

my companion

## Module - 5

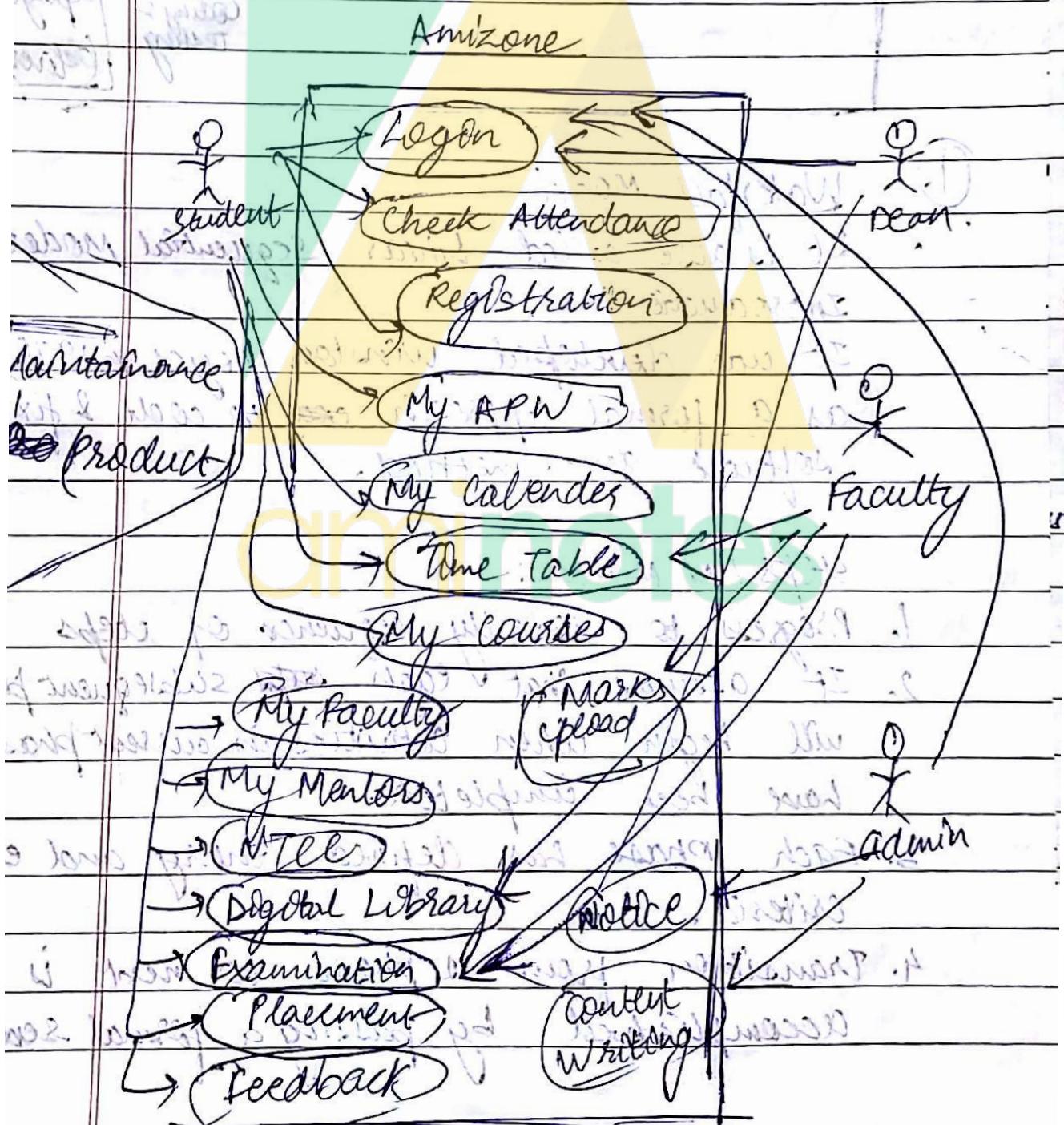
UML → Unified Modeling Language

1) Use Case Diagrams

→ Actor (User)

→ Use Cases (functionalities)

Q = Draw a use case diagram for Amizone system or any Banking system



## UML - Unified Modeling Language

It define a standard way to visualize the way a system has been designed. It is quite similar to blueprint used in other field of engineer.

~~UML is not a programming language. It is rather a visual language. We use UML diagram to portray the behaviour and structure of a system.~~

### Characteristics of UML

#### i) Iterative

- a) Instead of trying to define all the details of a model at once, several passes are made.
- b) each iteration adds more details.

