**Software Engineering Assignment**

Viraj Kisan Daule

CSE22162

1. *Scenario Based Methods:*

A model based approach in Software Engineering concept which depicts how a user interacts with the system and the sequence of activities that may occur in the process.

The following depicts the various method.

* 1. **Use Case Scenarios**- It depicts the possible usage of features by a user within the system.
  2. **User Stories**- Describes the features and functionalities that the target users may require in the software in the form a story with a format.
  3. **Storyboards**- A visual representation of a sequence of activities that the software will be implementing.
  4. **Activity** **Scenarios**- Focuses on particular users based on their type to find out how will the activity flow will occur.
  5. **Misuse Scenarios**- Scenarios studied to find out the ways in which the software or its components may be misused.
  6. **Business Scenarios**- A business perspective on the software that studies business processes and goals.
  7. **Exception Scenarios**- Describes situations where a software needs to handle errors in the system.

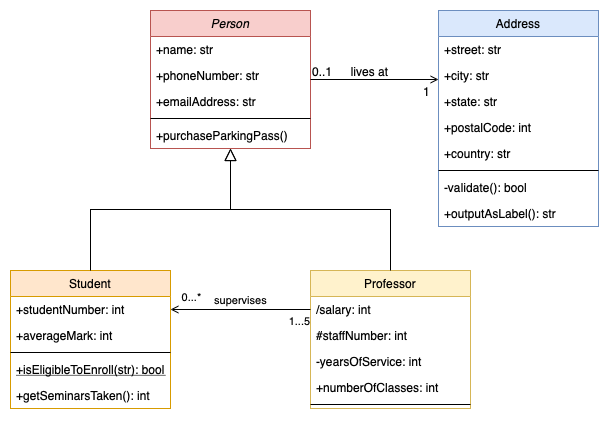
1. *UML Models:*

A visual representation of the model and its scenario based methods.

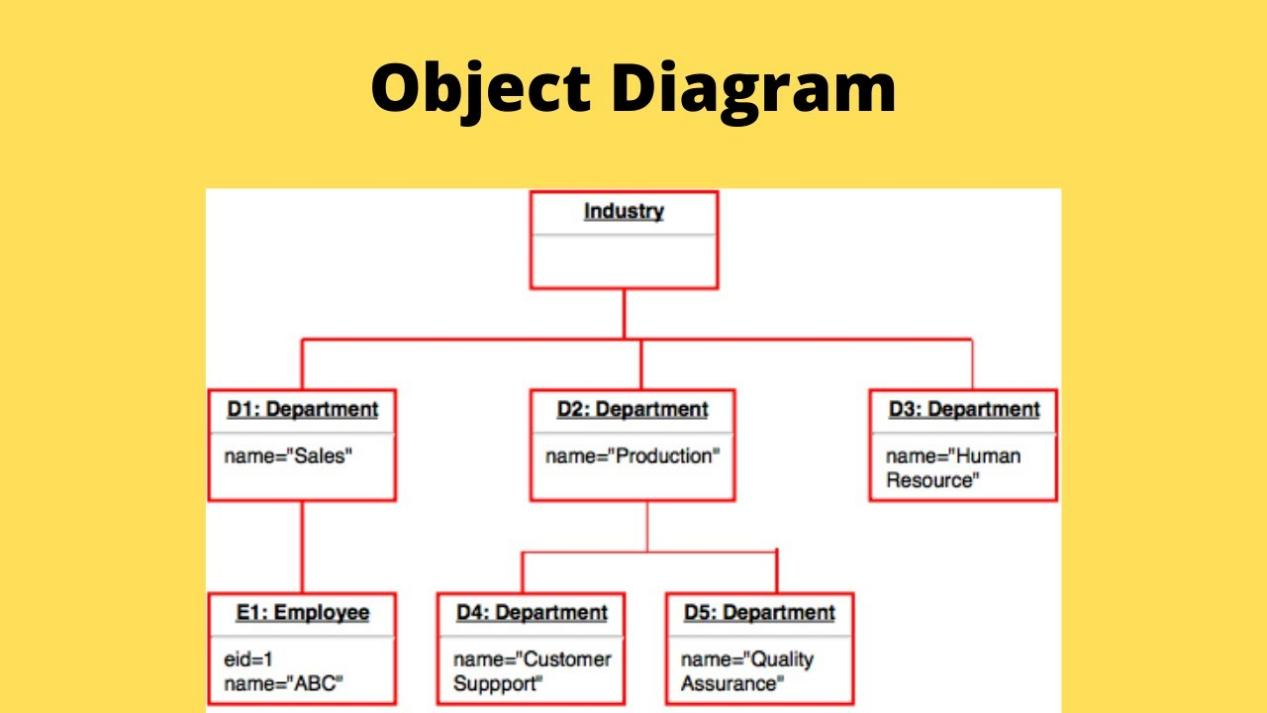
It can be categorized into the following 2 categories.

