

KARNATAK LAW SOCIETY'S  
**GOGTE INSTITUTE OF TECHNOLOGY**  
UDYAMBAG, BELAGAVI-590008  
(An Autonomous Institution under Visvesvaraya Technological University, Belagavi)  
**(APPROVED BY AICTE, NEW DELHI)**



*Course Activity Report*  
*Submitted in the partial fulfillment for the academic requirement of*  
***Vth Semester B.E.***  
***in***

***SOFTWARE DESIGN AND MODELLING LABORATORY***

*Submitted by*  
**Jyotiprasad Patil– 2GI18CS059**  
**Varun Joshi– 2GI18CS058**  
**Gangadhar Kulam – 2GI18CS049**

**GUIDE**

*Prof. Gajendra Deshpande*  
DEPARTMENT OF COMPUTER SCIENCE

**2020– 21**

Department of Computer science and Engineering

**COURSE PROJECT REPORT (Academic Year 2020-21)**

**SUBJECT: SOFTWARE DESIGN AND MODELLING LABORATORY****SUBJECT CODE: 18CSL58****Date: 28/12/2020****PROBLEM STATEMENT:****Report on Link Management System****Team Members Details:**

Sl.No.	Name	USN
1	Jyotiprasad Patil	2GI18CS059
2	Varun Joshi	2GI18CS058
3	Gangadhar Kulam	2GI18CS049

**Marks allocation:**

	Batch No. : 32					
1	Seminar Title: Link Management System	Marks Range	USN			
			2GI18CS059	2GI18CS058	2GI18CS049	
2	Abstract (PO2)	0-2				
3	Application of the topic to the course (PO2)	0-3				
4	Literature survey and its findings (PO2)	0-4				
5	Methodology, Results and Conclusion (PO1,PO3,PO4)	0-6				
6	Report and Oral presentation skill (PO9,PO10)	0-5				
	Total	20				

**Signature Of Teacher Incharge**

## Table of Contents:

Objective

.....

Abstract

.....

ER diagram

.....

Use Case Diagram

.....

Activity Diagram

.....

Sequence Diagram

.....

Class Diagram

.....

Test Cases

.....

Conclusion

## **Link Management System**

### **Objective –**

As we all know that the world is undergoing unprecedented times and almost all the schools and colleges have become virtual and students are taught using varied technologies like Google Meet and Zoom.

These classes are often recorded so that students can again access them for their revision. But at times managing the links of these recorded classes becomes tiresome for the teachers and students also face difficulties while trying to find a particular lecture. So our project basically aims at building an application which will facilitate teachers and students to manage these links while helping them to be more productive.

### **Abstract –**

Our Project aims at creating a web-based application which will be able to store all the links of the recorded classes subject wise and year/class wise for a particular institute. In this application, the teacher will be able to log into the system and then be able to add or delete the links of his classes, and hence it will be easier for the students to find all the links in a particular place. The teachers will also be able to add the description of the topics which were taught in the class in front of the links. This will also be convenient for the teachers because they will not have to send the links in the WhatsApp groups which creates a chaos and is hard to maintain. The links can be varied and the video of the link can be uploaded to YouTube or Google Drive. This seminar report presents an object-oriented approach to software development based on modelling objects from the real world and then using the model to build a language-independent design organized around those objects. Object-oriented modelling and design promote better understanding of requirements, cleaner designs, and more maintainable systems. The process of managing day to day records in system. A set of object-oriented concepts and a language-independent graphical notation that is used to analyze problem requirements, design a solution to the problem is described.

### **Functional Requirements of the System –**

#### ► Database for each teacher's login –

The teachers will be able to log in to the software and then upload the link of any particular class which they wish too.

#### ► Enabling the teacher to add the link

Enabling the teacher to delete the link in case if there is some problem with the video or other unavoidable circumstances

▶ Enabling the students to access the links

The students should be able to access the links by visiting the web application whenever they need it.

▶ Sending Notifications, if permission is granted to the students regarding new links

The students will be sent notifications regarding new links if they have given the permission of the same.

