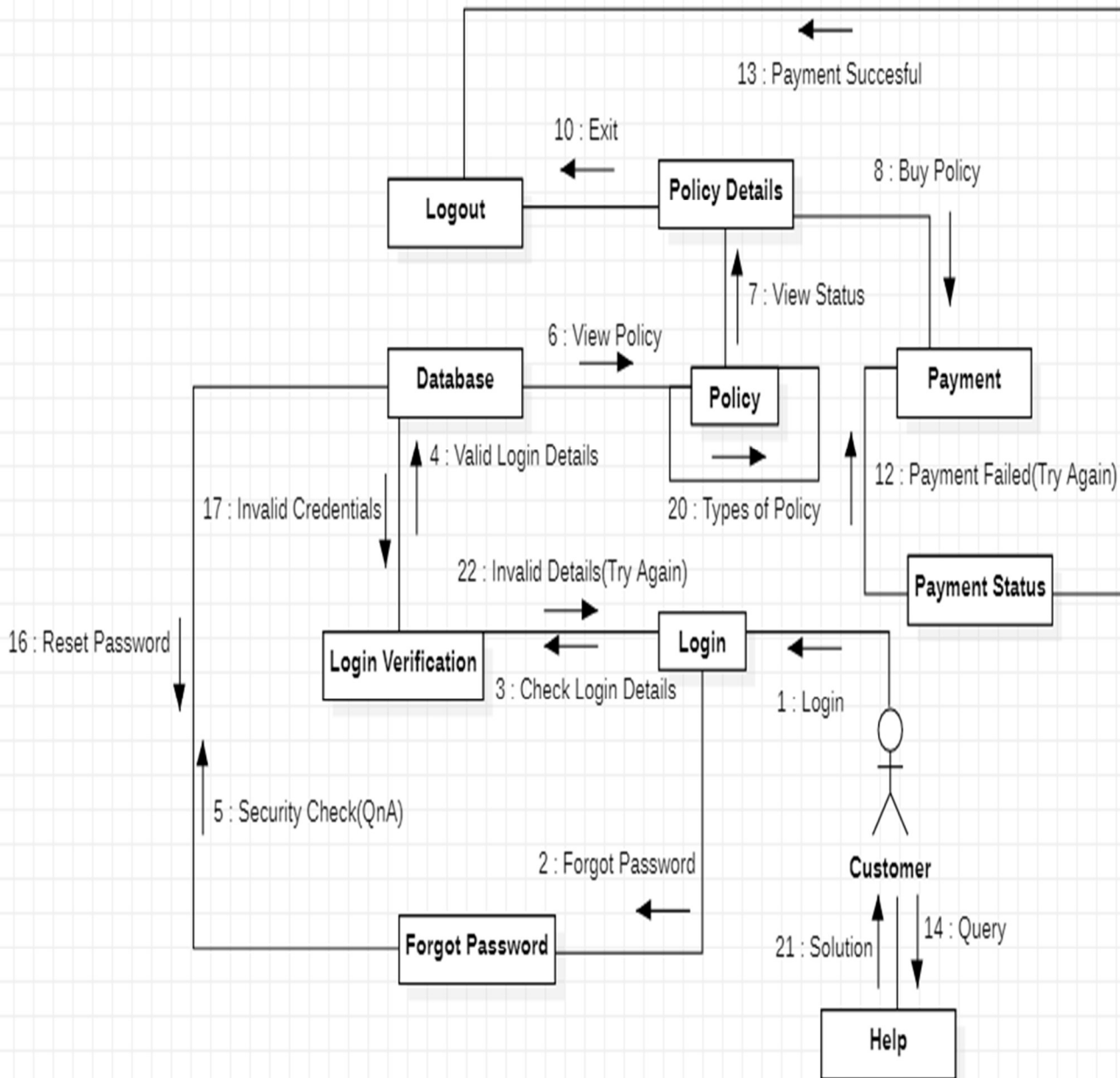
SEQUENCE DIAGRAM



Definition: Sequence Diagram

- ❖ The sequence diagram represents the flow of messages in the system and is also termed as an event diagram. It helps in envisioning several dynamic scenarios.
- ❖ It portrays the communication between any two lifelines as a time-ordered sequence of events, such that these lifelines took part at the run time. It incorporates the iterations as well as branching
- ❖ UML Sequence Diagrams are interaction diagrams that detail how operations are carried out. They capture the interaction between objects in the context of a collaboration.
- ❖ Sequence diagrams are time focus and they show the order of the interaction visually by using the vertical axis of the diagram to represent the time at which messages are sent.
- ❖ Sequence diagram captures:
 - The interaction that takes place in a collaboration that either realizes a use case or an operation (instance diagrams or generic diagrams)
 - High-level interactions between user of the system and the system, between the system and other systems, or between sub-systems known as system sequence diagrams.