#### ii) Incremental

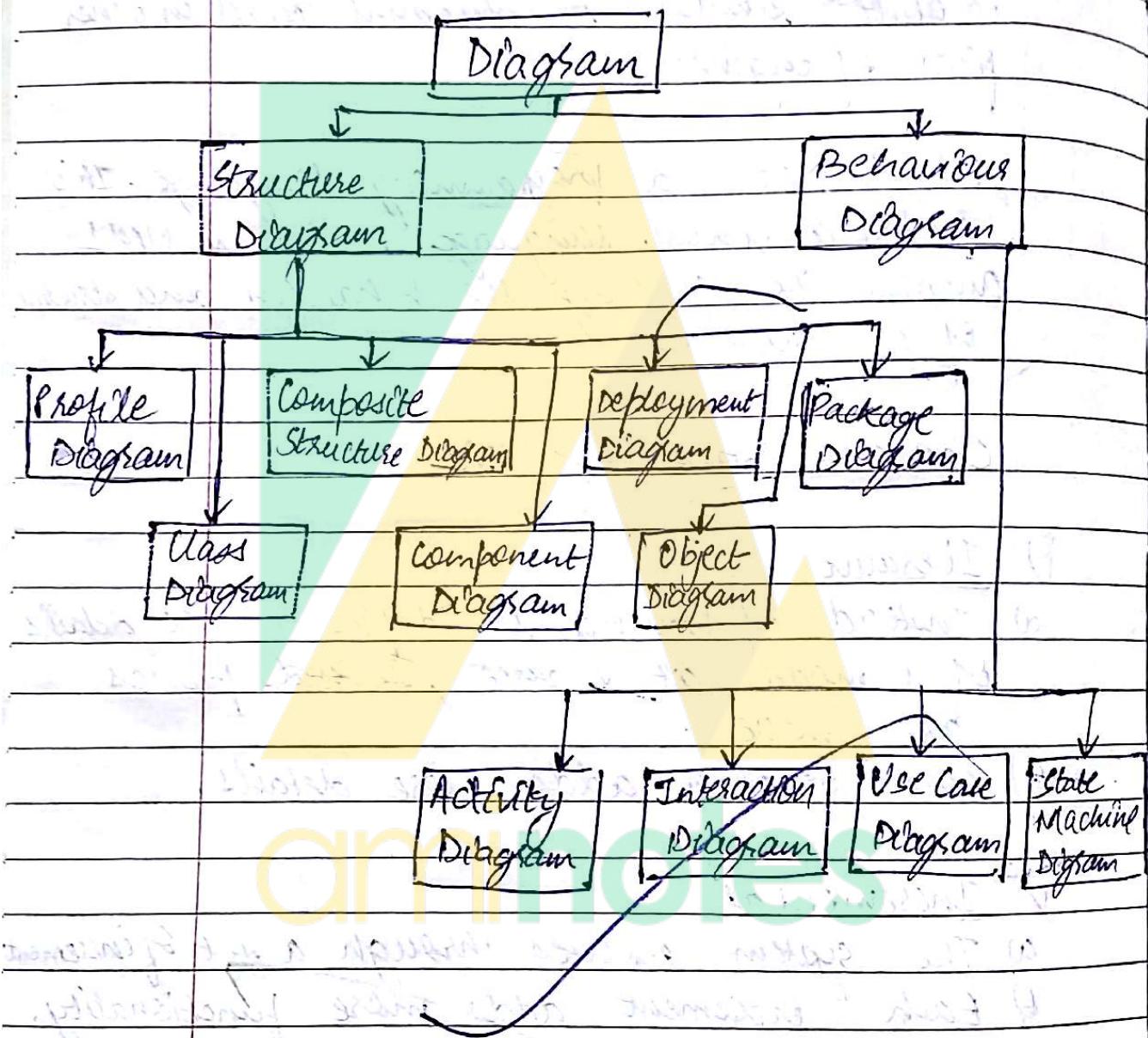
- a) The system evolves through a set of increment
- b) Each increment adds more functionality.

#### iii) User Centric

- a) Analysts specify functionality with use cases
- b) Customer confirm use cases
- c) Designers and implementors realize Use Cases
- d) Tester verify the system with Use Cases.

iii) Well defined architecture

- a) The system is partitioned into ~~sys~~ subsystems
- b) Logical and Physical views of the system are separated.



## Use Case Diagram

A use case diagram describes a system's functional requirement in terms of Use Case. It is a model of the system's intended functionality (Use Case) and its environment (actors).

It enables us to relate what we need from a system to how the system delivers on those needs.

### Procedure / Steps to Draw

- i) Identify the actors of the system
- ii) For each category of user, identify all roles played by users relevant to the system
- iii) Identify what are the users required the system to be performed to achieve these goals
- iv) Create Use Case for every goals
- v) Structure the Use Cases
- vi) Prioritize, review, estimate and validate the uses.

