PARIKSHA ONLINE

A Mid Semester Exam Conduction Web Application by using Advance Java and Cloud Services

A Minor Project-II Report (IT-608)

Submitted by:

SHREYA SAHU (0103IT171103)

NAINCY SAHU (0103IT171063)

Group No.- 9

in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

IN

INFORMATION TECHNOLOGY

at

LAKSHMI NARAIN COLLEGE OF TECHNOLOGY

KALCHURI NAGAR, RAISEN ROAD, BHOPAL (INDIA) - 462021

SESSION JAN - JUNE 2020

DECLARATION

We hereby declare that the project entitled “**PARIKSHA ONLINE**” submitted for the B.Tech. (Information Technology) degree is **our** original work and the project has not formed the basis for the award of any other degree, diploma, fellowship or any other similar titles.

Name & Signature of the students with date

Place:

(1)……………………………………

Date:

(2)……………………………………

CERTIFICATE

This is to certify that the project titled “**PARIKSHA ONLINE**” is the bona fide work carried out by **NAINCY SAHU (0103IT171063) and SHREYA SAHU (0103IT171103)** are students of B.Tech. (Information Technology) of Lakshmi Narain College of Technology, Bhopal affiliated to Rajiv Gandhi Proudyogiki Vishwavidyal, Bhopal, Madhya Pradesh (India) during the academic year 2019-20, in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology (**Information Technology**) and that the project has not formed the basis for the award previously of any other degree, diploma, fellowship or any other similar title.

Signature & Seal of HOD,
Information Technology
Lakshmi Narain College of Technology,
Bhopal

Signature of the Guide
with Date

Abstract:

At present there are no such online automated examination system that can fulfill the needs of universities like adding or removing students and faculties, categorizing them on the basis of their semester, year, branch, and college. Creating and scheduling different papers for students from different branches and year and then analyzing the results and performance of the students. Some universities have already tried to conduct online mid semester examination using their own servers but that too failed. This is because when more number of student access the system and system utilizes maximum processor and transactions during concurrent examination timing, their server are sometimes incapable of handling such a huge traffic. This encourages the use of cloud technology in the field of online examinations as well. Examination management application deployed on cloud can be accessed by users through web browser on-per-use-basis. Cloud environment enables maximum utilization of resources during actual online exam time and reduces or scale down the resource utilization when examination is completed. So any number of applicants can appear for exam with ease and cost associated with infrastructure is minimal.

ACKNOWLEDGEMENT

We would like to express our deepest appreciation to all those who provided us the possibility to complete this report. A special gratitude we give to our project coordinator, **Mr. Akshay Jadhav**, whose contribution in stimulating suggestions and encouragement, helped us to coordinate the project especially in writing this report.

Furthermore we would also like to acknowledge with much appreciation the crucial role of our project guide, **Prof.Abhishek Dwivedi** who gave the permission to use all required advice and the necessary help to complete the minor project. Last but not least, many thanks go to the project coordinators **Dr. Manish Shrivastava and Prof. Kamlesh Chandravanshi** whose have invested their full effort in guiding the team in achieving the goal. We appreciate the guidance given by other supervisor as well as the panels especially in our project presentation that has improved our presentation skills thanks to their comment and advices.

At last, we would thank to all our families, friends and our colleagues at LNCT bhopal who have taken sincere pain to boast our morale in the moments of despair and acted as a source of inspiration for completing the project.

Table of Content

CHAPTER 1

INTRODUCTION

1.1 Problem Definition

The traditional System of exam have many drawbacks such as time consuming, Difficulty of analyzing the test manually, More observers are required to take exam of many students, Results are not accurate since calculations is done manually, The chance of losing exam's result is higher in current systems, Checking of result is time consuming since it done manually, Limitation of no of student can give examination at a time.

The on-line exam system is to take online exam in an efficient manner and no time wasting for checking the paper. The main objective of this is to efficiently evaluate the candidate thoroughly through a fully automated system that not only saves lot of time but also gives fast results. But existing online system fails when large number of students from different branches, give exam simultaneously, it is very difficult to deal with such a huge database storing results of students from various branches and different affiliated colleges and then categorize them.