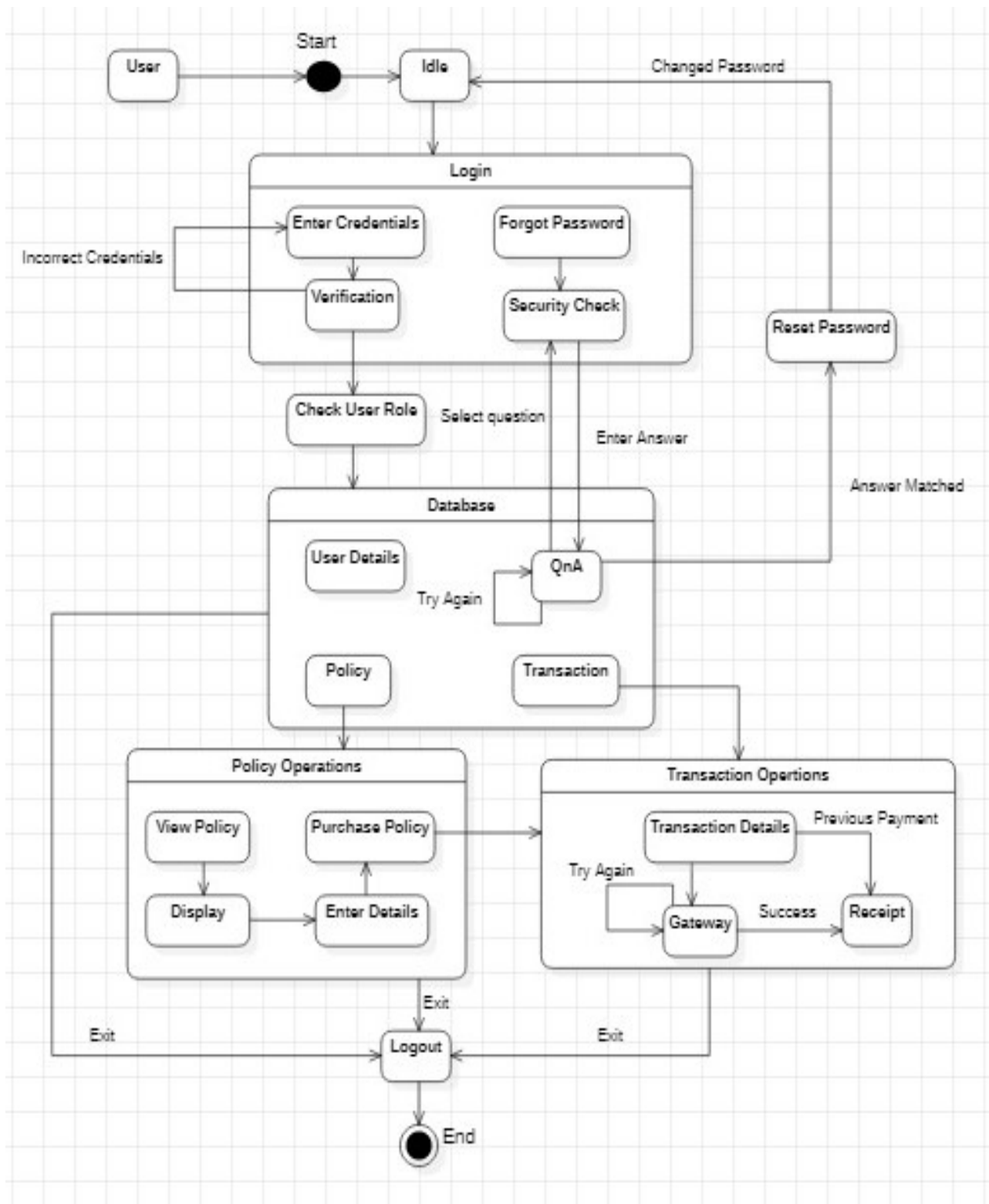
COLLABORATION DIAGRAM



Definition: Collaboration Diagram

- ❖ The collaboration diagram is used to show the relationship between the objects in a system.
- ❖ Both the sequence and the collaboration diagrams represent the same information but differently. Instead of showing the flow of messages, it depicts the architecture of the object residing in the system as it is based on object-oriented programming.
- ❖ The collaboration diagram, is also known as a communication diagram.
- ❖ Each message constitutes a sequence number, such that the top-level message is marked as one and so on.
- ❖ It is used in the following scenario: -
 - To model collaboration among the objects or roles that carry the functionalities of use cases and operations.
 - To model different scenarios within the use case or operation, involving a collaboration of several objects and interactions.
 - To support the identification of objects participating in the use case.

