	Hope Foundation's
	Finolex Academy of Management and Technology, Ratnagiri
	Department of Information Technology

Subject name: Software Design Lab			Subject Code: ITL601
Class	TE IT	Semester –VI (CBCGS)	Academic year: 2018-19
Name of Student	**Kazi Jawwad A Rahim**		**QUIZ Score : 06**
Roll No	**27**	Assignment/Experiment No.	06
Title: **Designing of Sequence Diagrams**			

---

1. Lab objectives applicable: LOB4.

2. Lab outcomes applicable: LO3

3. Learning Objectives:
1. To understand basics of sequence diagram in UML.
2. Assign objects in sequence diagrams to classes
3. Map messages to operations.

4. Practical applications of the assignment/experiment: Diagrams are used in order to give an insight for the activities involved in the functionality of a system.

5. Prerequisites:
1. SRS

6. Hardware Requirements:
   Windows operating system (Windows 7 or higher)

7. Software Requirements:
   UML designing tool such as IBM Rational Rose/StarUML

---

8. Quiz Questions (if any): (Online Exam will be taken separately batch-wise, attach the certificate/ Marks obtained)
1. What is the purpose of sequence diagram?
2. What is object and messages?

---

9. Experiment/Assignment Evaluation:

Sr. No.	Parameters	Marks obtained	Out of
1	Technical Understanding (Assessment may be done based on Q & A **or** any other relevant method.) Teacher should mention the other method used -		6
2	Neatness/presentation		2
3	Punctuality		2
Date of performance (DOP)		**Total marks obtained**	
			10

Signature of the faculty

---

10. Theory:

- **The Sequence Diagram:**

Sequence diagrams are interaction diagrams that illustrate the ordering of messages according to time. It represents the behavioral aspects of a system. Sequence diagram shows the interactions between the objects by means of passing messages from one object to another with respect to time in a system.

Notations: These diagrams are in the form of two-dimensional charts. The objects that initiate the interaction are placed on the x–axis. The messages that these objects send and receive are placed along the y–axis, in the order of increasing time from top to bottom.

Messages

Messages are shown as an arrow from the life-line of sender object to the life-line of receiver object and labeled with the message name. Chronological order of the messages passing throughout the objects' life-line show the sequence in which they occur. There may exist some different types of messages:

- **Synchronous messages:** Receiver start processing the message after receiving it and sender needs to wait until it is made. A straight arrow with close and fill arrow-head from sender life-line bar to receiver end, represent a synchronous message.

- **Asynchronous messages:** For asynchronous message sender needs not to wait for the receiver to process the message. A function call that creates thread can be represented as an asynchronous message in sequence diagram. A straight arrow with open arrow-head from sender life-line bar to receiver end, represent an asynchronous message.

- **Return message:** For a function call when we need to return a value to the object, from which it was called, then we use return message. But, it is optional, and we are using it when we are going to model our system in much detail. A dashed arrow with open arrow-head from sender life-line bar to receiver end, represent that message.

- **Response message:** One object can send a message to self. We use this message when we need to show the interaction between the same object.
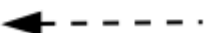
Message Type	Notation
Synchronous message	→
Asynchronous message	→
Response message	←- - - - - ·

Figure-01

- **Where to use Interaction Diagrams?**

We have already discussed that interaction diagrams are used to describe dynamic nature of a system. Now we will look into the practical scenarios where these diagrams are used. To understand the practical application we need to understand the basic nature of sequence and collaboration diagram.

The main purposes of both the diagrams are similar as they are used to capture the dynamic behavior of a system. But the specific purposes are more important to clarify and understood.