1. **Structural Models**- Static aspect of the model

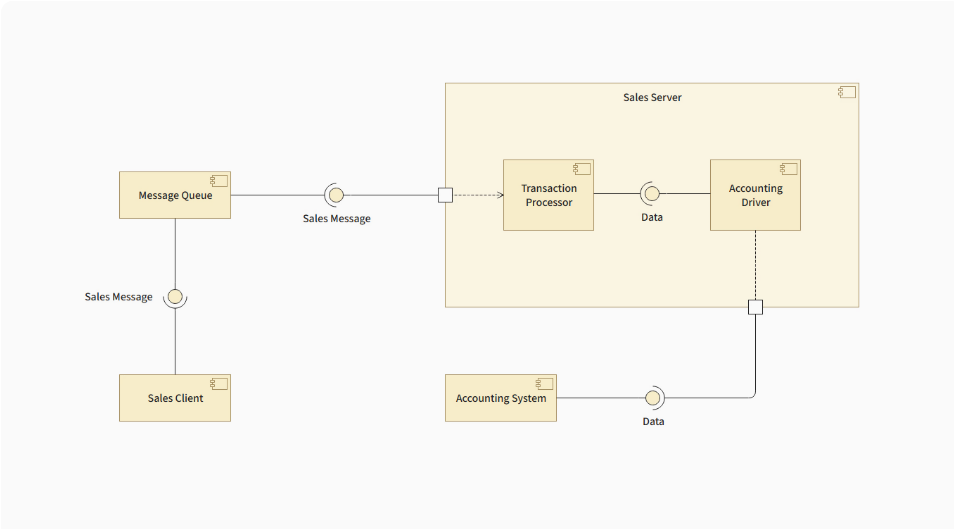
* **Class Diagram**: Shows classes, attributes, methods, and their relationships.



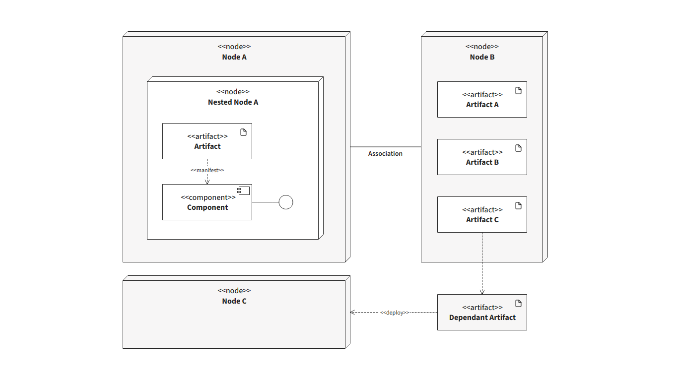
* **Object Diagram** - Displays specific instances of classes and their relationships at a point in time.



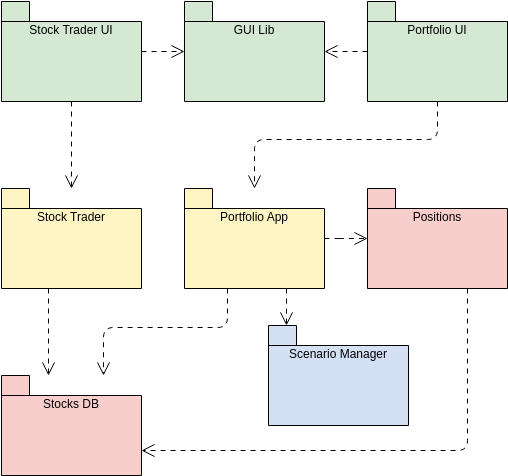
* **Component Diagram** - Visualizes system components and their interactions.



