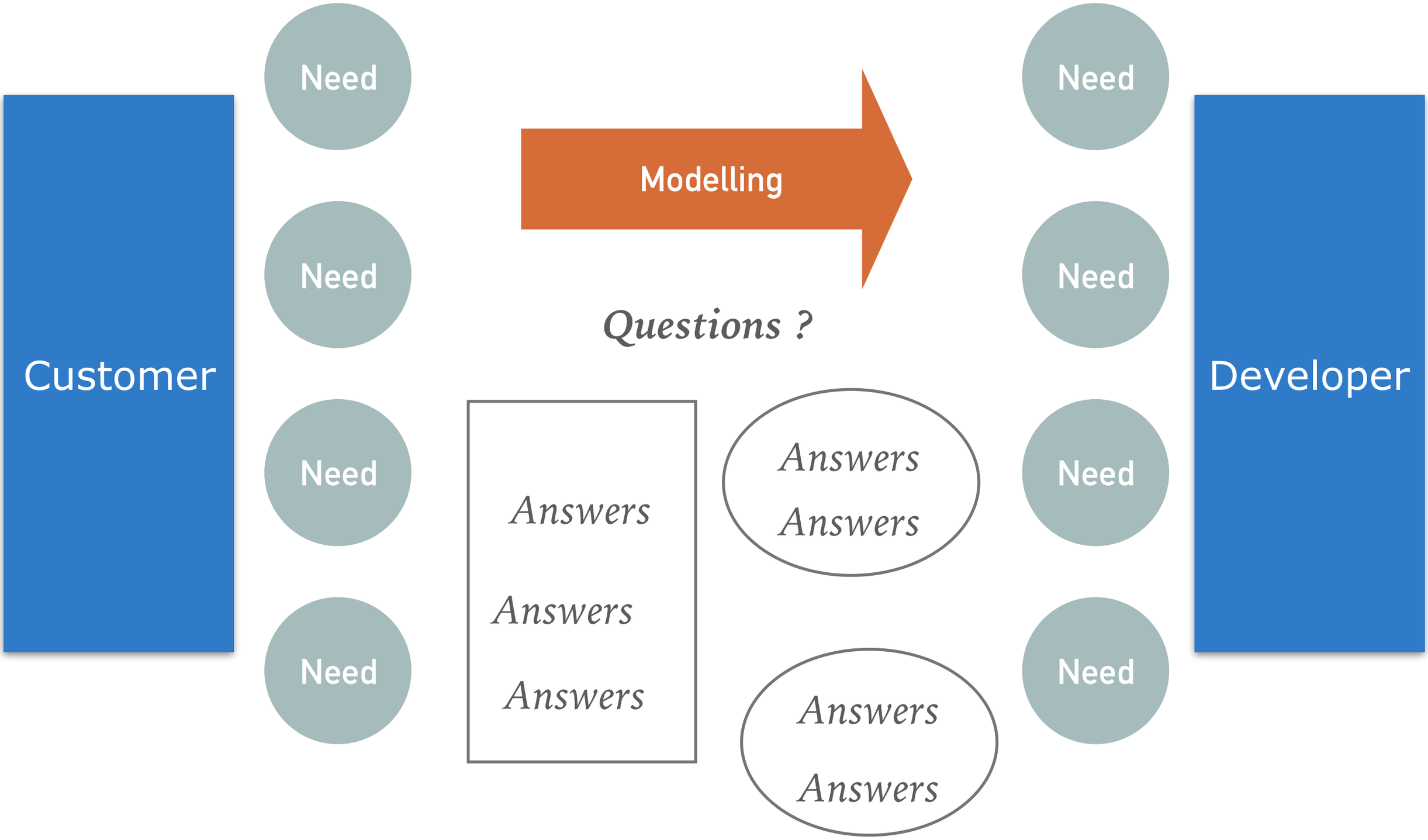
MODELLING II

Software Design Approach

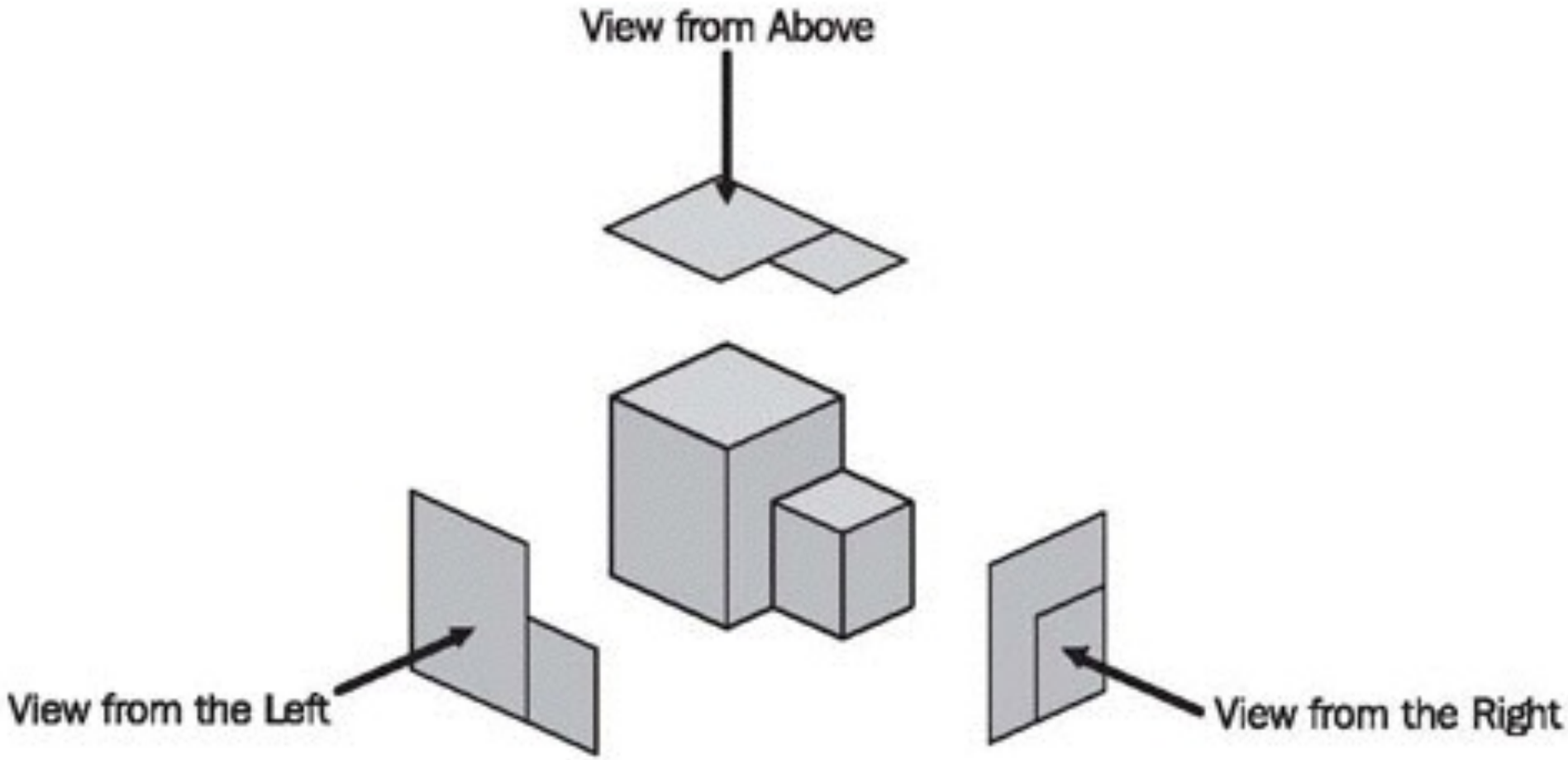
AGENDA

- Big Picture
- Project Documents
- Intro to UML

BIG PICTURE


Need
Need
Need
Need
Need
Need
Modelling
Questions ?
Customer
Developer
Answers
Answers
Answers
