

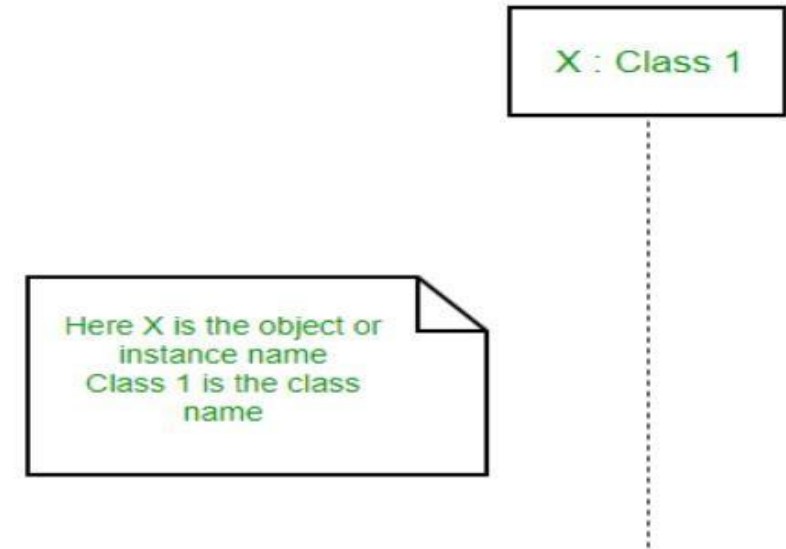


SEQUENCE DIAGRAM

Prof. Maitreyee Ganguly

Definition

- A sequence diagram depicts interaction between objects in a sequential order i.e. the order in which these interactions occur.
- Sequence diagrams describe how and in what order the objects in a system function.

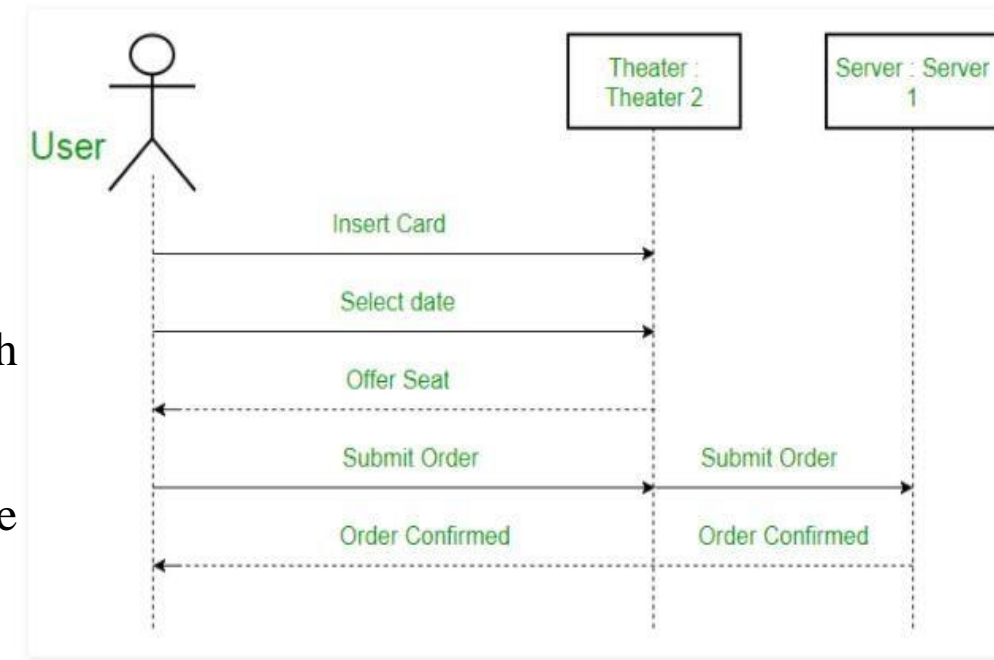


Sequence Diagram Notations

- Lifeline
- A lifeline represents an individual participant in the sequence diagram.
- Lifeline elements are located at the top in a sequence diagram.
- The standard in UML for naming a lifeline follows the following format –
Instance Name: Class Name

Actors

- An actor in a UML diagram represents a type of role where it interacts with the system and its objects.
- It is important to note here that an actor is always outside the scope of the system we aim to model using the UML diagram.
- We use actors to depict various roles including human users and other external subjects.
- We represent an actor in a UML diagram using a stick person notation.
- We can have multiple actors in a sequence diagram.
- For example – Here the user in the seat reservation system is shown as an actor who it exists outside the system and is not a part of the system.



