A PROJECT REPORT

ON

“ONLINE ASSIGNMENT SYSTEM”

Submitted to

UNIVERSITY OF MUMBAI

In Partial Fulfilment of the Requirement for the Award of

BACHELOR’S DEGREE IN
COMPUTER ENGINEERING

BY

Tai Kainath Imtiyaz Rubina	16CO16
Ansari Rayan Ahmed Sajjad Farzana	17DCO65
Khan Juned Rais Shahida	16CO29

UNDER THE GUIDANCE OF
PROF. Muktar Ansari

DEPARTMENT OF COMPUTER ENGINEERING
Anjuman-I-Islam’s Kalsekar Technical Campus
SCHOOL OF ENGINEERING & TECHNOLOGY
Plot No. 2 3, Sector - 16, Near Thana Naka,
Khandagaon, New Panvel - 410206

2019-2020

AFFILIATED TO

UNIVERSITY OF MUMBAI

A PROJECT II REPORT
ON

"ONLINE ASSIGNMENT SYSTEM"

Submitted to
UNIVERSITY OF MUMBAI

In Partial Fulfilment of the Requirement for the Award of

BACHELOR'S DEGREE IN
COMPUTER ENGINEERING

BY

Tai Kainath Imtiyaz Rubina	16CO16
Ansari Rayan Ahmed Sajjad Farzana	17DCO65
Khan Juned Rais Shahida	16CO29

UNDER THE GUIDANCE OF
PROF. Muktar Ansari

DEPARTMENT OF COMPUTER ENGINEERING
Anjuman-I-Islam's Kalsekar Technical Campus
SCHOOL OF ENGINEERING & TECHNOLOGY
Plot No. 2 3, Sector - 16, Near Thana Naka,
Khandagaon, New Panvel - 410206

2019-2020
AFFILIATED TO

UNIVERSITY OF MUMBAI

Anjuman-i-Islam's Kalsekar Technical Campus

Department of Computer Engineering

SCHOOL OF ENGINEERING & TECHNOLOGY

Plot No. 2 3, Sector - 16, Near Thana Naka,
Khandagaon, New Panvel - 410206

CERTIFICATE

This is certify that the project entitled

"ONLINE ASSIGNMENT SYSTEM"

submitted by

Tai Kainath Imtiyaz Rubina	**16CO16**
Ansari Rayan Ahmed Sajjad Farzana	**17DCO65**
Khan Juned Rais Shahida	**16CO29**

is a record of bonafide work carried out by them, in the partial fulfilment of the requirement for the award of Degree of Bachelor of Engineering (Computer Engineering) at ***Anjuman-I-Islam's Kalsekar Technical Campus, Navi Mumbai*** under the University of MUMBAI. This work is done during year 2019-2020, under our guidance.

Date: / /

(Prof. Muktar Ansari)
Project Supervisor

(Prof.Kalpana R. Bodke)
Project Coordinator

(Prof.Tabrez Khan)
HOD, Computer Department

DR. ABDUL RAZAK HONNUTAGI
Director

External Examiner

Acknowledgements

We would like to take the opportunity to express our sincere thanks to our guide **Prof.Muktar Ansari**, Assistant Professor, Department of Computer Engineering, AIKTC, School of Engineering, Panvel for his invaluable support and guidance throughout our project research work. Without his kind guidance & support this was not possible.

We are grateful to him for his timely feedback which helped us to track and schedule the process effectively. His time, ideas and encouragement that he gave help us to complete our project efficiently.

We would like to express deepest appreciation towards **DR. ABDUL RAZAK HONNUTAGI**, Director, AIKTC, Navi Mumbai, **Prof. Tabrez Khan**, Head of Department of Computer Engineering and **Prof. Prof. Kalpana R. Bodke**, Project Coordinator whose invaluable guidance supported us in completing this project.

At last we must express our sincere heartfelt gratitude to all the staff members of Computer Engineering Department who helped us directly or indirectly during this course of work.

Tai Kainath Imtiyaz Rubina
Ansari Rayan Ahmed Sajjad Farzana
Khan Juned Rais Shahida

Project I Approval for Bachelor of Engineering

This project entitled *Önline Assignment System¨* by ***Tai Kainath Imtiyaz Rubina , Ansari Rayan Ahmed Sajjad Farzana and Khan Juned Rais Shahida*** is approved for the degree of ***Bachelor of Engineering in Department of Computer Engineering.***

Examiners

1. ....
2. ....

Supervisors

1. ....
2. ....

Chairman

....

Declaration

We declare that this written submission represents our ideas in our own words and where others ideas or words have been included, We have adequately cited and referenced the original sources. We also declare that We have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Tai Kainath Imtiyaz Rubina
16CO16

Ansari Rayan Ahmed Sajjad Farzana
17DCO65

Khan Juned Rais Shahida
16CO29

ABSTRACT

Online Assignment System is a web-based module which allows student and faculty to communicate about assignment and queries. Student can submit assignment and faculty can correct the answer and rate the best answer which would be uploaded as the best answer and the students can further refer that answer as notes. There will be certain keyword if the student's answer matches the maximum keyword provided by the faculty then the faculty will display that result as the best answer and the answer will be highlighted. Faculty can also generate question paper from the assignment questions. Many colleges are adopting this module as it is efficient and decrease the paper-work and human efforts.

Keywords: Web-based Assignment Submission; University Question and Answer Repository

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The problem arises from manual system used by colleges is that it has no record about the time or marks of student assignment. Some of the student hand over their assignment after the due date. Some of the assignments can be lost by students as well as faculty and as we do not have any record about it students have to write the assignment once again. As there is no record of assignment faculty would nd it difficult to check whether the student have submitted the assignment or not. After the faculty correct the assignment and return it back, some student does not get their assignment back as anyone can take it or misplace it, to avoid this we are creating an online assignment system. The main motive of this project is to create a paperless environment by developing online assignment submission. This project also reduces the unnecessary workload of lecturers and student. The scope for the Online Assignment System is that student can submit their assignment anytime and anywhere there is no need of the presences of lecturer as the student is submitting the assignment online. The collection of the assignments will be kept by the lecturer and the student can view their marks anytime they want. This system is very useful as well as innovative because there is no paper involved and even this system is user friendly. Student and Faculty can use this system anytime anywhere. The answer is corrected automatically with the help of automated search engine and the keyword provided by the faculty. Faculty does not need to check the assignment answer manually as all the checking will be done through key word matching. The best answer will be highlighted as the most rated answer and that answer will be referred as notes by the student. This system generated question paper from the assignment question which student can referred for their examination. The generation of question paper is done by faculty and the faculty has to select the question from assignment question which he/she is going to add in the question paper.

1.1 Purpose

The problem arises from manual system used by colleges is that it has no record about the time or marks of student assignment. Some of the student hand over their assignment after the due date. Some of the assignments can be lost by students as well as faculty and as we do not have any record about it students have to write the assignment once again. As there is no record of assignment faculty would find it difficult to check whether the student have submitted the assignment or not. After the faculty correct the assignment and return it back, some student does not get their assignment back as anyone can take it or misplace it, to avoid this we are creating a online assignment system. The main objective of this project is to create a paperless environment by developing online assignment submission. This project also reduces the unnecessary workload of lecturers and students

1.2 Project Scope

The scope for the Online Assignment System is that student can submit their assignment anytime and anywhere there is no need of the presences of lecturer as the student is submitting the assignment online. The collection of the assignments will be kept by the lecturer and the student can view their marks anytime they want.[2]

1.3 Project Goals and Objectives

1.3.1 Goals

1. To allow faculty members to add assignment online.

2. To generate question bank using assignment questions.

3. It also save papers.

1.3.2 Objectives

1. TO reduce wastage of papers.

2. Organized way of storing assignment and to reduce faculty and. students work load.

3. Evaluate students answer similarity with model answer.

4. Random question bank generation.

1.4 Organization of Report

Chapter **Introduction** shows how this idea popped up and motivation we got to develop this project. We checked if there any system exist for this problem. We found paper based and computer based system. We studied their advantages, disadvantages and got to know how we can build solution to overcome those disadvantages.

Chapter **Literature Survey** includes summary, advantages, disadvantages and ways we can improve those disadvantages of reference paper we studied. Review of literature helps to understand need of project, how project can improve situations and it helps developers to understand what exactly need to develop. Literature review helps clients to know in what areas project can be used.

Chapter **Project Planning** is given so that other developers or clients can know what technologies, tools and software its estimated development cost, expected profit can be known from this chapter. **System design** chapter is provided with six diagrams to understand modules, users and architecture of project. Use case diagram is given to understand functionality of a system with users and usecases. To visualize database ER diagram is shown. Class diagram is provided to understand structure of project and to understand how data is passing through modules Data Flow Diagram(DFD) is given.

Chapter **Implementation describes** each and every module of project in details. Also to understand interaction logic between object in system sequence diagram is shown. Activity diagram shows control flow from one activity to another. Flow chart for every module is given that shows overall structure of the process or system, traces the flow of information and work through it, and highlights key processing and decision points.

Chapter **System Testing** and Screenshots of project discusses Test cases used for testing the system, to check validation. The results occurred are given in this chapter. The analysis done after development is described here.

Last chapter **Conclusion and Future Scope** describes how we can make project scope more broad. What are the limitations of system and conclusion.

Chapter 2

Literature Survey

2.1 Question Paper Generator and Answer Verifier

In these paper, there is three module
Login/Registration Module: In these if the user wish to use the system he/she has to login first.
The Question Addition Module: In these faculty will adds Assignment question and the student has to answer it in given time period and the respective faculty has to also add the answer in his/her own words.
Paper Assignment Generation Module: The teacher will have to select the subject, and the number of questions that have to be added to the assignment.

2.1.1 Advantages of Paper

a. This module will be instantly activated once the student has completed solving all the questions and has submitted the assignment.

2.1.2 Disadvantages of Paper

a. The system corrects the answer as the length of the answer is such that the length of the answer is not appropriate, indicating that the student was not able to write important information and if the length of answer is too long it indicates that the student has filled too much information respectively.

b. If length of answer written by student matches the required length, then the student will get suitable marks or else student will get less marks.

2.1.3 How to overcome the problems mentioned in Paper

a. Our system corrects the answer in the form of keywords matching if the student answer matches the maximum keyword then the answer is displayed as correct.

b. Our system is irrespective of the length, only checks the answer based on keyword.

2.2 Design of Adaptive Question Bank Development and Management System

In this paper, we are designing an adaptive question bank management system that is intelligently picking questions from rich database (question bank) and representing the question model according to the inputs or parameters provided by the question paper designer (QPD). This question bank uses concept map developed using graphical method that uses hierarchical knowledge of particularr domain. The concept map integrated with question bank (question database) will ensure the question modeling process based on degree of certain criteria like Bloom's Taxonomy, difficulty level, marking scheme etc.

2.2.1 Advantages of Paper

a. This intelligence will reduce the pressure from the question paper designer and will free them from responsibility of setting learning objectives properly.

2.2.2 Disadvantages of Paper

a. This system contains questions which is added to the question bank. They are divided into different types like question based on points, difficulty level, type and subject.

2.2.3 How to overcome the problems mentioned in Paper

a. Our system will provide all sort of question, it depend on student which type of question he/she will attempt.

b. Student does not have to add various dimensions, they have to only select the courses and the assignment will be displayed.

2.3 Smart University-Student Information Management System.

The proposed system provides one single point of contact for all administrative system of university. Improved communication gap between university and students. Student can access all his information at just one tap on his smartphone. The system offers reliability, time savings and easy control. It can be used as a base for creating and enhancing applications for all above mentioned modules. This will greatly simplify and speed up the result preparation and management process. It reduces the work and resources required in traditional process. The proposed system provides the new way of computing and displaying operations with responsive and attractive user interface on mobile devices.

2.3.1 Advantages of Paper

a. This system will greatly simplify and speed up the organization and management process.

b. Human errors in calculation are eliminated.

2.3.2 Disadvantages of Paper

a. In this system student can challenge the lecturer by demanding the photocopy of the answer sheet and can also apply for revaluation based on the result provided by to them

2.3.3 How to overcome the problems mentioned in Paper

a. Our system only checks the answer based on key- word matching, the answer will be displayed on the dashboard on need of demanding the photocopy

b. It reduces human error

c. It eliminates paper work. Hence, our system is eco-friendly.

2.4 Technical Review

Our application is fabricated with following technologies : for back-end – Django 2.2.5, for Webscraping – beautifulsoup 4.8.2,Google 2.0 for cosine similarity ntlk 3.4.5, for front-end – web technologies.

2.4.1 Django.

Django is a high-level Python Web framework that encourages rapid development and clean, pragmatic design. Built by experienced developers, it takes care of much of the hassle of Web development, so you can focus on writing your app without needing to reinvent the wheel. It's free and open source.

Reasons to use Django.

a. Simple syntax.

b. Easily Understandable

c. Extensive Support Libraries.

d. Integration Feature.

2.4.2 Google 2.0

Google API is a library that helps in searching a query and fetching the relevant links. It provides a search function which takes in parameters like search query, number of results, etc. We used google to search question title and get relevant links. This links are then scraped with beautiful soup.

Reasons to use Google 2.0.

a. Simple

b. Instead of putting so much effort for a trivial task google package has been made.

c. Its almost a one liner solution to find links of all the google search result directly.

2.4.3 Beautifulsoup 4.8.2

Beautiful Soup is a library that makes it easy to scrape information from web pages. It sits a top an HTML or XML parser, providing Pythonic idioms for iterating, searching, and modifying the parse tree.

Reason to use Beautifulsoup 4.8.2

a. It provides a few simple methods.

b. It doesn't take much code to write an application.

c. Beautiful Soup sits on top of popular Python.

2.4.4 NLTK 3.4.5

The Natural Language Toolkit (NLTK) is a platform used for building Python programs that work with human language data for applying in statistical natural language processing (NLP). It contains text processing libraries for tokenization,parsing, classification, stemming, tagging and semantic reasoning. It also includesgraphical demonstrations and sample data sets as well as accompanied by a cook book and a book which explains the principles behind the underlying language processing tasks that NLTK supports.

Reason to use NTLK 3.4.5

a. Easily Understandable

b. Extensive Support Libraries.

c. Integration Feature.

2.4.5 Web Technologies

Web Technologies is a combination of many languages like HTML, CSS, Js, SQL,PHP, Bootstrap etc.

Reasons to use Web Technologies.

a. Web Technologies are programming languages such as HTML and CSS, which are well known among IT professionals.

b. They run on the device's own web browser through a simple URL.

c. It run on any operating system.

d. Dynamic and Interactive Web pages.

e. Responsive Websites.

Chapter 3

Project Planning

3.1 Members and Capabilities

Table 3.1: Table of Capabilities

SR. No	Name of Member	Capabilities
1	Juned	Python,Database
2	Rayan	Database,UI Design
3	Kainath	Python,UI Desgin

3.2 Roles and Responsibilities

Table 3.2: Table of Responsibilities

SR. No	Name of Member	Role	Responsibilities
1	Juned	Co-Team Leader	Algorithm Design
2	Rayan	Co-Team Leader	UI Design
3	Kainath	Co-Team Leader	Algorithum Desgin

3.3 Assumptions and Constraints

Assumptions:

1. User types will be faculty and students.
2. All the resources will be provided during development
3. Consistent feedback from mentor during development
4. Users will have satisfactory hardware and software required to run the application

Constraints:

1. Since most of the technologies used in project is open source development budget constraint is very low

2. PaaS hosting which supports application developed with Python
3. The user should be able to understand how the system works.

3.4 Project Management Approach

We have use waterfall model for the development of our project. In the Waterfall approach, the whole process of software development is divided into separate phases. The outcome of one phase acts as the input for the next phase sequentially. This means that any phase in the development process begins only if the previous phase is complete. The waterfall model is a sequential design process in which progress is seen as flowing steadily downwards (like a waterfall) through the phases of Analysis, Design, Construction, Testing, Production/Implementation and Maintenance.

3.5 Ground Rules for the Project

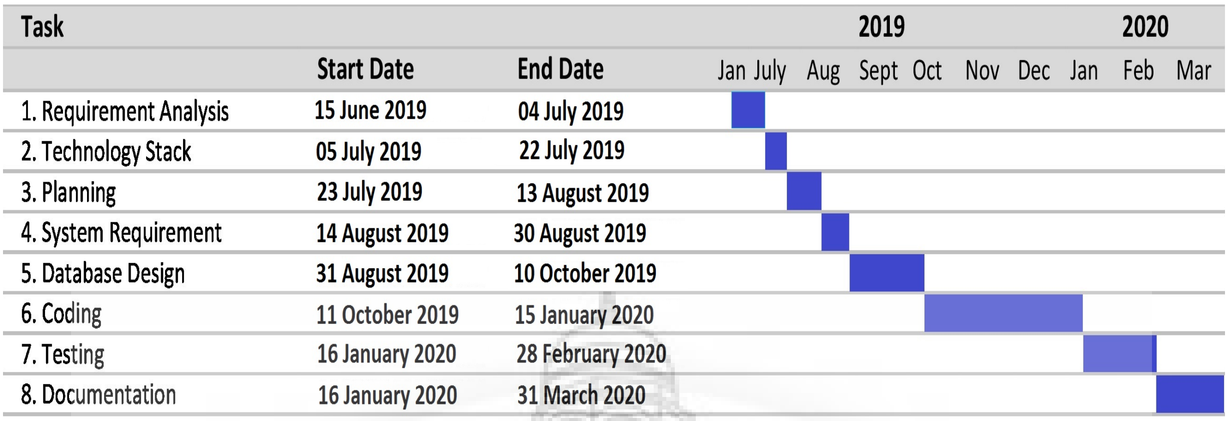
We consider the following ground rules:

1. Project should also build from user prospective
2. We make sure that everyone in the group understands in the group understands and no one left behind
3. We make sure everyone in the group will initiate by sharing ideas, how to solve problems and things need to be improved
4. We will be honest and ,we will try are best to complete the project before deadline.

3.6 Project Budget

- Web Services: Open Source
- Client side validation:Open Source
- Database: Open source
- Libraries like Spacy , Gensim for text comparison :Open Source

3.7 Project Timeline



Task	Start Date	End Date
1. Requirement Analysis	15 June 2019	04 July 2019
2. Technology Stack	05 July 2019	22 July 2019
3. Planning	23 July 2019	13 August 2019
4. System Requirement	14 August 2019	30 August 2019
5. Database Design	31 August 2019	10 October 2019
6. Coding	11 October 2019	15 January 2020
7. Testing	16 January 2020	28 February 2020
8. Documentation	16 January 2020	31 March 2020

Chapter 4

Software Requirements Specification

4.1 Overall Description

4.1.1 Product Perspective

Online Assignment System is a web based system implementing client-server model. Online Assignment System provides simple mechanism for solving problem related to assignment management. This system can act like an online answers repository during exam preparations.

4.1.2 Product Features

The following are the main features that are included in Online Assignment System:

- **Faculty Dashboard:** Allowing faculty members to add course and upload assignments, generate model answers and question banks using assignments. Faculty can also view all submitted answers from students of each question. Similarity of student answers with the model answer is also visible.
- **Student Dashboard**: The student dashboard allows students to easily add and update answers of the assignments. They can put their answers in draft or published status depending on their choice. Assignments are fetched with respect to students branch and year. Filter answers feature on main page allows students to view questions by filtering it with respect to course.
- Login and registration page allows new users to register themselves and start using application without any hassle.
- **Similarity and Model answer**: This module allows faculty to generate model answer from different website, and if the model answer is not automatically generated then faculty members can add model answers manually. Similarly uses cosine similarity algorithm to check similarity between student's answer and faculty's answer.

- Students can refer to the submitted answers later if required through the main page without login.

4.1.3 User Classes and Characteristics

It is considered that the users do have the basic knowledge of operating the internet and to have access to it.

User Types:

STUDENT:

- Student Login.
- Add Answers.
- Update Answers.
- Delete Answers.
- View Assignments.
- Update answer status- Draft/Publish.

FACULTY:

- Add course
- Add assignments.
- Add Questions
- Generate Model Answer
- View submitted answers.
- reject Student answers
- view answer similarity
- Generate Question Bank and print.

ADMIN

- Add Years and branches
- Manage users.
- Manage database.

4.1.4 Operating Environment

Operating environment for the Online Assignment System is as listed below.
The hardware, software and technology used should have following specifications:

- Computer with minimum 12 Mbps or above internet connection.
- 4 GB RAM or above
- Pentium processor with speed of 3 GHz.
- Operating system: Windows 7 or above / Linux Mint.
- database: sqlite3.
- platform: Python-Django.
- Web browsers: Chrome or Mozilla Firefox.

4.1.5 Design and Implementation Constraints

The information in Online Assignment System must be stored in the database that is accessible by the website.

- SQ Lite3 will be used as SQL engine and database.
- The Online Assignment System is running 24 hours a day.
- Users may access from any computer that has latest internet browser like Google Chrome or Mozilla Firefox.
- Users must have their correct usernames and passwords to enter into their online accounts and do actions.

4.2 System Feature

4.2.1 System Feature

Description and Priority

1. First priority goes to the login and registration form. users will register according to its user type i.e Faculty and Student type.
2. Second priority is to provide dashboard to the user depending on its user type.
3. Student will have access to Student Dashboard and Faculty will have access to Faculty Dashboard
4. In the Dashboard, only those content will be fetched from the database which is accessible to the user.

5. Different options on the left side of dashboard will be provided to the user, so he can do necessary task.

Functional Requirements

Student:

- Student Login.
- Add Answers.
- Update Answers.
- Delete Answers.
- View Assignments.
- Update answer status- Draft/Publish.

Faculty:

- Add course
- Add assignments.
- Add Questions
- Generate Model Answer
- View submitted answers.
- reject Student answers
- view answer similarity
- Generate Question Bank and print.

Admin:

- Add Years and branches
- Manage users.
- Manage database.

4.2.2 Module Description

Types of user:

- Students
- Faculty
- Admin

- **Faculty Module:**
  (a) Add Course : This module will allow faculty members to add their course
  (b) Add Assignments: It will allow faculty members to add assignments of course
  (c) Add Questions: This will allow to add questions in assignments.
  (d) Generate Model answer: Generate model answer will search question-title on google and fetch relevant information and store it in model answer field.
  (e) View submitted answers: It will display student roll number, name, answer and similarly score compared to model answer. it will also provide REJECT button to reject student's answer
  (f) Generate Question Bank: This module will allow faculty members to generate and print question bank. Question bank generator will randomly select questions from selected assignments

- **ADMIN Module:**
  (a) Admin will have access to Django Admin. Admin will have granular control over all the data in database.
  (b) It can also manage users.

- **Student Module:**
  (a) Add Answer: Students will be able to add answers through this module. They will select assignments and add answers.
  (b) Update Answer: Students can update their answers through this module.
  (c) Delete Answer: Students can delete answers through this module.
  (d) Answer Status: Draft/Publish status allow students to hide or display their answer from other users.
  (e) View Assignments: This will allow students to view all the assignments.

4.3 External Interface Requirements

4.3.1 User Interfaces

The user interface is very user-friendly and intuitive. It will allow to use Online Assignment System with minimal training. Most of the the functions are implemented in such a way that they are self explanatory. Faculty dashboard is very intuitive which allow faculty members to manage assignment very easily. Each Faculty member can view its course, assignments and questions and perform required actions.Student dashboard allows students to easily add answers to the assignments with ease.

4.3.2 Hardware Interfaces

Since this application is cloud-hosted, the hardware requirements are very minimal.Any device with latest configurations and internet will be able access Online Assignment System. Though computer or laptop is recommended for ease of input.

- 4 GB RAM or above
- Pentium 3 GHz processor
- Good internet connection

4.3.3 Software Interfaces

- Windows 7 or above or equivalent.
- Latest web browser like Google Chrome or Mozilla Firefox to load and view web pages.

4.3.4 Communications Interfaces

- Good internet connection is mandatory to access the application.
- Keyboard and mouse for input.

4.4 Nonfunctional Requirements

4.4.1 Performance Requirements

- It must be able to handle many users at a given time.
- Fast client-server data communication for faster access to website
- The overall performance of the software will reliable and enable the users to work efficiently

- saved in the database and thus only authenticated users are allowed to the login
- Lower risk of data-loss from database.

4.4.2 Safety Requirements

If there is extensive damage to a wide portion of the database due to catastrophic failure, such as a disk crash, all the data will be remain safe. The database is available in root of application source code which can be saved from time to time by admin.

4.4.3 Security Requirements

- System will use secured database.
- Users can view or edit only their content. They cannot modify other user's content without login details of other user.
- System will have different types of users and every user has access constraints

Chapter 5

System Design

5.1 System Requirements Definition

The overall performance of the software will reliable and enable the users to work efficiently. It must be able to handle many users at a given time. System will have different types of users and every user has access constraints. It also provide fast client-server data communication for faster access to website. It provide lower risk of data-loss from database. Users can view or edit only their content, they cannot modify other user's content without login details of other user.

5.1.1 Functional requirements

1. First users i.e. faculty/student should register to the system
2. After successful registration the user will login to the system
3. **Faculty Dashboard:-**
   (a) Faculty can add/update/delete course with respect to branch.
   (b) Faculty can add/update/delete assignment with respect to course.
   (c) Faculty can add/update/delete question with respect to assignment.
4. **Student Dashboard :-**
   (a) The dashboard will display all the course with respect to year and branch.
   (b) Student can select their course.
   (c) Student can select their assignment with respect to course.
   (d) Student can select question and can add answer accordingly.
5. **Assignment Repository:-**
   (a) All added question will display.
   (b) All added answer with respect to question after selecting question will display.

(c) Evaluation of answer using natural language processing.