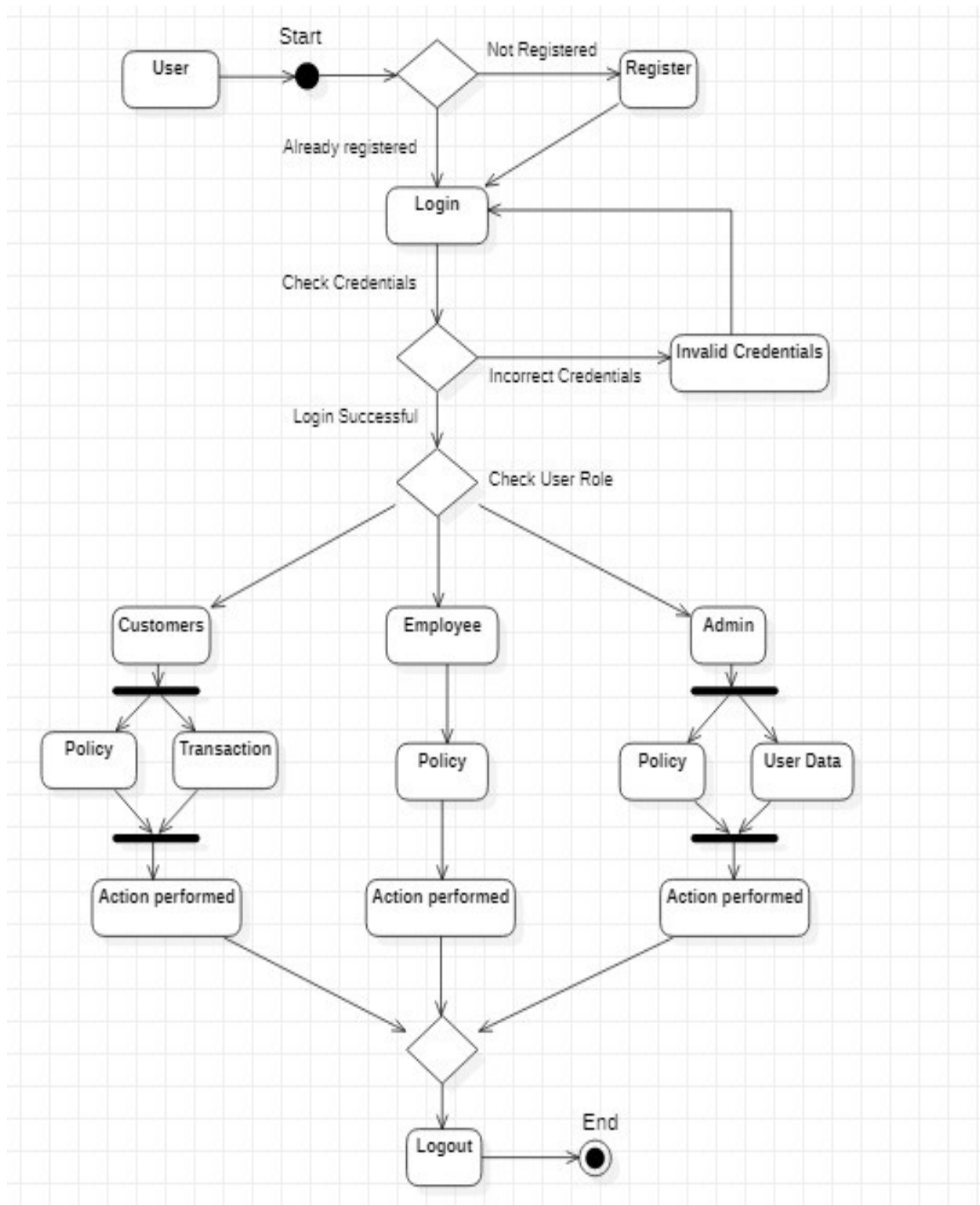
STATE CHART DIAGRAM



Definition: State Chart Diagram

- ❖ A state diagram is used to represent the condition of the system or part of the system at finite instances of time.
- ❖ It's a behavioral diagram and it represents the behavior using finite state transitions. It is also referred to as State-machine Diagrams or State-chart Diagrams.
- ❖ State diagrams are usually applied to objects but can be applied to any element that has behavior to other entities such as: actors, use cases, methods, subsystems systems and etc.
- ❖ A state diagram is used to model the dynamic behavior of a class in response to time and changing external stimuli.
- ❖ Uses of state chart diagram -
 - We use it to state the events responsible for change in state (we do not show what processes cause those events).
 - We use it to model the dynamic behavior of the system.
 - To understand the reaction of objects/classes to internal or external stimuli.