Answers
Answers
Answers

VIEW VS MODEL

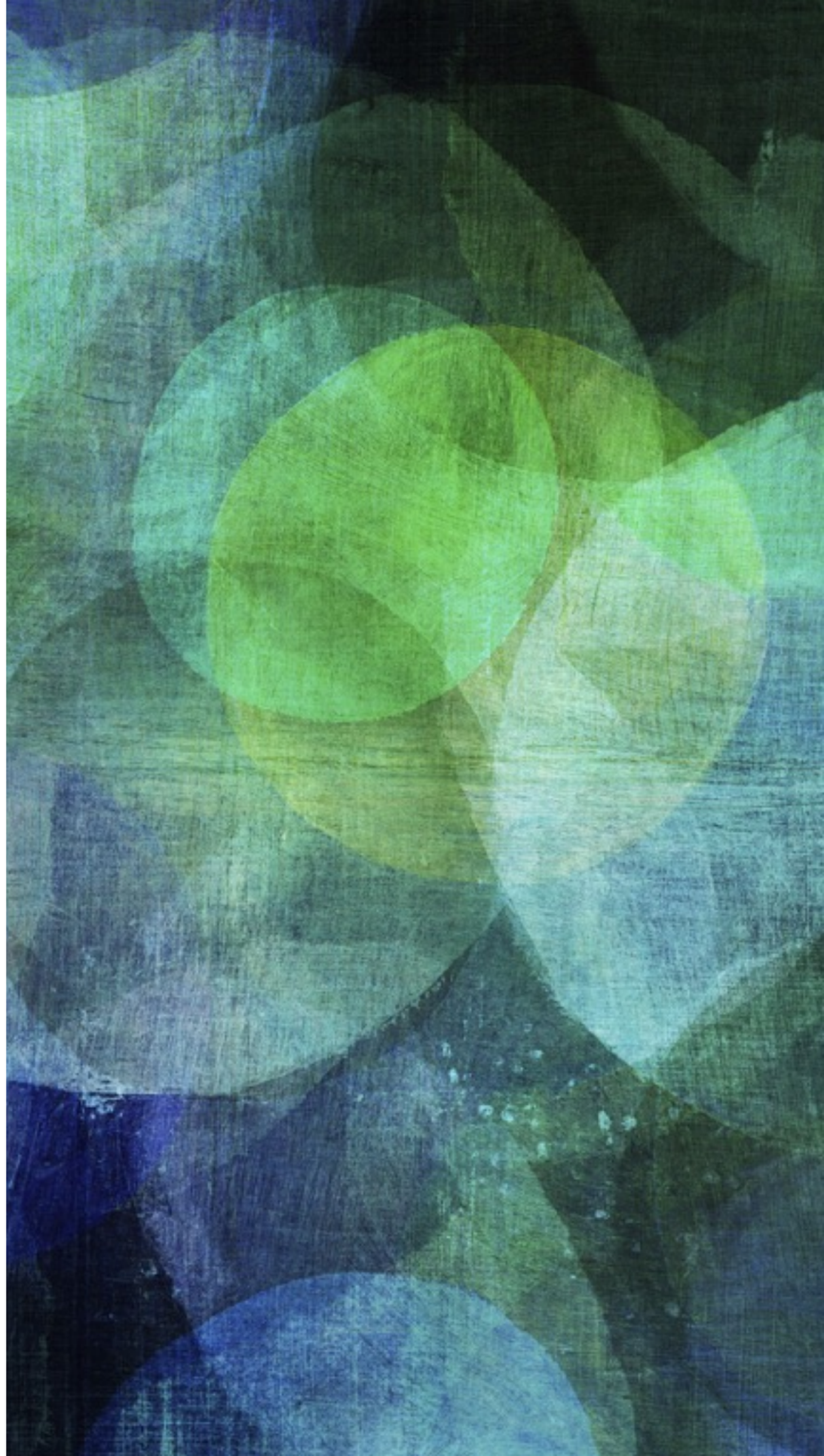


DOCUMENTS

Business Oriented

VISION DOCUMENT

Documents



VISION DOCUMENT – WHAT

It is a very basic document which is built initially to present the project for the first time

VISION DOCUMENT – WHY ?