(d) Question bank will be generated from the assignment

Use-case Diagram

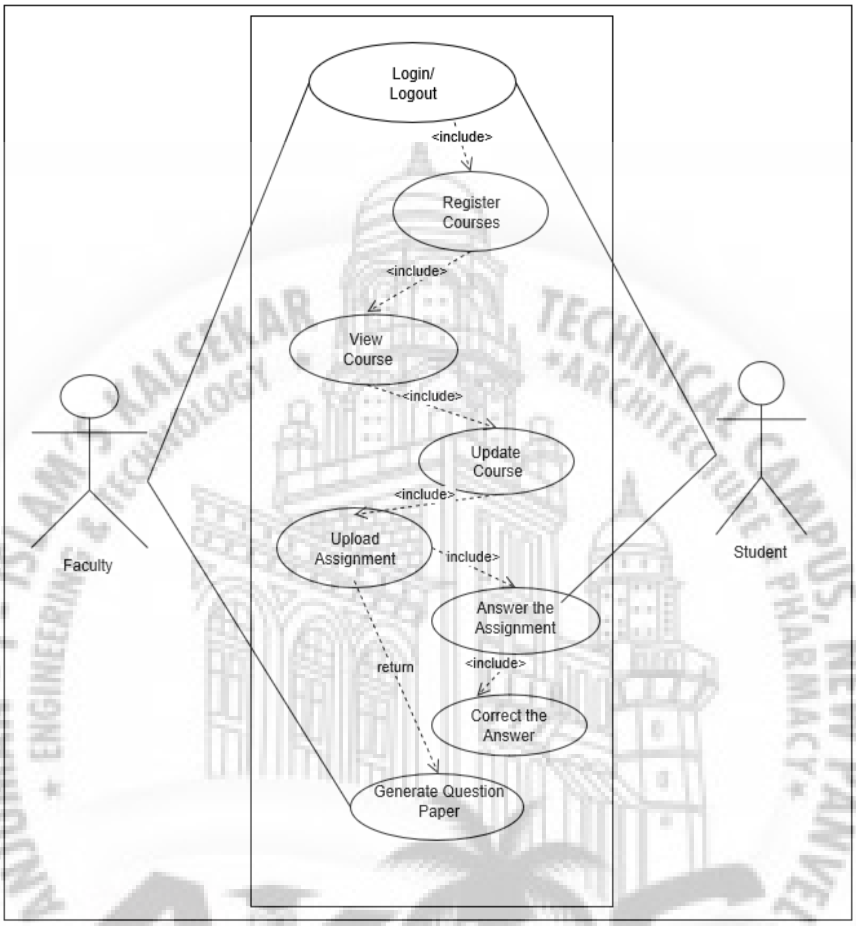


Figure 5.1: Use Case for Online Assignment System

Use case diagram is given to understand the functionality of the system with users.

1. Faculty/student will login into the system.

2. They can register into various course according to their semester.

3. They can view and update course accordingly .

4. Faculty in charge to the particular course can upload the assignment.

5. Student can answer the assignment and accordingly to keywords answer will be corrected.

6. Through assignment a question paper will be generated .

5.2 System Architecture Design

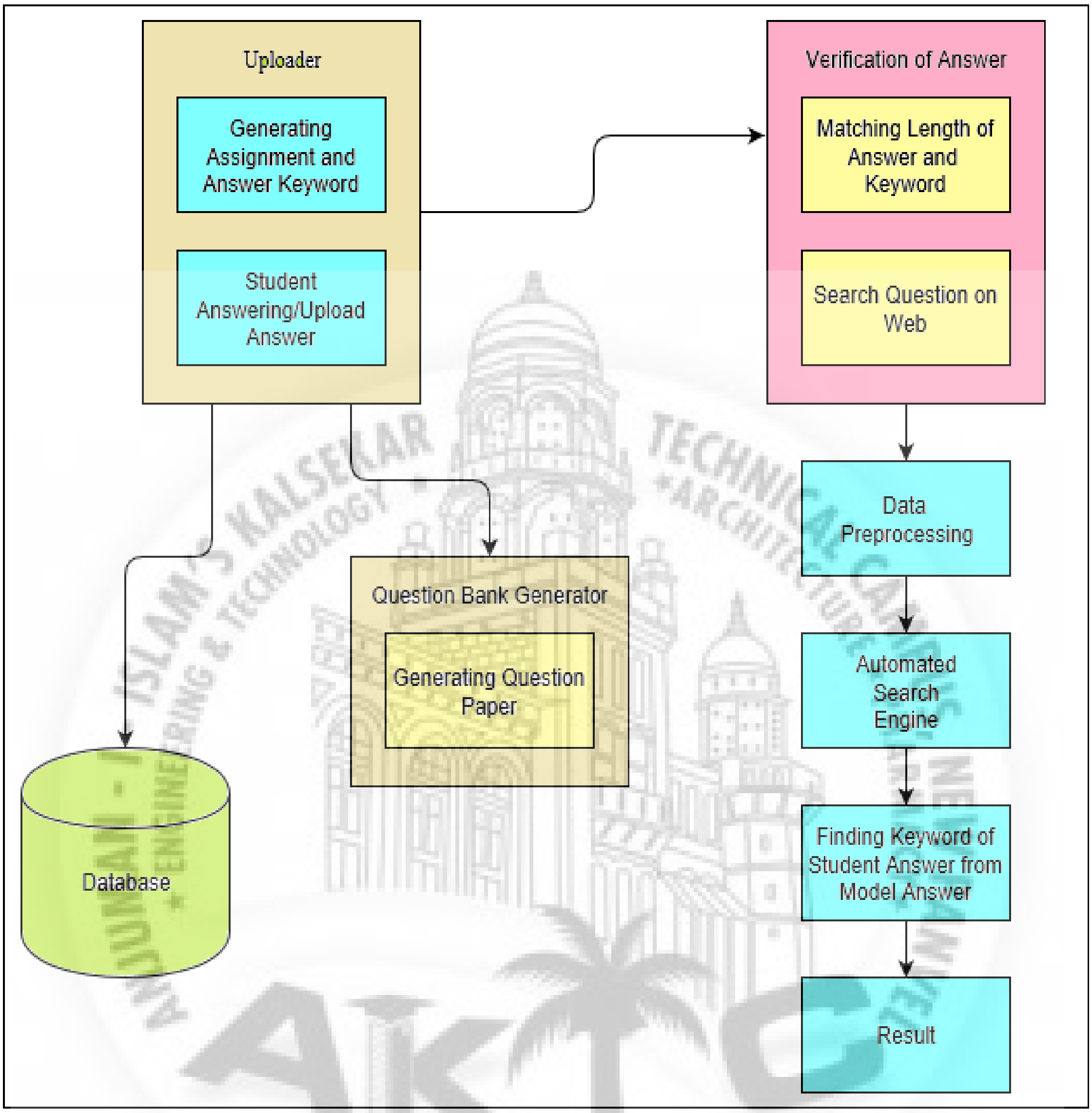

Uploader
Generating Assignment and Answer Keyword
Student Answering/Upload Answer
Verification of Answer
Matching Length of Answer and Keyword
Search Question on Web
Data Preprocessing
Question Bank Generator
Generating Question Paper
Automated Search Engine
Database
Finding Keyword of Student Answer from Model Answer
Result

Figure 5.2: System Architecture

Question Bank Generator:-
Question bank Generator helps faculty members to generate Question Bank by selecting uploaded assignment questions. Depending on the number of questions selected, questions will be randomly taken from the selected assignment and inserted in the question bank
Verification of Answer:-
In this module, we evaluate students answer by com- paring it with the model answer. The comparison is done with the help of cosine similarity algorithm. After the evaluation, the accuracy measure will be returned for that answer.
The Model Answer is generated in following steps:
●The question is searched over the web with the help of Google Search API.

• The result returned from the Google Search API contains links of many websites. This links will be parsed in PRE-PROCESSING model. The content of the websites will be extracted, stripped off their html tags and other irrelevant contents and later merged together. After pre-processing, the model answer is stored in the database for comparsion with student answer.

5.3 Sub-system Development

Mainly there are three modules in our system that is Admin, Faculty and Student.

5.3.1 Student

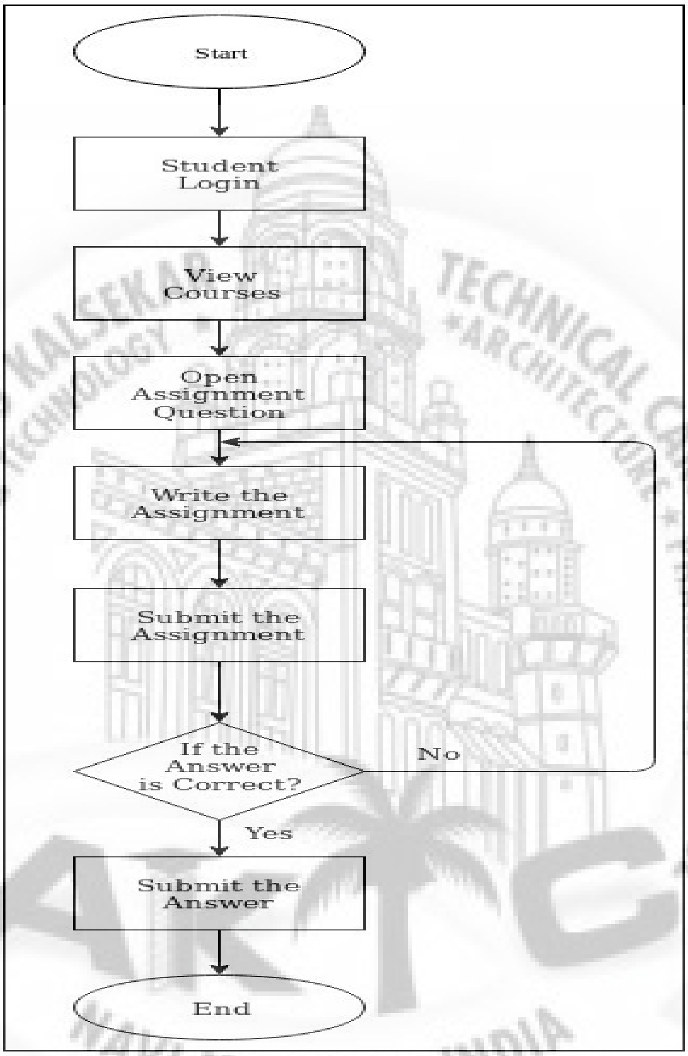


Figure 5.3: Flowchart chart for Student

Flow Chart for Student showing the work flow of student about how the student answers the assignment and if the answers is incorrect he/she has to again write the answers.

5.3.2 Faculty

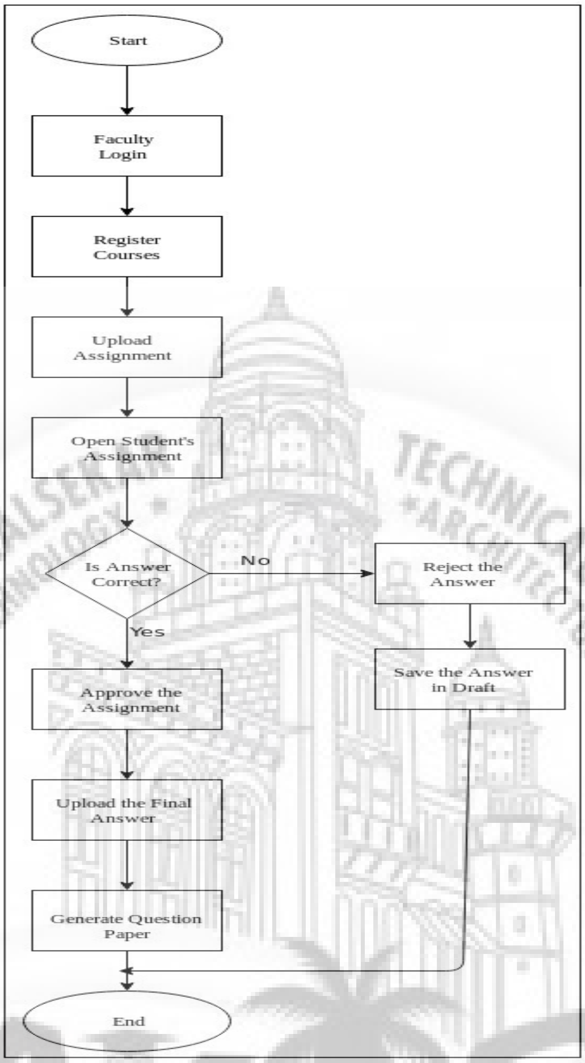


Figure 5.4: Flowchart chart for Faculty

Flow Chart for faculty is showing the work flow of faculty, how the faculty uploads assignment question, corrects the assignment answer and even generates the question paper.

5.4 Systems Integration

In order to achieve the goal of system the developed modules need to be integrated properly with other modules. The admin module ,faculty module and student module should be integrated properly in order to achieve our final output that is it should generate a question bank through assignment and verify the answer according to the keywords.

5.4.1 Class Diagram

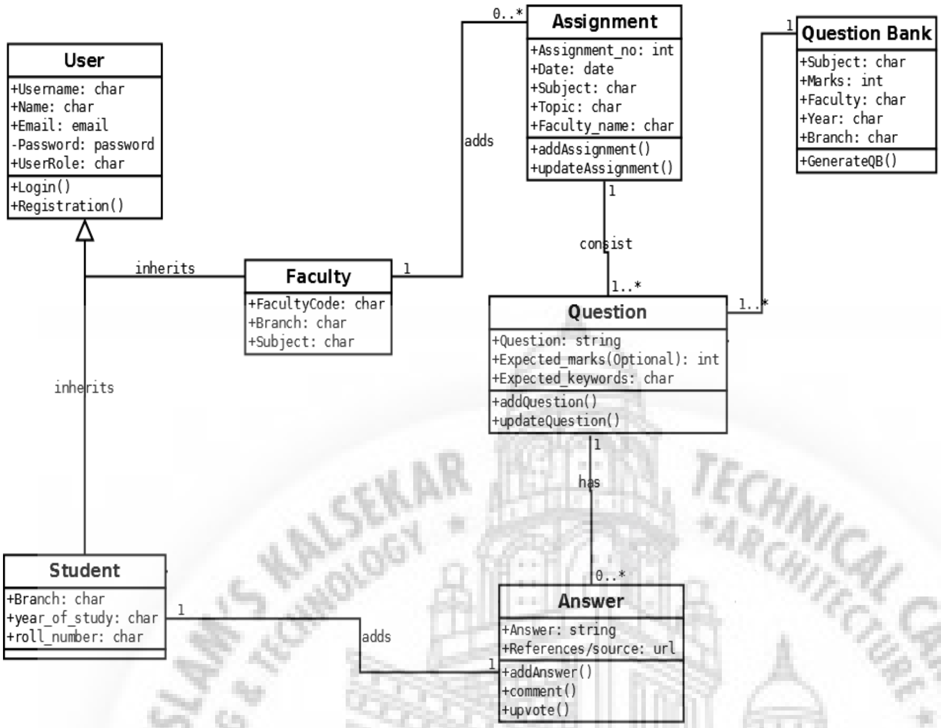


Figure 5.5: Class Diagram

Class diagram showing each modules interconnection and relation between how one module is interacting with others.

5.4.2 Data Flow Diagram

A data-flow diagram (DFD) is a way of representing a flow of a data of a process or a system. The DFD also provides information about the outputs and inputs of each entity and the process itself

- DFD Level 0

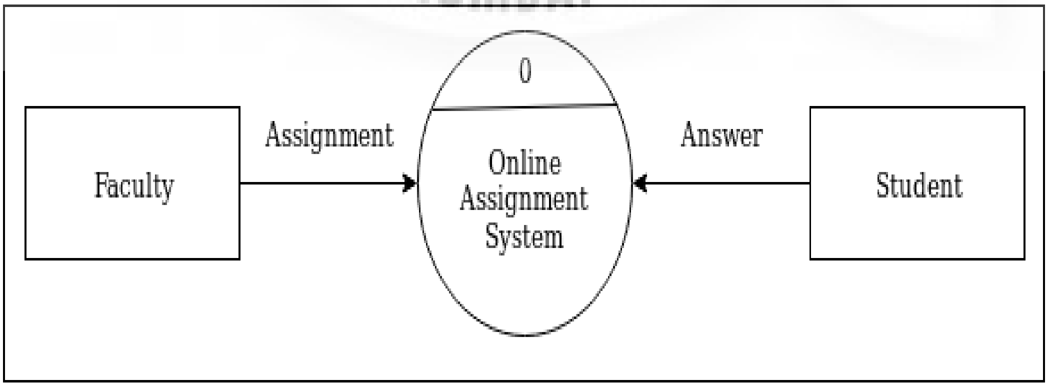


Figure 5.6: DFD Level 0 for Online Assignment System

DFD Level 0 depicting main outcome of the system, faculty provides assignment question to the student and the students answers the assignment question

- DFD Level 1

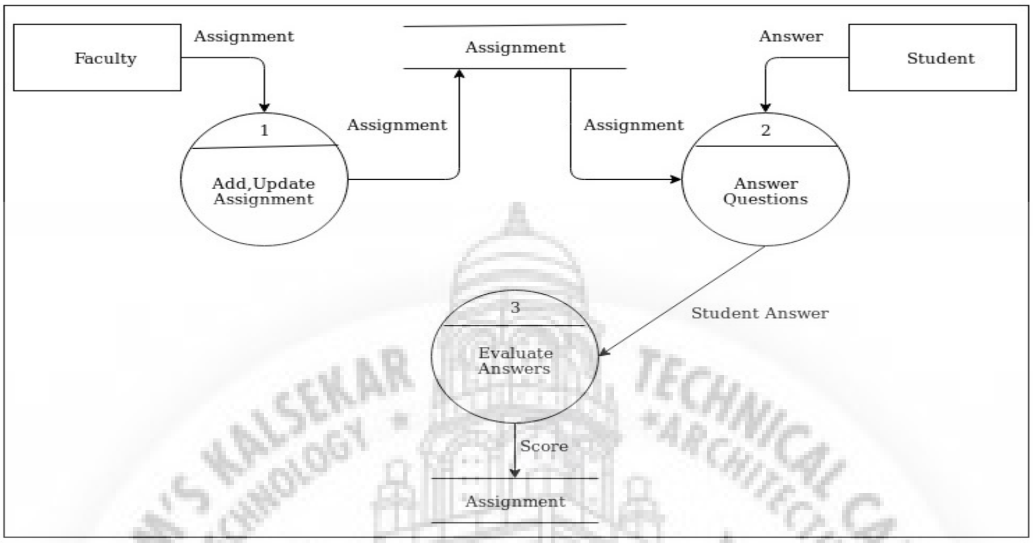


Figure 5.7: DFD Level 1 for Online Assignment System

DFD Level 1 showing the main process flow of student and faculty i.e how the faculty adds the assignment, how student response the answers and hoe the answer is evaluated and all the details is stored in Database

- DFD level 2

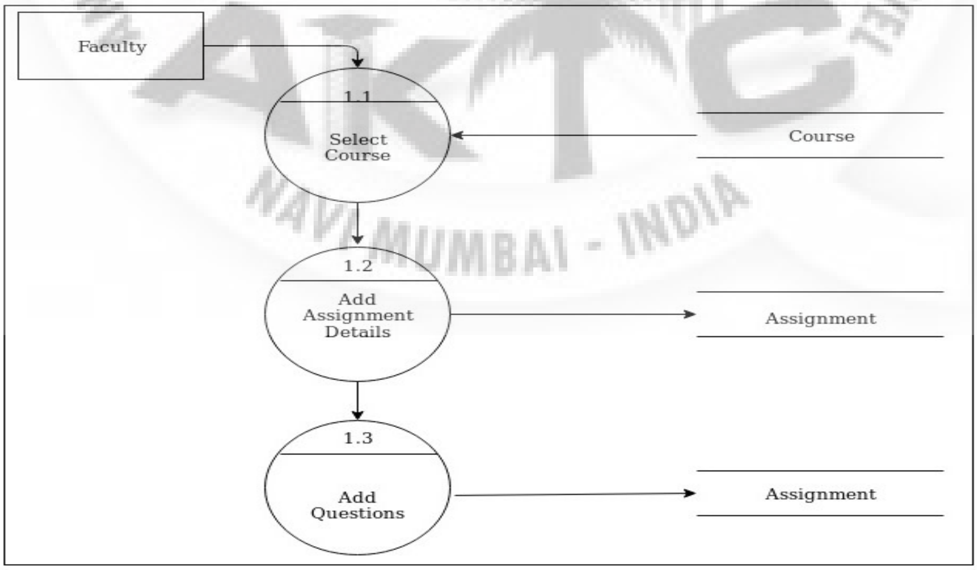


Figure 5.8: DFD Level 2 for Faculty

DFD Level 2 for faculty showing how the main process flow about how the faculty adds assignment details, and also add questions

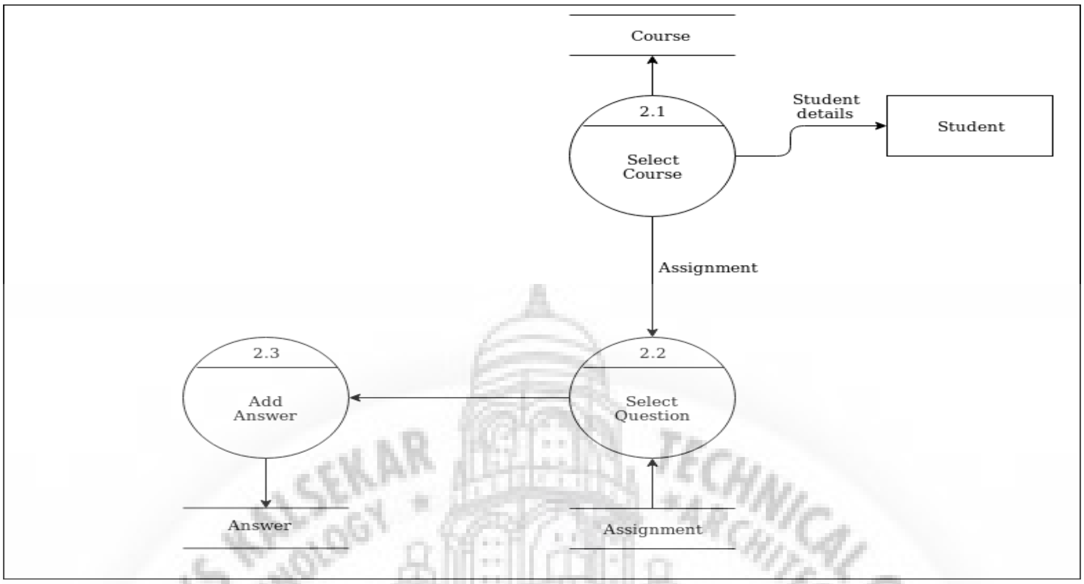


Figure 5.9: DFD Level 2 for Student

DFD Level 2 for student showing the process flow of how student will select the course, select question and adds answer

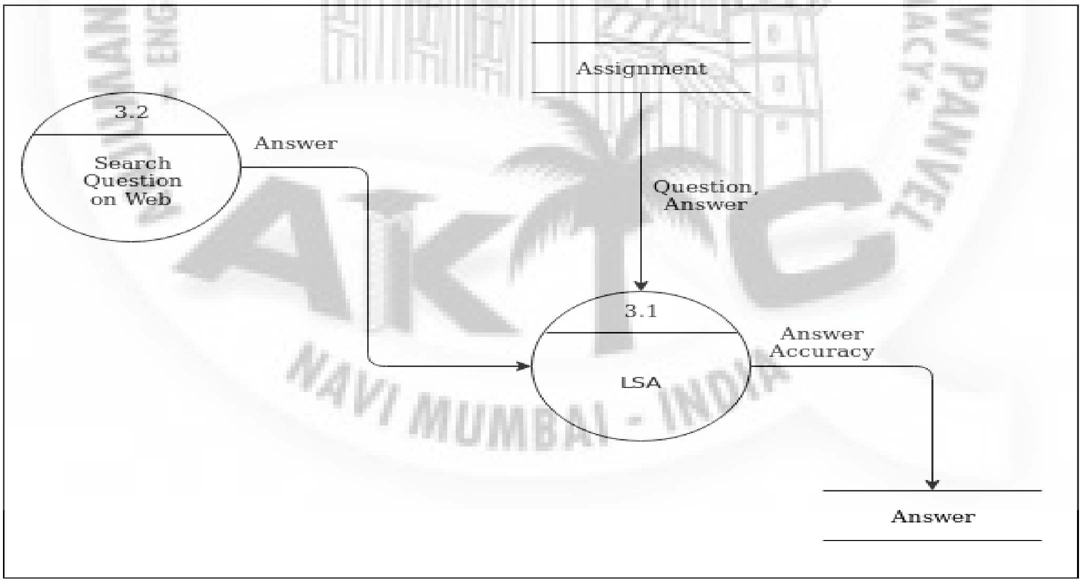


Figure 5.10: DFD Level 2 for Online Assignment System

After uploading the answer system will automatically evaluate the answer using the inbuilt auto evaluation method. Evaluation system will search the question on web and then compare it with the answer and then it will display its accuracy

5.4.3 Sequence Diagram

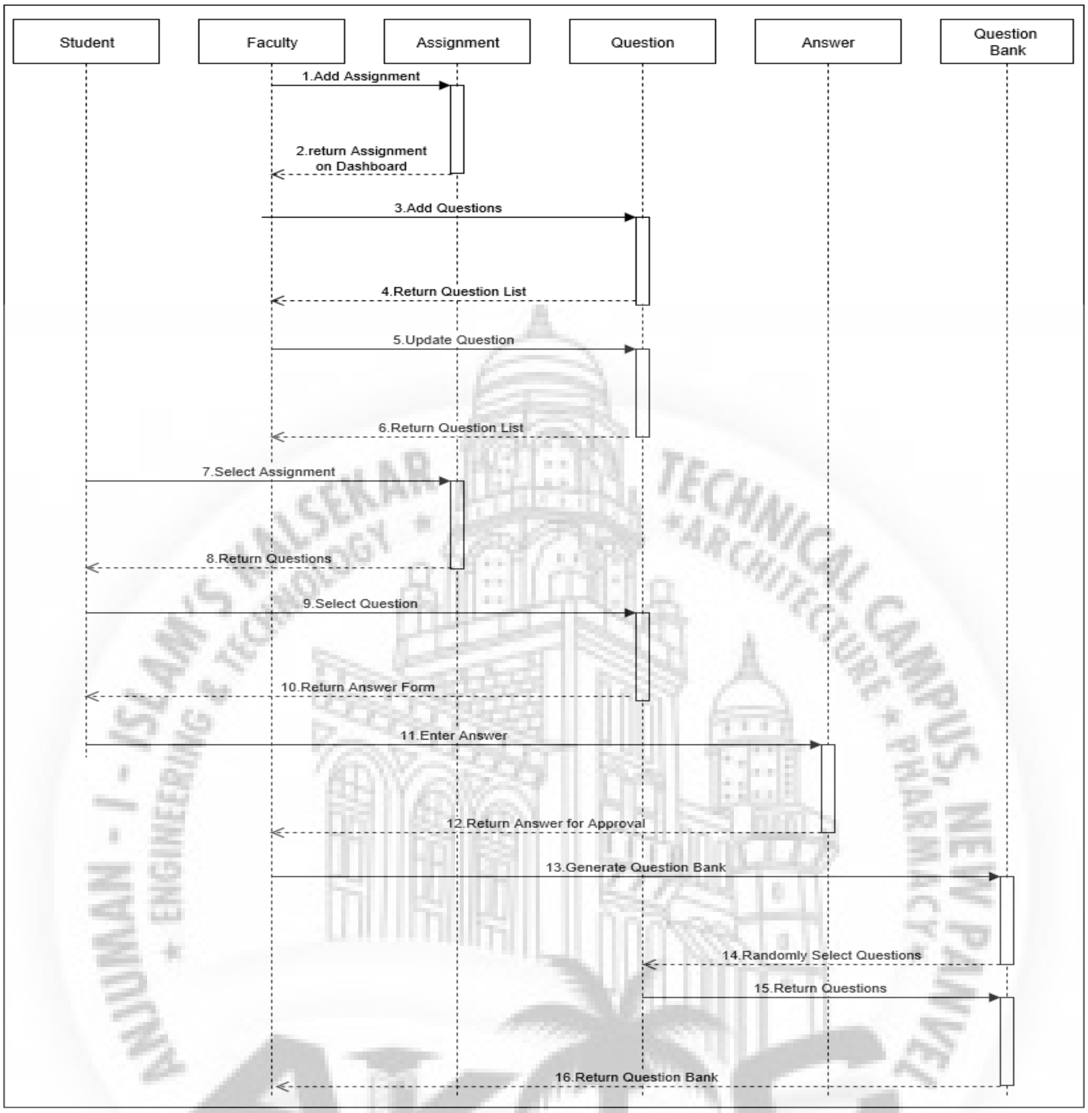


Figure 5.11: Sequence diagram for Online Assignment System

Sequence Diagram for Online Assignment System showing that how the process flow of faculty and student is working in sequence .