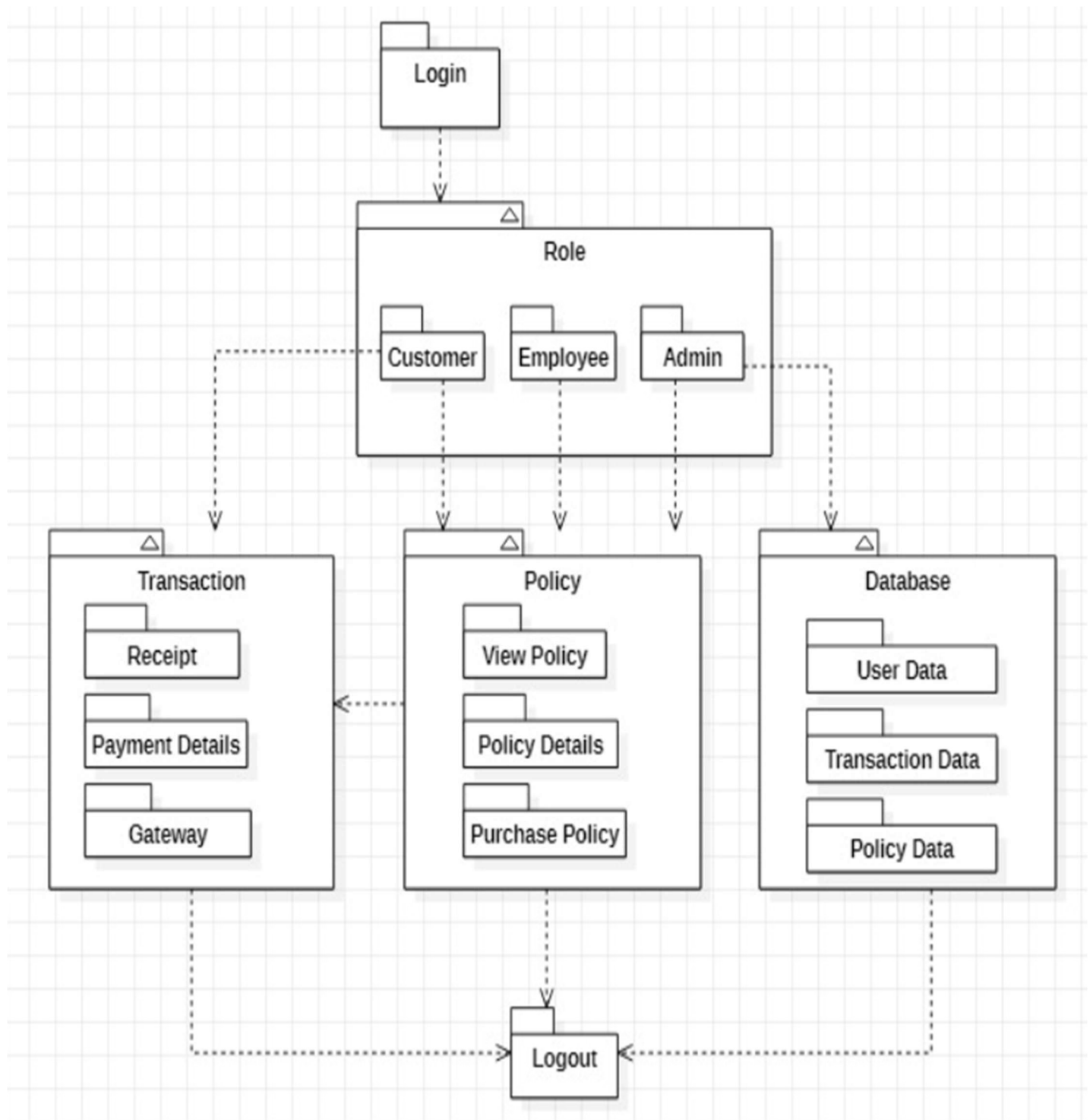
ACTIVITY DIAGRAM



Definition: Activity Diagram

- ❖ Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system.
- ❖ The control flow is drawn from one operation to another. This flow can be sequential, branched, or concurrent. Activity diagrams deal with all type of flow control by using different elements such as fork, join etc.
- ❖ Activity diagrams are not only used for visualizing the dynamic nature of a system, but are also used to construct the executable system by using forward and reverse engineering techniques.
- ❖ An activity diagram portrays the control flow from a start point to a finish point showing the various decision paths that exist while the activity is being executed.
- ❖ We can depict both sequential processing and concurrent processing of activities using an activity diagram. They are used in business and process modelling where their primary use is to depict the dynamic aspects of a system.