At present there are no such online automated examination system that can fulfill the needs of university like adding or removing students and faculties, categorizing them on the basis of their semester, year, branch, and college. Creating and scheduling different papers for students from different branches and year. And then analyzing the results and performance of the students by categorizing them on the basis of their semester, year, branch and college.

Some universities have already tried to conduct online mid sem examination using their own servers but that too failed. This is because when more number of student access the system and system utilizes maximum processor and transactions during concurrent examination timing, their server are sometimes incapable of handling such a huge traffic.

1.2 Project Overview

Pariksha Online is an online mid sem examination website for universities to conduct the online exam of large number of students simultaneously. This is possible by hosting our website on the cloud and using its services like auto scaling and auto load balancing, it increases the response time of the website. So any number of students can appear in the exam simultaneously with best response time. Thus it helps to overcome the existing error in the manual system.

Examination management application deployed on cloud can be accessed by users through web browser on-per-use-basis. Thus it removes the headache of server management and the university can easily conduct the examination of any number of students. The end–users in an examination usually consist of students, academic staff and administrative staff.

In case of particular entrance exam which is to be conducted online for institute and around 50000 candidates are going to appear simultaneously, cloud computing can provide you option to utilize maximum resources only during actual examination process and ramp down utilization of hardware and resources after examination is completed. If online examination is going to be conducted for 3 hours then online examination system can be deployed in cloud environment like AWS with auto scale facility for the virtual machines. Auto scale facility of cloud environment enables to increase virtual machines automatically when more users are trying to access the website.

In this way cloud environment enables maximum utilization of resources during actual online exam time and reduces or scale down the resource utilization when examination is completed. So any number of applicants can appear for exam with ease and cost associated with infrastructure is minimal. Online examination would also provide you advantage of accurate results and analytics and other administration paperwork of manual examination management gets reduced drastically.

Role of students include: to appear in online examination, attend online test, submit answers, know status of result, view declared results. The admin role and responsibility include: to enroll and verify student detail, add or remove a student. The faculty role and responsibility include: online setting of subject wise, college wise and semester wise, to view the question, Setting of question papers using already stored questions, removing a question paper that is no longer needed, examination scheduling, and to view and analyze the result of students by categorizing them on the basis of their semester, year, branch, and college.

1.3 Objective

- Online examination will reduce the hectic job of assessing the answers given by the candidates.
- Being an integrated Online Examination System it will reduce paperwork.

- ➢ Categorizing the students on the basis of their semester, year, branch, and college.
- ➢ Creating and scheduling the different papers for the students of different categories.
- ➢ Responses by the candidates will be checked automatically and instantly.
- ➢ Can generate the analyzed reports which summarize the overall performance of the students .

1.4 Scope

This project would be very useful for educational institute especially universities where regular evaluation of students' is required. The list of modules are the following :

Sr. No.	**Categories**	**Description**
1	Admin	Super user, add faculty and students.
2	Student	Student can give the exam.
3	Faculty	Faculty can set and schedule the paper.

Scope of this Web – Application is for conducting online 'objective' type examination and providing immediate results. The system should satisfy the following requirements :

- ❖ Administrative Aspect :
  - ➢ Adding or expelling faculty.
  - ➢ Adding or expelling students.
  - ➢ Providing them with UserId and Password.
- ❖ Faculty Aspect :
  - ➢ Create/Edit/Delete questions.
  - ➢ Create/Edit/Delete paper.
  - ➢ Scheduling the paper.
  - ➢ Viewing the results.
  - ➢ Analyzing the performance.
- ❖ Student Aspect :
  - ➢ Appearing for the exam.
  - ➢ Viewing the result of exam.

Chapter 2

Literature Survey

2.1 Existing System

The traditional System of exam that is using pen and paper which have many drawbacks such as time consuming, difficulty of analyzing the test manually, more observers are required to take exam of many students, results are not accurate since calculations is done manually, checking of result is time consuming since it done manually, limitation of no of student can give examination at a time.

The on-line exam system is to take online exam in an efficient manner and no time wasting for checking the paper. The main objective of this is to efficiently evaluate the candidate thoroughly through a fully automated system that not only saves lot of time but also gives fast results. These all above mentioned features are provided in some websites like onlineexambuilder.com, classmarker.com, etc. But this also fails when large number of students can give exam simultaneously.