Who needs this system ? and Why ?

What problems that is solve ? and How ?

What happens when problem is solved ?

What is the system in lines ?

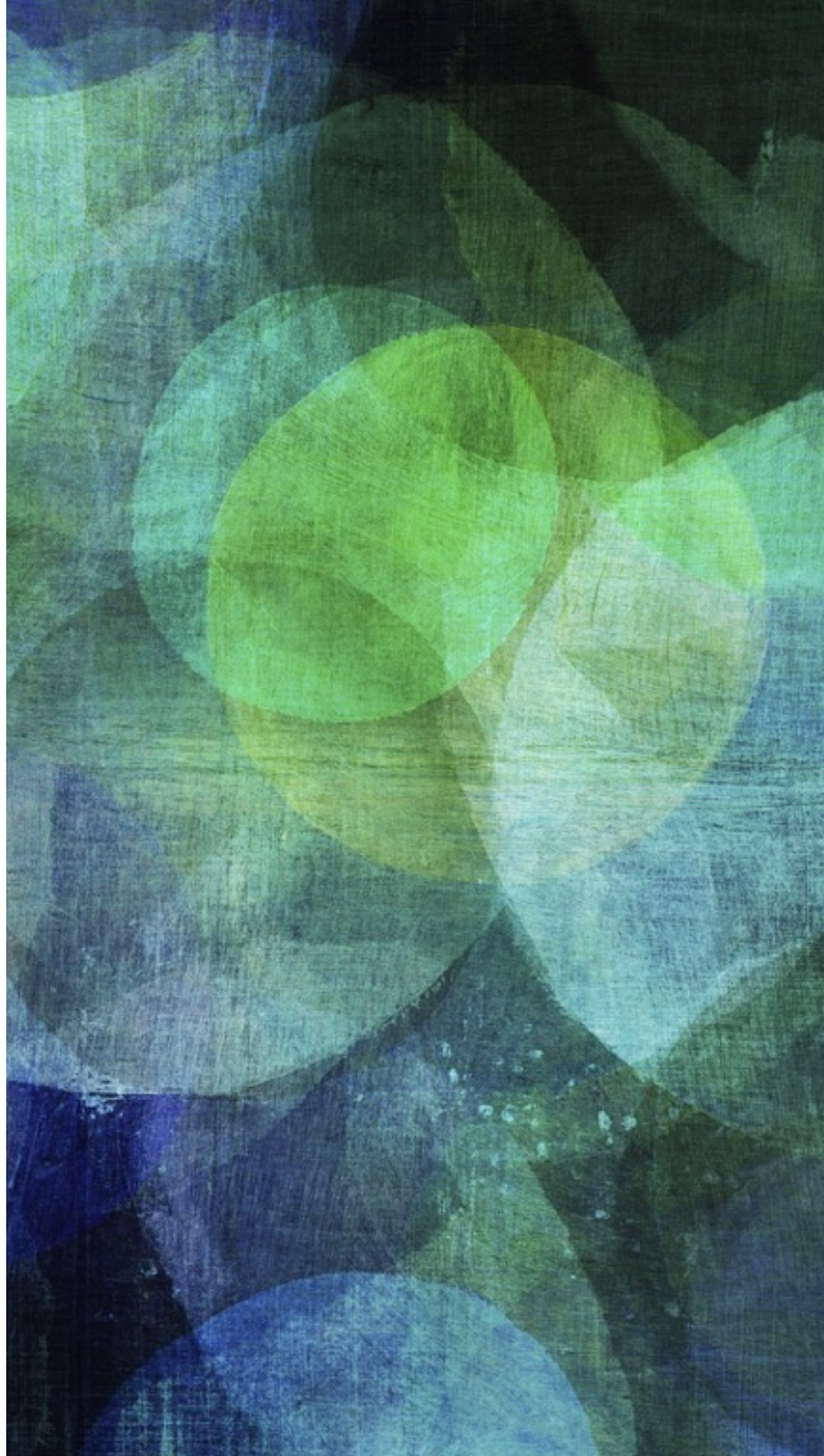
Who would use the system ?

What are the main functions of the system ?

Business Oriented

STACK HOLDER REGISTER

Documents



STAKE HOLDER REGISTER – WHAT ?

It keeps involved persons with their roles and contacts

STAKE HOLDER REGISTER – WHY ?

Who is involved ?

Who to reach him/her ?