* **Deployment Diagram** - Represents the physical deployment of software on hardware nodes.

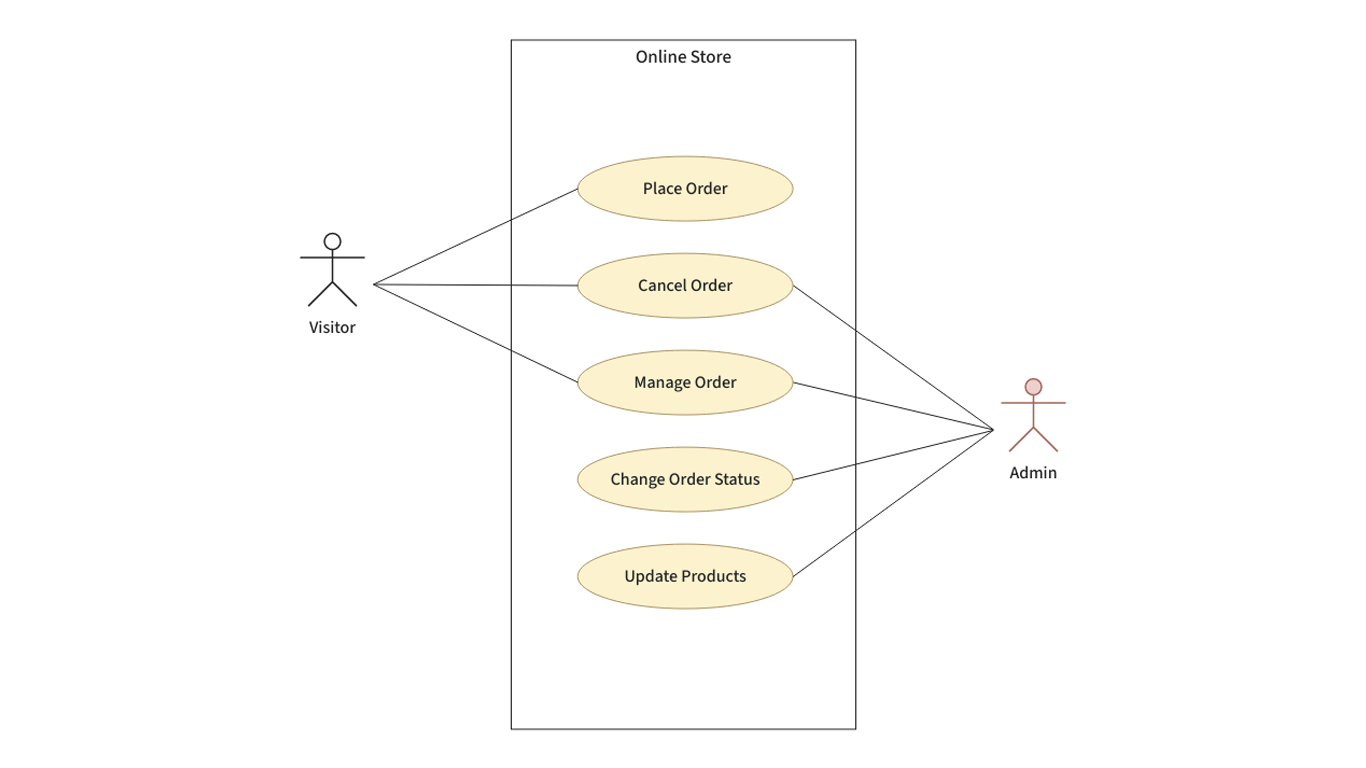


* **Package Diagram** - Organizes related classes into packages for modular design.