The existing online exam system are not capable of fulfilling the needs of university like adding or removing students and faculties, categorizing them on the basis of their semester, year, branch, and college. Creating and scheduling different papers for students from different branches and year. And then analyzing the results and performance of the students by categorizing them on the basis of their semester, year, branch and college.

2.2 Proposed System

In this semester we will split the faculty module into faculty and admin module and will also complete the student module.

- **Admin Module**: The admin role and responsibility includes add or remove a student, verify students detail, add or remove a faculty. And then providing them userID and password for login.
- **Faculty Module**: The faculty role and responsibility includes online setting of subject wise, college wise and semester wise, to view the question, Setting of question papers using already stored questions, removing a question paper that is no longer needed, examination scheduling, and to view and analyse the result of students by categorizing them on the basis of their semester, year, branch, and college.
- **Student Module**: Role of students includes appearing in online examination, attend online test, submit answers, know status of result, view declared results.

We will also include analysis functionality in faculty module in which the faculty can view the analysis of the student of particular branch, year, college and also of particular subject. The maximum mark of exam will be 20. Analysis will be done on the basis of the marks of the students in the particular subject. This will show the percentage of students who score in the below mentioned criteria:

i. A --- 20 – 17
ii. B --- 16 – 13
iii. C --- 12 – 9
iv. D --- 8 – 5
v. E --- 4 – 0

2.3 Feasibility Study:

Economic feasibility :

Economic analysis is most frequently used for evaluation of the effectiveness of the system. More commonly known as cost/benefit analysis the procedure is to determine the benefit and saving that are expected from a system and compare them with costs, decisions is made to design and implement the system. In the system, the organization is most satisfied by economic feasibility. Because, if the organization implements this system, it need not require any additional hardware resources as well as it will be saving lot of time.

Technical feasibility:

The system offers greater levels of user friendliness combined with greater processing speed. Therefore, the cost of maintenance can be reduced.

Behavioral feasibility :

People are inherently resistant to change and computer has been known to facilitate changes. An estimate should be made of how strong the user is likely to move towards the development of computerized system. These are various levels of users in order to ensure proper authentication and authorization and security of sensitive data of the organization.

Chapter 3

System Analysis and Design

3.1 Requirement Specification

Hardware Specifications:

- Processor (CPU) with 2 gigahertz (GHz) frequency or above
- A minimum of 4 GB of RAM
- Monitor Resolution 1024 X 768 or higher
- A minimum of 40 GB of available space on the hard disk
- Internet Connection Broadband (high-speed) Internet connection with a speed of 4 Mbps or higher
- Keyboard and a Microsoft Mouse or some other compatible pointing device
- Chrome* 36+ /
- Edge* 20+ /
- Mozilla Firefox 31+ /
- Internet Explorer 11+ (Windows only)

Browser Configuration:

Your browser must be configured as follows:

- JavaScript must be enabled
- Cookies must be enabled.
- Pop-up windows must be enabled.

Software Specifications:

- Operating System : Windows 7+
- Apache Tomcat Server / Glassfish
- MySql Server Database
- Java version 8 +