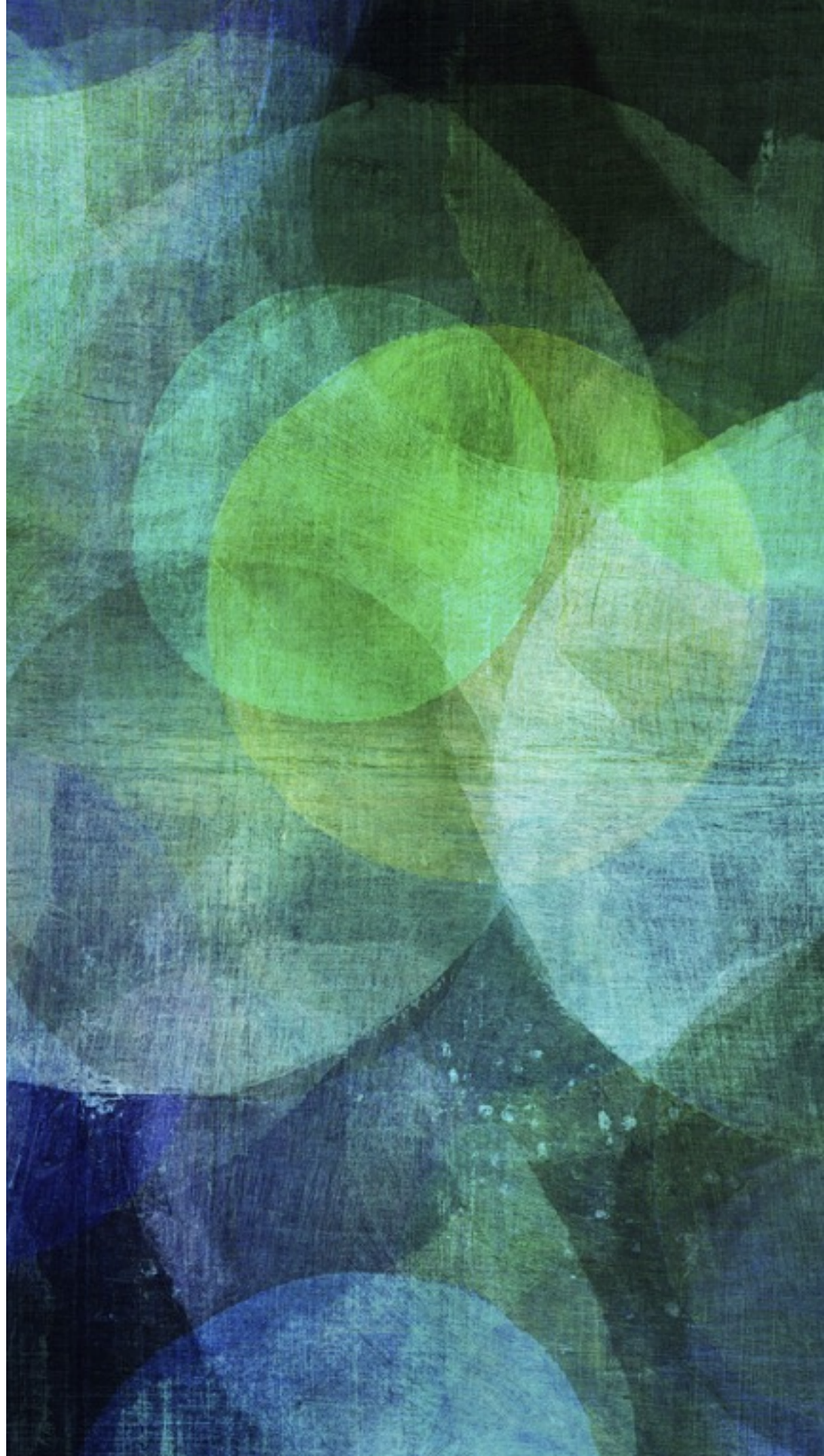
What is he/she responsible for ?

What is the sequence of communication ?

Technical Oriented

SOFTWARE REQUIREMENTS SPECIFICATIONS

Documents



SOFTWARE REQUIREMENTS SPECIFICATIONS – WHAT ?

It is used to maintain System requirements (Functional and Non Functional)

Use Cases / User Story in details

SOFTWARE REQUIREMENTS SPECIFICATIONS – WHAT ?

What is the system ?

What are technologies/platforms/3rd party service used ?

What are the user stories ?