▶ Not Enabling the Students to manipulate the links –

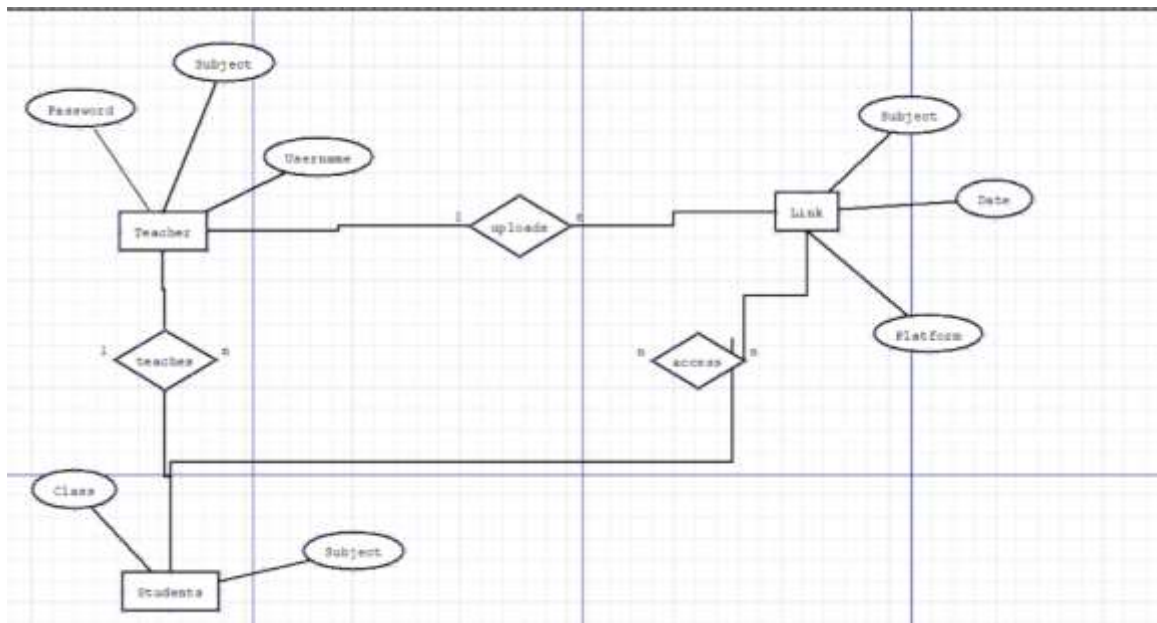
The students will not be allowed to manipulate with the links and the links can only be changed by the respective teachers

**Non-Functional Requirements of the System –**

- ▶ Should be able to handle at least a minimum of 500 people on the application at a particular time
- ▶ Should be scalable, that is it should be possible to increase the application's functioning to a larger scale.
- ▶ The Software should be portable, that is moving from one OS to another should be easier.
- ▶ Teachers should change the initially assigned password immediately after the first login

**ER diagram:**

An Entity–relationship model (ER model) describes the structure of a database with the help of a diagram, which is known as Entity Relationship Diagram.



### **Use Case Diagram:**

It is a graphic depiction of the interaction among the element of Link Management System. It represents the external user's interaction with the Link Manage system that shows the relationship between the user and the different use cases(functionalities) in which the user is involved. The main actors of the Link Management System are:

- 1) System Administrator
- 2) Teachers
- 3) Students

#### **Use cases for teachers are:**

- 1) Registration
- 2) Login
- 3) Uploading/Deleting Links
- 4) Adding Description