Messages

- Communication between objects is depicted using messages.
- The messages appear in a sequential order on the lifeline. We represent messages using arrows.

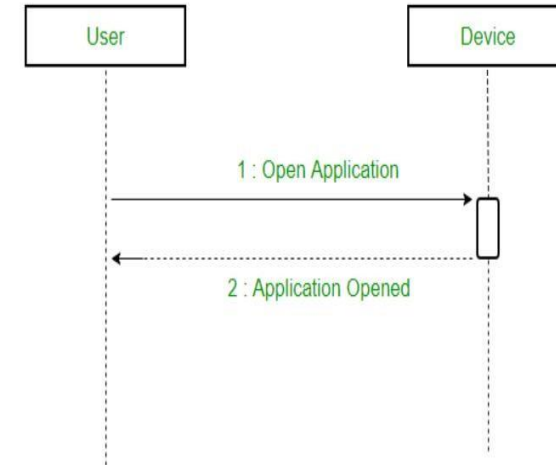
Types of messages

▪ **Synchronous messages** –

- A synchronous message waits for a reply before the interaction can move forward.
- The sender waits until the receiver has completed the processing of the message. The caller continues only when it knows that the receiver has processed the previous message i.e. it receives a reply message.
- We use a solid arrow head to represent a synchronous message.

Asynchronous Messages –

- An asynchronous message does not wait for a reply from the receiver.
- The interaction moves forward irrespective of the receiver processing the previous message or not.
- We use a lined arrow head to represent an asynchronous message.

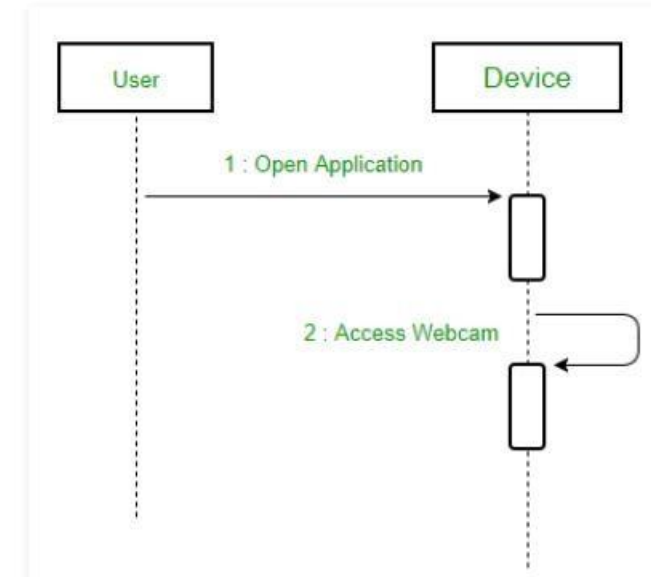


→ Synchronous

→ Asynchronous

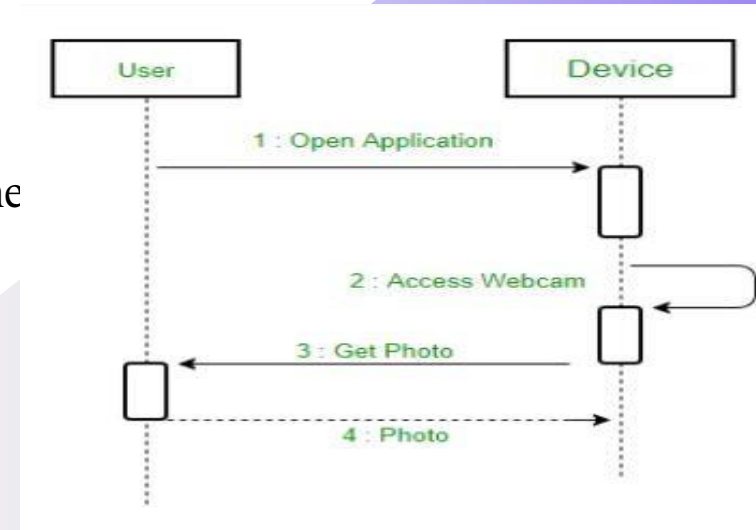
Self Message:

- Certain scenarios might arise where the object needs to send a message to itself. Such messages are called Self Messages and are represented with a U shaped arrow.
- For example – Consider a scenario where the device wants to access its webcam. Such a scenario is represented using a self message.



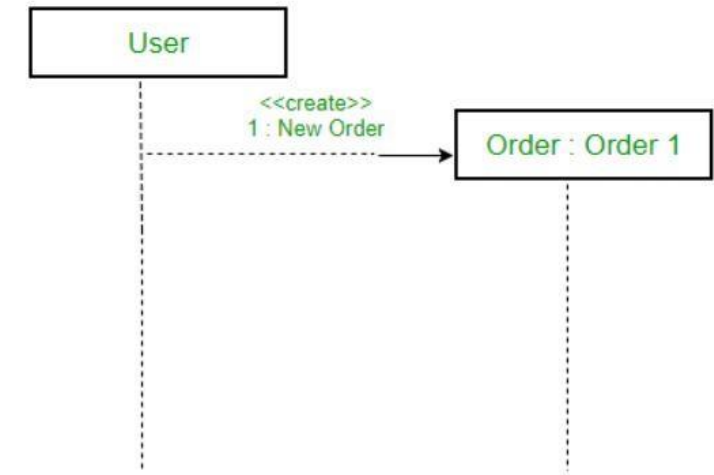
Reply Message :

- Reply messages are used to show the message being sent from the receiver to the sender.
- We represent a return/reply message using an open arrowhead with a dotted line.
- For example – Consider the scenario where the device requests a photo from the user. Here the message which shows the photo being sent is a reply message.



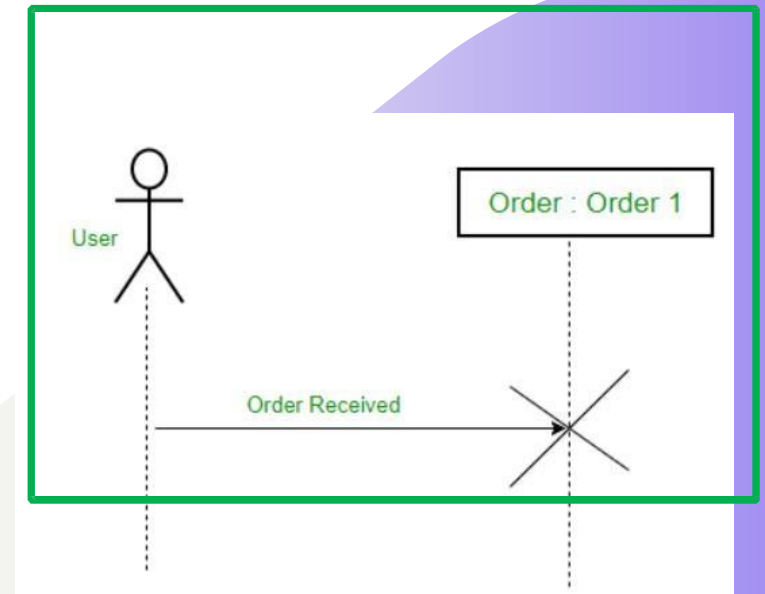
Create Message

- We use a Create message to create a new object in the sequence diagram.
- It is represented with a dotted arrow and creates a word labeled on it to specify that it is the create Message symbol.
- For example – The creation of a new order on an e-commerce website would require a new object of Order class to be created.