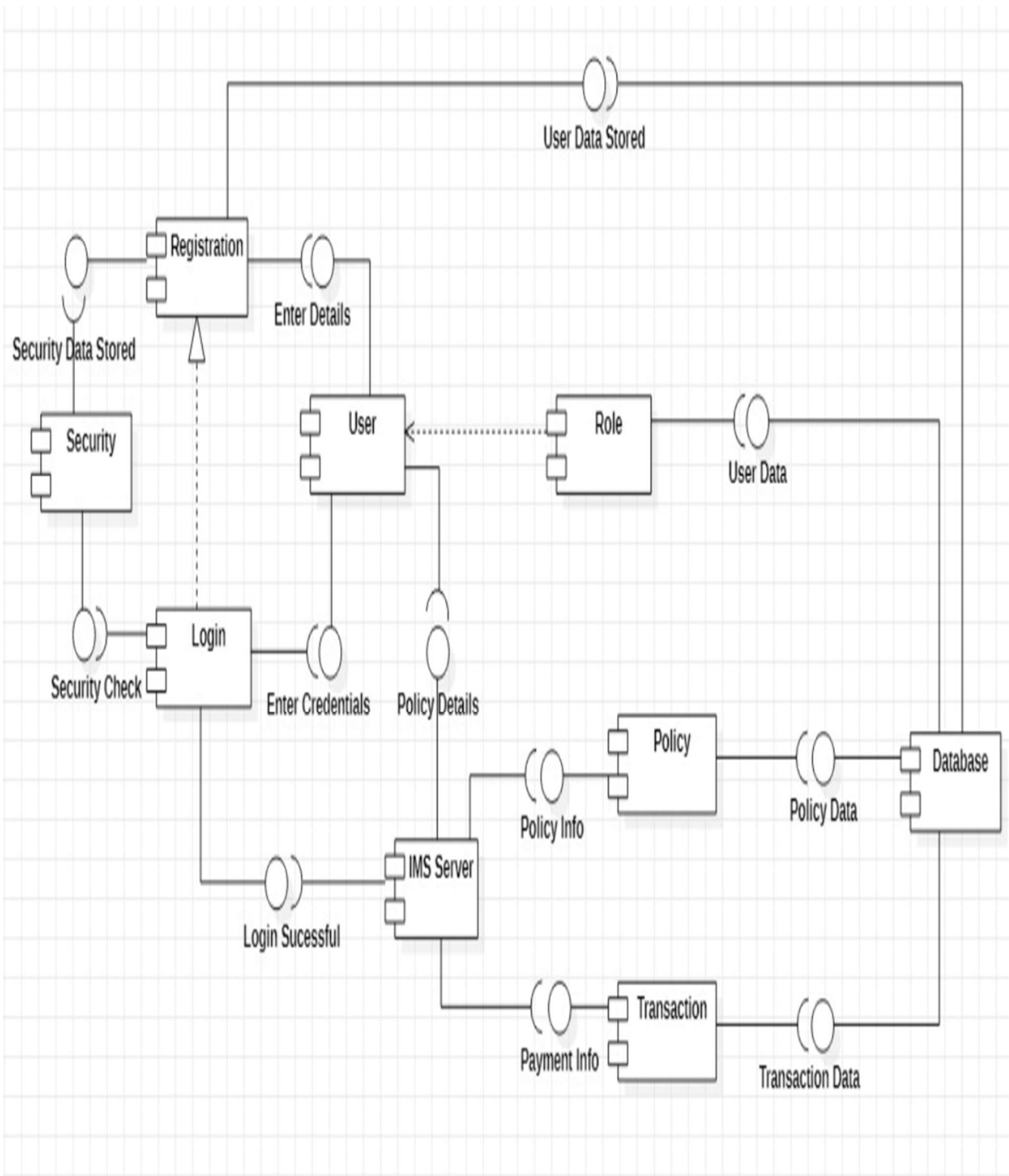
PACKAGE DIAGRAM



Definition: Package Diagram

- ❖ Package diagram, a kind of structural diagram, shows the arrangement and organization of model elements in a project. Package diagram can show both structure and dependencies between sub-systems or modules, showing different views of a system,
- ❖ A package is a collection of logically related UML elements. Packages appear as rectangles with small tabs at the top. The package name is on the tab or inside the rectangle. The dotted arrows are dependencies.
- ❖ One package depends on another if changes in the other could possibly force changes in the first.
- ❖ Package diagrams are used to
 - Structure high level system elements.
 - Organizing large system which contains diagrams, documents and other key deliverables.
 - Simplify complex class diagrams, it can group classes into packages.