Chapter 6

Implementation

Login.

There are three modules in our system,to get into the system the user must first login

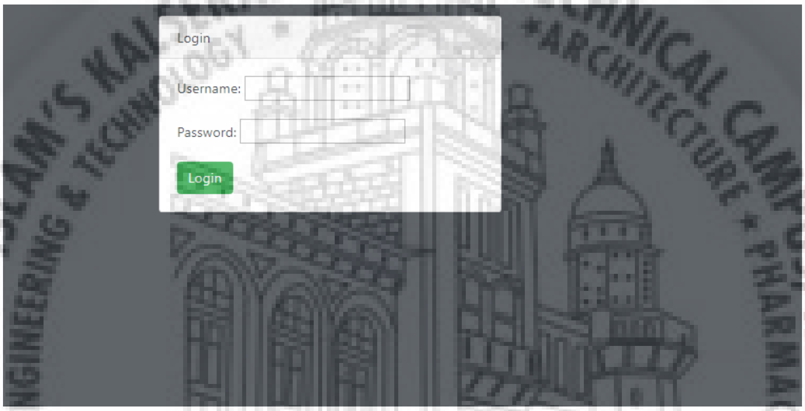


Figure 6.1: Login.

```
// ////////////////////////////// Templates //////////////////////////////////
<!DOCTYPE html>
<html lang=”en”>
<head>
 <meta charset=”utf-8”>
  <meta http-equiv=”X-UA-Compatible” content=”IE=edge”>
  <meta name=”viewport” content=”width=device-width, initial-scale=1, shrink-to-
      fit=no”>
  <meta name=”description” content=””>
  <meta name=”author” content=””>
  <title>OAS-Login</title>
  {% load static %}
  <!-- Custom fonts for this template -->
  <link href=”{% static ’vendor/fontawesome-free/css/all.min.css’ %}” rel=”
      stylesheet” type=”text/css”>
  <!-- Custom styles for this template -->
  <link href=”{% static ’css/sb-admin.css’ %}” rel=”stylesheet”>
</head>
<body class=”bg-dark”>
  <div class=”container”>
    <div class=”card card-login mx-auto mt-5”>
      <div class=”card-header”>Login</div>
      <div class=”card-body”>
          <form method=”post”>
            {% csrf_token %}
```

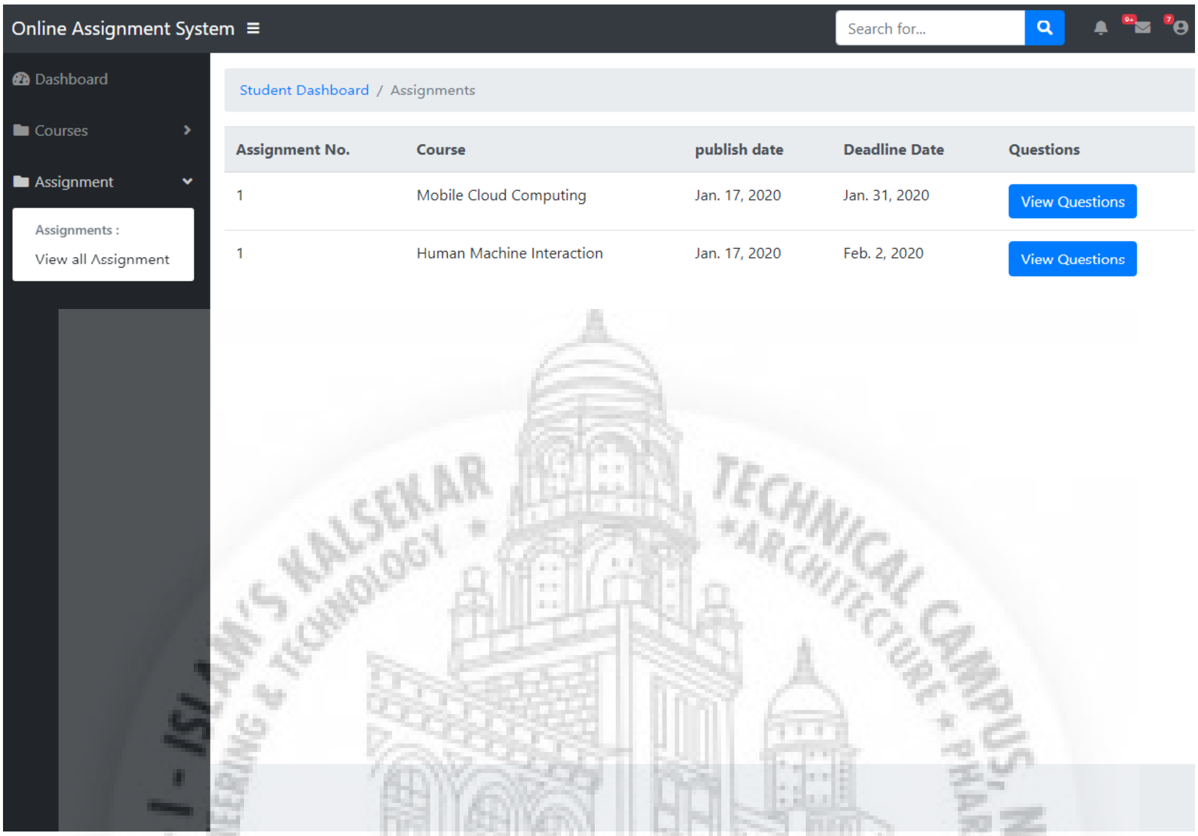
```
              {{ form.as_p }}
              <button  class="btn btn-success" type="submit">Login</button>
            </form>
        </div>
      </div>
    </div>
    <!-- Bootstrap core JavaScript-->
    <script src="{% static 'vendor/jquery/jquery.min.js' %}"></script>
    <script src="{% static 'vendor/bootstrap/js/bootstrap.bundle.min.js' %}"></script>
    <!-- Core plugin JavaScript-->
    <script src="{% static 'vendor/jquery-easing/jquery.easing.min.js'%}"></script>
</body>
</html>
// ////////////////////VIEWS ////////////////////////////////////////////////////////////
def login(request):
    return render(request , 'login.html')
```

6.1 Faculty

The faculty module consist of following section :

- Add Course : This module will allow faculty members to add their course
- Add Assignments: It will allow faculty members to add assignments of course
- Add Questions: This will allow to add questions in assignments.
- Generate Model answer: Generate model answer will search question-title on google and fetch relevant information and store it in model answer field.
- View submitted answers: It will display student roll number, name, answer and similarly score compared to model answer. it will also provide REJECT button to reject student's answer
- Generate Question Bank: This module will allow faculty members to generate and print question bank. Question bank generator will randomly select questions from selected assignments

6.1.1 Assignment.



Online Assignment System ≡

Search for...

Student Dashboard / Assignments

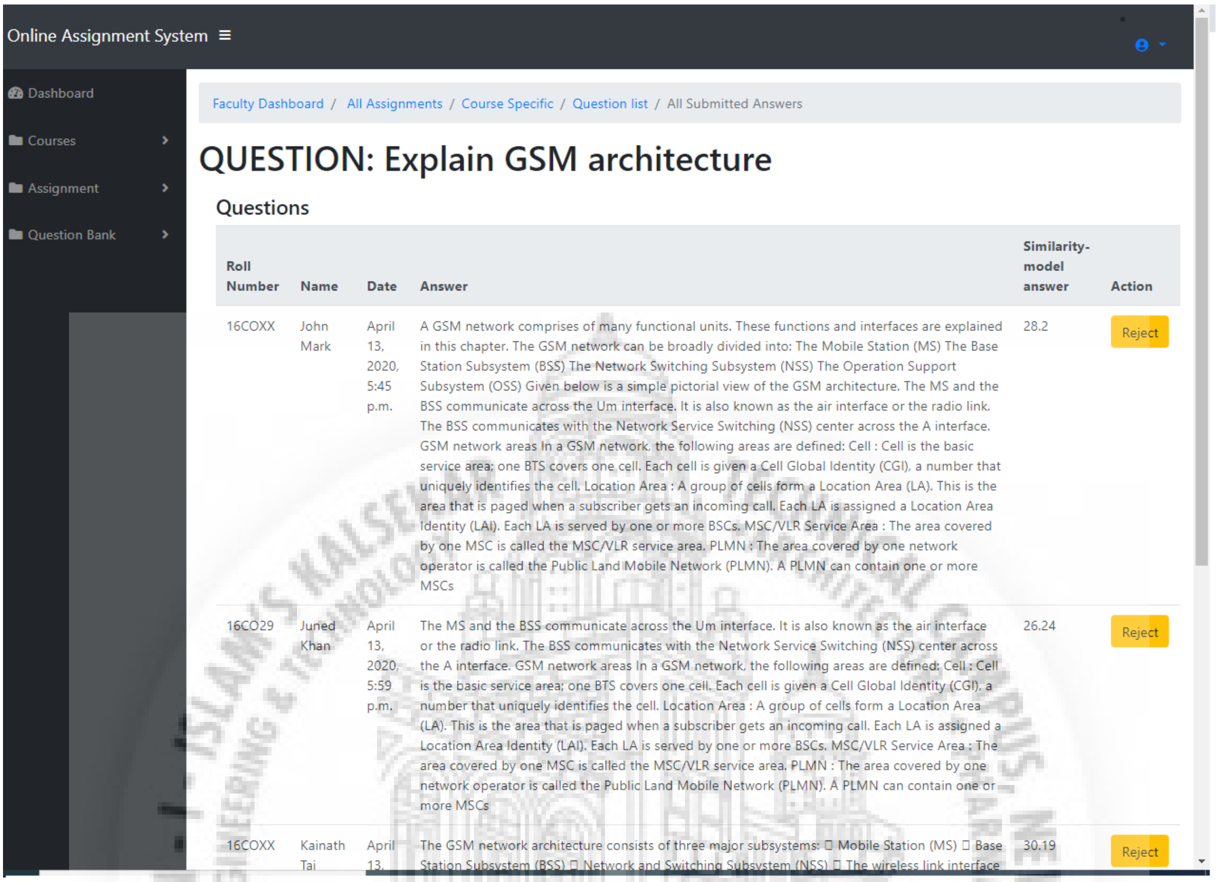
Assignment No.	Course	publish date	Deadline Date	Questions
1	Mobile Cloud Computing	Jan. 17, 2020	Jan. 31, 2020	View Questions
1	Human Machine Interaction	Jan. 17, 2020	Feb. 2, 2020	View Questions

Figure 6.2: Assignment.

```
///TEMPLATE///
{% extends "./base.html" %}
{% block breadcrumb%}
<li class="breadcrumb-item "><a href="{% url 'ViewCourse' %}">All Course</a></li
  >
<li class="breadcrumb-item active">Course Specific Assignment</li>
{%endblock%}
{% block content %}
  <h1>Course:{{course.name }}</h1>
  <p><strong>Semester:</strong> {{ course.semester }}</p>
  <p><strong>Branch:</strong> {{ course.branch }}</p>
  <p><strong>Year:</strong> {{course.year}}</p>
  <div style="margin-left:20px;margin-top:20px">
    <h4>Assignments related to this course</h4>
    {% if course.assignment_set.all %}
    <table class="table">
      <thead class="thead-light">
        <tr>
          <th scope="col">Assignment Number</th>
          <th scope="col">Publish Date</th>
          <th scope="col">Deadline</th>
          <th scope="col">Edit</th>
          <th scope="col">Delete</th>
          <th scope="col">View</th>
        </tr>
      </thead>
      <tbody>
        {% for assignment in course.assignment_set.all %}
        <tr>
          <td>{{ assignment.number }}</td>
          <td>{{ assignment.publish_date }}</td>
          <td>{{ assignment.deadline_date }}</td>
          <td> <a href="{% url 'UpdateCourseAssignment' assignment.id %}">Update</a> </td>
          <td> <a onclick="return confirm('Are you sure?');" href="{% url 'DeleteCourseAssignment' assignment.id %}">Delete</a></td>
          <td><a href="{% url 'assignment_detail'  assignment.id %}">View Questions</a></td>
        </tr>
      {% endfor %}
      </tbody>
    </table>
    {% else %}
      <p>There are no assignments added yet !</p>
    {% endif %}
  </div>
/// URL ///
path('addAssignment/', views.AssignmentCreate.as_view(), name='CreateAssignment'),
    path('assignment_list/',views.AssignmentListView.as_view(),name='ViewAssignment'),
    path('assignment/<int:pk>/update/', views.AssigmentUpdate.as_view(), name='UpdateAssignment'),
    path('Courseassignment/<int:pk>/update/', views.CourseAssigmentUpdate.as_view(), name='UpdateCourseAssignment'),
    path('assignment/<int:pk>/delete/', views.AssignmentDelete.as_view(), name='DeleteAssignment'),
    path('Courseassignment/<int:pk>/delete/', views.CourseAssignmentDelete.as_view(), name='DeleteCourseAssignment'),
    path('assignment/<int:pk>/', views.AssignmentDetailView.as_view(), name='assignment_detail'),
/// VIEW ///
from .forms import CreateAssignmentForm
class AssignmentCreate(CreateView):
    model = Assignment
    form_class = CreateAssignmentForm
    template_name = 'Faculty_Dashboard/add_assignment.html'

    def get_form_kwargs(self):
        kwargs = super().get_form_kwargs()
        kwargs['user'] = self.request.user
        return kwargs
    def form_valid(self, form):
        faculty = Faculty.objects.get(user=self.request.user)
        form.instance.faculty = faculty
        return super(AssignmentCreate, self).form_valid(form)
    def get_success_url(self):
        return reverse('ViewAssignment')
class AssigmentUpdate(UpdateView):
    model = Assignment
    form_class = CreateAssignmentForm
    template_name = 'Faculty_Dashboard/update_assignment.html'
    def get_form_kwargs(self):
        kwargs = super().get_form_kwargs()
        kwargs['user'] = self.request.user
        return kwargs
    def form_valid(self, form):
        faculty = Faculty.objects.get(user=self.request.user)
        form.instance.faculty = faculty
        return super(AssigmentUpdate, self).form_valid(form)
    def get_success_url(self):
        return reverse('ViewAssignment')
class CourseAssigmentUpdate(UpdateView):
    model = Assignment
    form_class = CreateAssignmentForm
    template_name = 'Faculty_Dashboard/add_assignment.html'
    def get_form_kwargs(self):
        kwargs = super().get_form_kwargs()
        kwargs['user'] = self.request.user
        return kwargs
    def form_valid(self, form):
        faculty = Faculty.objects.get(user=self.request.user)
        form.instance.faculty = faculty
        return super(CourseAssigmentUpdate, self).form_valid(form)
    def get_success_url(self):
        return reverse_lazy('course_detail', kwargs={'pk': self.object.course_id})
class AssignmentListView(generic.ListView):
    model = Assignment
    template_name = "Faculty_Dashboard/assignment_list.html"
    context_object_name = 'assignment_list'
    paginate_by = 10
    def get_queryset(self):
        faculty = Faculty.objects.get(user=self.request.user)
        return Assignment.objects.filter(faculty=faculty)
class AssignmentDelete(DeleteView):
    model = Assignment
    template_name = "Faculty_Dashboard/assignment_list.html"
    success_url = reverse_lazy('ViewAssignment')
    def get(self, request, *args, **kwargs):
        return self.delete(request, *args, **kwargs)
#Returns to Course Assignment list after deleting
class CourseAssignmentDelete(DeleteView):
    model = Assignment
    template_name = "Faculty_Dashboard/course_detail.html"
    def get(self, request, *args, **kwargs):
        return self.delete(request, *args, **kwargs)
    def get_success_url(self):
        return reverse_lazy('course_detail', kwargs={'pk': self.object.course_id})
class AssignmentDetailView(generic.DetailView):
    model = Assignment
    template_name = "Faculty_Dashboard/assignment_detail.html"
```

6.1.2 Submitted Answer.



Faculty Dashboard / All Assignments / Course Specific / Question list / All Submitted Answers

QUESTION: Explain GSM architecture

Questions

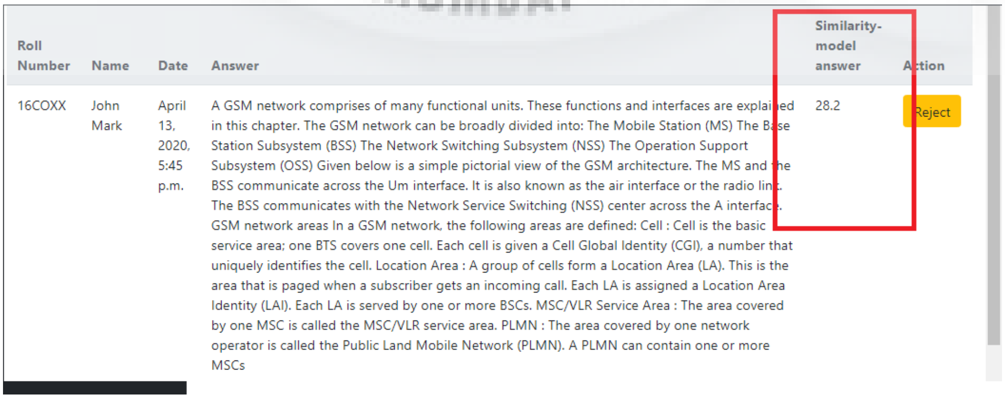
Roll Number	Name	Date	Answer	Similarity-model answer	Action
16COXX	John Mark	April 13, 2020, 5:45 p.m.	A GSM network comprises of many functional units. These functions and interfaces are explained in this chapter. The GSM network can be broadly divided into: The Mobile Station (MS) The Base Station Subsystem (BSS) The Network Switching Subsystem (NSS) The Operation Support Subsystem (OSS) Given below is a simple pictorial view of the GSM architecture. The MS and the BSS communicate across the Um interface. It is also known as the air interface or the radio link. The BSS communicates with the Network Service Switching (NSS) center across the A interface. GSM network areas In a GSM network, the following areas are defined: Cell : Cell is the basic service area; one BTS covers one cell. Each cell is given a Cell Global Identity (CGI), a number that uniquely identifies the cell. Location Area : A group of cells form a Location Area (LA). This is the area that is paged when a subscriber gets an incoming call. Each LA is assigned a Location Area Identity (LAI). Each LA is served by one or more BSCs. MSC/VLR Service Area : The area covered by one MSC is called the MSC/VLR service area. PLMN : The area covered by one network operator is called the Public Land Mobile Network (PLMN). A PLMN can contain one or more MSCs	28.2	Reject
16CO29	Juned Khan	April 13, 2020, 5:59 p.m.	The MS and the BSS communicate across the Um interface. It is also known as the air interface or the radio link. The BSS communicates with the Network Service Switching (NSS) center across the A interface. GSM network areas In a GSM network, the following areas are defined: Cell : Cell is the basic service area; one BTS covers one cell. Each cell is given a Cell Global Identity (CGI), a number that uniquely identifies the cell. Location Area : A group of cells form a Location Area (LA). This is the area that is paged when a subscriber gets an incoming call. Each LA is assigned a Location Area Identity (LAI). Each LA is served by one or more BSCs. MSC/VLR Service Area : The area covered by one MSC is called the MSC/VLR service area. PLMN : The area covered by one network operator is called the Public Land Mobile Network (PLMN). A PLMN can contain one or more MSCs	26.24	Reject
16COXX	Kainath Tai	April 13,	The GSM network architecture consists of three major subsystems: □ Mobile Station (MS) □ Base Station Subsystem (BSS) □ Network and Switching Subsystem (NSS) □ The wireless link interface	30.19	Reject

Figure 6.3: Submitted Answer.

```
/// Template ///
% extends ”./base.html” %}
{% block breadcrumb%}
<li class=”breadcrumb-item ”><a href=”{% url ’ViewAssignment’ %}”> All
    Assignments</a></li>
<li class=”breadcrumb-item ”><a href=”{% url ’course_detail’ question.assignment
    .course_id %}”>Course Specific</a></li>
<li class=”breadcrumb-item ”><a href=”{% url ’assignment_detail’ question.
    assignment.id %}”>Question list</a></li>
<li class=”breadcrumb-item active”>All Submitted Answers</li>
{%endblock%}
{% block content %}
<h1> QUESTION: {{question }}</h1>
  <div style=”margin-left:20px;margin-top:20px”>
    <h4>Questions</h4>
    <table class=”table”>
      <thead class=”thead-light”>
        <tr>
          <th scope=”col”>Roll Number</th>
          <th scope=”col”>Name</th>
          <th scope=”col”>Date</th>
          <th scope=”col”>Answer</th>
          <th scope=”col”>Similarity-model answer</th>
          <th scope=”col”>Action</th>
        </tr>
      </thead>
```

```
        <tbody>
          {% for answer in answers %}
          <tr>
            <td>{{ answer.username.student.rollnumber}}</td>
            <td>{{ answer.username.first_name}} {{ answer.username.last_name}}</td>
            <td>{{ answer.date}}</td>
            <td>{{answer.answer_txt}}</td>
            <td>{{answer.similarity}}</td>
            <td><a class="btn btn-warning" href="{% url 'RejectAnswer' qid=question.id ansID=answer.id %}">Reject</a></td>
          </tr>
        {% endfor %}
        </tbody>
      </table>
    </div>
{% endblock %}
{% block script %}
{% endblock %}
/// VIEW ///
def getSubmittedAnswer(request , id):
    question = get_object_or_404(Question, pk=id)
    answers = Answer.objects.filter(question = question , status = 'Published')
    return render(request, 'Faculty_Dashboard/submitted_answers.html', {'question': question, 'answers':answers })
def rejectAnswer(request , qid,ansID ):
    question = get_object_or_404(Question, pk=qid)
    answers = Answer.objects.filter(question = question , status = 'Published')
    answer = get_object_or_404(Answer , pk =ansID)
    answer.status = 'Draft'
    answer.save()
    return render(request, 'Faculty_Dashboard/submitted_answers.html', {'question': question, 'answers':answers })
/// URL ///
    path('submittedAnswers/<int:id>', views.getSubmittedAnswer, name='AllSubmittedAnswers'),
    path('rejectAnswer/<int:qid>/<int:ansID>', views.rejectAnswer, name='RejectAnswer'),
```

6.1.3 Similarity.

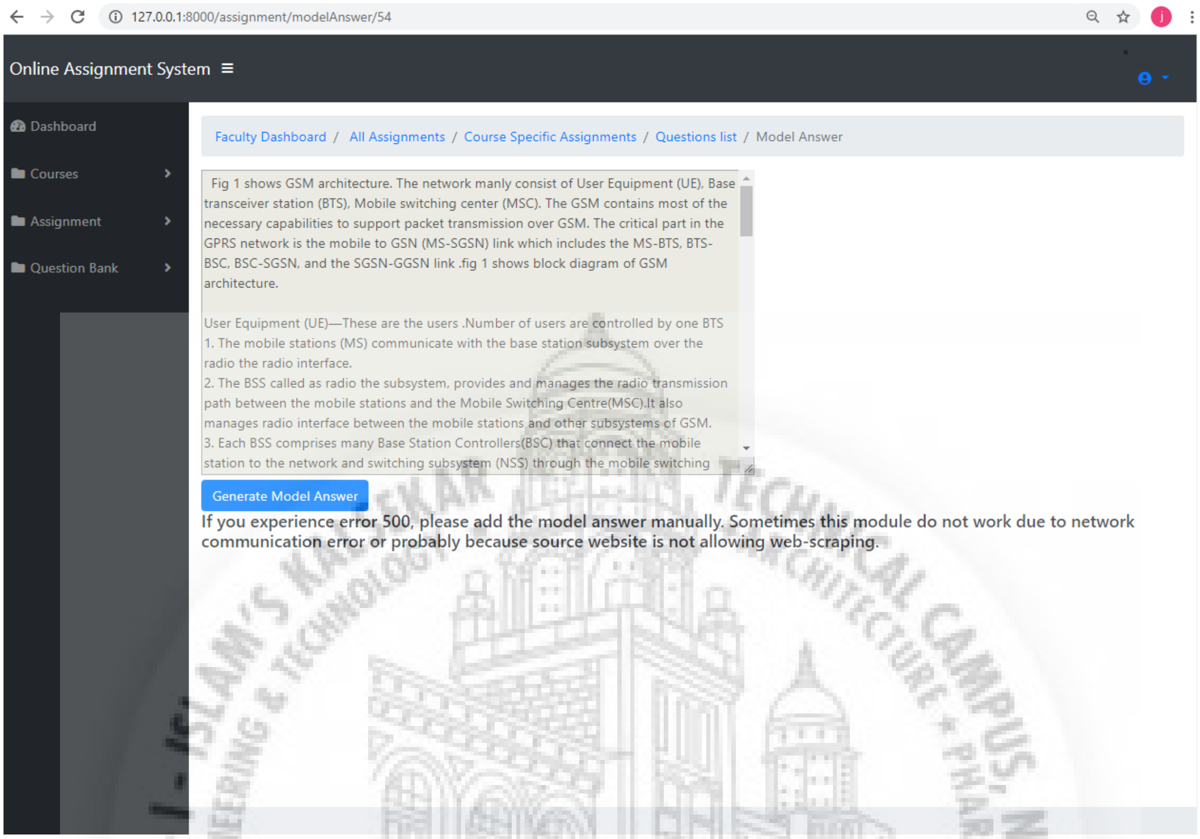


Roll Number	Name	Date	Answer	Similarity-model answer	Action
16COXX	John Mark	April 13, 2020, 5:45 p.m.	A GSM network comprises of many functional units. These functions and interfaces are explained in this chapter. The GSM network can be broadly divided into: The Mobile Station (MS) The Base Station Subsystem (BSS) The Network Switching Subsystem (NSS) The Operation Support Subsystem (OSS) Given below is a simple pictorial view of the GSM architecture. The MS and the BSS communicate across the Um interface. It is also known as the air interface or the radio link. The BSS communicates with the Network Service Switching (NSS) center across the A interface. GSM network areas In a GSM network, the following areas are defined: Cell : Cell is the basic service area; one BTS covers one cell. Each cell is given a Cell Global Identity (CGI), a number that uniquely identifies the cell. Location Area : A group of cells form a Location Area (LA). This is the area that is paged when a subscriber gets an incoming call. Each LA is assigned a Location Area Identity (LAI). Each LA is served by one or more BSCs. MSC/VLR Service Area : The area covered by one MSC is called the MSC/VLR service area. PLMN : The area covered by one network operator is called the Public Land Mobile Network (PLMN). A PLMN can contain one or more MSCs	28.2	Reject