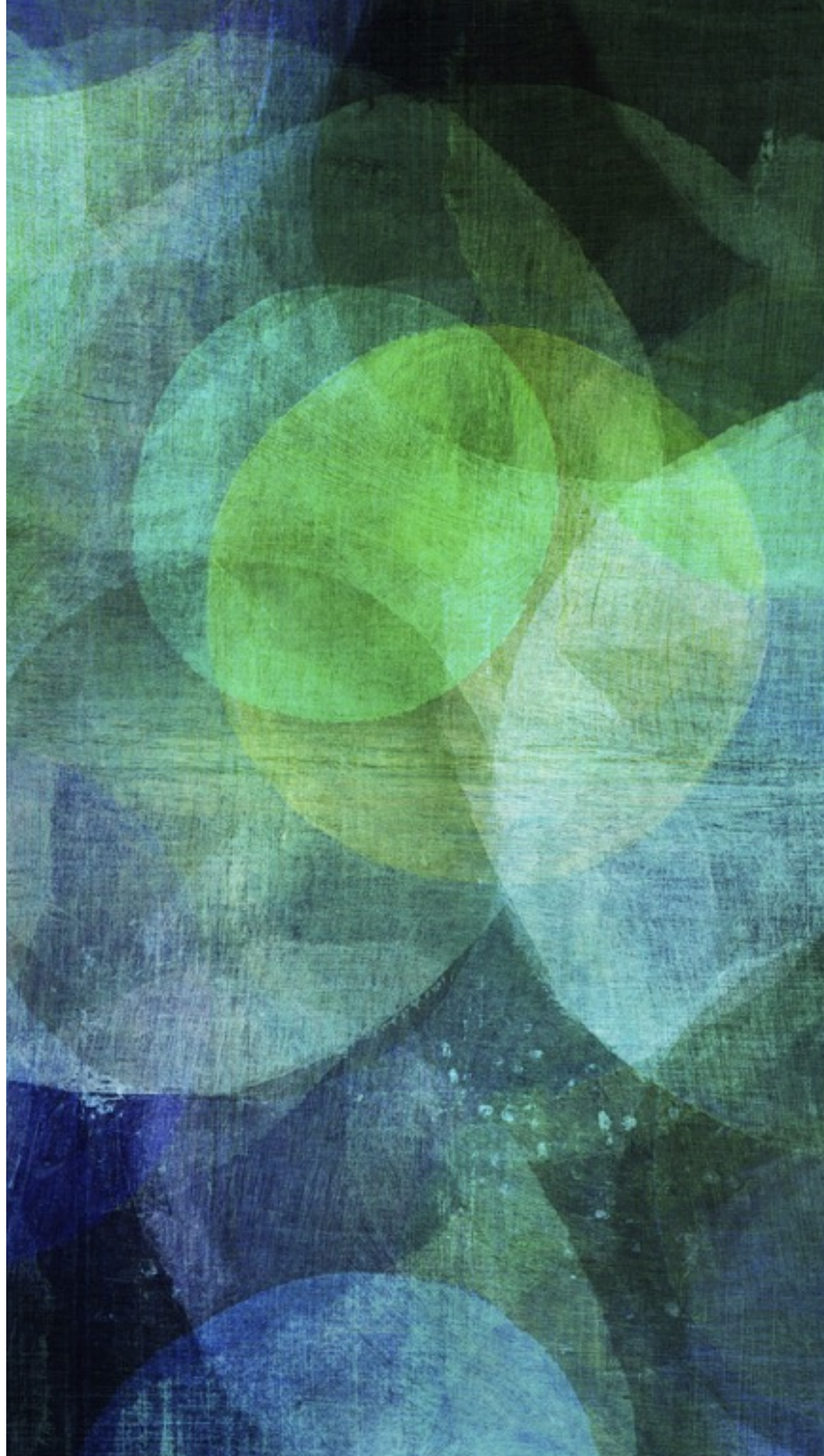
What is each user story in details ? who needs it ? how it would be implemented

What are the non-functional requirements

Technical Oriented

ANALYSIS & DESIGN

Documents



ANALYSIS AND DESIGN – WHAT ?

Basic UML diagrams

ANALYSIS AND DESIGN – WHAT ?

What are the main Players ? [Entity Diagram]

What are the main Processes ? [Interaction/Sequence diagram]

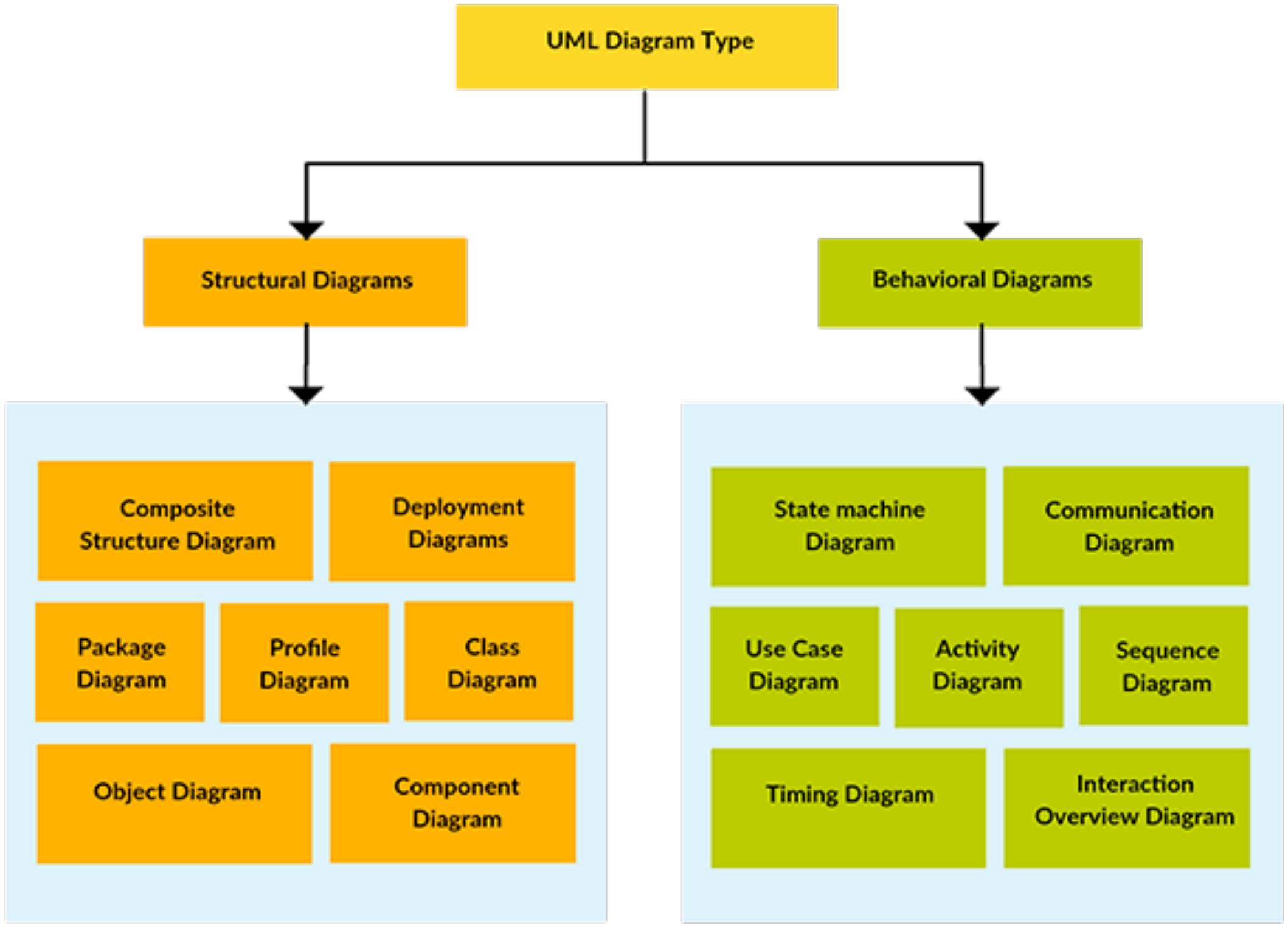
What are the main system states ? [Machine State diagram]