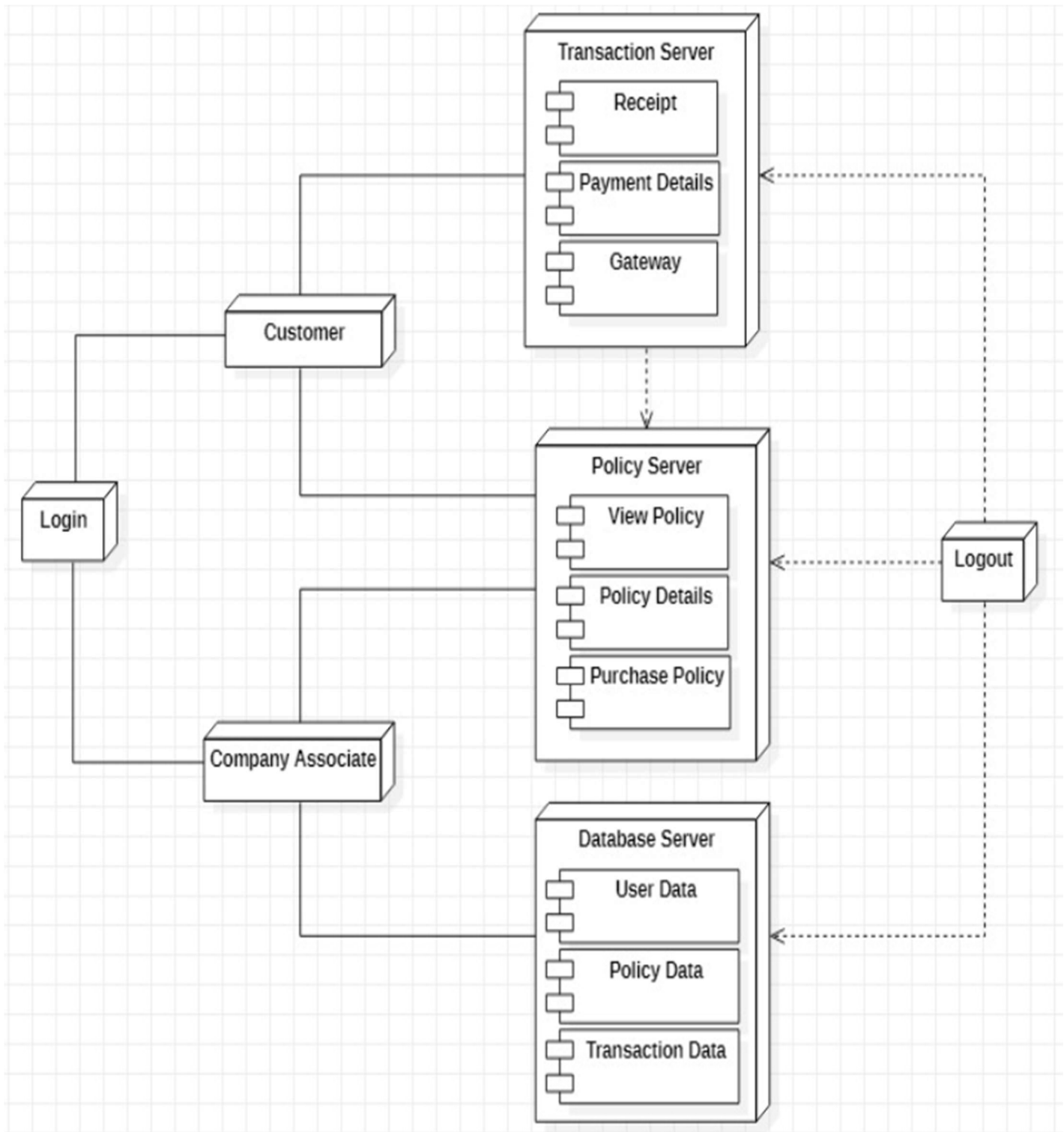
COMPONENT DIAGRAM



Definition: Component Diagram

- ❖ Component diagrams are used in modeling the physical aspects of object-oriented systems that are used for visualizing, specifying, and documenting component-based systems and also for constructing executable systems through forward and reverse engineering.
- ❖ Component diagrams are essentially class diagrams that focus on a system's components that often used to model the static implementation view of a system.
- ❖ Component diagram depicts how components are wired together to form larger components or software systems. They are used to illustrate the structure of arbitrarily complex systems.
- ❖ Component diagram is a special kind of diagram in UML. It describes the components used to make those functionalities. Component diagrams are used to visualize the physical components in a system. These components are libraries, packages, files, etc.

DEPLOYMENT DIAGRAM



Definition: Deployment Diagram

- ❖ Deployment diagram is a diagram that shows the configuration of run time processing nodes and the components that live on them.
- ❖ Deployment diagram is a kind of structure diagram used in modeling the physical aspects of an object-oriented system. They are often used to model the static deployment view of a system (topology of the hardware).
- ❖ Deployment diagram show the structure of the run-time system. They capture the hardware that will be used to implement the system and the links between different items of hardware.
- ❖ Deployment diagram model physical hardware elements and the communication paths between them. They can be used to plan the architecture of a system.
- ❖ Component diagrams are used to describe the components and deployment diagrams shows how they are deployed in hardware. These two diagrams are special diagrams used to focus on software and hardware components.

CONCLUSION

- ❖ Insurance Management System is an integral mode of approach that should be adopted to make the process of purchasing and managing insurances simple and effective.
- ❖ With this system, the load of paperwork will also be reduced and there shall be less loss of data. This would lead to optimum utilization of resources that benefits all the parties involved in the entire process.
- ❖ The structural and behavioral nature of the Insurance Management System is studied using different types of UML diagrams.
- ❖ The diagrams depicted above explain in detail the construction of the system and its working. The relationship between different parts of the system and their function is expressed using these diagrams.
- ❖ Using pictorial representation, the interaction between all Actors and the system and the interaction of individual Actors with the system is also discussed.

THANK

YOU