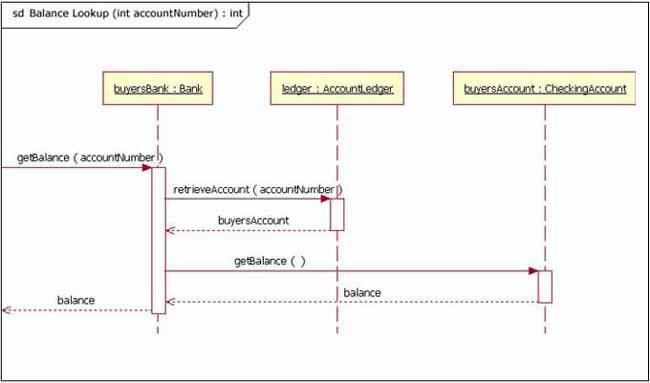


1. Behavioural Models- Dynamic aspect of the model

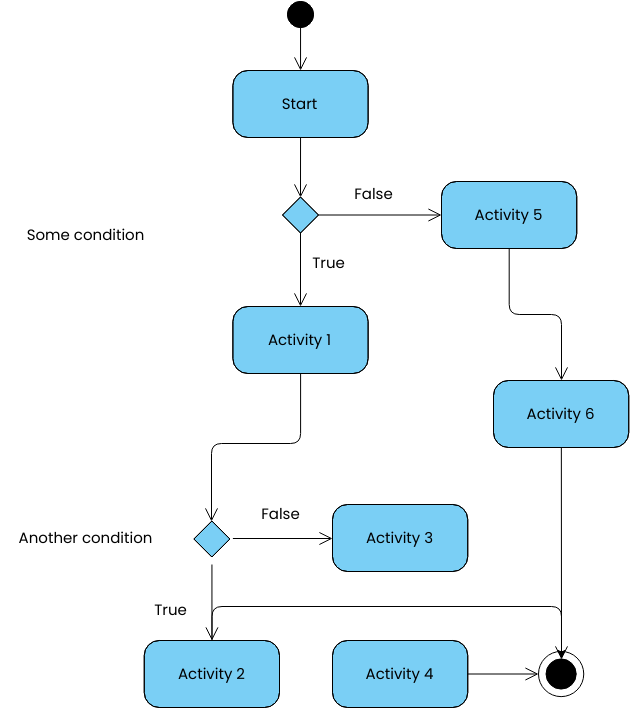
* **Use Case Diagram** - Illustrates how users interact with the system to achieve goals.



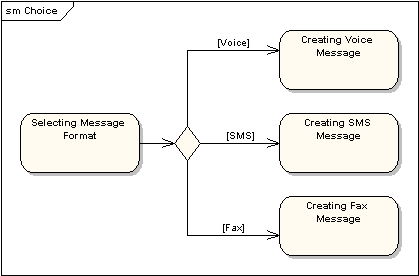
* **Sequence Diagram** - Shows the sequence of messages exchanged between objects in a scenario.



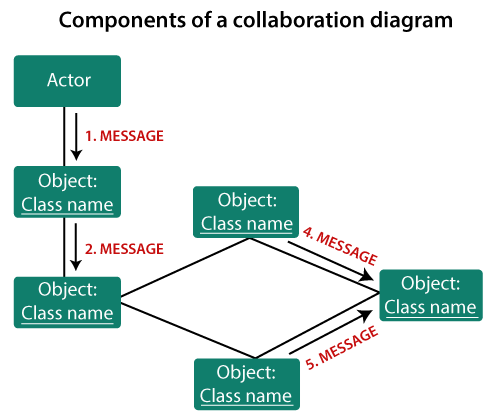
* **Activity Diagram** - Represents the flow of activities or processes within the system.