### Advantages

1. It is easy to understand and provide an excellent way for communicating with customers.
2. It can help to manage the complexity of large project by partitioning the problem into major user feature (i.e. Use Case) and by specifying

## Disadvantages

It doesn't capture the non-functional requirements easily.

Example :-

Bank ATM



## Class Diagram

A class diagram describes the structure of an object-oriented system by showing the classes in that system and relationships between the classes. It also shows constraints and attributes of classes.

## Procedure

1. Identify the objects in the problem domain and create classes for each of them
2. Add attribute for those classes
3. Add operation for those classes
4. Connect all the class with appropriate relationships.

## Advantages

1. Illustrate data model for Information System no matter how simple or complex.
2. Better understand the general overview of the schematics of an application.
3. Visually express any specific needs of a system and disseminate that information throughout the business.

## Disadvantages

1. Programmers may need to learn UML to build the class diagram in the first place.
2. The time spent building the class diagram may add to overall development time.
3. If class diagram is over complicated then it may be difficult to correlate with actual code.

Student	faculty
S - Name	f-name
S - enrollNo	f-id
S - add	f-add
login()	logout()
check att.()	upload()
check timetable()	edit()
exam()	check att()
Result()	

## Object Diagram

Object diagram are derived from class diagrams  
 so object diagram are dependent upon class diagrams.

Object diagram represent an instance of a class diagram. The basic concepts are similar for class diagrams and object diagrams. Object diagram also represent the static view of a system but static view of a system is a snapshot of the system at a particular moment. Object diagram used to render a set of object and their relationship as on an instant.

### Procedure

1. Identify the mechanism you'd like to model. A mechanism represent some function or behaviour of the part of system.
2. For each mechanism, identify classes, interfaces, and other element that participate in this collaboration. Identify the relationship among these thing as well.
3. Consider one scenario that walks through this mechanism. Freeze that scenario at a moment in time and render each object that participate in the mechanism.
4. Expose the state and attribute values of each

such object, as necessary to understand the scenario.

### Advantages

1. The ability to tackle more challenging problem domains.
2. Improved communication among user, analysts, designers and programmers.

### Example :- Writer

Writer	Document	Ethel : Writer	Draft : Doc.
name : string	name : string	Name = Ethel	Name - All
age : int	Page No : int	age = 34	about me P.No = 34

Master : Document
Name = all about me
P.No = 28

### Activity Diagram

Activity diagram is another important behavioural diagram in UML diagram to describe dynamic aspect of the system. Activity diagram is essentially an advanced version of flow-chart that modeling the flow from one activity to another activity.

### Procedure :-

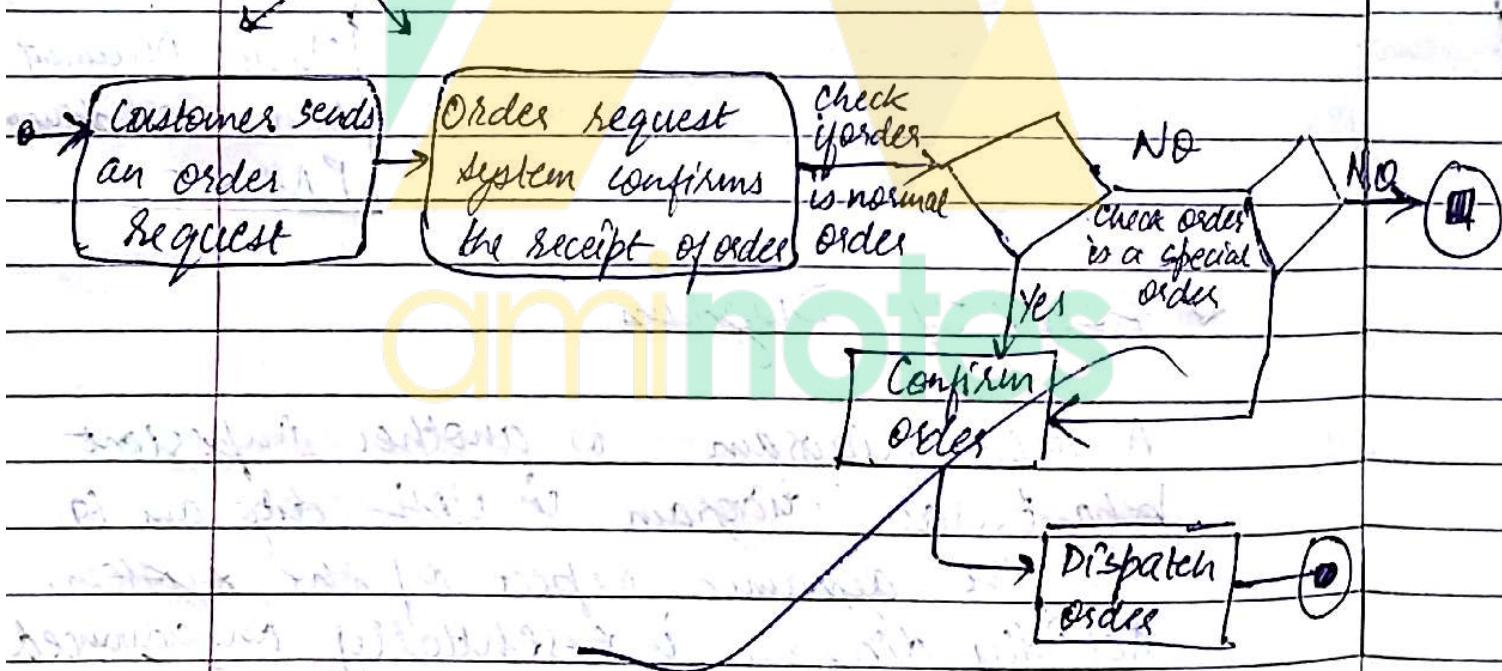
1. Collect information source.