Figure 6.4: Similarity.

```
from googlesearch import search
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
def getLinks(query):
    results = []
    for j in search(query, tld=”co.in”, num=1, stop=1, pause=5):
        print(j)
        results.append(j)
    return results
import requests
from bs4 import BeautifulSoup
def getAnswer(question):
    links = getLinks(question + ”ques10”)
    for link in links:
        URL = link
        result = requests.get(URL)
        soup = BeautifulSoup(result.content, ’html5lib’)
        #div = soup.find_all([’div.content span.clear-fix p’, ’div.content span.clear-fix li’])
        div = soup.find_all(”span”, {”class”: ”clearfix”})
        for para in div:
            answer = ’’.join(para.text)
        return answer
#      print(answer)
# getAnswer(”Explain Microprocessor”)
def checkSimilairty(userAnswer,modelAnswer):
    X =userAnswer
    Y =modelAnswer
    # tokenization
    X_list = word_tokenize(X)
    Y_list = word_tokenize(Y)
    sw = stopwords.words(’english’)
    l1 =[];l2 =[]
    # remove stop words from string
    X_set = {w for w in X_list if not w in sw}
    Y_set = {w for w in Y_list if not w in sw}
    # form a set containing keywords of both strings
    rvector = X_set.union(Y_set)
    for w in rvector:
        if w in X_set: l1.append(1) # create a vector
        else: l1.append(0)
        if w in Y_set: l2.append(1)
        else: l2.append(0)
    c = 0
    # cosine formula
    for i in range(len(rvector)):
            c+= l1[i]*l2[i]
    cosine = c / float((sum(l1)*sum(l2))**0.5)
    return round(cosine*100,2)
```

6.1.4 Model Answer.



127.0.0.1:8000/assignment/modelAnswer/54

Online Assignment System ≡

Faculty Dashboard / All Assignments / Course Specific Assignments / Questions list / Model Answer

Fig 1 shows GSM architecture. The network manly consist of User Equipment (UE), Base transceiver station (BTS), Mobile switching center (MSC). The GSM contains most of the necessary capabilities to support packet transmission over GSM. The critical part in the GPRS network is the mobile to GSN (MS-SGSN) link which includes the MS-BTS, BTS-BSC, BSC-SGSN, and the SGSN-GGSN link .fig 1 shows block diagram of GSM architecture.

User Equipment (UE)—These are the users .Number of users are controlled by one BTS
1. The mobile stations (MS) communicate with the base station subsystem over the radio the radio interface.
2. The BSS called as radio the subsystem, provides and manages the radio transmission path between the mobile stations and the Mobile Switching Centre(MSC).It also manages radio interface between the mobile stations and other subsystems of GSM.
3. Each BSS comprises many Base Station Controllers(BSC) that connect the mobile station to the network and switching subsystem (NSS) through the mobile switching

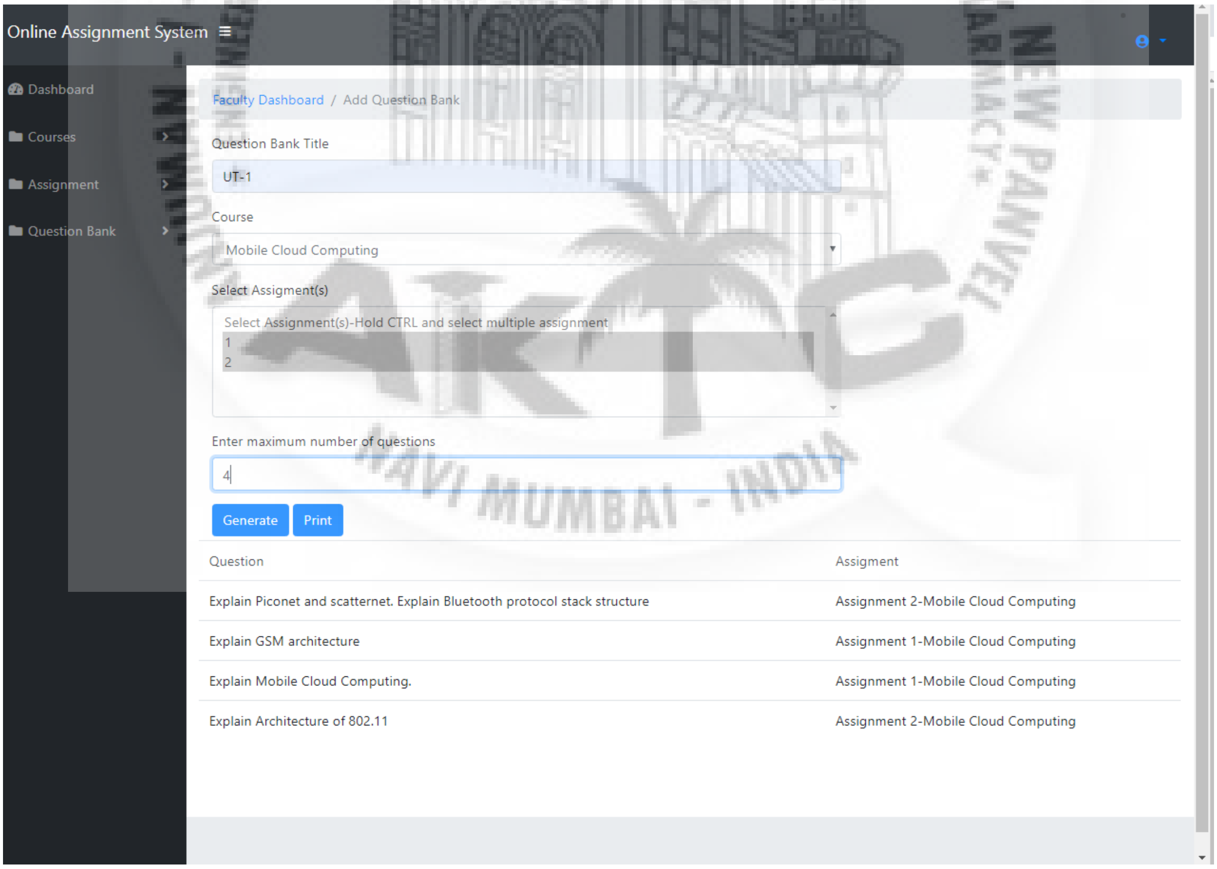
If you experience error 500, please add the model answer manually. Sometimes this module do not work due to network communication error or probably because source website is not allowing web-scraping.

Figure 6.5: Model Answer.

```
/// TEMPLATE///
{% extends "./base.html" %}
{% block breadcrumb %}
<li class="breadcrumb-item "><a href="{% url 'ViewAssignment' %}"> All Assignments</a></li>
<li class="breadcrumb-item "><a href="{% url 'course_detail' question.assignment.course_id %}">Course Specific Assignments</a></li>
<li class="breadcrumb-item active"><a href="{% url 'assignment_detail' question.assignment.id %}">Questions list </a></li>
<li class="breadcrumb-item active">Model Answer</li>
{% endblock %}
{% block content %}
<textarea name="modelAnswer" rows="15" cols="80" disabled>
  {{modelAnswer}}
</textarea>
<br>
<a class="btn btn-primary" href="{% url 'generateModelAnswer' id %}">Generate Model Answer</a>
<div class="">
  <h5>If you experience error 500, please add the model answer manually. Sometimes this module do not work due to network communication error or
  probably because source website is not allowing web-scraping.</h5>
</div>
{% endblock %}
/// URL \\\
 path('generateModelAnswer/<int:id>', views.generateModelAnswer, name='generateModelAnswer'),
    path('modelAnswer/<int:id>', views.modelAnswer, name='viewModelAnswer')
```

```
/// VIEW ///
def generateModelAnswer(request, id):
    question = Question.objects.get(id=id)
    modelAnswer = eval.getAnswer(question.title)
    question.modelAnswer = modelAnswer
    question.save()
    return render(request, 'Faculty_Dashboard/modelAnswer.html', {'question': question, 'modelAnswer':modelAnswer, 'id':question.id })
def modelAnswer(request, id):
    question = Question.objects.get(id=id)
    if(question.modelAnswer):
        modelAnswer = question.modelAnswer
        return render(request, 'Faculty_Dashboard/modelAnswer.html', {'question': question, 'modelAnswer':modelAnswer, 'id':question.id })
    else:
        modelAnswer = "Model answer not generated. Click on Generate Model Answer, it might take upto 5-10 seconds depending upon response from source site. Please don't repeatedly click generate button to avoid getting blocked by source site."
        return render(request, 'Faculty_Dashboard/modelAnswer.html', {'question': question, 'modelAnswer':modelAnswer, 'id':question.id })
```

6.1.5 Question Bank Generator.


Online Assignment System ≡
Dashboard
Courses
Assignment
Question Bank
Faculty Dashboard / Add Question Bank

Question Bank Title: UT-1

Course: Mobile Cloud Computing

Select Assigment(s): Select Assignment(s)-Hold CTRL and select multiple assignment
1
2

Enter maximum number of questions: 4

Question	Assigment
Explain Piconet and scatternet. Explain Bluetooth protocol stack structure	Assignment 2-Mobile Cloud Computing
Explain GSM architecture	Assignment 1-Mobile Cloud Computing
Explain Mobile Cloud Computing.	Assignment 1-Mobile Cloud Computing
Explain Architecture of 802.11	Assignment 2-Mobile Cloud Computing

Figure 6.6: Question Bank Generator.

```
/// VIEW ///
```

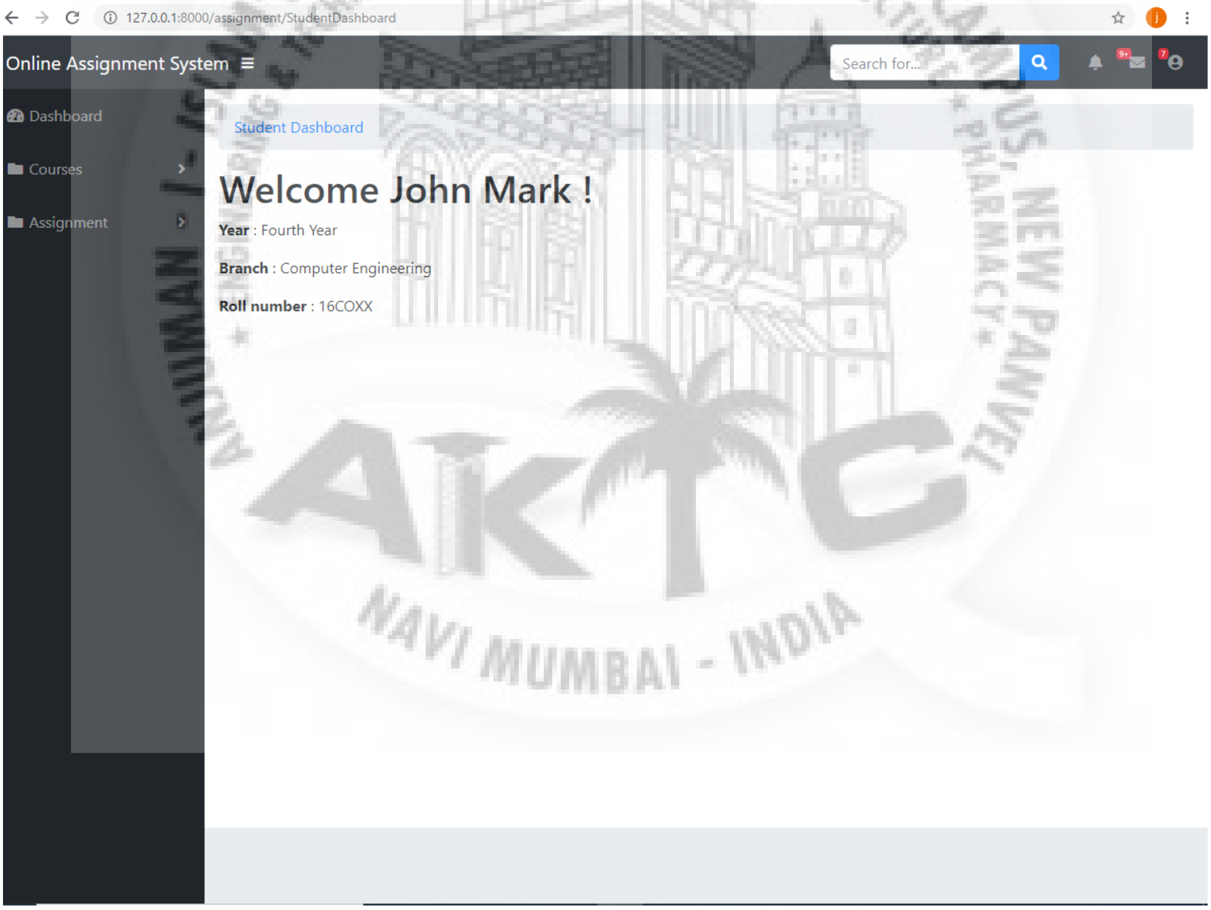
```
def questionBank(request):
    faculty = Faculty.objects.get(user=request.user)
    courses  = Course.objects.filter(faculty=faculty)
    return  render(request , ’Faculty_Dashboard/question_bank.html’, {’courses’:courses})
def all_json_models(request, id):
    course = Course.objects.get(id=id)
    assignments = Assignment.objects.all().filter(course=course)
    json_models = serializers.serialize(”json”, assignments)
    return HttpResponse(json_models, content_type=”application/javascript”)
def generateQB(request):
    if(request.method==”GET”):
        course = request.GET[’course’]
        maxQuestions = request.GET[’maxQuestions’]
        assignmentList = request.GET.getlist(’assignment’)
        print(maxQuestions)
        course = Course.objects.get(id=course)
        assignment=Assignment.objects.filter(course=course,id__in=assignmentList)
        questions=Question.objects.filter(assignment__in=assignment).order_by(’?’)[:int(maxQuestions)]
        faculty = Faculty.objects.get(user=request.user)
        courses  = Course.objects.filter(faculty=faculty)
        return  render(request,”Faculty_Dashboard/question_bank.html”,{’question_list’:questions,’courses’: courses})
/// URL ///
    path(’addQuestionBank/’,views.questionBank,name=’QuestionBank’),
    path(’addQuestionBank/selectedCourse/<int:id>/all_json_assignments’,views.all_json_models, name=”SendCourse”),
    path(’addQuestionBank/getQuestions/’,views.generateQB, name=’GetQuestions’),
```

6.2 Student

Student module consist following section:

- Add Answer: Students will be able to add answers through this module. They will select assignments and add answers.
- Update Answer: Students can update their answers through this module.
- Delete Answer: Students can delete answers through this module.
- Answer Status: Draft/Publish status allow students to hide or display their answer from other users.
- View Assignments: This will allow students to view all the assignments.

6.2.1 Dashboard.



Welcome John Mark !

Year : Fourth Year

Branch : Computer Engineering

Roll number : 16COXX

Figure 6.7: Dashboard.

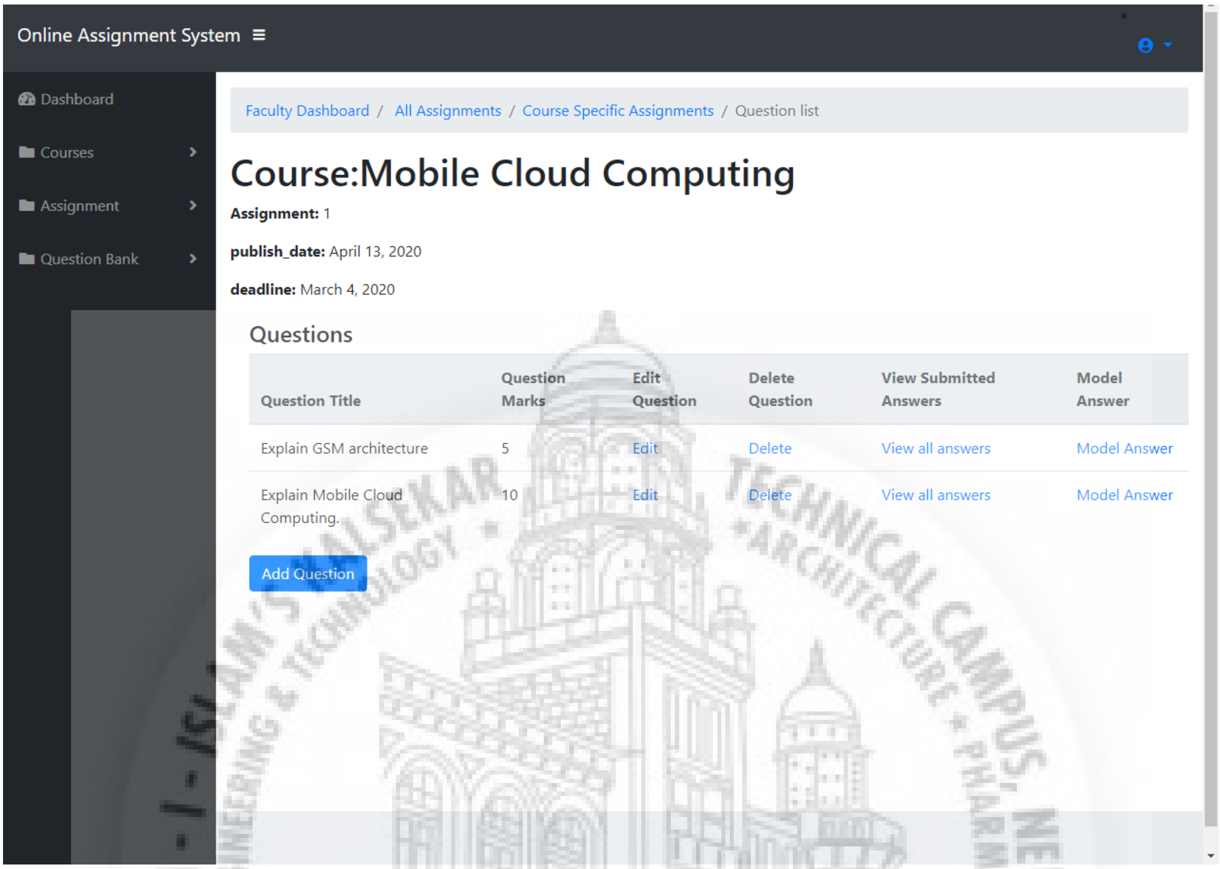
```
    path('StudentDashboard',views.StudentDashboard,name='StudentDashboard'),
########################views##############
@login_required
def StudentDashboard(request):
    return render(request , 'Student_Dashboard/index.html')
```

```
####################templates##############
<!DOCTYPE html>
<html lang=”en”>
<head>
  <meta charset=”utf-8”>
  <meta http-equiv=”X-UA-Compatible” content=”IE=edge”>
  <meta name=”viewport” content=”width=device-width, initial-scale=1, shrink-to-fit=no”>
  <meta name=”description” content=””>
  <meta name=”author” content=””>
  <title>Student Dashboard</title>
  {% load static %}
  <!-- Custom fonts for this template-->
  <link href=”{% static 'vendor/fontawesome-free/css/all.min.css' %}” rel=”stylesheet” type=”text/css”>
  <!-- Page level plugin CSS-->
  <link href=”{% static 'vendor/datatables/dataTables.bootstrap4.css' %}” rel=”stylesheet”>
  <!-- Custom styles for this template-->
  <link href=”{%  static 'css/sb-admin.css' %}” rel=”stylesheet”>
</head>
<body id=”page-top”>
  <nav class=”navbar navbar-expand navbar-dark bg-dark static-top”>
    <a class=”navbar-brand mr-1” href=”{% url 'index' %}”>Online Assignment System</a>
    <button class=”btn btn-link btn-sm text-white order-1 order-sm-0” id=”sidebarToggle” href=”#”>
      <i class=”fas fa-bars”></i>
    </button>
    <!-- Navbar Search -->
    <form class=”d-none d-md-inline-block form-inline ml-auto mr-0 mr-md-3 my-2 my-md-0”>
      <div class=”input-group”>
        <input type=”text” class=”form-control” placeholder=”Search for...” aria-label=”Search” aria-describedby=”basic-addon2”>
        <div class=”input-group-append”>
          <button class=”btn btn-primary” type=”button”>
            <i class=”fas fa-search”></i>
          </button>
        </div>
      </div>
    </form>
    <!-- Navbar -->
    <ul class=”navbar-nav ml-auto ml-md-0”>
      <li class=”nav-item dropdown no-arrow mx-1”>
        <a class=”nav-link dropdown-toggle” href=”#” id=”alertsDropdown” role=”button” data-toggle=”dropdown” aria-haspopup=”true” aria-expanded=”false”>
          <i class=”fas fa-bell fa-fw”></i>
          <span class=”badge badge-danger”>9+</span>
        </a>
        <div class=”dropdown-menu dropdown-menu-right” aria-labelledby=”alertsDropdown”>
          <a class=”dropdown-item” href=”#”>Action</a>
          <a class=”dropdown-item” href=”#”>Another action</a>
          <div class=”dropdown-divider”></div>
          <a class=”dropdown-item” href=”#”>Something else here</a>
        </div>
      </li>
      <li class=”nav-item dropdown no-arrow mx-1”>
        <a class="nav-link dropdown-toggle" href="#" id="messagesDropdown" role="button" data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">
          <i class="fas fa-envelope fa-fw"></i>
          <span class="badge badge-danger">7</span>
        </a>
        <div class="dropdown-menu dropdown-menu-right" aria-labelledby="messagesDropdown">
          <a class="dropdown-item" href="#">Action</a>
          <a class="dropdown-item" href="#">Another action</a>
          <div class="dropdown-divider"></div>
          <a class="dropdown-item" href="#">Something else here</a>
        </div>
      </li>
      <li class="nav-item dropdown no-arrow">
        <a class="nav-link dropdown-toggle" href="#" id="userDropdown" role="button" data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">
          <i class="fas fa-user-circle fa-fw"></i>
        </a>
        <div class="dropdown-menu dropdown-menu-right" aria-labelledby="userDropdown">
          <a class="dropdown-item" href="#">Settings</a>
          <a class="dropdown-item" href="#">Activity Log</a>
          <div class="dropdown-divider"></div>
          <a class="dropdown-item" type="submit" data-toggle="modal" data-target="#logoutModal">Logout</a>
        </div>
      </li>
    </ul>
  </nav>
  <div id="wrapper">
    <!-- Sidebar -->
    <ul class="sidebar navbar-nav">
      <li class="nav-item">
        <a class="nav-link" href="{% url 'StudentDashboard' %}">
          <i class="fas fa-fw fa-tachometer-alt"></i>
          <span>Dashboard</span>
        </a>
      </li>
      <li class="nav-item dropdown">
       <a class="nav-link dropdown-toggle" href="#" id="pagesDropdown" role="button" data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">
         <i class="fas fa-fw fa-folder"></i>
         <span>Courses</span>
       </a>
       <div class="dropdown-menu" aria-labelledby="pagesDropdown">
         <h6 class="dropdown-header">Course</h6>
         <a class="dropdown-item" href="{% url 'AllCourse' %}">View all Course</a>
       </div>
     </li>
     <li class="nav-item dropdown">
      <a class="nav-link dropdown-toggle" href="#" id="pagesDropdown" role="button" data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">
        <i class="fas fa-fw fa-folder"></i>
        <span>Assignment</span>
      </a>
      <div class="dropdown-menu" aria-labelledby="pagesDropdown">
```

```
      <h6 class=”dropdown-header”>Assignments :</h6>
      <a class=”dropdown-item” href=”{% url 'AllAssignment' %}”>View all Assignment</a>
      <!-- <a class=”dropdown-item” href=”{% url 'ViewCourse' %}”>View all Assignment</a>
      <h6 class=”dropdown-header”>Questions :</h6>
      <a class=”dropdown-item” href=”{% url 'CreateCourse' %}”>Add Questions</a>
      <a class=”dropdown-item” href=”{% url 'ViewCourse' %}”>View all Assignment</a> -->
    </div>
  </li>
  </ul>
  <div id=”content-wrapper”>
    <div class=”container-fluid”>
      <!-- Breadcrumbs-->
      <ol class=”breadcrumb”>
        <li class=”breadcrumb-item”>
          <a href=”{% url 'StudentDashboard' %}”>Student Dashboard</a>
        </li>
        {% block breadcrumb%}
        {%endblock%}
      </ol>
      <!-- Page Content -->
      <!--
      <h1>Blank Page</h1>
      <hr>
      <p>This is a great starting point for new custom pages.</p> -->
      {% block content %}
      {% endblock %}
    </div>
    <!-- /.container-fluid -->
    <!-- Sticky Footer -->
    <footer class=”sticky-footer”>
      <div class=”container my-auto”>
        <div class=”copyright text-center my-auto”>
          <span></span>
        </div>
      </div>
    </footer>
  </div>
  <!-- /.content-wrapper -->
</div>
<!-- /#wrapper -->
<!-- Scroll to Top Button-->
<a class=”scroll-to-top rounded” href=”#page-top”>
  <i class=”fas fa-angle-up”></i>
</a>
<!-- Logout Modal-->
<div class=”modal fade” id=”logoutModal” tabindex=”-1” role=”dialog” aria-labelledby=”exampleModalLabel” aria-hidden=”true”>
  <div class=”modal-dialog” role=”document”>
    <div class=”modal-content”>
      <div class=”modal-header”>
        <h5 class=”modal-title” id=”exampleModalLabel”>Ready to Leave?</h5>
        <button class=”close” type=”button” data-dismiss=”modal” aria-label=”Close”>
          <span aria-hidden=”true”> </span>
        </button>
      </div>
```

```
        <div class=”modal–body”>Select ”Logout” below if you are ready to end your current session.</div>
        <div class=”modal–footer”>
          <button class=”btn btn–secondary” type=”button” data–dismiss=”modal”>Cancel</button>
          <a class=”btn btn–primary” href= ”{% url ’logout’ %}?next=/”>Logout</a>
        </div>
      </div>
    </div>
  </div>
  <!–– Bootstrap core JavaScript––>
  <script src=”{% static ’vendor/jquery/jquery.min.js’ %}”></script>
  <script src=”{% static ’vendor/bootstrap/js/bootstrap.bundle.min.js’ %}”></script>
  <!–– Core plugin JavaScript––>
  <script src=”{% static ’vendor/jquery–easing/jquery.easing.min.js’ %}”></script>
  <!–– Custom scripts for all pages––>
  <script src=”{% static ’js/sb–admin.min.js’ %}”></script>
  {% block script %}
  {% endblock %}
</body>
</html>
{% extends ”./base.html” %}
{% block content %}
 <h1>Welcome {{user.first_name}} {{user.last_name}} !</h1>
  <p><b>Year</b> : {{user.student.year}}</p>
  <p><b>Branch</b> : {{user.student.branch}}</p>
  <p><b>Roll number</b> : {{user.student.rollnumber}}</p>
{% endblock %}
```

