**Godavari College Of Engineering, Jalgaon.**

**Subject Name:** PDE **Teacher Name:** Prof.Nilesh Chaudhari Sir

**Practical No**. : 2 **Date:**

**Class: S**.E **Roll No: 1951711245011**

**Title:**  Explain Sequence Diagram in Software Engineering.

**Sequence Diagram :-** A **sequence diagram** shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the logical view of the system under development. Sequence diagrams are sometimes called **event diagrams** or **event scenarios**.

A sequence diagram shows, as parallel vertical lines (*lifelines*), different processes or objects that live simultaneously, and, as horizontal arrows, the messages exchanged between them, in the order in which they occur. This allows the specification of simple runtime scenarios in a graphical manner.

**Sequence Diagrams 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 subsystems (sometimes known as system sequence diagrams)

### **Purpose of Sequence Diagram**

* Model high-level interaction between active objects in a system
* Model the interaction between object instances within a collaboration that realizes a use case
* Model the interaction between objects within a collaboration that realizes an operation
* Either model generic interactions (showing all possible paths through the interaction) or specific instances of a interaction (showing just one path through the interaction)

### **Sequence Diagrams at a Glance**

Sequence Diagrams show elements as they interact over time and they are organized according to object (horizontally) and time (vertically):

#### Object Dimension

* The horizontal axis shows the elements that are involved in the interaction
* Conventionally, the objects involved in the operation are listed from left to right according to when they take part in the message sequence. However, the elements on the horizontal axis may appear in any order

#### Time Dimension

* The vertical axis represents time proceedings (or progressing) down the page.

**Note that:**

Time in a sequence diagram is all a about ordering, not duration. The vertical space in an interaction diagram is not relevant for the duration of the interaction.

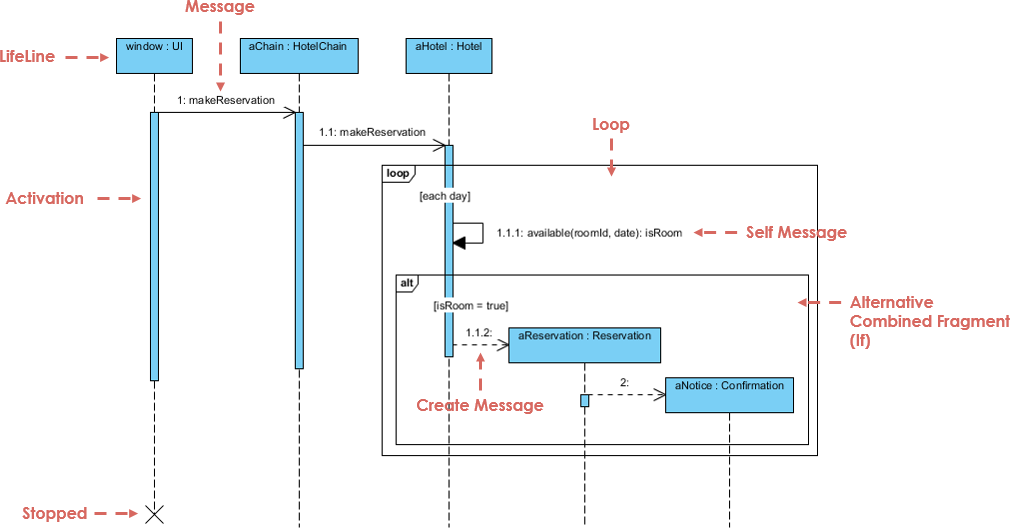
### **Sequence Diagram Notation**

|  |  |
| --- | --- |
| Notation Description | Visual Representation |
| Actor   * a type of role played by an entity that interacts with the subject (e.g., by exchanging signals and data) * external to the subject (i.e., in the sense that an instance of an actor is not a part of the instance of its corresponding subject). * represent roles played by human users, external hardware, or other subjects. |  |
| Lifeline   * A lifeline represents an individual participant in the Interaction. |  |
| Activations   * A thin rectangle on a lifeline) represents the period during which an element is performing an operation. * The top and the bottom of the of the rectangle are aligned with the initiation and the completion time respectively |  |
| Call Message   * A message defines a particular communication between Lifelines of an Interaction. * Call message is a kind of message that represents an invocation of operation of target lifeline. |  |
| Return Message   * A message defines a particular communication between Lifelines of an Interaction. * Return message is a kind of message that represents the pass of information back to the caller of a corresponded former message. |  |
| Self Message   * A message defines a particular communication between Lifelines of an Interaction. * Self message is a kind of message that represents the invocation of message of the same lifeline. |  |
| Recursive Message   * A message defines a particular communication between Lifelines of an Interaction. * Recursive message is a kind of message that represents the invocation of message of the same lifeline. It's target points to an activation on top of the activation where the message was invoked from. |  |
| Create Message   * A message defines a particular communication between Lifelines of an Interaction. * Create message is a kind of message that represents the instantiation of (target) lifeline. |  |
| Destroy Message   * A message defines a particular communication between Lifelines of an Interaction. * Destroy message is a kind of message that represents the request of destroying the lifecycle of target lifeline. |  |
| Duration Message   * A message defines a particular communication between Lifelines of an Interaction. * Duration message shows the distance between two time instants for a message invocation. |  |
| Note  A note (comment) gives the ability to attach various remarks to elements. A comment carries no semantic force, but may contain information that is useful to a modeler. |  |

### **Sequence Diagram Example: Hotel System**

Sequence Diagram is an interaction diagram that details how operations are carried out -- what messages are sent and when. Sequence diagrams are organized according to time. The time progresses as you go down the page. The objects involved in the operation are listed from left to right according to when they take part in the message sequence.

Below is a sequence diagram for making a hotel reservation. The object initiating the sequence of messages is a Reservation window.



**Conclusion:-** In this Assignment i learn what is Sequence Diagram and how to implement it.