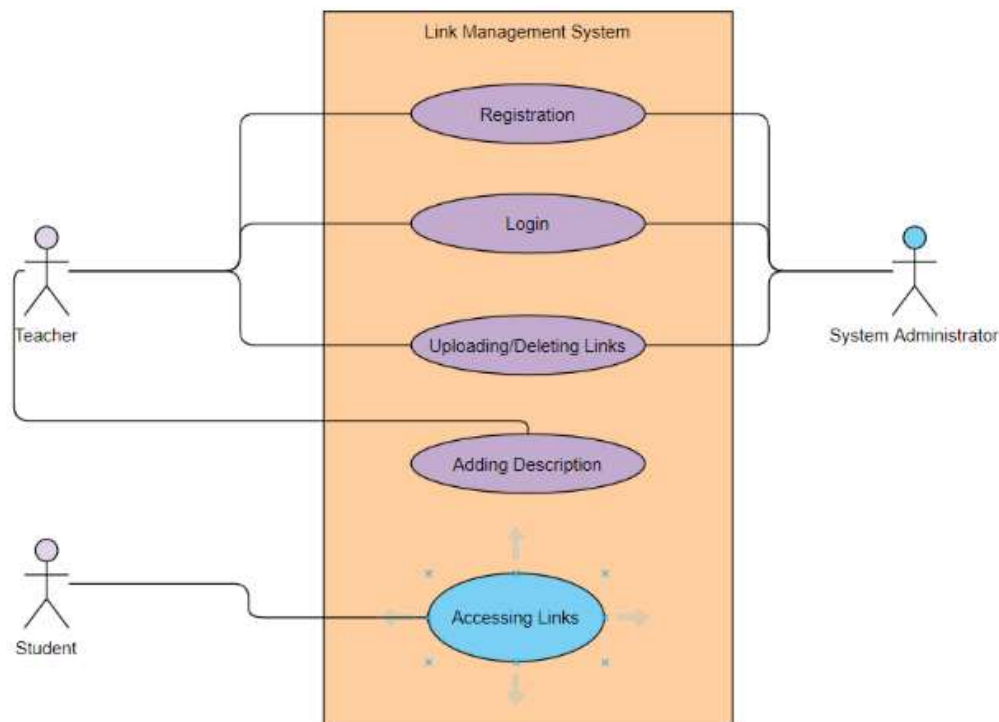
#### **Use cases for students are:**

- 1) Accessing Links

Use case for System Administrator is