2. Find activities and action
3. Adopt actors from business use case
4. Connect action
5. Refine activities
6. Verify the view

### Advantage -

1. Demonstrate the logic of an algorithm.
2. Describes the steps performed in a UML use case
3. Simplify and improve any process by clarifying complicated use case

### activities



## State Chart Diagram

The name of diagram itself clarify the purposes of the diagram and other details.

It describe different states of a component in a system. The states are specific to a component / object of a system. It describe a state machine that can be defined as a machine which defined as a machine which defines different state of an object and these states are controlled by external or internal events.

### Characteristics.

1. To model the dynamic aspects of a system.
2. To model the lifetime of a reactive system.
3. To describe different state of an object during its life time.
4. Define a state machine to model the state of an object.

### Process / Procedure

1. Identify the important objects to be analysed.
2. Identify the states
3. Identify the events.

### Advantages

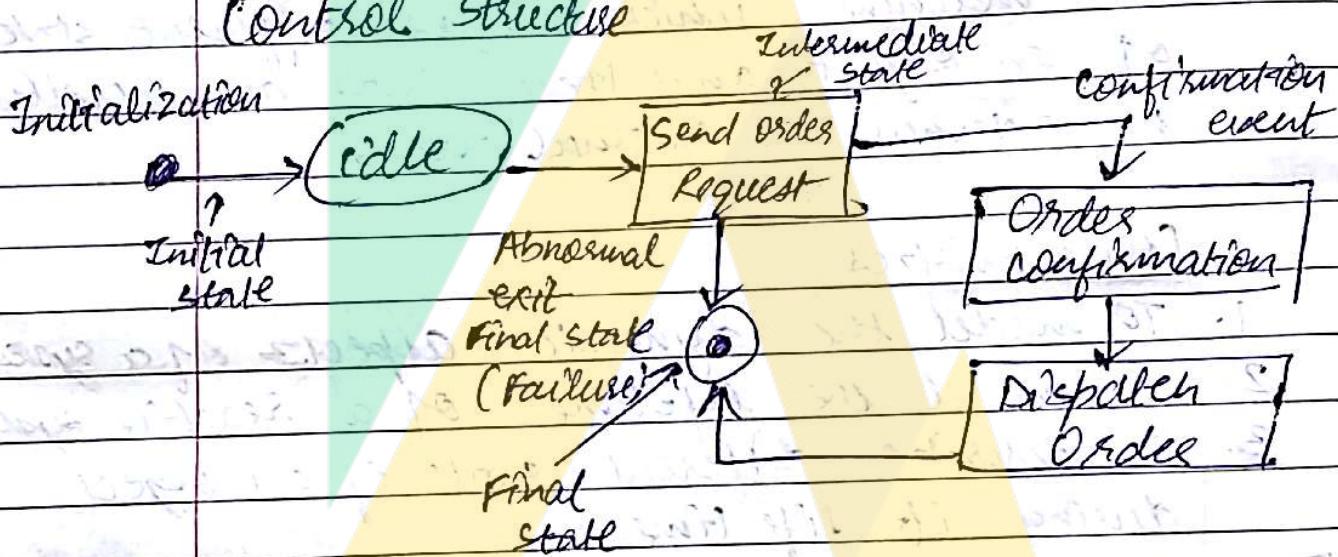
1. Domain SMEs can be kept aware of life cycle states of their key concern.
2. It makes diagram easier to read because

it modularizes the code into logical states.

Disadvantages -

1. Applicability, requirements analysis, requirements documentation.
2. Each state should be validation to determine if it is relevant to the solution scope which may be time consuming.

### Control Structure



aminoNotes

Q

10/9