What are the main Classes ? [Class Diagram]

What are the main Tables ? [ERD]

INTRO TO UML

UML


UML Diagram Type
Structural Diagrams
Behavioral Diagrams
Composite Structure Diagram
Deployment Diagrams
Package Diagram
Profile Diagram
Class Diagram
Object Diagram
Component Diagram
State machine Diagram
Communication Diagram
Use Case Diagram
Activity Diagram
Sequence Diagram
Timing Diagram
Interaction Overview Diagram

BASIC DIAGRAMS

- Entity Diagram
- Use Cases - User Stories
- State Machine Diagram
- Activity Diagram
- Sequence Diagram
- Class Diagram

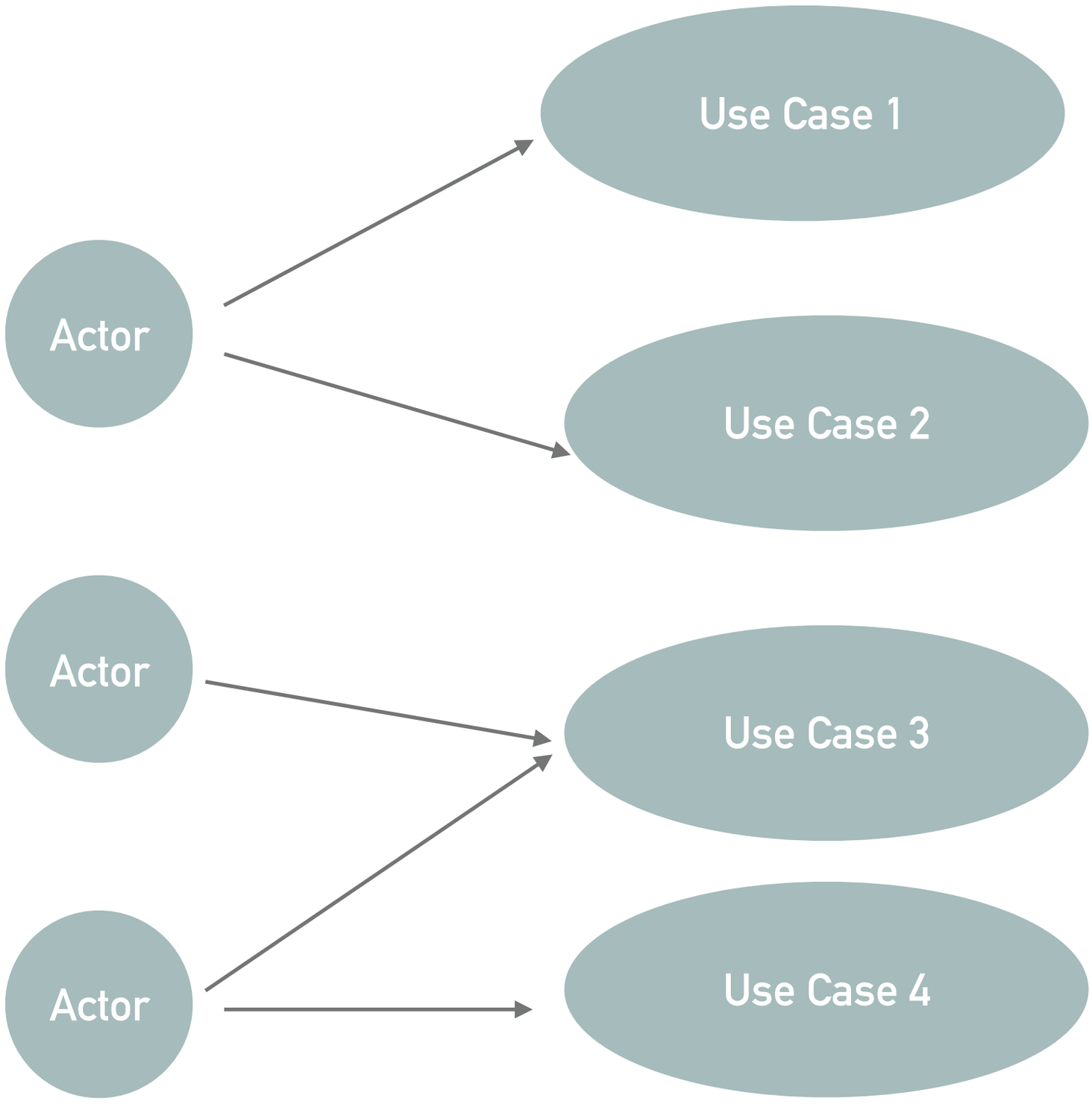
ENTITY DIAGRAM

Entity Name	Entity Name	Entity Name
Property Property Property	Property Property Property	Property Property Property

USE CASE DIAGRAM

Use case diagrams give a graphic overview of the actors involved in a system, different functions needed by those actors and how these different functions are interacted.

USE CASES DIAGRAM



LIST OF USER STORIES

It is an alternative way to describe: Actor <––> System function relationships.
It is used among Agile development environments.