6.2.2 List of Assignment.



Faculty Dashboard / All Assignments / Course Specific Assignments / Question list

Course:Mobile Cloud Computing

Assignment: 1

publish_date: April 13, 2020

deadline: March 4, 2020

Questions

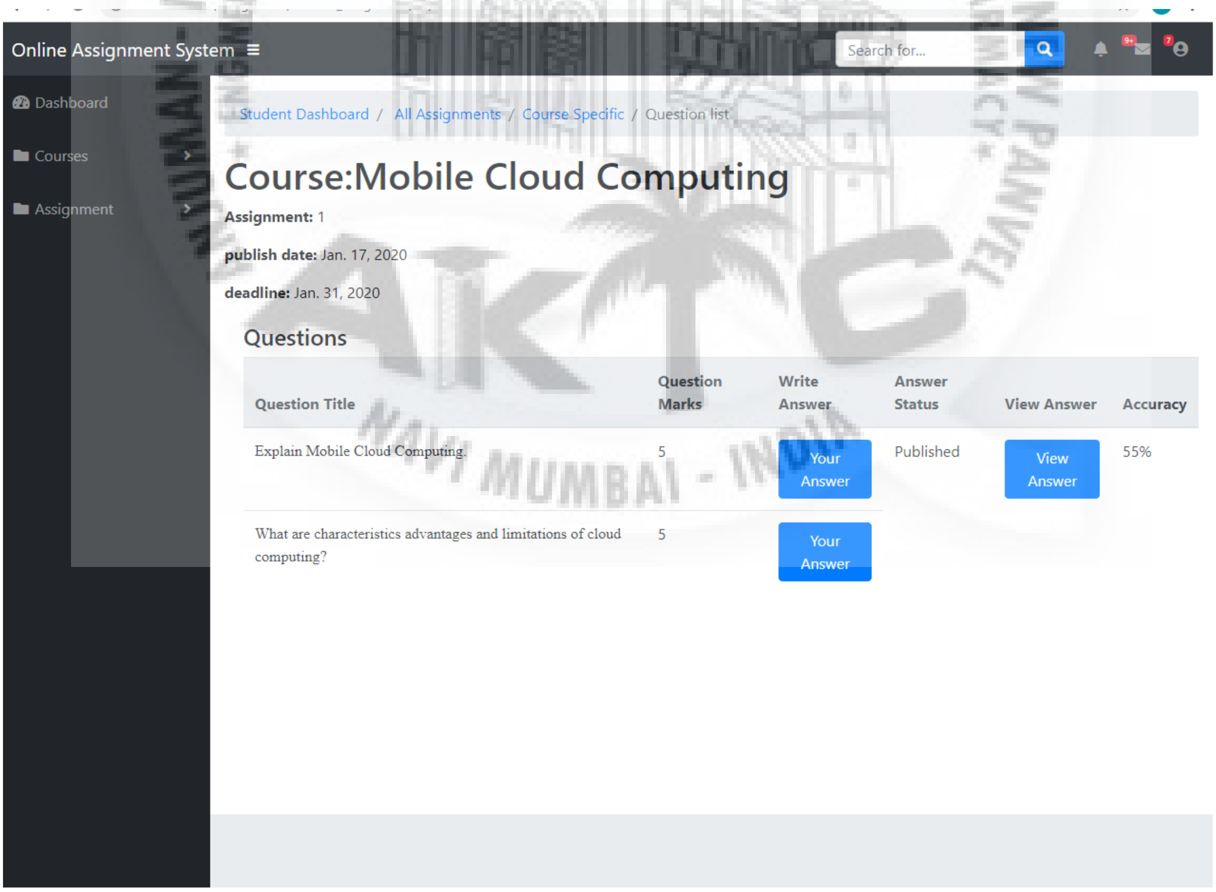
Question Title	Question Marks	Edit Question	Delete Question	View Submitted Answers	Model Answer
Explain GSM architecture	5	Edit	Delete	View all answers	Model Answer
Explain Mobile Cloud Computing.	10	Edit	Delete	View all answers	Model Answer

Figure 6.8: List of Assignment.

```
############################urls######################
path('student_assigment_list/',views.StudentAssignmentListView.as_view(),name='
    AllAssignment'),
###############################views##########################
class StudentAssignmentListView(generic.ListView):
    model = Assignment
    template_name = "Student_Dashboard/assignment_list.html"
    context_object_name = 'assignment_list'
    paginate_by = 10
    def get_queryset(self):
        student = Student.objects.get(user = self.request.user)
        branch = Branch.objects.get(name=student.branch)
        year = StudyYear.objects.get(year=student.year)
        return Assignment.objects.filter(course__year= year,course__branch=
            branch)
################################templates#######################
{% extends "./base.html" %}
{% block breadcrumb%}
<li class="breadcrumb-item active">Assignments</li>
{%endblock%}
{% block content %}
{% if assignment_list %}
  <table class="table">
    <thead class="thead-light">
      <tr>
        <th scope="col">Assignment No.</th>
        <th scope="col">Course</th>
```

```
          <th scope=”col”>publish date</th>
          <th scope=”col”>Deadline Date</th>
          <th scope=”col”>Questions</th>
        </tr>
      </thead>
      <tbody>
        {% for assignment in assignment_list %}
        <tr>
          <td>{{ assignment.number }}</td>
          <td>{{ assignment.course }}</td>
          <td>{{ assignment.publish_date }}</td>
          <td>{{ assignment.deadline_date }}</td>
          <td><a class=”btn btn-primary” href=”{% url ’student_assignment_detail’
              assignment.id %}”>View Questions</a></td>
        </tr>
      {% endfor %}
      </tbody>
    </table>
  {% else %}
    <p>There are no assignments added yet!</p>
  {% endif %}
{% endblock %}
```

6.2.3 Assignment Question.


Online Assistant System ≡
Search for...
Dashboard
Courses
Assignment
Student Dashboard / All Assignments / Course Specific / Question list

Course:Mobile Cloud Computing

Assignment: 1

publish date: Jan. 17, 2020

deadline: Jan. 31, 2020

Questions

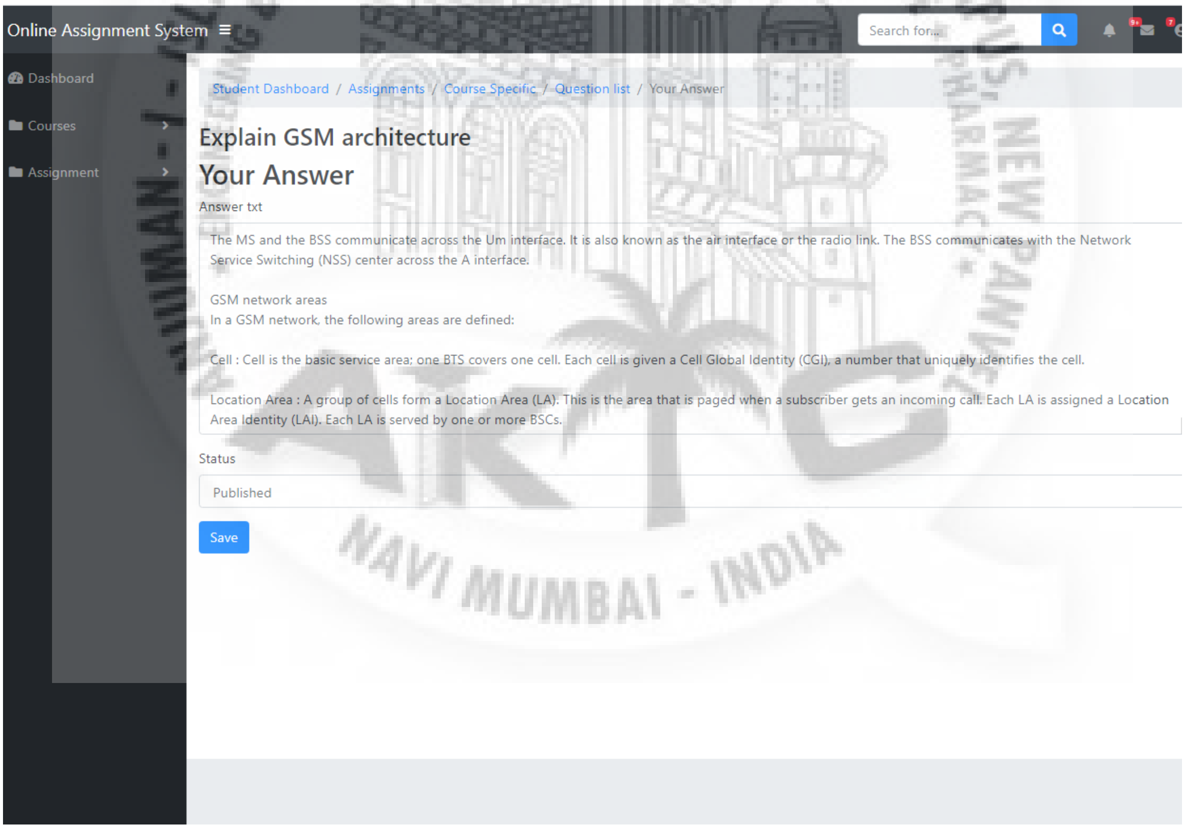
Question Title	Question Marks	Write Answer	Answer Status	View Answer	Accuracy
Explain Mobile Cloud Computing.	5	Your Answer	Published	View Answer	55%
What are characteristics advantages and limitations of cloud computing?	5	Your Answer			

Figure 6.9: Assignment Question.

```
###############################urls###################

        path(’student_assignment/<int:pk>/’, views.StudentAssignmentDetailView.as_view(), name=’student_assignment_detail’),
###############################views##################
class StudentAssignmentDetailView(generic.DetailView):
    model = Assignment
    template_name = ”Student_Dashboard/assignment_detail.html”
    def get_context_data(self, **kwargs):
        context = super(StudentAssignmentDetailView, self).get_context_data(**kwargs)
        answers = Answer.objects.filter(username = self.request.user)
        context[’answers’] = answers
        return context
#############################templates##############
{% extends ”./base.html” %}
{% block breadcrumb%}
<li class=”breadcrumb-item ”><a href=”{% url ’AllAssignment’ %}”> All Assignments</a></li>
<li class=”breadcrumb-item ”><a href=”{% url ’student_course_detail’ assignment.course_id %}”>Course Specific</a></li>
<li class=”breadcrumb-item active”>Question list</li>
{%endblock%}
{% block content %}
  <h1>Course:{{assignment.course }}</h1>
  <p><strong>Assignment:</strong> {{ assignment.number }}</p>
  <p><strong>publish date:</strong> {{ assignment.publish_date }}</p>
  <p><strong>deadline:</strong> {{assignment.deadline_date}}</p>
  <div style=”margin-left:20px;margin-top:20px”>
    <h4>Questions</h4>
    {% if assignment.question_set.all %}
    <table class=”table”>
      <thead class=”thead-light”>
        <tr>
          <th scope=”col”>Question Title</th>
          <th scope=”col”>Question Marks</th>
          <th scope=”col”>Write Answer</th>
          <th scope=”col”>Answer Status</th>
          <th scope=”col”>View Answer</th>
        </tr>
      </thead>
      <tbody>
        {% for question in assignment.question_set.all %}
        <tr>
          <td style=”font-family : serif;” >{{ question.title }}</td>
          <td>{{ question.marks }}</td>
          <td><a class=”btn btn-primary”  href=”{% url ’CreateAnswer’ pk=question.id %}” >Your Answer</a></td>
          {% for answer in answers %}
          {% ifequal  answer.question question %}
          <td>{{answer.status}}</td>
          <td><a class=”btn btn-primary”  href=”{% url ’student_answer’ apk=assignment.id pk=question.id %}”>View Answer</a></td>
          {% endifequal %}
          {% endfor %}
        </tr>
      {% endfor %}
      </tbody>
    </table>
    {% else %}
      <p>There are no questions added yet!</p>
    {% endif %}
  </div>
{% endblock %}
{% block script %}
{% endblock %}
from django.views import generic
class QuestionListView(generic.ListView):
    model = Question
    template_name = 'index.html'
    paginate_by = 10
    def get_context_data(self, **kwargs):
        context = super().get_context_data(**kwargs)
        context['branchs'] = Branch.objects.all()
        context['years'] = StudyYear.objects.all()
        context['courses']=Course.objects.all()
        return context
    def get_queryset(self):
        return Question.objects.order_by('-date')
```

6.2.4 Add Answer.


Online Assistant System ≡
Search for...
Dashboard
Courses
Assignment
Student Dashboard / Assignments / Course Specific / Question list / Your Answer
Save

Explain GSM architecture

Your Answer

Answer txt

The MS and the BSS communicate across the Um interface. It is also known as the air interface or the radio link. The BSS communicates with the Network Service Switching (NSS) center across the A interface.

GSM network areas
In a GSM network, the following areas are defined:

Cell : Cell is the basic service area; one BTS covers one cell. Each cell is given a Cell Global Identity (CGI), a number that uniquely identifies the cell.

Location Area : A group of cells form a Location Area (LA). This is the area that is paged when a subscriber gets an incoming call. Each LA is assigned a Location Area Identity (LAI). Each LA is served by one or more BSCs.

Status

Published

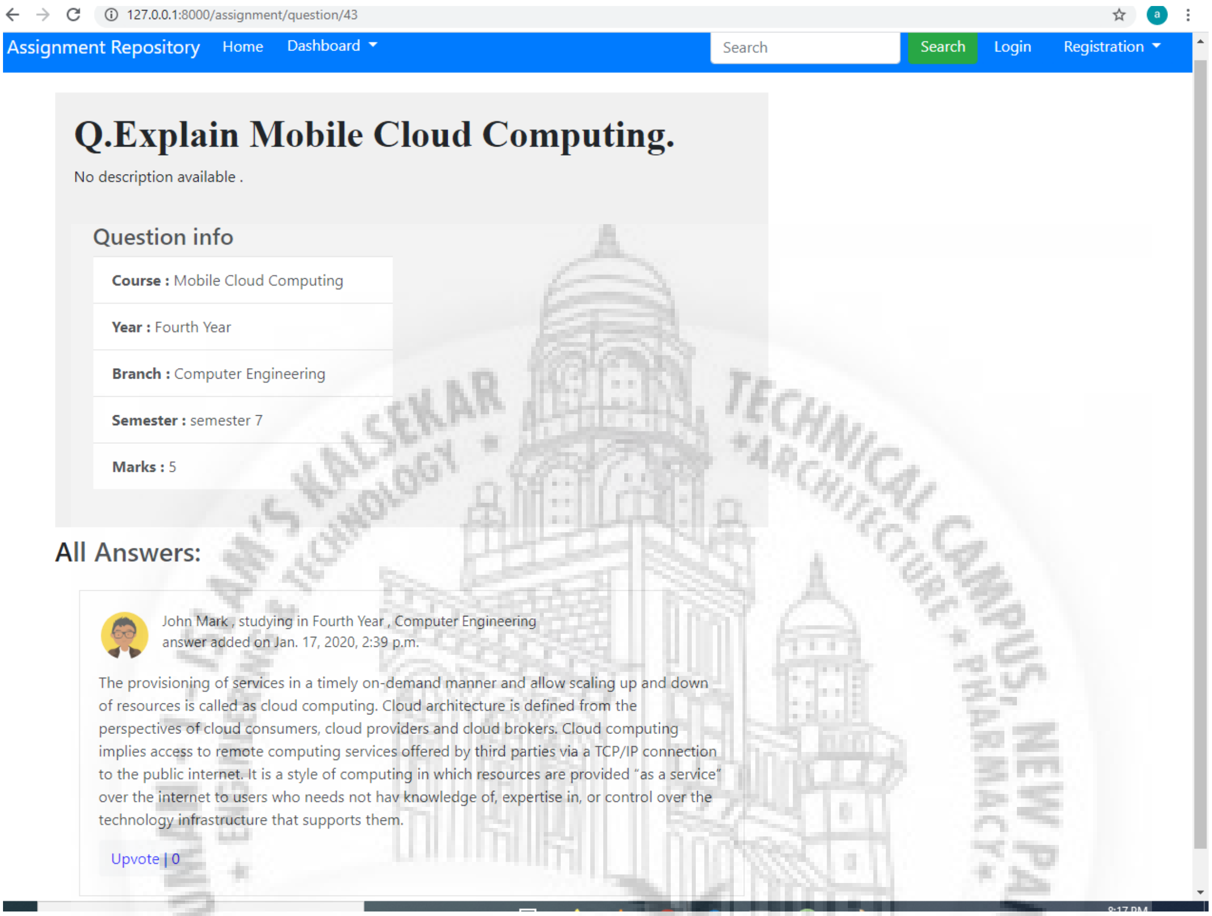
Figure 6.10: Student Add Answer.

```
############################ urls ###################
path('<int:pk>/createAnswer/', views.AnswerCreate, name='CreateAnswer'),
    path('<int:apk>/student_answer/<int:pk>/', views.getAnswer ,name='student_answer'),
################################ views #############################
```

```
from .forms import AddAssignmentAnswerForm
def AnswerCreate(request, pk):
    question = get_object_or_404(Question, pk=pk)
    user_answer = Answer.objects.filter(question = question, username = request.user).first()
    if request.method == 'POST':
      form = AddAssignmentAnswerForm(request.POST)
      if form.is_valid():
          answer = request.POST['answer_txt']
          status = request.POST['status']
          userAnswer, created = Answer.objects.get_or_create(username=request.user, question = question )
          userAnswer.answer_txt = answer
          userAnswer.status = status
          if(question.modelAnswer):
              userAnswer.similarity = eval.checkSimilairty(userAnswer.answer_txt,question.modelAnswer)
          userAnswer.save()
          return redirect('student_assignment_detail', question.assignment.id)
    elif user_answer:
      form = AddAssignmentAnswerForm(instance = user_answer)
    else:
      form = AddAssignmentAnswerForm()
    return render(request, 'Student_Dashboard/add_answer.html', {'question': question, 'form': form ,'pk': question.assignment.id , 'cpk': question.assignment.course.id })
def getAnswer(request,pk,apk):
    assignment = get_object_or_404(Assignment,pk=apk)
    question = get_object_or_404(Question,pk=pk)
    answer = get_object_or_404(Answer,question=question,username=request.user)
    return render(request, 'Student_Dashboard/answer_detail.html', {'question': question, 'answer':answer , 'assignment': assignment })
################################templates#####################
{% extends './base.html' %}
{% block breadcrumb%}
<li class="breadcrumb-item "><a href="{% url 'AllAssignment' %}">Assignments</a></li>
<li class="breadcrumb-item "><a href="{% url 'student_course_detail' cpk %}">Course Specific </a></li>
<li class="breadcrumb-item"><a href="{% url 'student_assignment_detail' pk %}">Question list</a></li>
<li class="breadcrumb-item active">Your Answer</li>
{%endblock%}
{% load widget_tweaks %}
{% block content %}
<h3>{{question.title}}</h3>
  <h2>Your Answer</h2>
  <form method="post">
    {% csrf_token %}
    {% for hidden in form.hidden_fields %}
      {{ hidden }}
    {% endfor %}
    {% for field in form.visible_fields %}
      <div class="form-group">
        <label for="{{ field.id_for_label }}">{{ field.label }}</label>
        {{ field|add_class:'form-control' }}
        {% for error in field.errors %}
          <span class="help-block">{{ error }}</span>
        {% endfor %}
      </div>
    {% endfor %}
    <div class="form-group">
      <input class="btn btn-primary" type="submit" value="Save">
    </div>
  </form>
{% endblock %}
```

6.3 Assignment Repository

6.3.1 Home Page of Assignment.



Q.Explain Mobile Cloud Computing.

No description available .

Question info

Course : Mobile Cloud Computing

Year : Fourth Year

Branch : Computer Engineering

Semester : semester 7

Marks : 5

All Answers:

John Mark , studying in Fourth Year , Computer Engineering
answer added on Jan. 17, 2020, 2:39 p.m.

The provisioning of services in a timely on-demand manner and allow scaling up and down of resources is called as cloud computing. Cloud architecture is defined from the perspectives of cloud consumers, cloud providers and cloud brokers. Cloud computing implies access to remote computing services offered by third parties via a TCP/IP connection to the public internet. It is a style of computing in which resources are provided "as a service" over the internet to users who needs not hav knowledge of, expertise in, or control over the technology infrastructure that supports them.

Upvote | 0

Figure 6.11: Home Page Answer.

```
 ###################################urls###################################
    path('question/<int:pk>', views.RepoQuestionDetailView.as_view(), name='repo
       -question-detail'),

###################################views##################################
class RepoQuestionDetailView(generic.DetailView):
    model = Question
    template_name = "questionAllAnswers.html"
    def get_context_data(self, **kwargs):
        context = super().get_context_data(**kwargs)
        context['answers'] = self.object.answer_set.all().annotate(count=Count('
           upvotes')).order_by('-count')
        return context
###################################templates#########################
{% extends "index_base.html" %}
{% load static %}
{% block content %}
<div class="container">
  <div class="row">
    <div class="col-md-8 my-3" style="padding : 5px;">
      <div style="background-color: #f0f0f0 ; padding:20px;">
        <h1 style="font-family : serif;"><b>Q.{{question.title}}</b></h1>
        {% if question.description %}
        {{question.description}}
        {% else %}
        <p>No description available.</p>
        {% endif %}
        <div class="QuestionInfo">
          <ul class="list-group list-group-flush">
            <h4>Question info</h4>
            <li class="list-group-item"><b>Course    : </b>{{question.assignment.course.name}}</li>
            <li class="list-group-item"><b>Year      : </b>{{question.assignment.course.year}}</li>
            <li class="list-group-item"><b>Branch    : </b>{{question.assignment.course.branch}}</li>
            <li class="list-group-item"><b>Semester  : </b>{{question.assignment.course.semester}}</li>
            <li class="list-group-item"><b>Marks     : </b>{{question.marks}}</li>
          </ul>
        </div>
      </div>
      <h3 class="my-2">All Answers:</h3>
      {% if answers %}
      {% for answer in answers %}
      {% ifequal answer.status 'Published' %}
      <div class="userAnswerBox">
          <div class="userDetails">
            <img class="UserPic" src="{% static 'images/avtar.png' %}">
            <span>{{answer.username.first_name}} {{answer.username.last_name}} , </span>
            <span>studying in {{answer.username.student.year}} , {{answer.username.student.branch}} </span>
          <p class="answerDate">answer added on {{answer.date}} </p>
          </div>
          <div class="userAnswer">
            {{answer.answer_txt}}
          </div>
          <div class="ratingButton">
              <span class=""><a class="btn btn-light upvote" style="Color:blue;" href="#">Upvote | {{answer.upvotesCount}}</a><span>
          </div>
      </div>
      {% endifequal %}
      {% endfor %}
      {% else %}
      <h4>No answers added yet!</h4>
      {% endif %}
    </div>
    <div class="col-md-3 mx-2 my-3" style="padding : 10px;">
      <!-- <h5>Related Questions</h5> -->
    </div>
</div>
</div>
{% endblock %}
from django.views import generic
class QuestionListView(generic.ListView):
    model = Question
    template_name = 'index.html'
    paginate_by = 10
    def get_context_data(self, **kwargs):
        context = super().get_context_data(**kwargs)
        context['branchs'] = Branch.objects.all()
        context['years'] = StudyYear.objects.all()
        context['courses']=Course.objects.all()
        return context
    def get_queryset(self):
        return Question.objects.order_by('-date')
```

Chapter 7

System Testing

The aim of the system testing process was to determine all defects in our project.The program was subjected to a set of test inputs and various observations were made and based on these observations it will be decided whether the program behaves as expected or not. Our Project went through two levels of testing
1. Unit testing
2. Integration testing

1. INTEGRATION TESTING In this type of testing we test various integration of the project module by providing the input .The primary objective is to test the module interfaces in order to ensure that no errors are occurring when one module invokes the other module.