3.2.1 Use Case Diagram

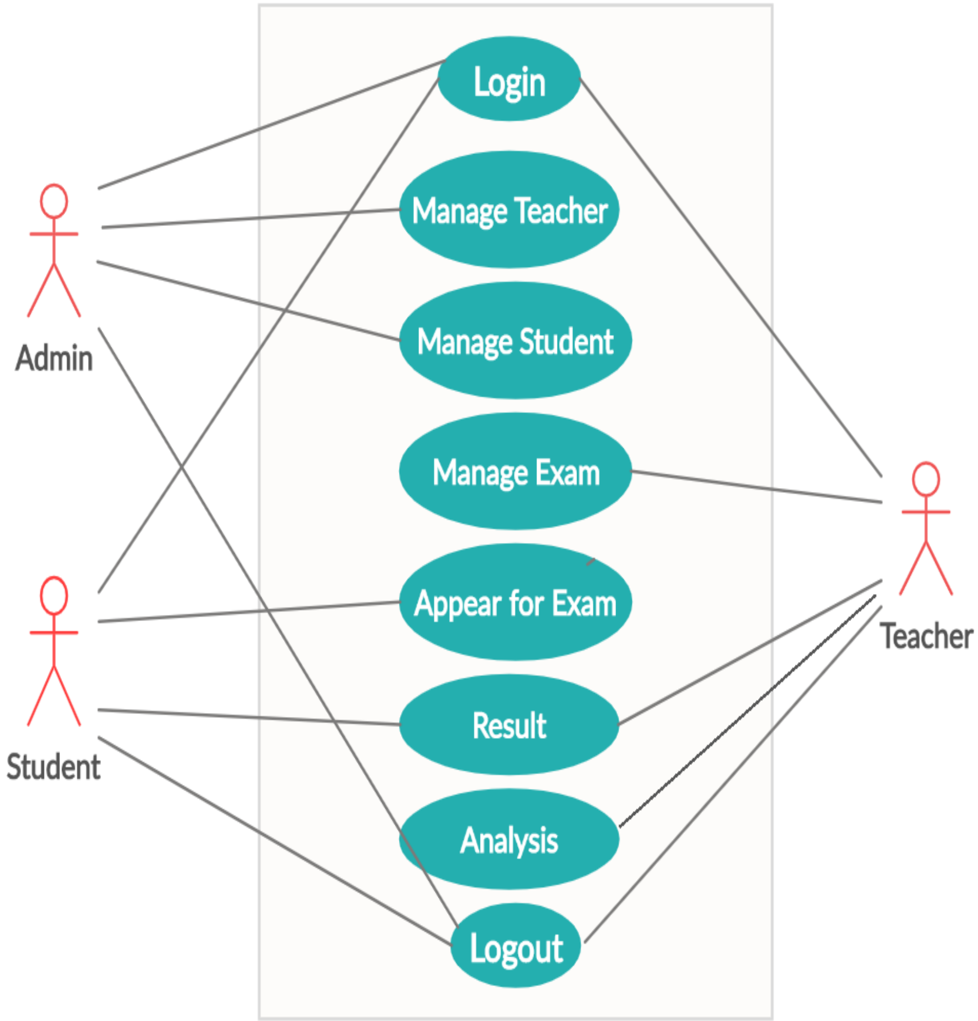


Fig. 1: Use Case Diagram

In the above use case diagram, there are three actors named student, teacher and admin. There are a total of eight use cases that represent the specific functionality of Pariksha Online. Each actor interacts with a particular use case. A student actor can login, appear for the exam, view the results and logout. This actor can perform only these interactions with the system even though other use cases are remaining in the system.

The second actor named teacher can interact with the functionalities - login and managing the exam paper of the system. This actor can also view the results and analyzed report of the performance of students. The third actor named admin can interact with the functionalities – login, managing the teacher data and managing the student data. These interactions of student, teacher and admin actor together sums up the entire website Pariksha Online.