USER STORY

<Story Name>

User <user role>

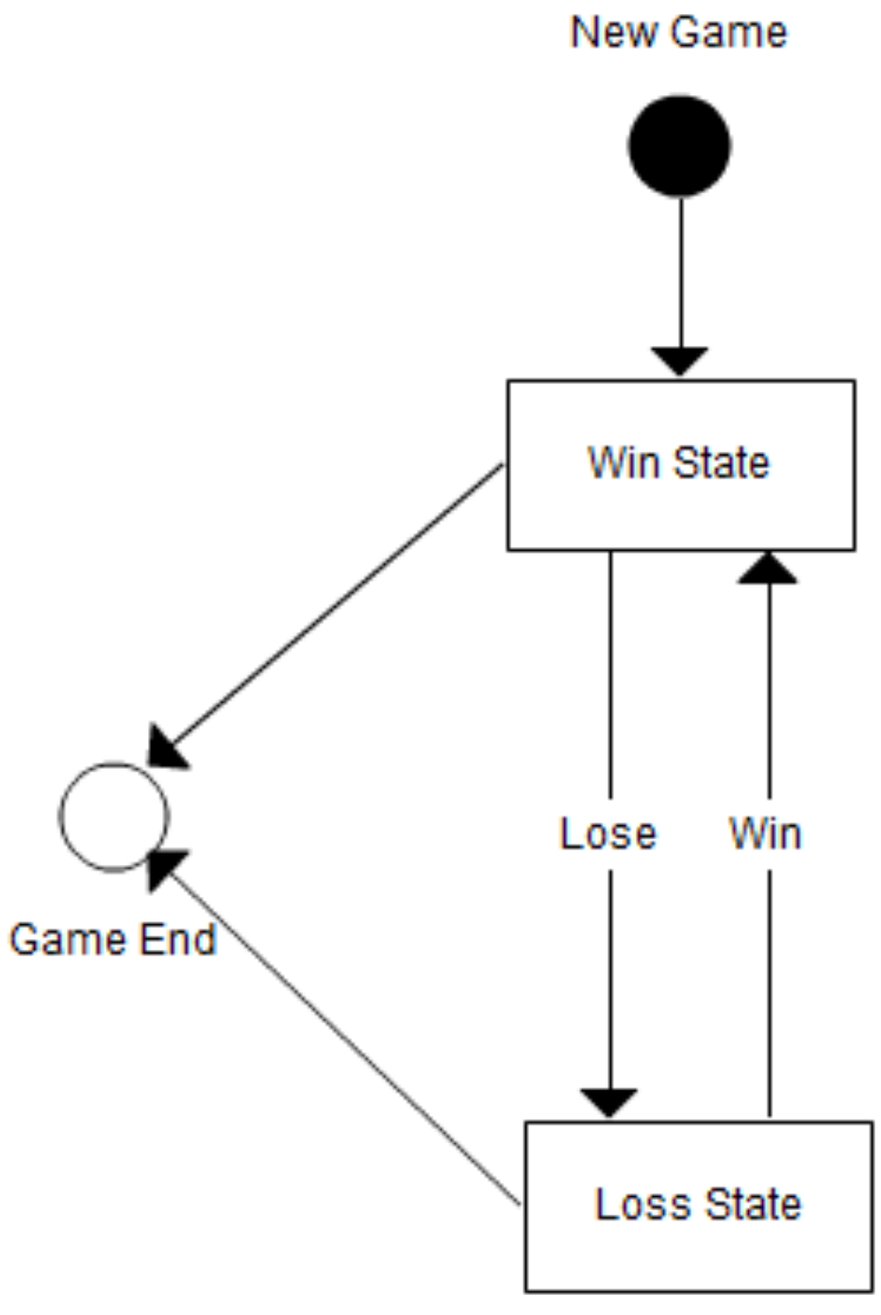
should be able to <action>

With <action details/acceptance criteria>

STATE MACHINE DIAGRAM

State machine diagrams are similar to activity diagrams, although notations and usage change a bit. They are sometime known as state diagrams or state chart diagrams as well. These are very useful to describe the behaviour of objects that act differently according to the state they are in at the moment