2. UNIT TESTING Unit testing is undertaken when a module has been created and successfully reviewed.In order to test a single module we need to provide a complete environment ie besides the module we would require
   • The procedures belonging to other modules that the module under test calls
   • Non local data structures that module accesses
   • A procedure to call the functions of the module under test with appropriate parameters Unit testing was done on each and every module that is described under module description of

7.1 Test Cases and Test Results

Test ID	Test Case Title	Test Condition	System Behavior	Expected Result
T01	Registration.	Student, Faculty Must be able to register themselves.	Registered successfully and dashboard option displayed depending on user type.	Registered successfully and dashboard option displayed depending on user type.

T02	Login.	Student, faculty must be able to login before doing any CRUD operation.	Login successfully with valid credentials and login failed with invalid credentials .	Login successfully with valid credentials and login failed with invalid credentials.
T03	Faculty – Add Course.	Add course to be add its assignment.	Course added successfully.	Course added successfully.
T04	Faculty- Add Assignment.	Add assignment for a course.	Assignment Added Successfully.	Assignment added successfully.
T05	Faculty- Add Questions.	Add questions in assignment.	Questions added row wise.	Questions added row wise.
T06	Faculty- Generate Model Answer.	Generate model answer wrt question title.	Model answer generated successfully.	Model answer generated successfully.
T07	Faculty – View all submitted.	View all submitted answers of each question to reject or view.	All answers displayed with reject option and similarity score.	All answers displayed with reject option and similarity score.
T08	Faculty- reject answer.	Reject students answer.	Students answer status updated to draft.	Students answer status updated to draft.
T09	Student– assignment view.	All assignments must fetched wrt students year and branch.	Assignments and courses are displayed as per students.	Courses and assignment should be visible Wrt students year and branch.
T10	Student- Select Assignment.	All assignments should be visibile in table format.	Assignments are visible along with view questions option.	Assignments are displayed row wise with questions option.
T011	Student – View Questions.	Question list to add answers.	All questions listed properly.	All questions must be listed.
T012	Student – Add answer.	Answer must be added properly.	Answer added successfully .	Answer should be added.
T013	Student- Answer status.	Answer should behave: Draft-not visible to anyone except owner. Publish: visible to everyone.	Draft: Answer only visible to student.Published: Displayed Publicly.	Draft: Answer only visible to student Published: Displayed Publicly.

T014	Faculty- Generated Question Bank .	Question Bank should be generated by randomly selecting questions from assignment.	Questions are randomly selected and number of questions is less than or equal to max questions.	Questions should be selected randomly and only less than or equal to max questions.
T015	Faculty- Print QB.	QB should be printed in pdf or paper.	New tab opened with print options or save as pdf.	Save as pdf option in new tab or print options.
T016	Filter Question on index page.	Questions should filtered wrt to course.	Only questions for selected course is displayed.	Working as expected.

7.2 Sample of a Test Case

Title: Login page – Authenticate user before accessing dashboard.
Description: Login page – Authenticate user before accessing dashboard.
Precondition: User is already registered.
Assumption: Login page – Authenticate user before accessing dashboard.

Test Steps:

1. Navigate to web application with address
2. In the 'Username' field, enter the username of the registered user.
3. Enter the password of the registered user
4. Click 'Log In'

Expected Result: User logged in successfully and redirected to home page. Dashboard with user type option visible on the navigation bar with log out option on far right of navbar
Actual Result: User(Faculty) was logged in and taken to home page. On clicking Dashobard, A 'faculty' option appeared, on clicking user was taken to faculty dashboard.

Chapter 8

Screenshots of Project

Login

There are three modules in our system, to get into the system the user must first login.

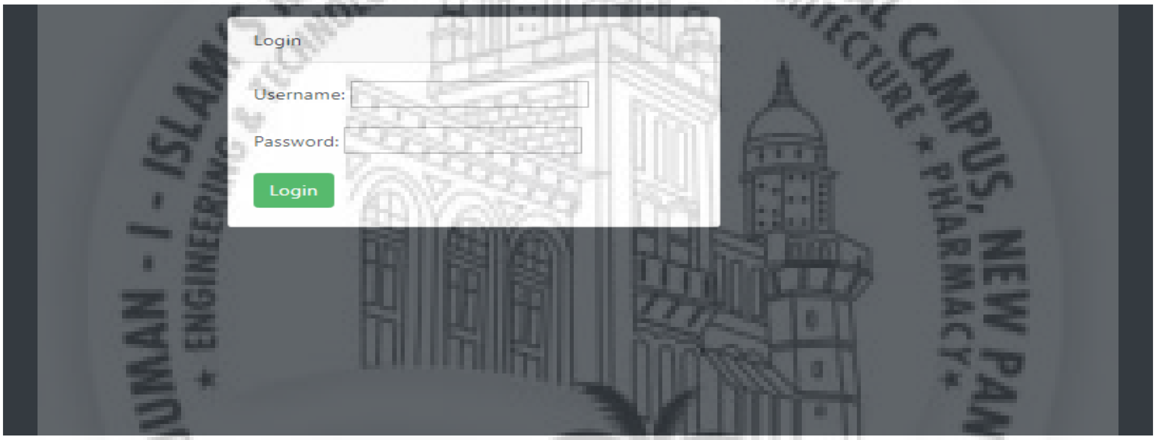
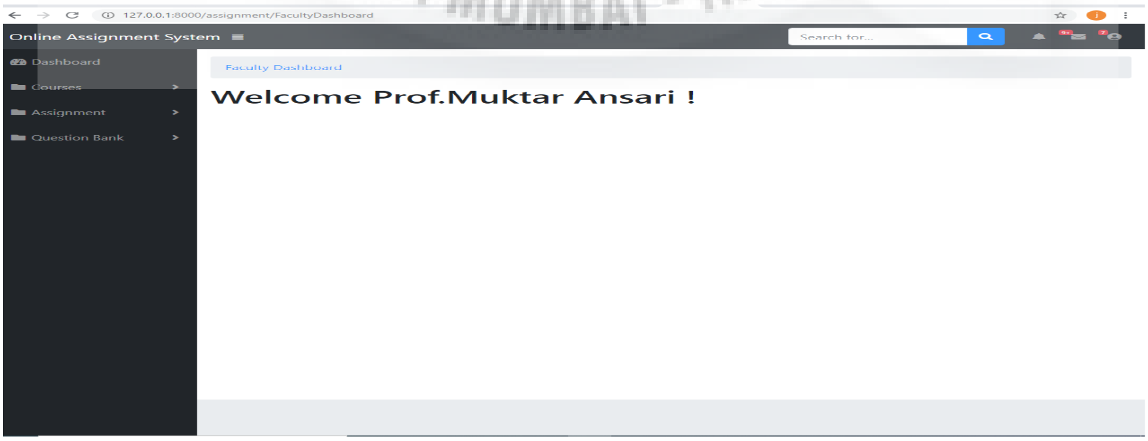


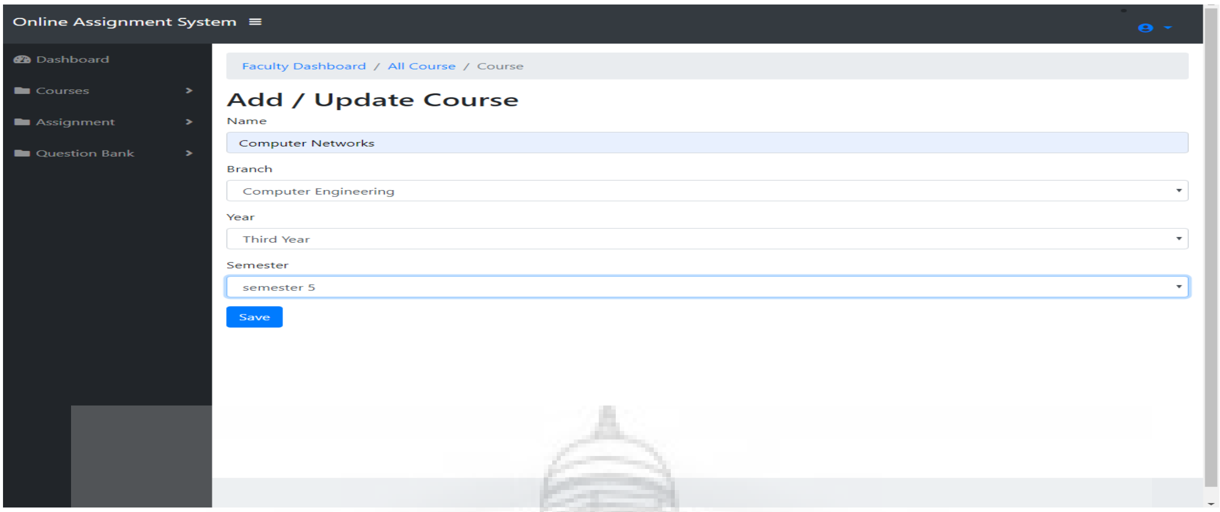
Figure 8.1: Login Page.

8.1 Faculty



Welcome Prof.Muktar Ansari !

Figure 8.2: Faculty Dashboard.



Faculty Dashboard / All Course / Course

Add / Update Course

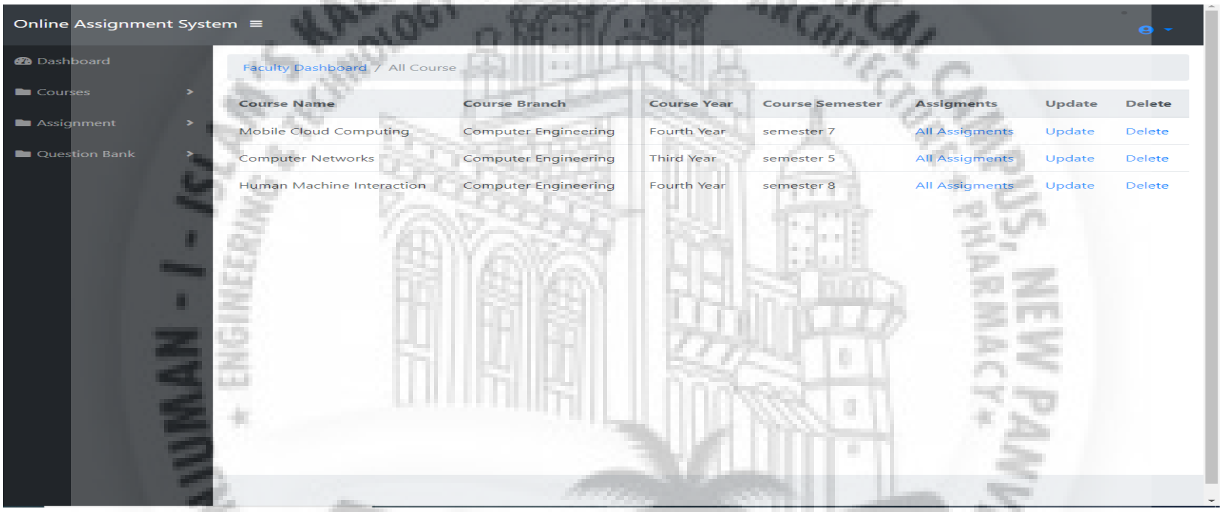
Name: Computer Networks

Branch: Computer Engineering

Year: Third Year

Semester: semester 5

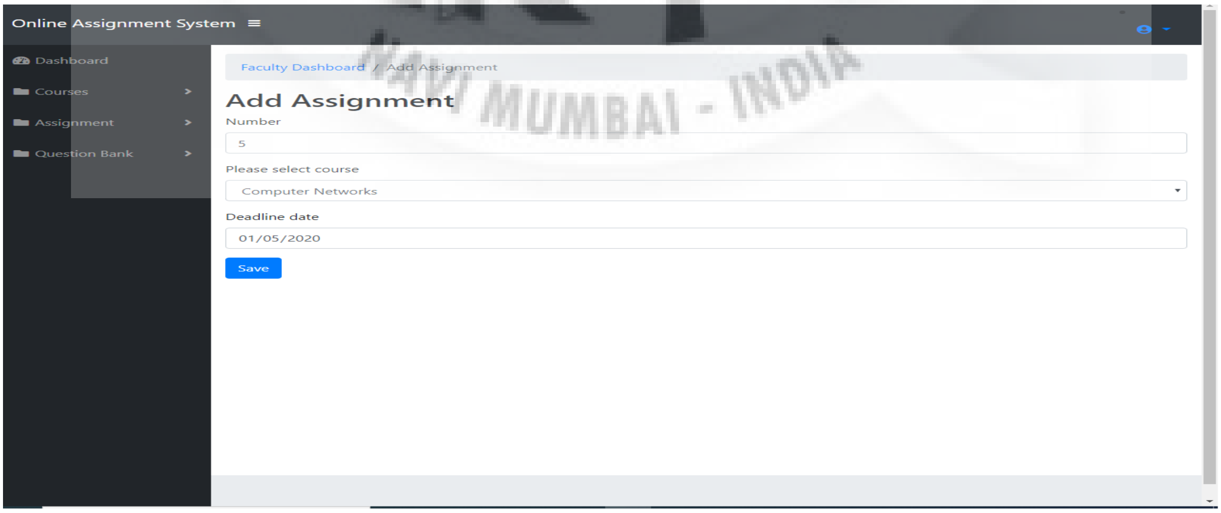
Figure 8.3: Add course.



Faculty Dashboard / All Course

Course Name	Course Branch	Course Year	Course Semester	Assigments	Update	Delete
Mobile Cloud Computing	Computer Engineering	Fourth Year	semester 7	All Assigments	Update	Delete
Computer Networks	Computer Engineering	Third Year	semester 5	All Assigments	Update	Delete
Human Machine Interaction	Computer Engineering	Fourth Year	semester 8	All Assigments	Update	Delete

Figure 8.4: Course Dashboard.



Faculty Dashboard / Add Assignment

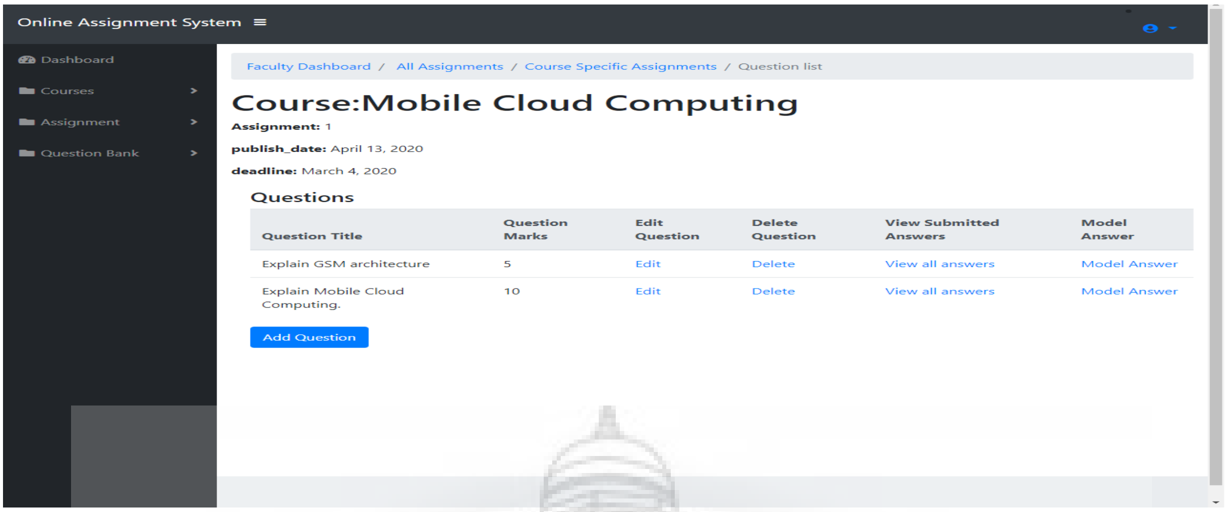
Add Assignment

Number: 5

Please select course: Computer Networks

Deadline date: 01/05/2020

Figure 8.5: Add Assignment.


Online Assignment System ≡
Dashboard
Courses >
Assignment >
Question Bank >
Faculty Dashboard / All Assignments / Course Specific Assignments / Question list
Add Question

Course:Mobile Cloud Computing

Assignment: 1

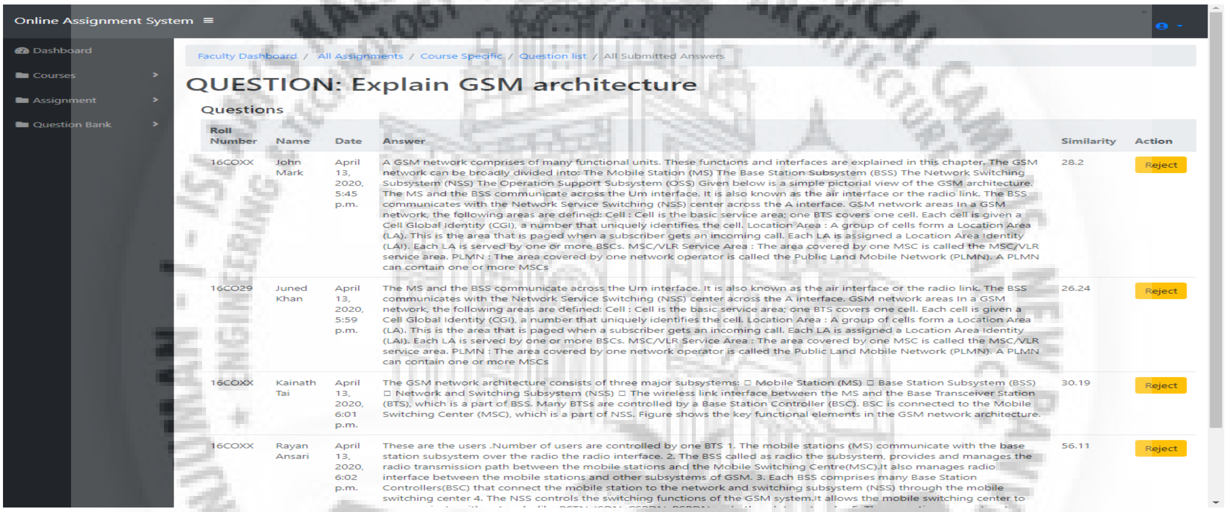
publish_date: April 13, 2020

deadline: March 4, 2020

Questions

Question Title	Question Marks	Edit Question	Delete Question	View Submitted Answers	Model Answer
Explain GSM architecture	5	Edit	Delete	View all answers	Model Answer
Explain Mobile Cloud Computing.	10	Edit	Delete	View all answers	Model Answer

Figure 8.6: Assignment Dashboard.

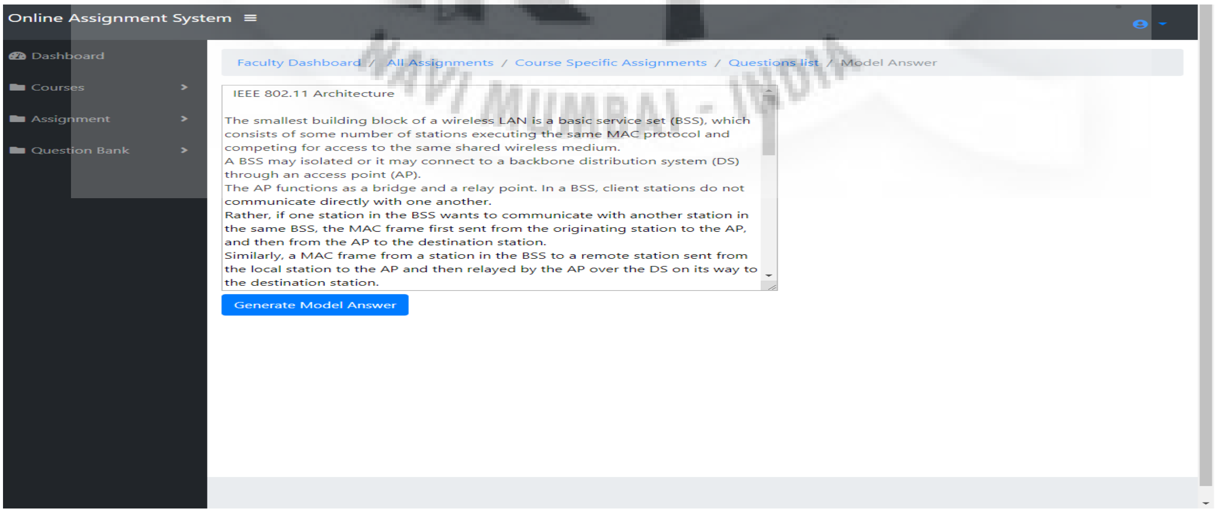

Online Assignment System ≡
Dashboard
Courses >
Assignment >
Question Bank >
Faculty Dashboard / All Assignments / Course Specific / Question list / All Submitted Answers

QUESTION: Explain GSM architecture

Questions

Roll Number	Name	Date	Answer	Similarity	Action
16COXX	John Mark	April 13, 2020, 5:45 p.m.	A GSM network comprises of many functional units. These functions and interfaces are explained in this chapter. The GSM network can be broadly divided into: The Mobile Station (MS) The Base Station Subsystem (BSS) The Network Switching Subsystem (NSS) The Operation Support Subsystem (OSS) Given below is a simple pictorial view of the GSM architecture. The MS and the BSS communicate across the Um interface. It is also known as the air interface or the radio link. The BSS communicates with the Network Service Switching (NSS) center across the A interface. GSM network areas In a GSM network, the following areas are defined: Cell : Cell is the basic service area; one BTS covers one cell. Each cell is given a Cell Global Identity (CGI), a number that uniquely identifies the cell. Location Area : A group of cells form a Location Area (LA). This is the area that is paged when a subscriber gets an incoming call. Each LA is assigned a Location Area Identity (LAI). Each LA is served by one or more BSCs. MSC/VLR Service Area : The area covered by one MSC is called the MSC/VLR service area. PLMN : The area covered by one network operator is called the Public Land Mobile Network (PLMN). A PLMN can contain one or more MSCs	28.2	Reject
16CO29	Juned Khan	April 13, 2020, 5:59 p.m.	The MS and the BSS communicate across the Um interface. It is also known as the air interface or the radio link. The BSS communicates with the Network Service Switching (NSS) center across the A interface. GSM network areas In a GSM network, the following areas are defined: Cell : Cell is the basic service area; one BTS covers one cell. Each cell is given a Cell Global Identity (CGI), a number that uniquely identifies the cell. Location Area : A group of cells form a Location Area (LA). This is the area that is paged when a subscriber gets an incoming call. Each LA is assigned a Location Area Identity (LAI). Each LA is served by one or more BSCs. MSC/VLR Service Area : The area covered by one MSC is called the MSC/VLR service area. PLMN : The area covered by one network operator is called the Public Land Mobile Network (PLMN). A PLMN can contain one or more MSCs	26.24	Reject
16COXX	Kainath Tai	April 13, 2020, 6:01 p.m.	The GSM network architecture consists of three major subsystems: □ Mobile Station (MS) □ Base Station Subsystem (BSS) □ Network and Switching Subsystem (NSS) □ The wireless link interface between the MS and the Base Transceiver Station (BTS), which is a part of BSS. Many BTSs are controlled by a Base Station Controller (BSC). BSC is connected to the Mobile Switching Center (MSC), which is a part of NSS. Figure shows the key functional elements in the GSM network architecture.	30.19	Reject
16COXX	Rayan Ansari	April 13, 2020, 6:02 p.m.	These are the users .Number of users are controlled by one BTS 1. The mobile stations (MS) communicate with the base station subsystem over the radio the radio interface. 2. The BSS called as radio the subsystem, provides and manages the radio transmission path between the mobile stations and the Mobile Switching Centre(MSC).It also manages radio interface between the mobile stations and other subsystems of GSM. 3. Each BSS comprises many Base Station Controllers(BSC) that connect the mobile station to the network and switching subsystem (NSS) through the mobile switching center 4. The NSS controls the switching functions of the GSM system.It allows the mobile switching center to	56.11	Reject

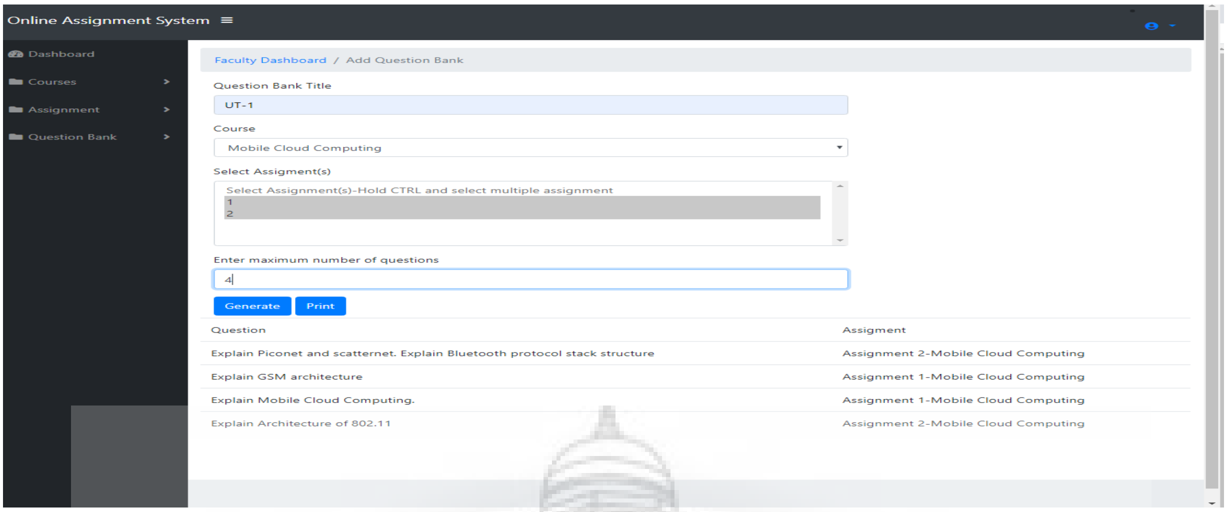
Figure 8.7: Submitted Assignment Answers.


Online Assignment System ≡
Dashboard
Courses >
Assignment >
Question Bank >
Faculty Dashboard / All Assignments / Course Specific Assignments / Questions list / Model Answer
Generate Model Answer

IEEE 802.11 Architecture

The smallest building block of a wireless LAN is a basic service set (BSS), which consists of some number of stations executing the same MAC protocol and competing for access to the same shared wireless medium.
A BSS may isolated or it may connect to a backbone distribution system (DS) through an access point (AP).
The AP functions as a bridge and a relay point. In a BSS, client stations do not communicate directly with one another.
Rather, if one station in the BSS wants to communicate with another station in the same BSS, the MAC frame first sent from the originating station to the AP, and then from the AP to the destination station.
Similarly, a MAC frame from a station in the BSS to a remote station sent from the local station to the AP and then relayed by the AP over the DS on its way to the destination station.