3.2.2 E-R Diagram

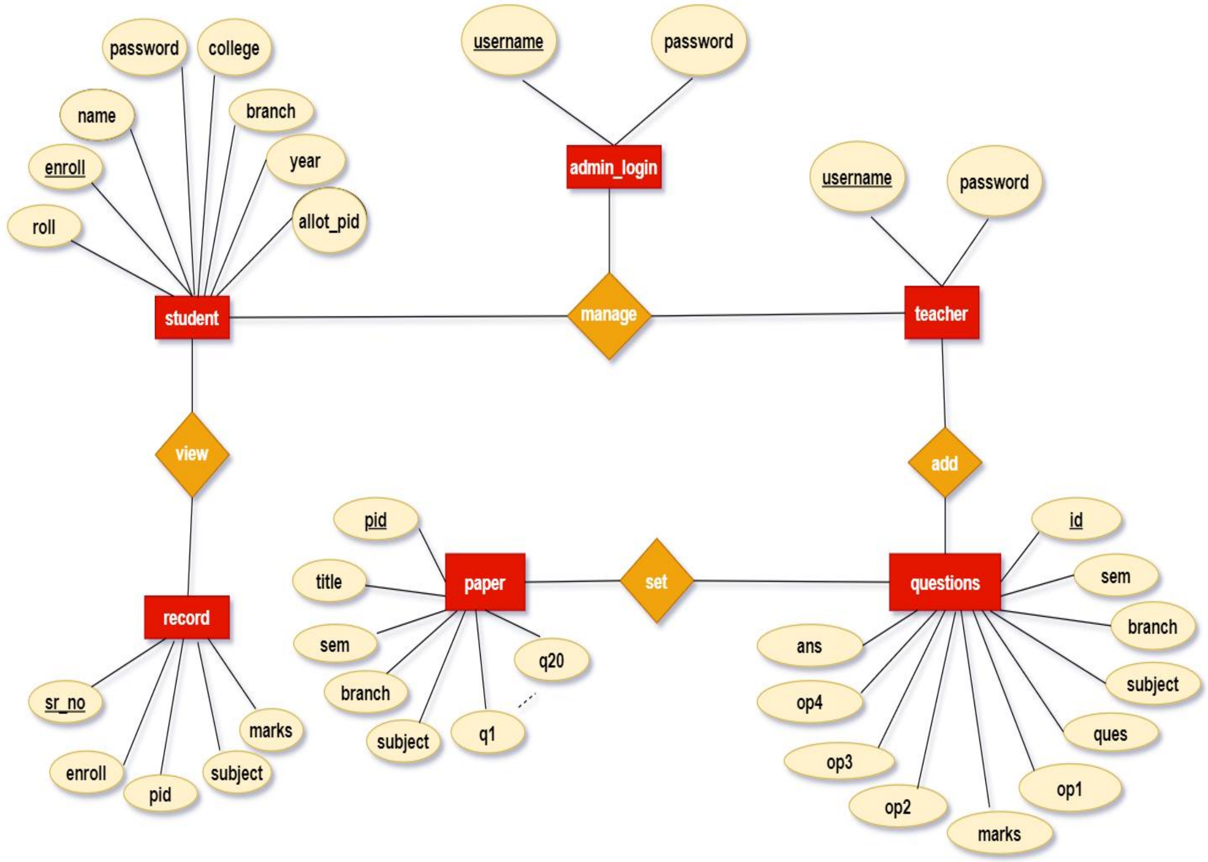


Fig. 2: E R Diagram

This ER diagram represents the model of Pariksha Online. The ER diagram of Pariksha Online shows all the visual instrument of database tables and the relations between student, teacher, questions, etc.It uses structured data and to define the relationships between structure data groups of Pariksha Online functionalities. The main entities and their attributes of the system are as follows:

- **admin_login entity:** Attributes of admin_login are username and password.
- **teacher entity:** Attributes of teacher are username and password.
- **student entity:** Attributes of student are roll, enroll, name, password, college , branch, year, allot_id.
- **questions entity:** Attributes of questions are id, sem, branch, subject, ques, op1, op2, op3, op4, marks, ans.
- **paper entity:** Attributes of paper are pid, title, sem, branch, subject, q1, …, q20.
- **record entity:** Attributes of record are sr_no, enroll, pid, subject, marks.