Delete Message

- We use a Delete Message to delete an object.
- When an object is deallocated memory or is destroyed within the system we use the Delete Message symbol.
- It is represented by an arrow terminating with an x.
- For example – In the scenario below when the order is received by the user, the object of the order class can be destroyed.

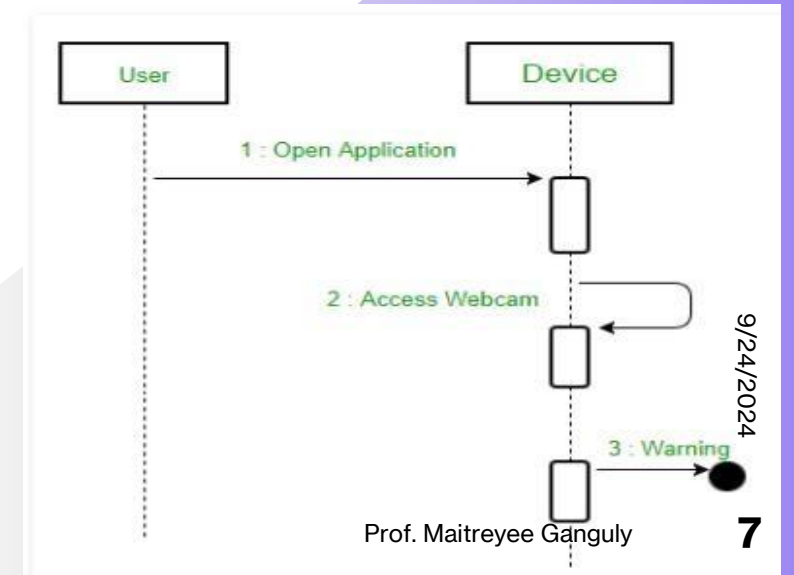
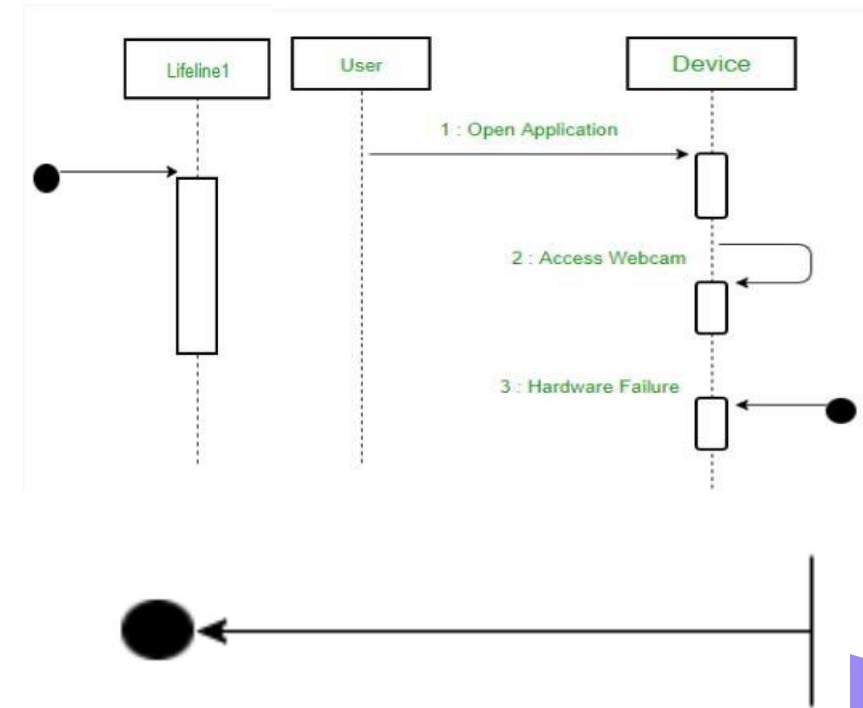