Figure 8.8: Model Answer.



Faculty Dashboard / Add Question Bank

Question Bank Title: UT-1

Course: Mobile Cloud Computing

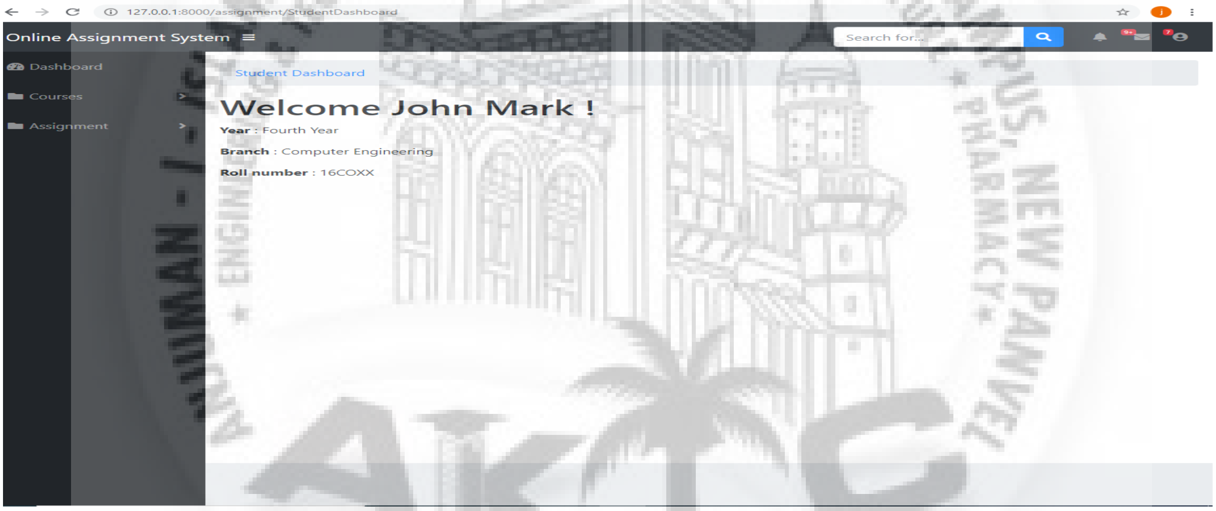
Select Assigment(s): Select Assignment(s)-Hold CTRL and select multiple assignment
1
2

Enter maximum number of questions: 4

Question	Assigment
Explain Piconet and scatternet. Explain Bluetooth protocol stack structure	Assignment 2-Mobile Cloud Computing
Explain GSM architecture	Assignment 1-Mobile Cloud Computing
Explain Mobile Cloud Computing.	Assignment 1-Mobile Cloud Computing
Explain Architecture of 802.11	Assignment 2-Mobile Cloud Computing

Figure 8.9: Question Bank

8.2 Student


127.0.0.1:8000/assignment/StudentDashboard
Online Assignment System ≡
Search for...
Dashboard
Courses >
Assignment >

Student Dashboard

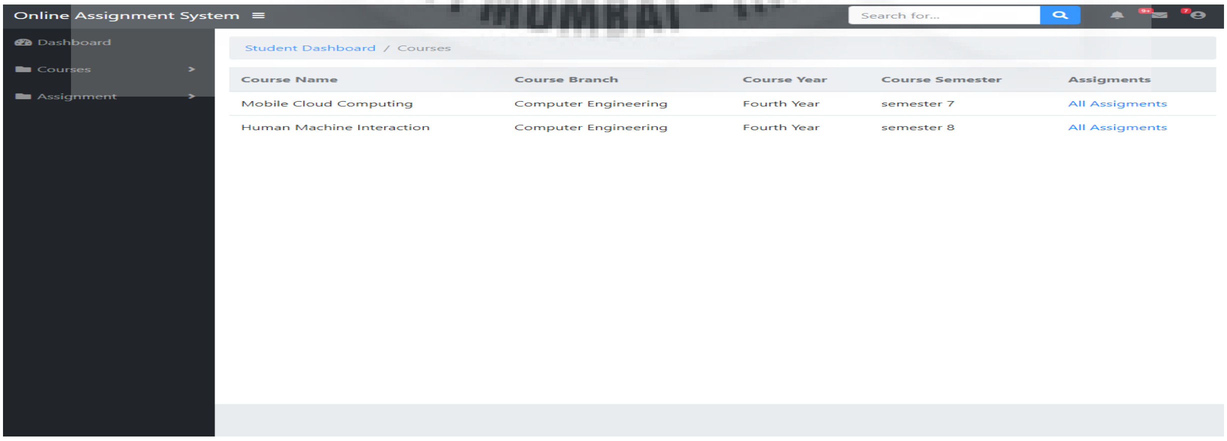
Welcome John Mark !

Year : Fourth Year

Branch : Computer Engineering

Roll number : 16COXX

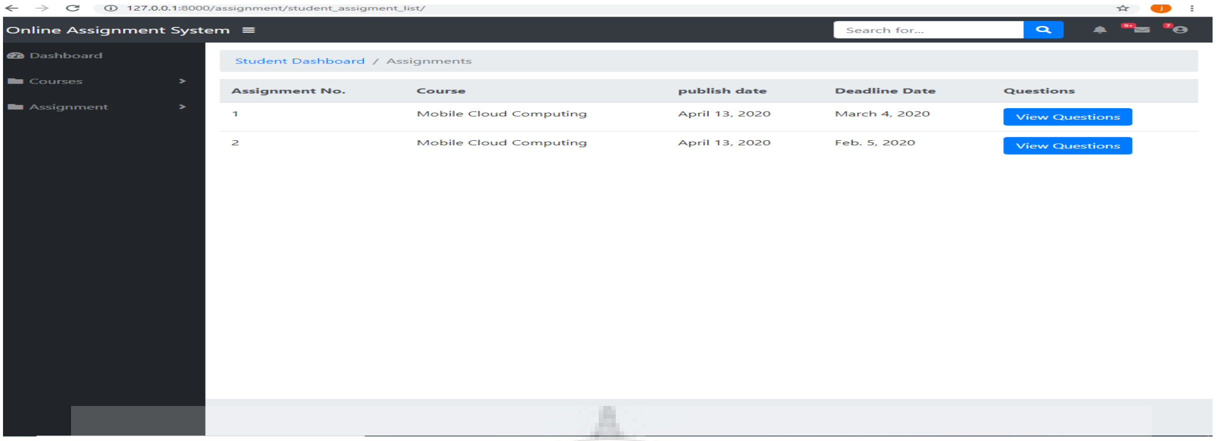
Figure 8.10: Student Dashboard.



Student Dashboard / Courses

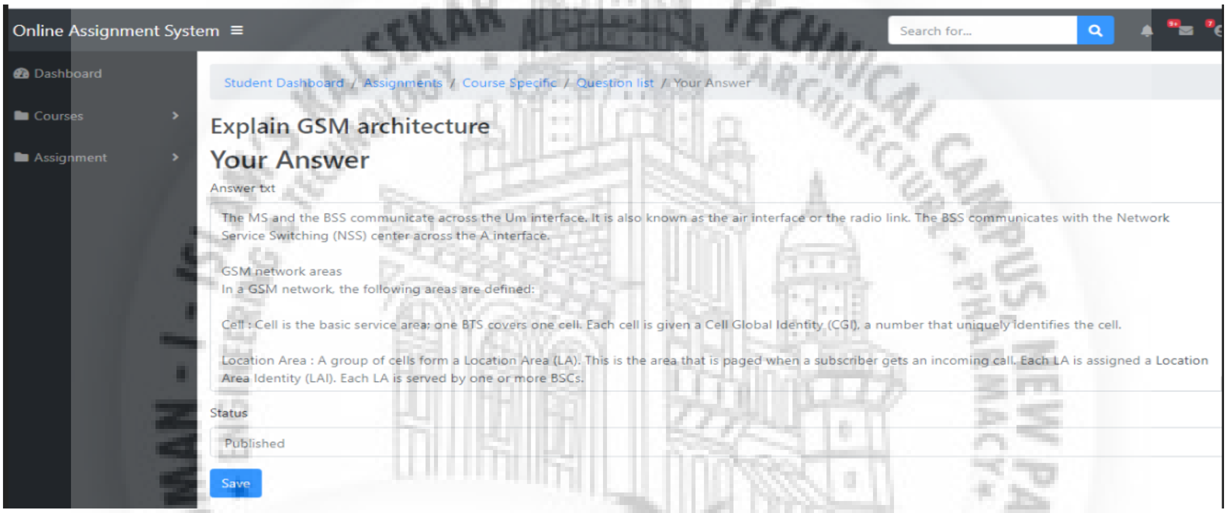
Course Name	Course Branch	Course Year	Course Semester	Assigments
Mobile Cloud Computing	Computer Engineering	Fourth Year	semester 7	All Assigments
Human Machine Interaction	Computer Engineering	Fourth Year	semester 8	All Assigments

Figure 8.11: Course Wise Assignment.


127.0.0.1:8000/assignment/student_assigment_list/
Online Assignment System
Search for...
Dashboard
Courses
Assignment
Student Dashboard / Assignments

Assignment No.	Course	publish date	Deadline Date	Questions
1	Mobile Cloud Computing	April 13, 2020	March 4, 2020	View Questions
2	Mobile Cloud Computing	April 13, 2020	Feb. 5, 2020	View Questions

Figure 8.12: Assignment List.


Online Assignment System
Search for...
Dashboard
Courses
Assignment
Student Dashboard / Assignments / Course Specific / Question list / Your Answer

Explain GSM architecture

Your Answer

Answer txt

The MS and the BSS communicate across the Um interface. It is also known as the air interface or the radio link. The BSS communicates with the Network Service Switching (NSS) center across the A interface.

GSM network areas
In a GSM network, the following areas are defined:

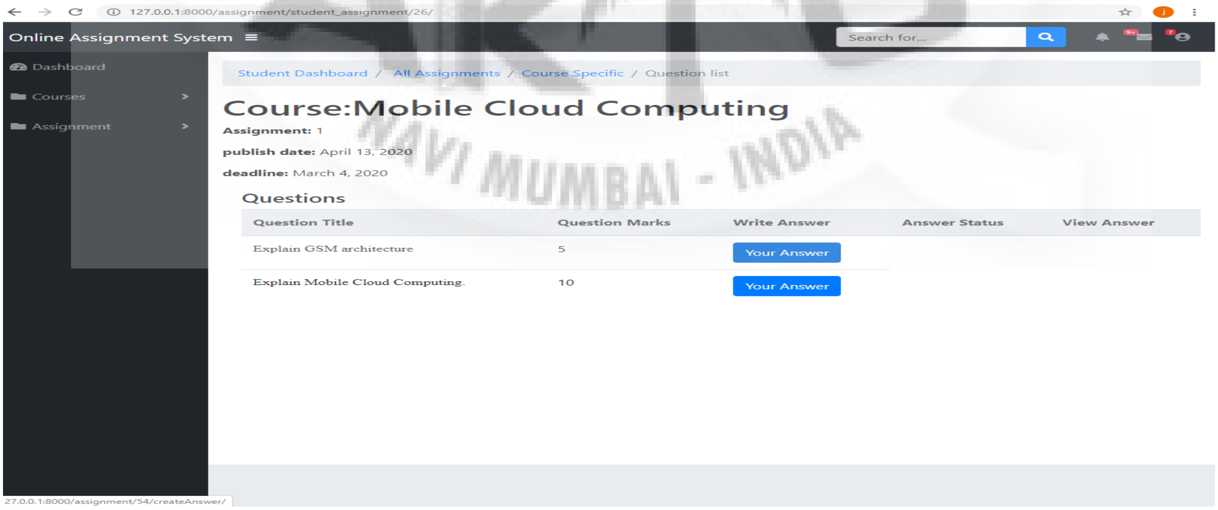
Cell : Cell is the basic service area; one BTS covers one cell. Each cell is given a Cell Global Identity (CGI), a number that uniquely identifies the cell.

Location Area : A group of cells form a Location Area (LA). This is the area that is paged when a subscriber gets an incoming call. Each LA is assigned a Location Area Identity (LAI). Each LA is served by one or more BSCs.

Status

Published

Figure 8.13: Add Answer.


127.0.0.1:8000/assignment/student_assignment/26/
Online Assignment System
Search for...
Dashboard
Courses
Assignment
Student Dashboard / All Assignments / Course Specific / Question list

Course:Mobile Cloud Computing

Assignment: 1

publish date: April 13, 2020

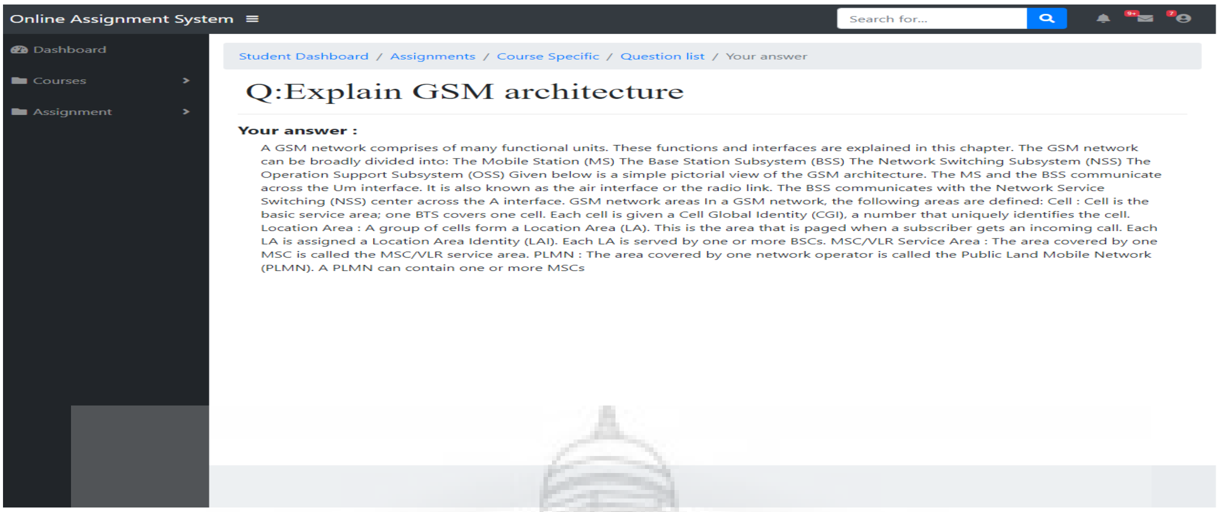
deadline: March 4, 2020

Questions

Question Title	Question Marks	Write Answer	Answer Status	View Answer
Explain GSM architecture	5	Your Answer		
Explain Mobile Cloud Computing.	10	Your Answer		

27.0.0.1:8000/assignment/54/createAnswer/

Figure 8.14: Answers List.

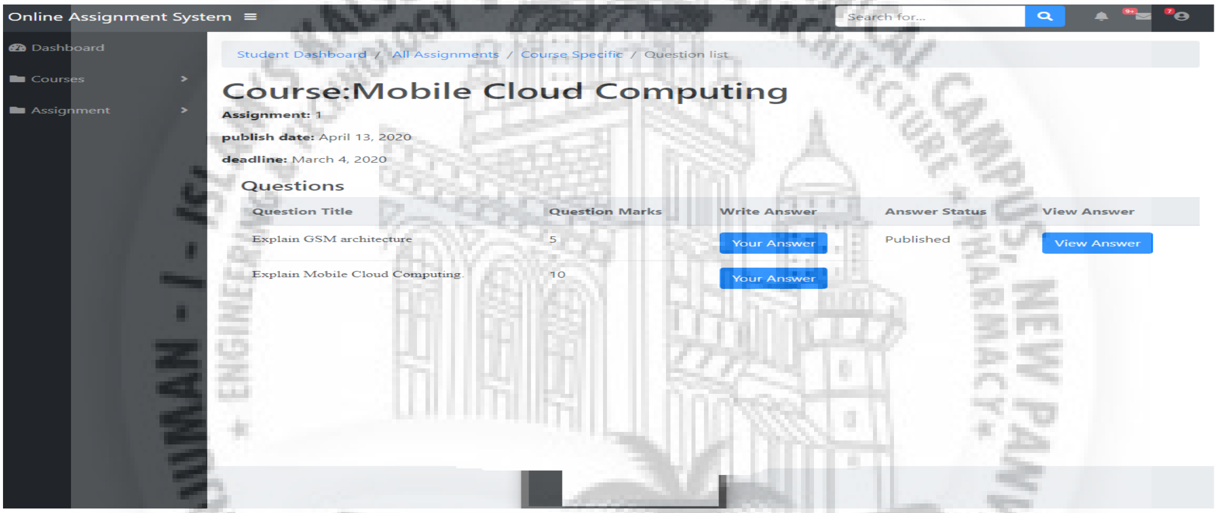

Online Assignment System
Search for...
9+
7
Dashboard
Courses
Assignment
Student Dashboard / Assignments / Course Specific / Question list / Your answer

Q:Explain GSM architecture

Your answer :

A GSM network comprises of many functional units. These functions and interfaces are explained in this chapter. The GSM network can be broadly divided into: The Mobile Station (MS) The Base Station Subsystem (BSS) The Network Switching Subsystem (NSS) The Operation Support Subsystem (OSS) Given below is a simple pictorial view of the GSM architecture. The MS and the BSS communicate across the Um interface. It is also known as the air interface or the radio link. The BSS communicates with the Network Service Switching (NSS) center across the A interface. GSM network areas In a GSM network, the following areas are defined: Cell : Cell is the basic service area; one BTS covers one cell. Each cell is given a Cell Global Identity (CGI), a number that uniquely identifies the cell. Location Area : A group of cells form a Location Area (LA). This is the area that is paged when a subscriber gets an incoming call. Each LA is assigned a Location Area Identity (LAI). Each LA is served by one or more BSCs. MSC/VLR Service Area : The area covered by one MSC is called the MSC/VLR service area. PLMN : The area covered by one network operator is called the Public Land Mobile Network (PLMN). A PLMN can contain one or more MSCs

Figure 8.15: Read only view of answer.


Online Assignment System
Search for...
9+
7
Dashboard
Courses
Assignment
Student Dashboard / All Assignments / Course Specific / Question list

Course:Mobile Cloud Computing

Assignment: 1

publish date: April 13, 2020

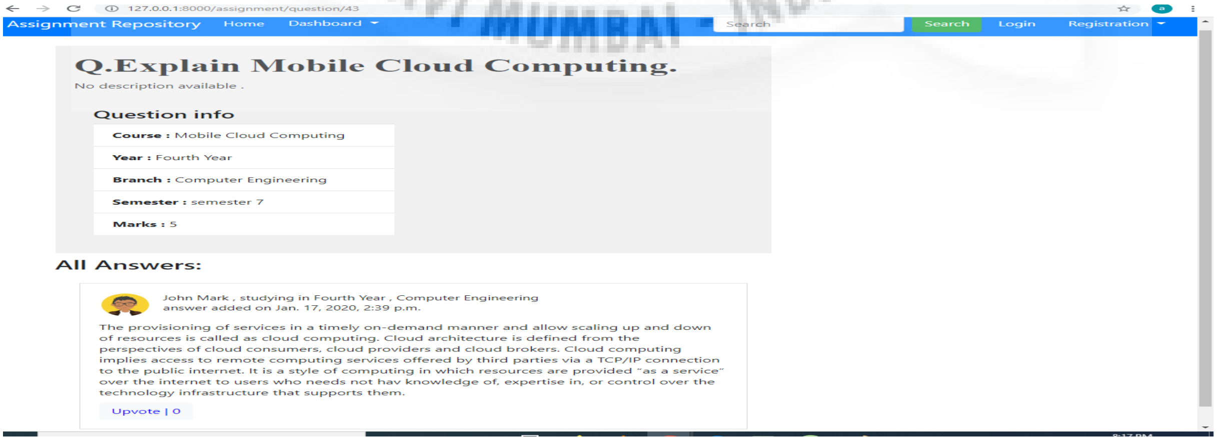
deadline: March 4, 2020

Questions

Question Title	Question Marks	Write Answer	Answer Status	View Answer
Explain GSM architecture	5	Your Answer	Published	View Answer
Explain Mobile Cloud Computing	10	Your Answer		

Figure 8.16: Answers Status.

8.3 Assignment Repository



Q.Explain Mobile Cloud Computing.

No description available .

Question info

Course : Mobile Cloud Computing

Year : Fourth Year

Branch : Computer Engineering

Semester : semester 7

Marks : 5

All Answers:

John Mark , studying in Fourth Year , Computer Engineering
answer added on Jan. 17, 2020, 2:39 p.m.

The provisioning of services in a timely on-demand manner and allow scaling up and down of resources is called as cloud computing. Cloud architecture is defined from the perspectives of cloud consumers, cloud providers and cloud brokers. Cloud computing implies access to remote computing services offered by third parties via a TCP/IP connection to the public internet. It is a style of computing in which resources are provided "as a service" over the internet to users who needs not hav knowledge of, expertise in, or control over the technology infrastructure that supports them.

Figure 8.17: Home Page of Answer.

Chapter 9

Conclusion and Future Scope

9.1 Conclusion

A web based Online Assignment System facilitated submission of assignments by students. The system consists of modules where students could upload their answer. Notification are sent to students when new assignment is added and also when lecturer approve or disapprove the assignment.
The assignment answer is corrected using keyword matching, online assignment system also generates question bank from the remaining assignment question, online assignment system is user friendly and also reduce work load of faculty and student, even it minimizes paper work.
Evaluation of web based online submission of assignment was done through surveying among university students and lecturers. Test results of online submission concluded that system is very friendly and would be very helpful to UOM.

9.2 Future Scope

- We can add various additional modules to our current system, making it more efficient in the way the system can accept data that is to be corrected.
- To add module that would accept voice data from a microphone and correct the same without any human assistance, using the same algorithm.

- To add module to accept answers from an image source, by using image to text conversion algorithms, entire papers can be scanned, and can be corrected.

References

[1] Student Perceptions of the impact of web-based homework on course interaction and learning in Introductory Accounting. Issue in Informa- tion Systems, 9(1), 223 – 232

[2] Black board Learn, Release 9.1- Our Next Generation Teaching and Learning Platform; https://online.uedu/les/blackboardlearn/blackboard.learn.brochure.pdf

[3] Salomon, G. (2000).;). Education and technology in the age of information. HaifaandTel Aviv: Haifa University Publishers and Zmora-Bitan

[4] Our Next Generation Teaching and Learning Platform https://online.fiu.edu/files/blackboardlearn/blackboardlearnbrochure

[5] Get started with WebRTC http://www.netmagazine.com/tutorials/get-startedwebrtc

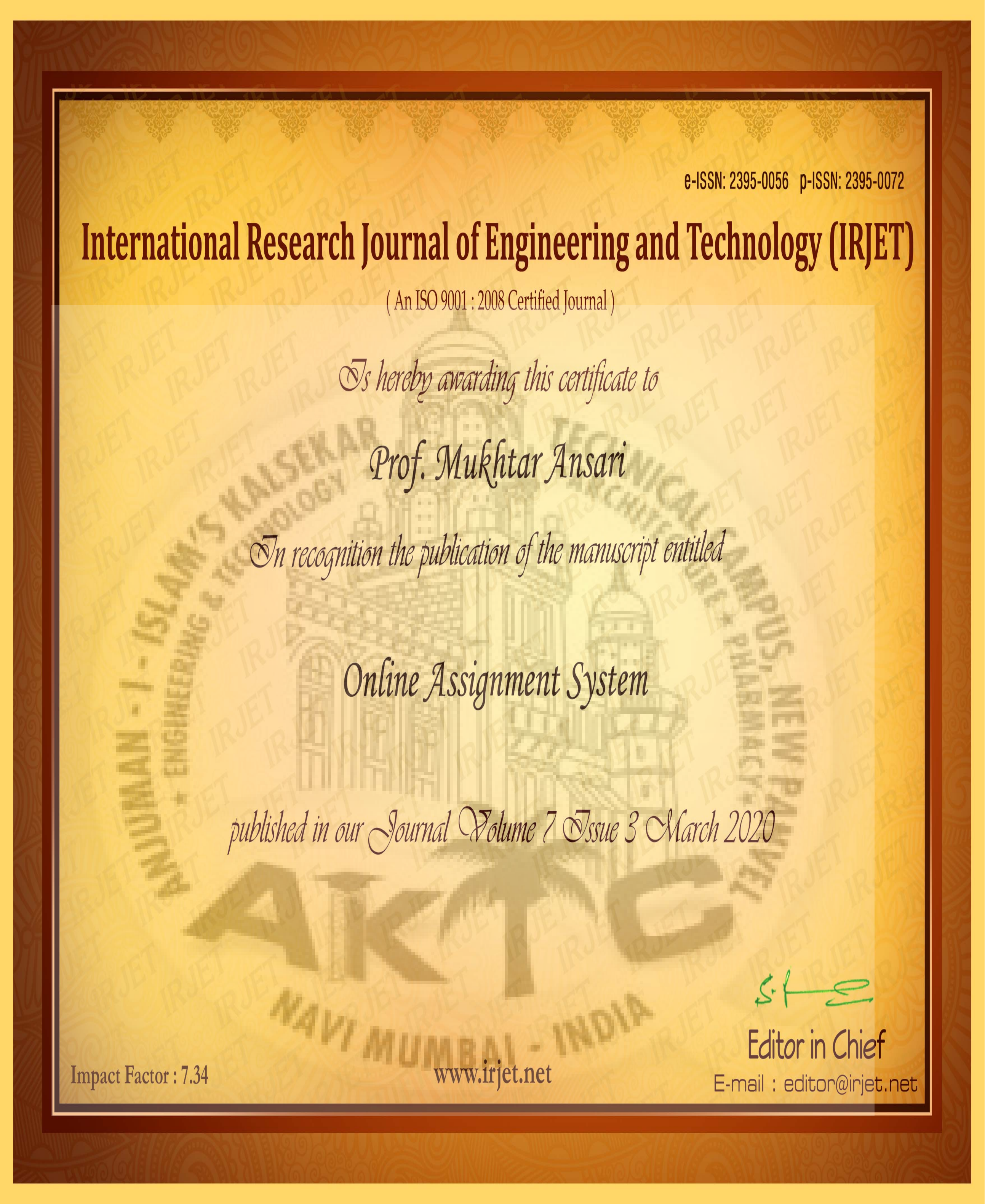
[6] BillBell.(2018,February)StackOverflow.[Online]. https://stackoverflow.com/questions/1746501/can-someone-give-an-example-of-cosine-similarity-in-a-very-simple-graphical-wa