1) Maintenance

2) Some Super User Privileges



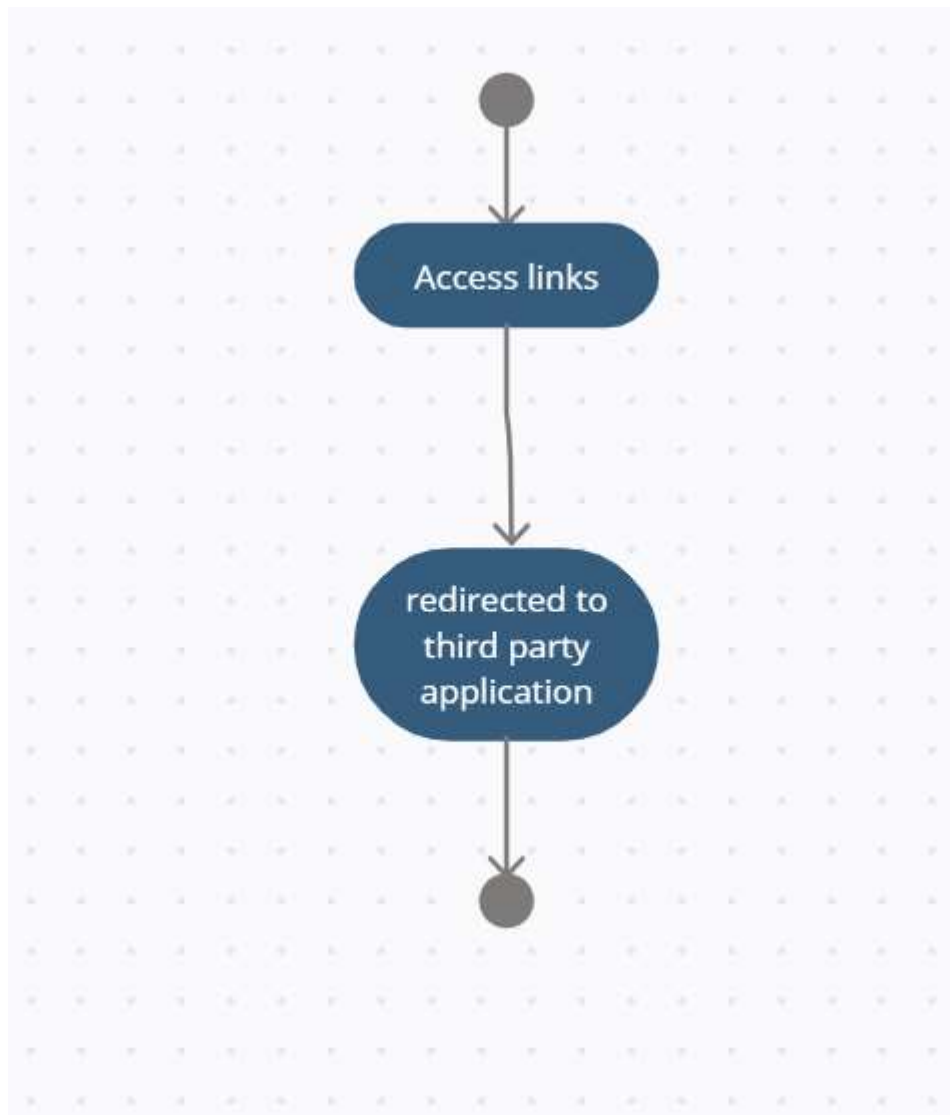
### Activity diagram:

Activity diagram is another important diagram in UML to describe the dynamic aspects of the system. Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system.

The control flow is drawn from one operation to another. This flow can be sequential, branched, or concurrent. Activity diagrams deal with all type of flow control by using different elements such as fork, join, etc

### Activity diagram for student:

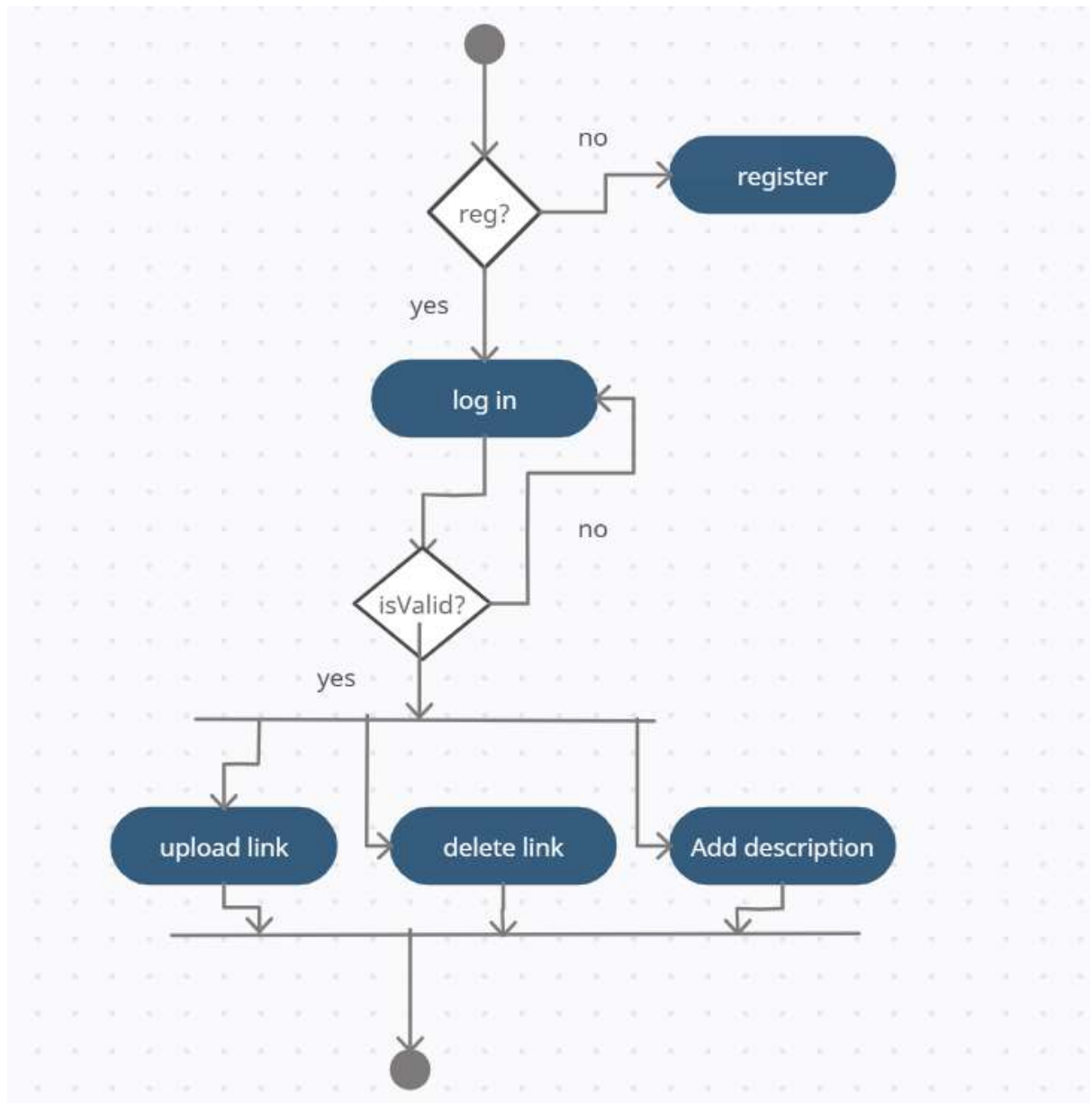
- Student can access all the links (can see) without sign in or sign up  
Student can only open the link if has its credentials or authorization