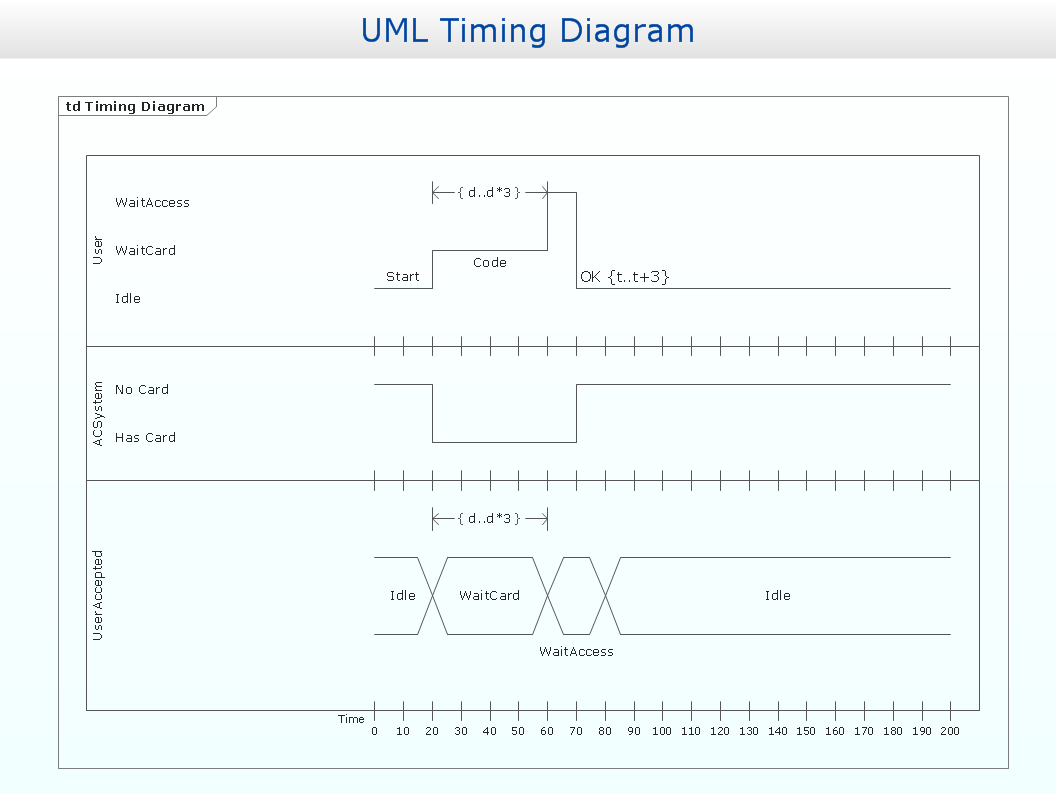
* **State Diagram** - Depicts object states and transitions between those states.



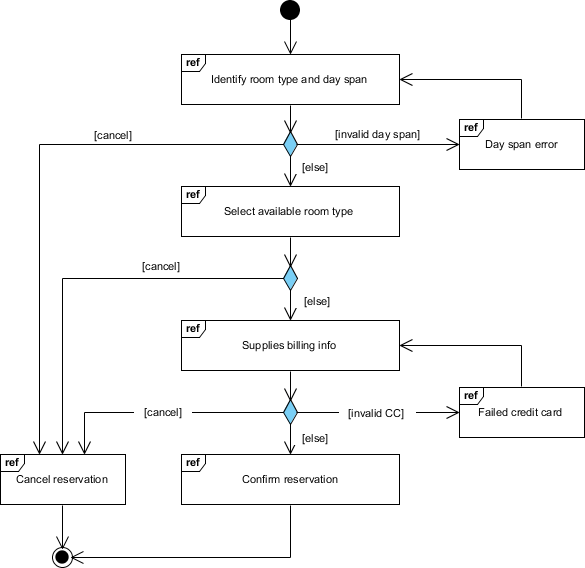
* **Collaboration Diagram** - Emphasizes object relationships and interactions during communication.



* **Timing Diagram** - Visualizes object state changes over time.



* **Interaction Overview Diagram** - Combines activity and sequence diagrams to show interaction flows.