3.2.3 Activity Flow Diagram

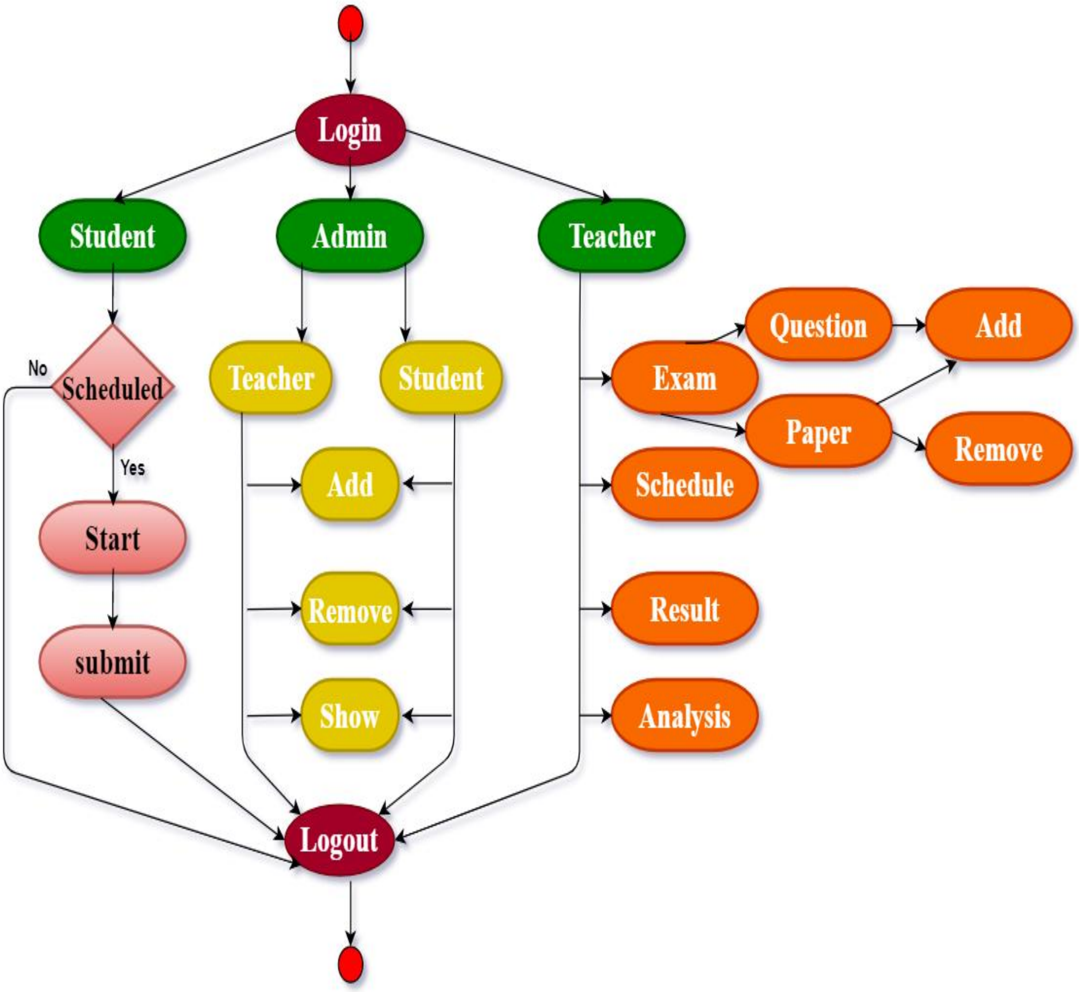


Fig. 3: Activity Flow Diagram

This is the Activity Flow Diagram of Pariksha Online, where we have elaborated the mid sem conduction process for a university. It is modeled for three types of clients/users i.e. Student, Admin, and Teacher. After a student has successfully login using assigned username and password, if exam is already scheduled by teacher then student can proceed to attempt the quiz and view the result, else he should logout directly. Similarly after successful login of admin he can manage both teacher and student performing all basic operations like add, remove and show, and then logout. Same with teacher, after successful login teacher can add questions to question bank, schedule a paper for a particular class of students depending on branch, year, semester. Techer can also view the result of students along with analyzing their performance.

CHAPTER 4

Proposed Work

4.1Module description

Admin Module:

The admin role and responsibility includes add or remove a student, verify students detail, add or remove a faculty. And then providing them userID and password for login.

Faculty Module:

The faculty role and responsibility includes online setting of subject wise, college wise and semester wise, to view the question, Setting of question papers using already stored questions, removing a question paper that is no longer needed, examination scheduling, and to view and analyse the result of students by categorizing them on the basis of their semester, year, branch, and college.

We will also include analysis functionality in faculty module in which the faculty can view the analysis of the student of particular branch, year and college and also of particular subject. The maximum mark of exam will be 20. Analysis will be done on the basis of the marks of the students in the particular subject. This will show the percentage of students who score in the below mentioned criteria:

i.	A	---	20 – 17
ii.	B	---	16 – 13
iii.	C	---	12 – 9
iv.	D	---	8 – 5
v.	E	---	4 – 0

Student Module:

Role of students includes appearing in online examination, attend online test, submit answers, know status of result, view declared results.

4.2 Database Description

The Database diagram of Pariksha Online shows all the visual instrument of database tables and the relations between student, teacher, questions, etc.

It uses structured data and to define the relationships between structure data groups of Pariksha Online functionalities. The main entities and their attributes of the system are as follows:

- **admin_login entity:** Attributes of admin_login are username and password.
- **teacher entity:** Attributes of teacher are username and password.

- **student entity:** Attributes of student are roll, enroll, name, password, college , branch, year, allot_id.
- **questions entity:** Attributes of questions are id, sem, branch, subject, ques, op1, op2, op3, op4, marks, ans.
- **paper entity:** Attributes of paper are pid, title, sem, branch, subject, q1, …, q20.
- **record entity:** Attributes of record are sr_no, enroll, pid, subject, marks.