**Activity diagram for Teacher:**

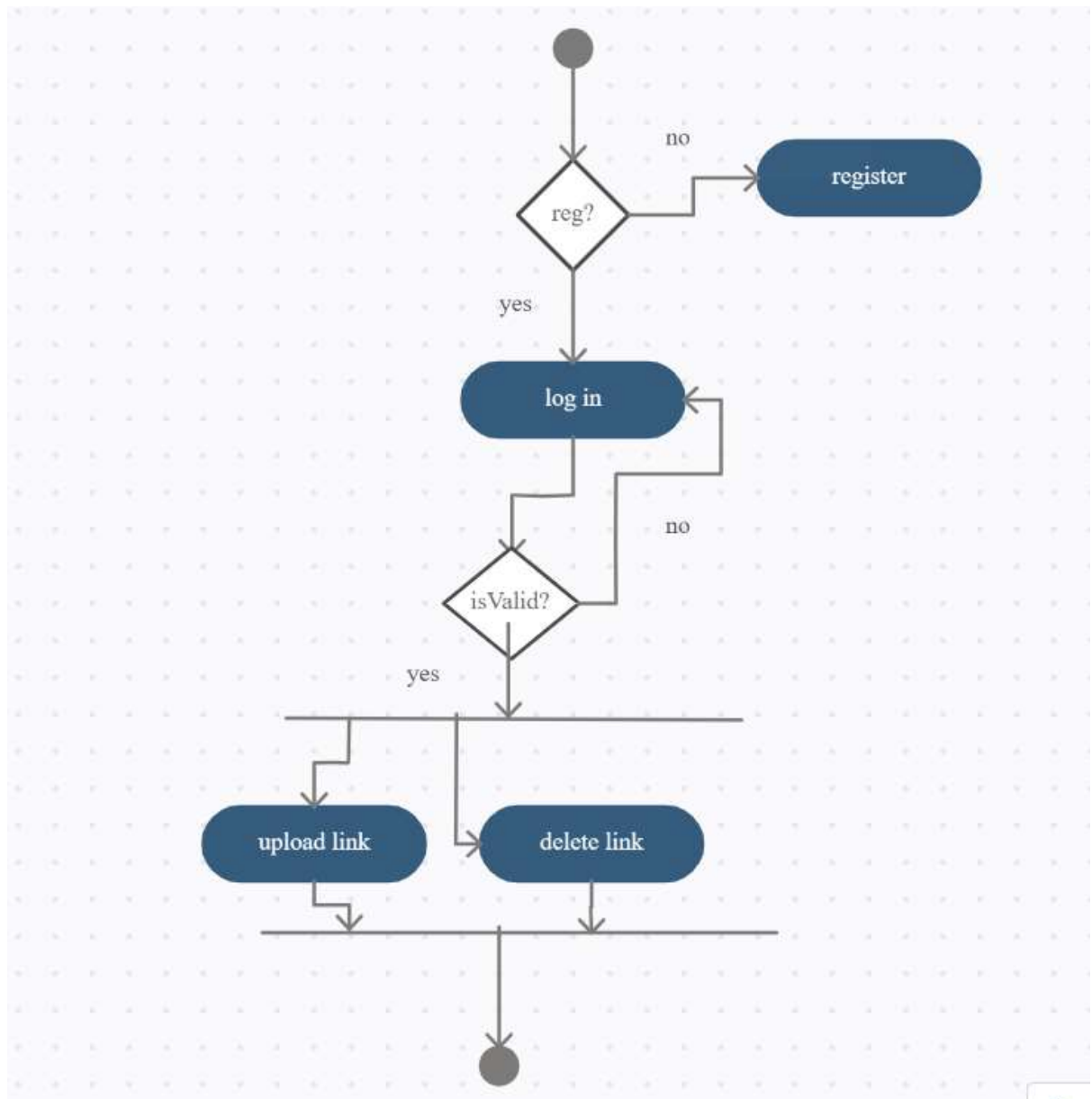
- Teacher have to log in if register
- If not Teacher have to register to the application
- Once the validation done Teacher can add or delete the links
- Teacher can add description for the link