[7] Rabab A. Wahab, S. A.-A., 2015; Student's Attitude Towards Online Assignment Submission at College of Health Sciences. Bahrain, International Conference On E-Learning

Achievements

1. Publications

   (a) *Online Assignment System*;Tai Kainath,Ansari Rayan,Khan Juned, Prof.Mukhtar Ansari,International Research Journal of Engineering and Technology (IRJET) , March-2020(`www.irjet.net`)



e-ISSN: 2395-0056 p-ISSN: 2395-0072

International Research Journal of Engineering and Technology (IRJET)

(An ISO 9001 : 2008 Certified Journal)

Is hereby awarding this certificate to

Prof. Mukhtar Ansari

In recognition the publication of the manuscript entitled

Online Assignment System

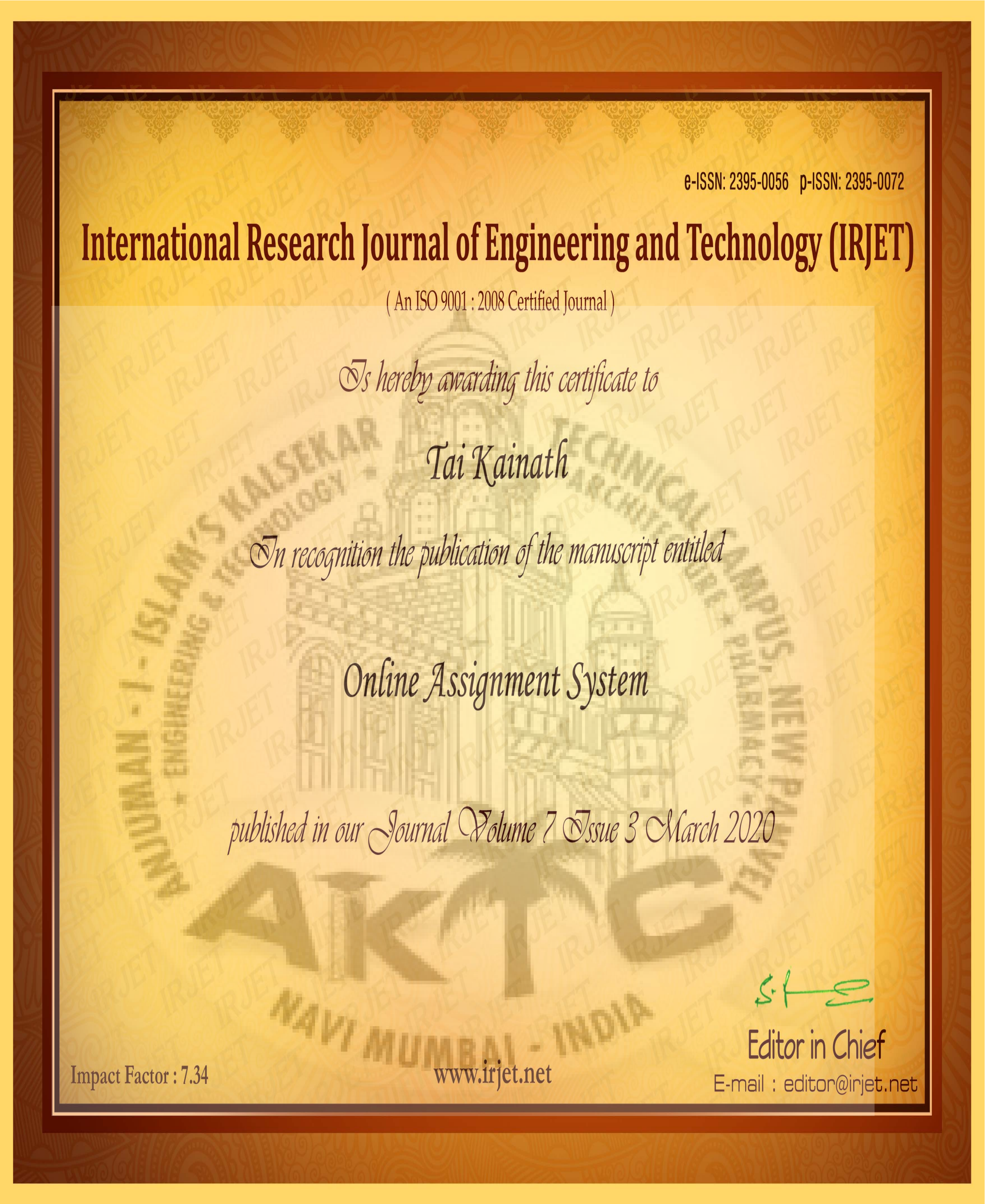
published in our Journal Volume 7 Issue 3 March 2020

Impact Factor : 7.34

www.irjet.net

Editor in Chief

E-mail : editor@irjet.net



e-ISSN: 2395-0056 p-ISSN: 2395-0072

International Research Journal of Engineering and Technology (IRJET)

(An ISO 9001 : 2008 Certified Journal)

Is hereby awarding this certificate to

Tai Kainath

In recognition the publication of the manuscript entitled

Online Assignment System

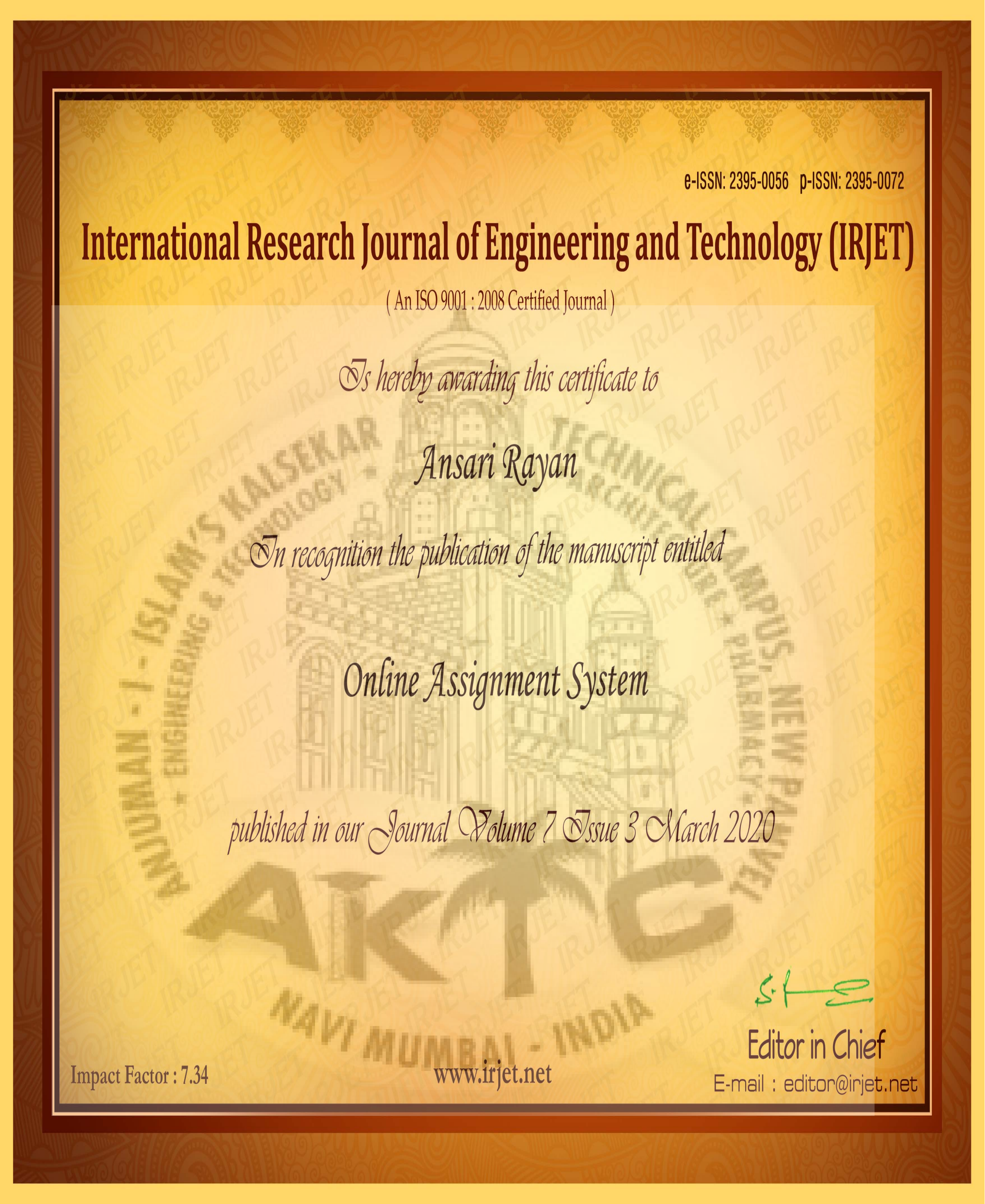
published in our Journal Volume 7 Issue 3 March 2020

Impact Factor : 7.34

www.irjet.net

Editor in Chief

E-mail : editor@irjet.net



e-ISSN: 2395-0056 p-ISSN: 2395-0072

International Research Journal of Engineering and Technology (IRJET)

(An ISO 9001 : 2008 Certified Journal)

Is hereby awarding this certificate to

Ansari Rayan

In recognition the publication of the manuscript entitled

Online Assignment System

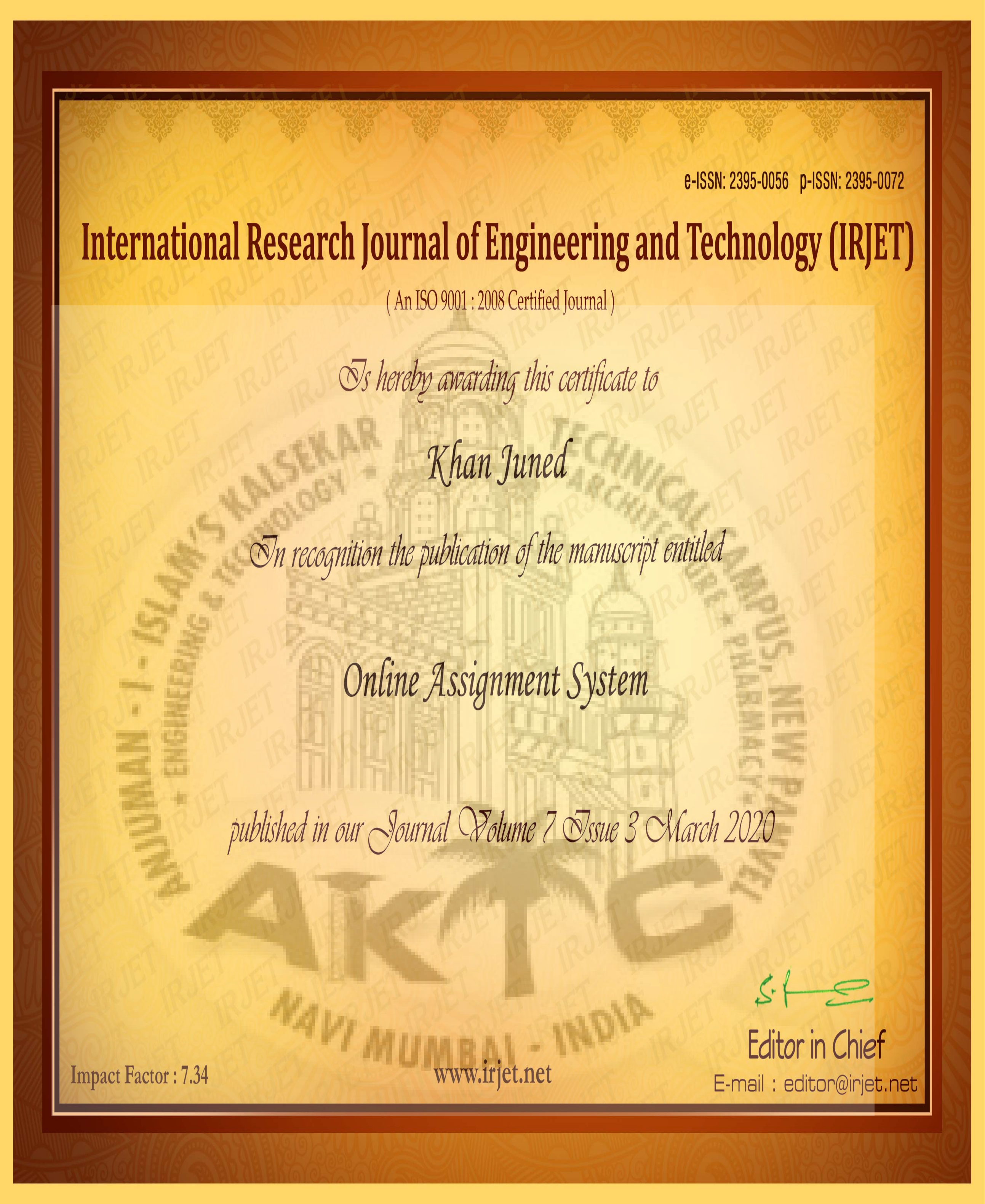
published in our Journal Volume 7 Issue 3 March 2020

Impact Factor : 7.34

www.irjet.net

Editor in Chief

E-mail : editor@irjet.net



e-ISSN: 2395-0056 p-ISSN: 2395-0072

International Research Journal of Engineering and Technology (IRJET)

(An ISO 9001 : 2008 Certified Journal)

Is hereby awarding this certificate to

Khan Juned

In recognition the publication of the manuscript entitled

Online Assignment System

published in our Journal Volume 7 Issue 3 March 2020

Impact Factor : 7.34

www.irjet.net

Editor in Chief

E-mail : editor@irjet.net

International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 07 Issue: 03 | Mar 2020 www.irjet.net p-ISSN: 2395-0072

Online Assignment System

Tai Kainath[1], Ansari Rayan[2], Khan Juned[3], Prof. Mukhtar Ansari[4]

[1]Tai Kainath, Student, Dept of Computer Engineering, Anjuman Islam's Kalsekar Technical Campus, Maharashtra, Navi Mumbai
[2]Ansari Rayan, Student, Dept of Computer Engineering, Anjuman Islam's Kalsekar Technical Campus, Maharashtra, Navi Mumbai
[3]Khan Juned, Student, Dept of Computer Engineering, Anjuman Islam's Kalsekar Technical Campus, Maharashtra, Navi Mumbai
[4]Prof. Mukhtar Ansari, Student, Dept of Computer Engineering, Anjuman Islam's Kalsekar Technical Campus, Maharashtra, Navi Mumbai

---***---

Abstract – *Online Assignment System is a web-based module which allows student and faculty to communicate about assign- ment and queries. Student can submit assignment and faculty can correct the answer and rate the best answer which would be uploaded as the best answer and the students can further refer that answer as notes. There will be certain keyword if the student's answer matches the maximum keyword provided by the faculty then the faculty will display that result as the best answer and the answer will be highlighted. Faculty can also generate question paper from the assignment questions. Many colleges are adopting this module as it is efficient and decrease the paper-work and human efforts.*

Key Words: **Web-based Assignment Submission; University Question and Answer Repository**

1.INTRODUCTION

The problem arises from manual system used by colleges is that it has no record about the time or marks of student assignment. Some of the student hand over their assignment after the due date. Some of the assignments can be lost by students as well as faculty and as we do not have any record about it students have to write the assignment once again. As there is no record of assignment faculty would nd it difficult to check whether the student have submitted the assignment or not. After the faculty correct the assignment and return it back, some student does not get their assignment back as anyone can take it or misplace it, to avoid this we are cre- ating an online assignment system. The main motive of this project is to create a paperless environment by developing online assignment submission. This project also reduces the unnecessary workload of lecturers and student. The scope for the Online Assignment System is that student can submit their assignment anytime and anywhere there is no need of the presences of lecturer as the student is submitting the assignment online. The collection of the assignments will be kept by the lecturer and the student can view their marks anytime they want. This system is very useful as well as innovative because there is no paper involved and even this system is user friendly. Student and Faculty can use this system anytime anywhere. The answer is corrected automatically with the help of automated search engine and the keyword provided by the faculty. Faculty does not need to check the assignment answer manually as all the checking will be done through key word matching. The best answer will be highlighted as the most rated answer and that answer will be referred as notes by the student. This system generated question paper from the assignment question which student can referred for their examination. The generation of question paper is done by faculty and the faculty has to select the question from assignment question which he/she is going to add in the question paper.

2. Literature Survey

The teacher teaching the particular subject will decide which question will be added to the assignment. In file system, the created pdf of assignment is stored using same algorithm for generation of online assignment. This system is activated only after the student completely solves the question and submits the assignment. This system instantly calculates and display the score to the student and at the same time the system uploads the score in the college database

The paper [?] In this paper, there are three modules

- Login/ Registration Module: In this, if the user wish to use the system he/she has to login first.
- The Question Addition Module : In this, faculty will add Assignment question and the student has to answer it in given time period and the respective faculty has to also add the answer in his/her own words.
- Assignment Generation Module : The lecturer teaching the particular subject will decide the number of question that have to be added to the assignment.In file system ,the created pdf of assignment is stored using the same algorithm for generation of online assignment. This system is activated only after the student completely solves the question and submits the assignment. This system instantly calculate and display the score to the student and at the same time

the system uploads the score in the college database.

2.1 Weakness

- The system corrects the answer as the length of the answer is such that the length of the answer is not appropriate, indicating that the student was not able to write important information and if the length of answer is too long it indicates that the student has filled too much information respectively

- If length of answer written by student matches the required length, then the student will get suitable marks or else student will get less marks

2. 2 How to overcome

System corrects the answer in the form of keywords matching if the student answer matches the maximum keyword then the answer is displayed as correct. Our system is irrespective of the length, only checks the answer based on keyword.

The paper [?] There are three users in this system:

- Question Designer (QD): Question designer consists of questions based on different field / subject / subjects as per the specification given by the institute along with the scheme of evaluation.
- Question Paper Designer (QPD): Each submitted question will be verified by Administrator which is also called as special committee of institute for recurrence for equivalence, precision and realistic. History of questions in question paper is main- tained so there should not be any reiterating issues. A question that is to be added in the question paper, are divided in different types like question based on marks, difficulty level, type and topic. QPD is based on important features and parameters to smart system so that we can edit the question paper in given time.
- Administrator (for authentication): After the verification is done, the organization identifies the question paper designer (QPD) and the registered people as question designers (QD) and then provides them login credentials.

Based on the parameters of Question Paper Designer, it generates question paper by selecting questions from back-end. In our system we are inculcate intelligence by compress the concept locate with effectiveness of algorithm, good combination of questions is generated as result.

2.3 Weakness

- This system contains questions which is added to the question bank. They are divided into different types like question based on points, difficulty level, type and subject.

2.4 How to Overcome

- Our system will provide all sort of question, it depends on student which type of question he/she will attempt. Student does not have to add various dimensions, they have to only select the courses and the assignment will be displayed.

The paper [?]The proposed system provides only one intro- duction to all administrative systems of the University. This system greatly simplifies and accelerates the organization and management process. This crosses the boundaries of a desktop-based system because our cloud can be accessed via Android, iOS, and the Web. Good communication between university and students with the help of instant notification with email and SMS. Students can access all their information on their smartphone with just one tap. Human errors in computation can be eliminated. A large amount of papers was saved. Hence, the environmentally friendly system. The system provides reliability, time saving and easy control. It can be used as a basis for creating and enhancing applications for all of the above modules. This greatly simplifies and accelerates the process of preparation and management of results. Our system is protected with the latest security measures using Google Firebase Authentication. This reduces the amount of work and resources needed in the traditional process. Time consuming is reduced and manual calculations are also ignored, reports can be received regularly and to the extent of consumer demand. The proposed system provides a new way of computing and performing operations on mobile devices with a responsive and attractive user-interface.

2.5 Weakness

In this system student can challenge the lecturer by demanding the photocopy of the answer sheet and can also apply for revaluation based on the result provided by to them.

2.6 How to Overcome

Our system only checks the answer based on key- word matching, the answer will be displayed on the dashboard on need of demanding the photocopy

3. Methodology

3.1 System Architecture

3.1.1 Uploader

Uploader module allows faculty members to add assignments and allows students to upload answers for the assignments. The assignments and answers will be store in database for future usage. This stored answer can later be fetched by Question Bank Generator to generate question bank

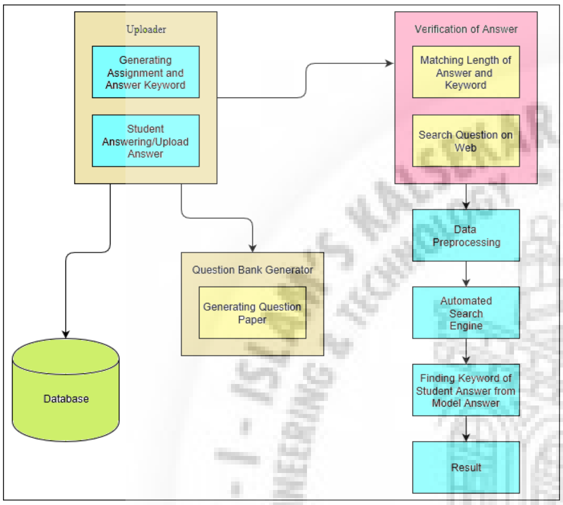

Uploader
Generating Assignment and Answer Keyword
Student Answering/Upload Answer
Verification of Answer
Matching Length of Answer and Keyword
Search Question on Web
Data Preprocessing
Automated Search Engine
Finding Keyword of Student Answer from Model Answer
Result
Question Bank Generator
Generating Question Paper
Database

Fig -1: System Architecture

3.1.2 Question Bank Generator

Question bank Generator helps faculty members to generate Question Bank by selecting uploaded assignment questions. Depending on the number of questions selected, questions will be randomly taken from the selected assignment and inserted in the question bank

3.1.3 Verification of Answer

- In this module, we evaluate students answer by com- paring it with the model answer. The comparison is done with the help of cosine similarity algorithm. After the evaluation, the accuracy measure will be returned for that answer.
- The Model Answer is generated in following steps:
  - The question is searched over the web with the help of Google Search API.
  - The result returned from the Google Search API contains links of many websites. This links will be parsed in PRE-PROCESSING model. The content of the websites will be extracted, stripped off their html tags and other irrelevant contents and later merged together. After pre-processing, the model answer is stored in the database for comparison with students answer.

3.2 Algorithm

3.2.1 Cosine Similarity

- We can directly compare two paragraphs and check the similarity between them by counting the maximum number of common words between them.

$$Cos\theta = \frac{\vec{a}\cdot\vec{b}}{\|\vec{a}\|\,\|\vec{b}\|} = \frac{\sum_1^n a_i b_i}{\sqrt{\sum_1^n a_i^2}\,\sqrt{\sum_1^n b_i^2}}$$

where, $\vec{a}\cdot\vec{b} = \sum_1^n a_i b_i = a_1 b_1 + a_2 b_2 + \cdots + a_n b_n$ is the dot product of the two vectors.

Fig-2: Equation for Cosine Similarity

- The problem with this approach is, when the size of paragraphs increases, the number of common words increases. This can affect the accuracy of result.
- To overcome this flaw, we are using Cosine Similar- ity algorithm.
- Cosine Similarity can help determine the similarity between two paragraphs irrespective of their size. In cosine similarity, the paragraphs are represented as vectors in a multi-dimensional space and the angle between this vector is measured. If the angle formed between this vector is higher, then there is lesser the similarity between two paragraphs.
- Cosine Similarity example There are two very short texts to compare this:

1) Julie loves me more than Linda loves me.
2) Jane likes me more than Julie loves me.

We want to know that these texts are similar, purely in terms of word count (and ignoring word order). We begin by making a list of words from both texts:

- me, Julie, loves, Linda, then, more, likes, Jane.

Now, we count how many times each of these words appear in each lesson:

me	2	2
Jane	0	1
Julie	1	1
Linda	1	0
likes	0	1
loves	2	1
more	1	1
then	1	1

We are not interested in words ourselves, though. We are only interested in the two vertical vectors that matter. For example, there are two examples of 'me' in each lesson. We are going to decide how close these two texts are to each other by calculating a function of those two vectors, that is, the cosine of the angle between them. There are two vectors, again:

- a: [2, 0, 1, 1, 0, 2, 1, 1]

- b: [2, 1, 1, 0, 1, 1, 1, 1]

The cosine of the angle between them is about 0.822. These vectors are 8-dimensional. One virtue of using the cosine analogy is clearly that it transforms a question that is beyond a person's ability to imagine.

In this case you can think of it as an angle of 35 degrees which is some 'distance' from zero or complete agreement.

4. Conclusion

A web based Online Assignment System facilitated submission of assignments by students. The system consists of modules where students could upload their answer. Notification are sent to students when new assignment is added and also when lecturer approve or disapprove the assignment.

The assignment answer is corrected using keyword matching, online assignment system also generates question bank from the remaining assignment question, online assignment system is user friendly and also reduce work load of faculty and student, even it minimizes paper work.

Evaluation of web based online submission of assignment was done through surveying among university students and lecturers. Test results of online submission concluded that system is very friendly and would be very helpful to UOM.

REFERENCES

[1] Student Perceptions of the impact of web-based homework on course interaction and learning in Introductory Accounting. Issue in Informa- tion Systems, 9(1), 223 – 232
[2] Black board Learn, Release 9.1- Our Next Generation Teaching and Learning Platform; https://online.uedu/les/blackboardlearn/blackboard.learn.brochure.pdf
[3] Salomon, G. (2000).;). Education and technology in the age of information. HaifaandTel Aviv: Haifa University Publishers and Zmora-Bitan
[4] Our Next Generation Teaching and Learning Platform https://online.fiu.edu/files/blackboardlearn/blackboardlearnbrochure
[5] Get started with WebRTC http://www.netmagazine.com/tutorials/get- startedwebrtc
[6]BillBell.(2018,February)StackOverflow.[Online]. https://stackoverflow.com/questions/1746501/can-someone-give-an-example-of-cosine-similarity-in-a-very-simple-graphical-wa
[7] Rabab A. Wahab, S. A.-A., 2015; Student's Attitude Towards Online Assignment Submission at College of Health Sciences. Bahrain, International Conference On E-Learning