Sequence diagrams are used to capture the order of messages flowing from one object to another. And the collaboration diagrams are used to describe the structural organizations of the objects taking part in the interaction. A single diagram is not sufficient to describe the dynamic aspect of an entire system so a set of diagrams are used to capture is as a whole.
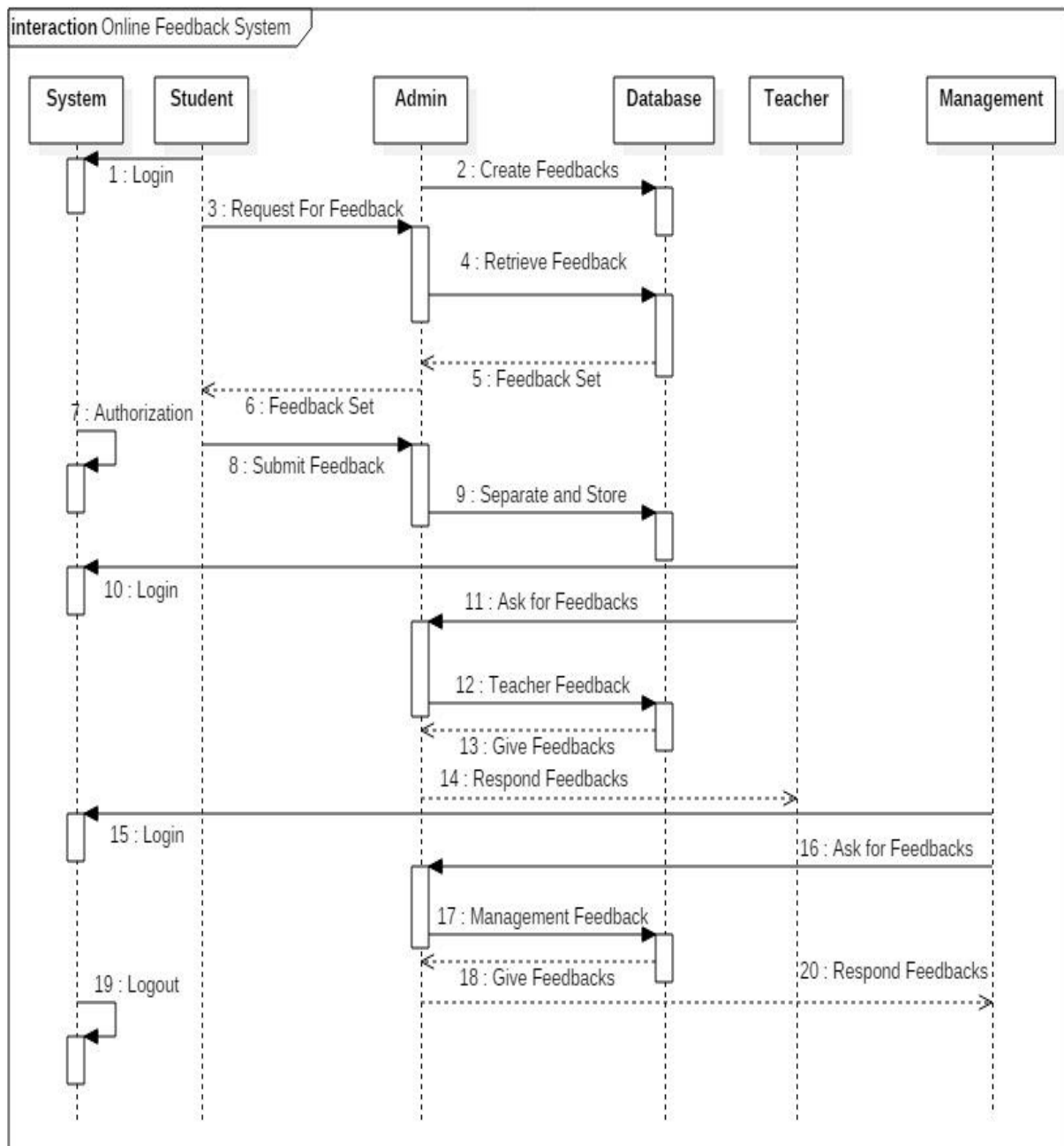
The interaction diagrams are used when we want to understand the message flow and the structural organization. Now message flow means the sequence of control flow from one object to another and structural organization means the visual organization of the elements in a system.

In a brief the following are the usages of interaction diagrams:

1. To model flow of control by time sequence.

2. To model flow of control by structural organizations.

3. For forward engineering.

11. Source code / Diagrams:

Sequence Diagram

References:

[1]. https://www.geeksforgeeks.org/unified-modeling-language-uml-introduction/
[2]. https://www.tutorialspoint.com/uml/
[3]. https://www.ibm.com/developerworks/rational/library/769.html