**Activity diagram for Admin:**

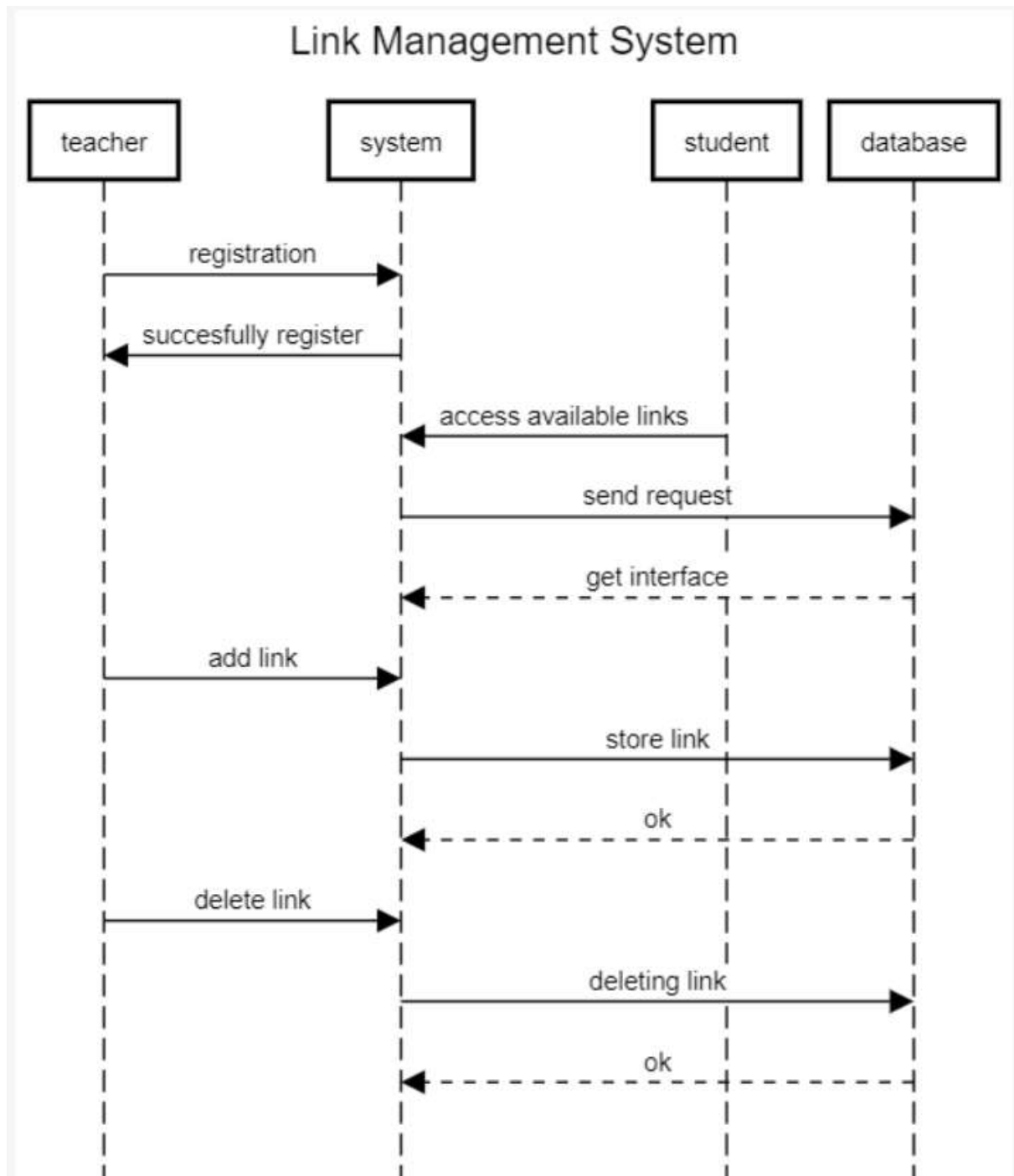
- Admin have to log in if register
- If not admin have to register to the application
- Once the validation done admin can add or delete the links



### Sequence diagram:

UML Sequence Diagrams are interaction diagrams that detail how operations are carried out. They capture the interaction between objects in the context of a collaboration. Sequence Diagrams are time focus and they show the order of the interaction visually by using the vertical axis of the diagram to represent time what messages are sent and when.

Sequence diagram for our system:



### Test Cases –

#### Test Cases for Teachers

- 1) **Registration of teacher –**

A teacher should be able to register himself/herself with the system.

#### Steps –

- 1) The teacher should select the option of teacher on the homepage

2) The teacher should enter his/her details and should create a password for future logins.

Expected Result –

A page displaying that the registration is successful.

- 2) **Login for the teacher –**

A teacher should be able to login himself/herself with the system.

Steps –

- 1) The teacher should select the option of teacher on the homepage
- 2) The teacher should enter his/her details and should be able to log into the system

Pre Condition –

The teacher should have already registered with the system

Expected Result –

A page displaying that the login is successful

- 3) **Adding, Editing and Modifying Links**

A teacher should be able to add, modify, or delete previously added links.

Steps –

- 1) The teacher should select the option of teacher on the homepage
- 2) After logging in the teacher should add the appropriate links with description.

Pre Condition –

The teacher should have already logged into the system

Expected Result –

A page displaying that the desired operation is successful.

### **Test cases for System Admin–**

All the test cases of teachers will be applicable for the System Administrators except that of editing the links

### **Test case for students –**

- 1) Viewing the content through the Links

A student should be able to view the links and also should be able to access the content.

Steps –

- 1) The student should select the option of student on the homepage
- 2) The student should open the portal and check if any new links have been added

Expected Result –

A page displaying all the links.

**Conclusion -**

From this report we come to the conclusion that object oriented approach to software development is quite convenient while designing an application before coding it.

This approach helps us to list down all the requirements we expect from the application and also the inputs which are needed to be fed into the application