EXPERIMENT 8

Aim :

To prepare Sequence Diagram for the Application Software – **Bill Management System.**

Theory :

What is Sequence Diagram ?

A Sequence Diagram is a type of interaction diagram because it describes how – and in what order – a group of objects works together. These diagrams are used by software developers and business professionals to understand requirements for a new system or to document an existing process. Sequence diagrams are sometimes known as Event Diagrams or Event Scenarios. Note that there are two types of Sequence Diagrams : UML diagrams and code – based diagrams. The latter is sourced from programming code.

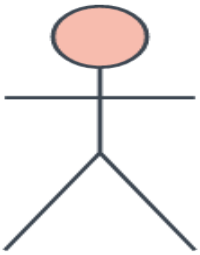
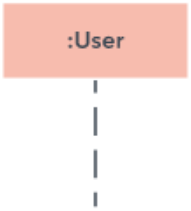
Benefits of Sequence Diagram :

- Represents the details of a UML use case.
- Models the logic of a sophisticated procedure, function, or operation.
- See how objects and components interact with each other to complete a process.
- Plan and understand the detailed functionality of an existing or future scenario.

Use Cases for Sequence Diagram :

- **Usage Scenario :** A usage scenario is a diagram of how your system could potentially be used. It's a great way to make sure that you have worked through the logic of every usage scenario for the system.
- **Method Logic :** Just as you might use a UML sequence diagram to explore the logic of a use case, you can use it to explore the logic of any function, procedure, or complex process.
- **Service Logic:** If you consider a service to be a high – level method used by different clients, a sequence diagram is an ideal way to map that out.

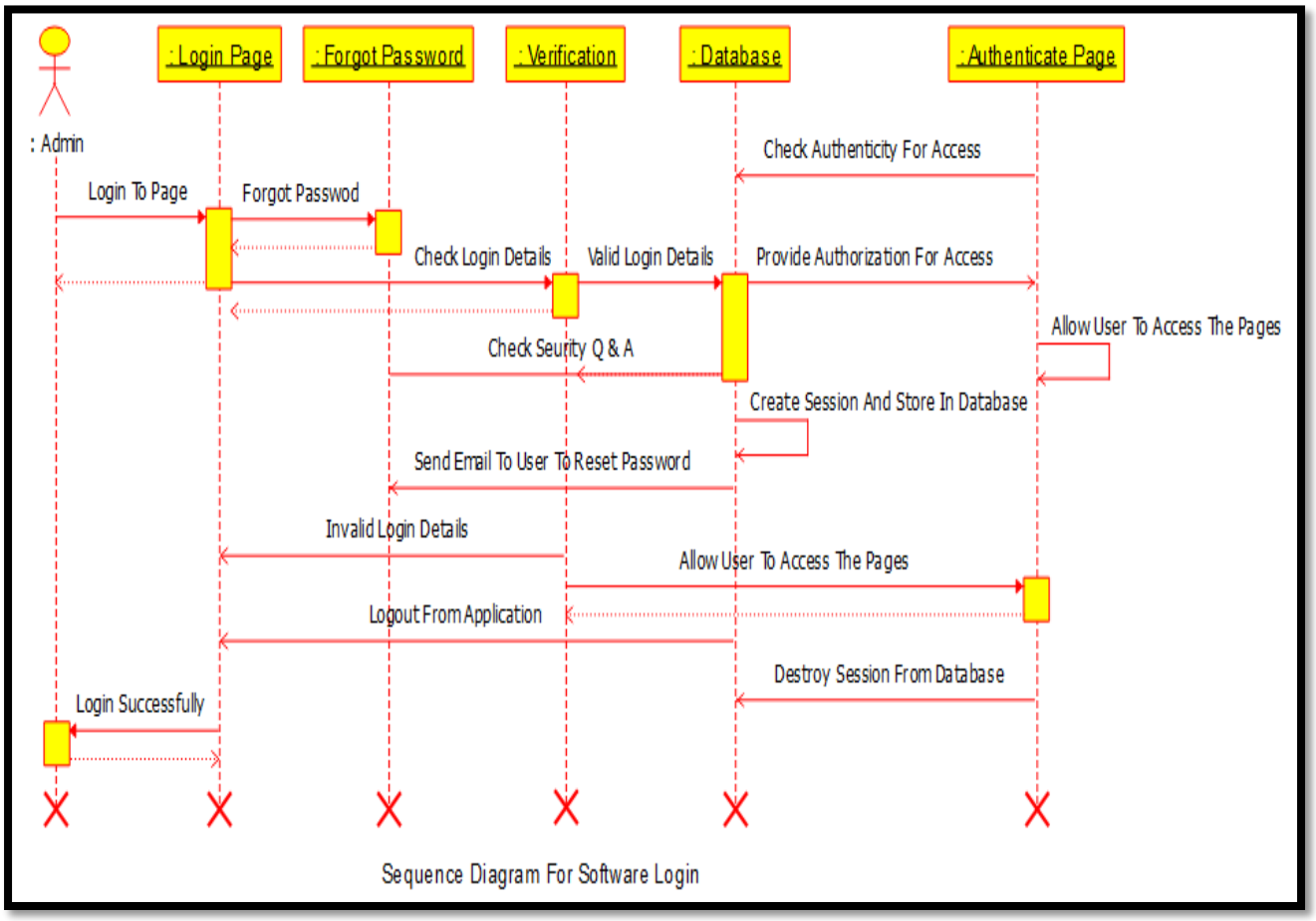
Basic Symbols & Components :

Symbol	Name	Description
	Object symbol	Represents a class or object in UML. The object symbol demonstrates how an object will behave in the context of the system. Class attributes should not be listed in this shape.
	Activation box	Represents the time needed for an object to complete a task. The longer the task will take, the longer the activation box becomes.
	Actor symbol	Shows entities that interact with or are external to the system.
	Package symbol	Also known as a frame, this rectangular shape has a small inner rectangle for labelling the diagram.
	Lifeline symbol	Represents the passage of time as it extends downward. This dashed vertical line shows the sequential events that occur to an object during the charted process. Lifelines may begin with a labelled rectangle shape or an actor symbol.
	Option loop symbol	Used to model if/then scenarios, i.e., a circumstance that will only occur under certain conditions.
	Alternative symbol	Symbolizes a choice between two or more message sequences. To represent alternatives, use the labelled rectangle shape with a dashed line inside.

Common Message Symbols :

Symbol	Name	Description
	Synchronous message symbol	Represented by a solid line with a solid arrowhead. This symbol is used when a sender must wait for a response to a message before it continues. The diagram should show both the call and the reply.
	Asynchronous message symbol	Represented by a solid line with a lined arrowhead. Asynchronous messages don't require a response before the sender continues. Only the call should be included in the diagram.
	Asynchronous return message symbol	Represented by a dashed line with a lined arrowhead.
	Asynchronous create message symbol	Represented by a dashed line with a lined arrowhead. This message creates a new object.
	Reply message symbol	Represented by a dashed line with a lined arrowhead, these messages are replies to calls.
	Delete message symbol	Represented by a solid line with a solid arrowhead, followed by an X. This message destroys an object.

Sequence Diagram for the Application Software – **Bill Management System :**



Sequence Diagram For Software Login

Conclusion :

The Sequence Diagram for the Application Software – **Bill Management System** was prepared successfully.