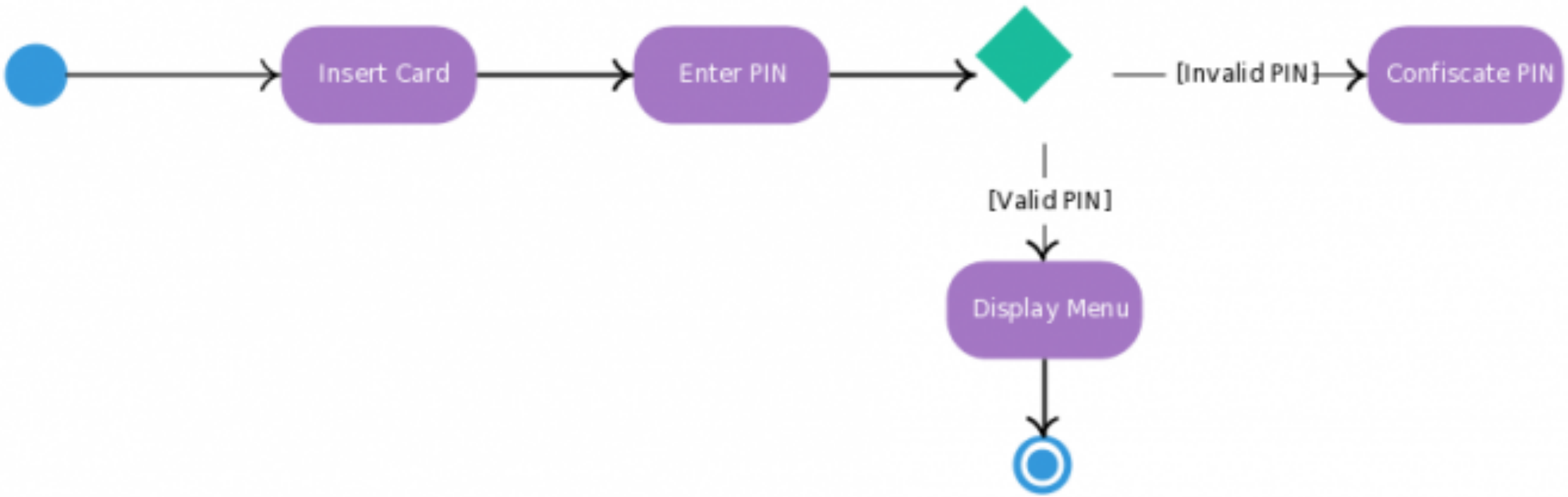
STATE MACHINE DIAGRAM



ACTIVITY DIAGRAM

Activity diagrams represent workflows in a graphical way. They can be used to describe business workflow or the operational workflow of any component in a system. Sometimes activity diagrams are used as an alternative to State machine diagrams.

ACTIVITY DIAGRAM – USE CASE DETAILS

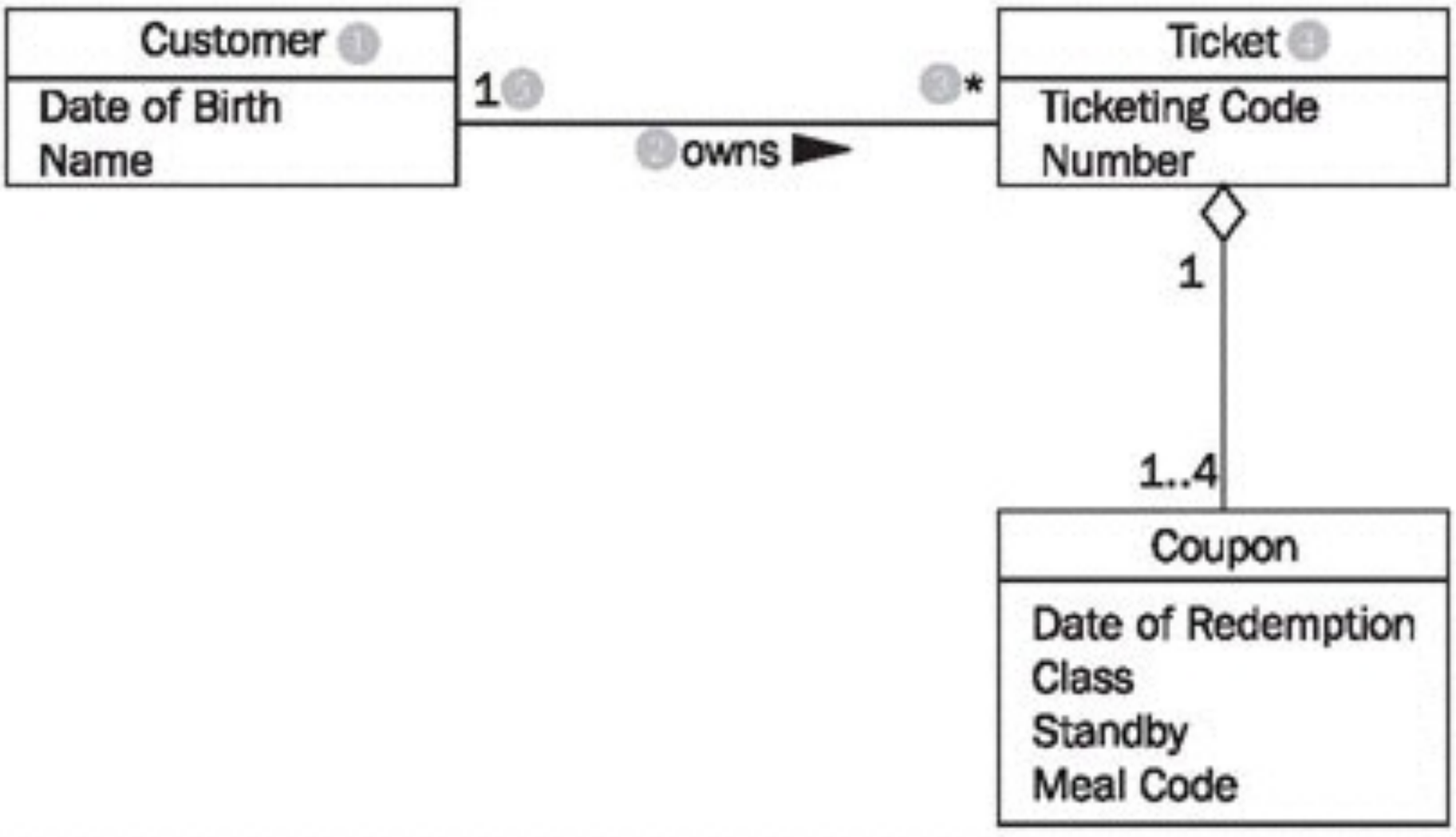


CLASS DIAGRAM

It is the main building block of any object oriented solution. It shows the classes in a system, attributes and operations of each class and the relationship between each class.

In most modelling tools, a class has three parts, name at the top, attributes in the middle and operations or methods at the bottom. In large systems with many related classes, classes are grouped together to create class diagrams

CLASS DIAGRAM



SEQUENCE DIAGRAM

Sequence diagrams in UML show how objects interact with each other and the order those interactions occur. It's important to note that they show the interactions for a particular scenario. The processes are represented vertically and interactions are show as arrows.

SEQUENCE DIAGRAM