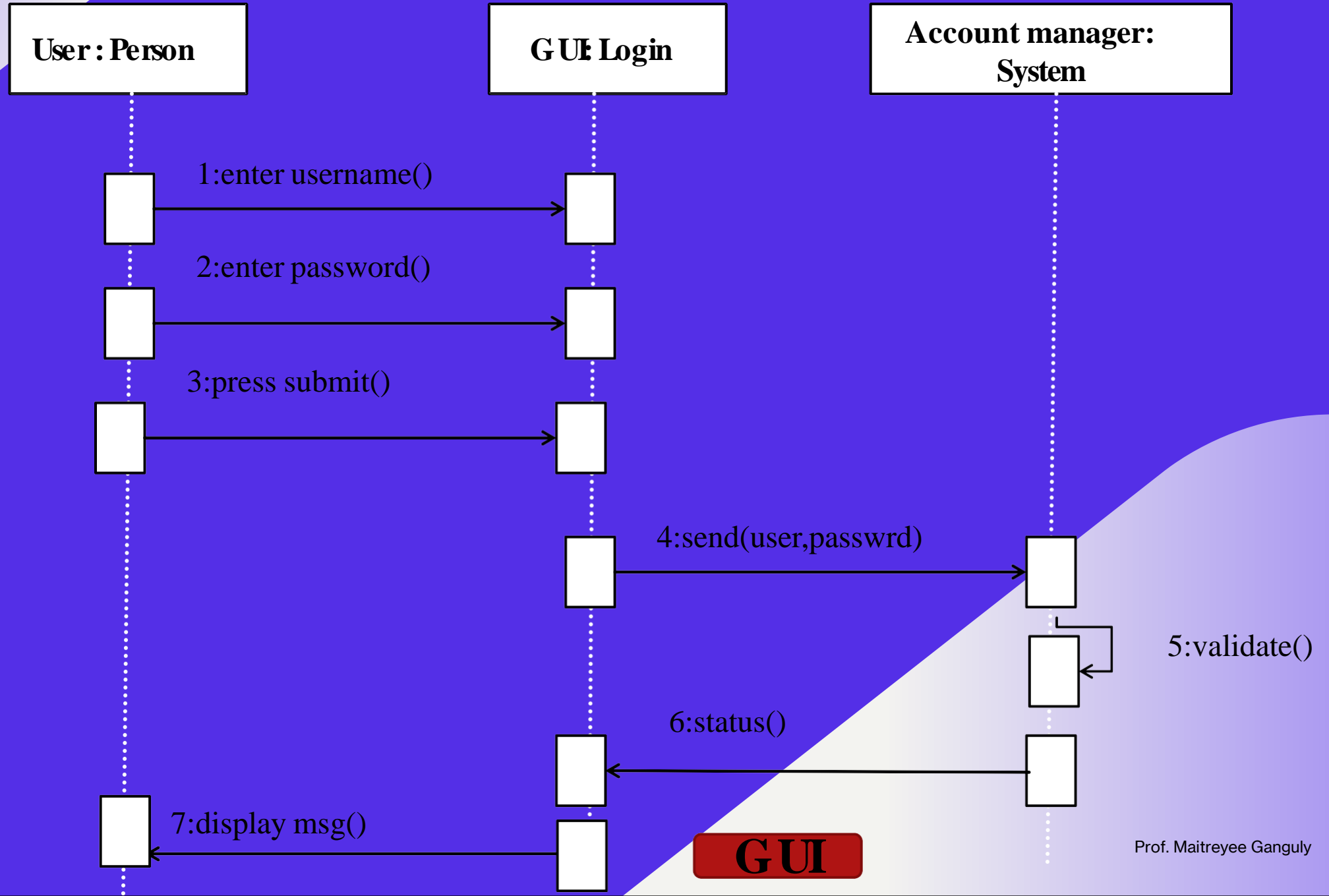
Found Message

- A Found message is used to represent a scenario where an unknown source sends the message.
- It is represented using an arrow directed towards a lifeline from an endpoint.
- For example: Consider the scenario of a hardware failure. It can be due to multiple reasons and we are not certain as to what caused the hardware failure.