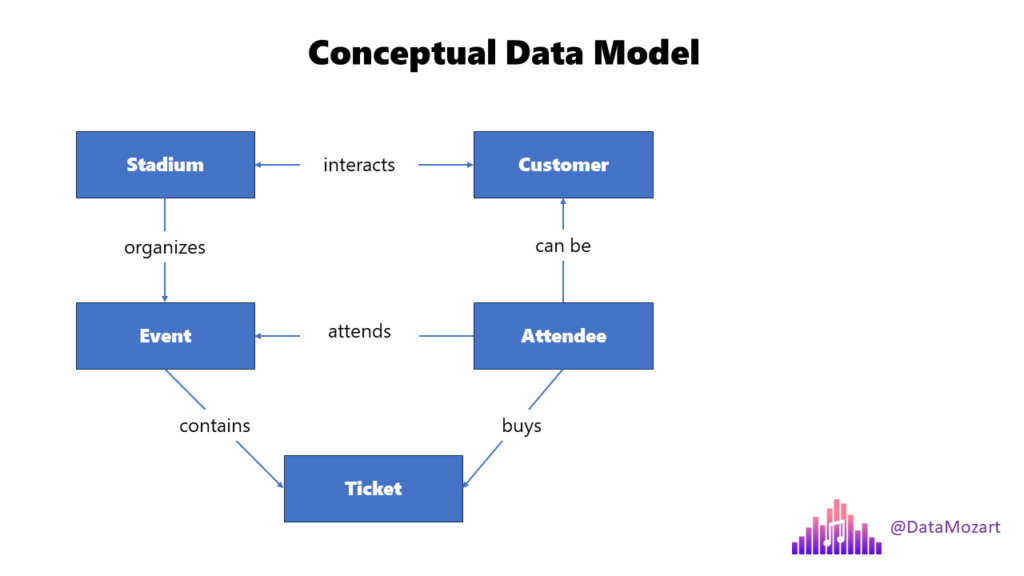


1. *Data Models:*

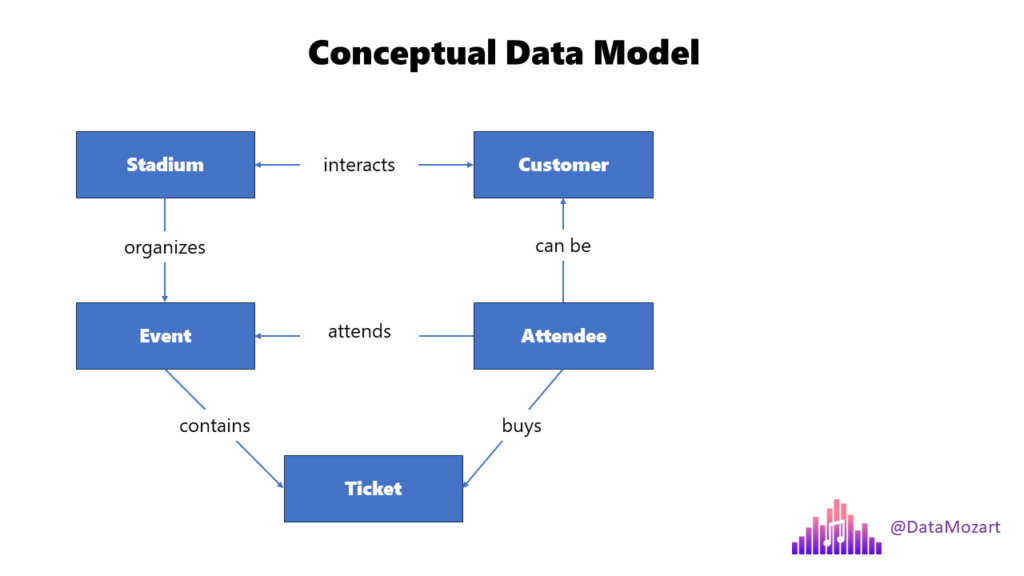
A conceptual representation defining how data is structured, organized, stored, and manipulated within a system. It provides a clear framework for understanding the relationships between different data elements and serves as a blueprint for designing databases and managing information.

Its types include:

1. **Conceptual**- High-level representation of data and relationships focusing on business concepts.



1. **Logical**- Detailed view of data with attributes, relationships, and data types without storage concerns.



1. **Physical**- Describes the actual storage of data using tables, columns, indexes, and constraints.