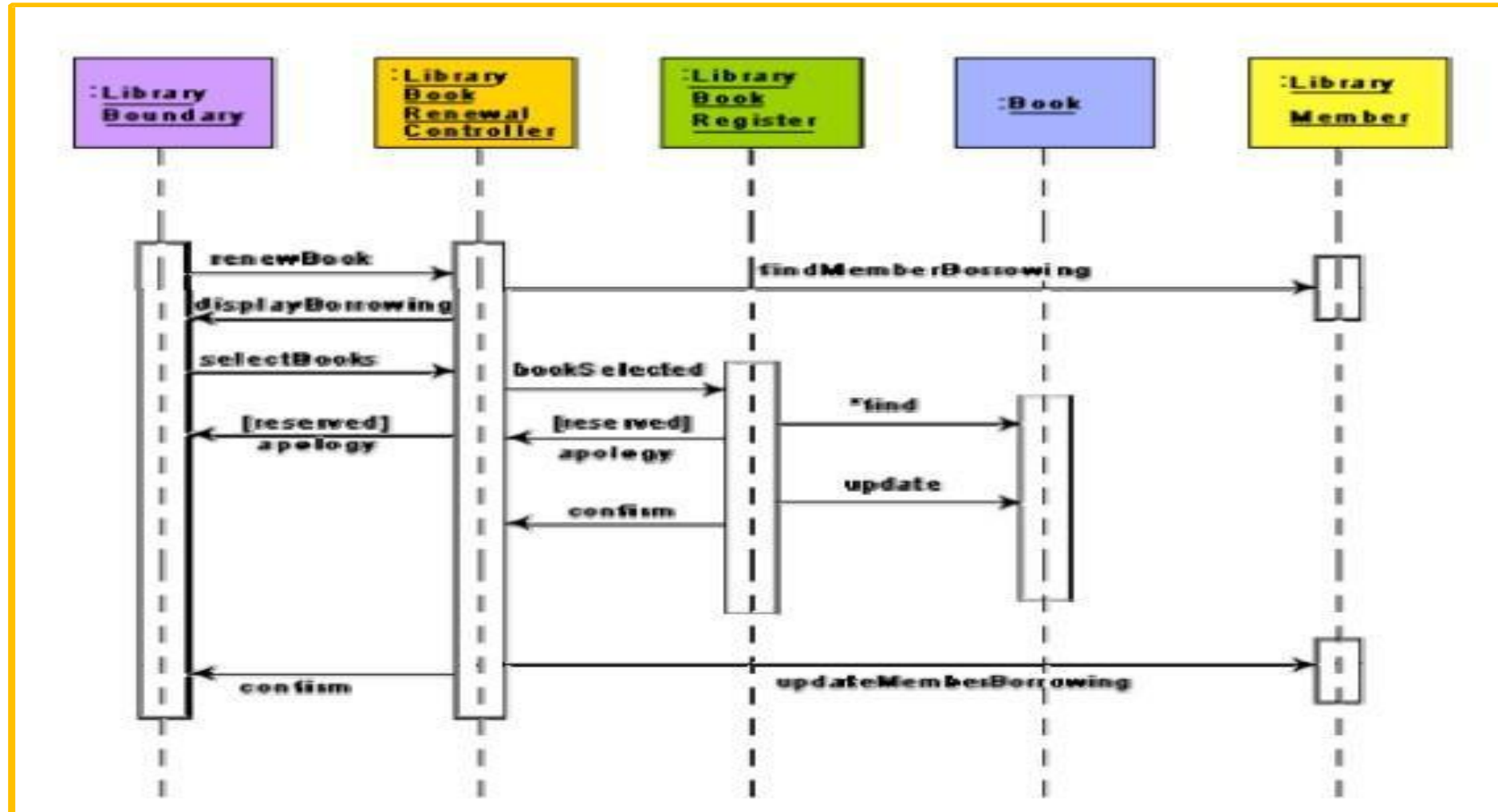
Lost Message

- A Lost message is used to represent a scenario where the recipient is not known to the system.
- It is represented using an arrow directed towards an endpoint from a lifeline.
- For example: Consider a scenario where a warning is generated.
- The warning might be generated for the user or other software/object that the lifeline is interacting with. Since the destination is not known beforehand, we use the Lost Message symbol.





The sequence diagram for the book renewal use case for the Library Automation Software



Sequence diagram for the renew book use case

THANK YOU