4.2.1 Table admin_login

Field Name	Data Type	Constraints / Description
username	varchar(30)	Primary Key
password	varchar(15)	

4.2.2 Table teacher

Field Name	Data Type	Constraints / Description
username	varchar(20)	Primary Key
password	varchar(20)	

4.2.3 Table student

Field Name	Data Type	Constraints / Description
roll	int(11)	
enroll	varchar(12)	Primary Key
name	varchar(45)	
password	varchar(15)	
college	varchar(45)	
branch	varchar(5)	
year	int(11)	
allot_pid	int(11)	

4.2.4 Table questions

Field Name	Data Type	Constraints / Description
id	int(11)	Auto increment, Primary Key
sem	int(11)	

branch	varchar(5)	
subject	varchar(45)	
ques	varchar(250)	
op1	varchar(100)	
op2	varchar(100)	
op3	varchar(100)	
op4	varchar(100)	
ans	int(11)	
marks	int(11)	

4.2.5 Table paper

Field Name	**Data Type**	**Constraints / Description**
pid	int(11)	Auto increment, Primary Key
title	varchar(50)	
sem	int(11)	
branch	varchar(5)	
subject	varchar(30)	
q1…q20	int(11)	

4.2.6 Table record

Field Name	**Data Type**	**Constraints / Description**
sr_no	int(11)	Auto increment, Primary Key
enroll	varchar(12)	
pid	int(11)	
subject	varchar(10)	
marks	int(11)	

Chapter 5

Coding Standards

Some of the coding standards are given below:

1. Limited use of global:

These rules tell about which types of data that can be declared global and the data that can't be.

2. Standard headers for different modules:

For better understanding and maintenance of the code, the header of different modules should follow some standard format and information. The header format must contain below things that is being used in various companies:

- Name of the module
- Date of module creation
- Author of the module
- Modification history
- Synopsis of the module about what the module does
- Different functions supported in the module along with their input output parameters
- Global variables accessed or modified by the module.
- Naming conventions for local variables, global variables, constants and functions:

Some of the naming conventions are given below:

- Meaningful and understandable variables name helps anyone to understand the reason of using it.
- Local variables should be named using camel case lettering starting with small letter (e.g. localData) whereas Global variables names should start with a capital letter (e.g. GlobalData). Constant names should be formed using capital letters only (e.g. CONSDATA).
- It is better to avoid the use of digits in variable names.
- The names of the function should be written in camel case starting with small letters.
- The name of the function must describe the reason of using the function clearly and briefly.

Indentation:

Proper indentation is very important to increase the readability of the code. For making the code readable, programmers should use White spaces properly. Some of the spacing conventions are given below:

- There must be a space after giving a comma between two function arguments.
- Each nested block should be properly indented and spaced.
- Proper Indentation should be there at the beginning and at the end of each block in the program.
- All braces should start from a new line and the code following the end of braces also start from a new line.
- Error return values and exception handling conventions:

All functions that encountering an error condition should either return a 0 or 1 for simplifying the debugging.

On the other hand, coding guidelines give some general suggestions regarding the coding style that to be followed for the betterment of understandability and readability of the code. Some of the coding guidelines are given below :

- Avoid using a coding style that is too difficult to understand:

Code should be easily understandable. The complex code makes maintenance and debugging difficult and expensive.

- Avoid using an identifier for multiple purposes:

Each variable should be given a descriptive and meaningful name indicating the reason behind using it. This is not possible if an identifier is used for multiple purposes and thus it can lead to confusion to the reader. Moreover, it leads to more difficulty during future enhancements.

- Code should be well documented:

The code should be properly commented for understanding easily. Comments regarding the statements increase the understandability of the code.

- Length of functions should not be very large:

Lengthy functions are very difficult to understand. That's why functions should be small enough to carry out small work and lengthy functions should be broken into small ones for completing small tasks.

- Try not to use GOTO statement:

GOTO statement makes the program unstructured, thus it reduces the understandability of the program and also debugging becomes difficult.

5.1 Algorithms and Pseudo Code

(QuestionsDao.java)

```
package com.javadao;

import com.javabean.QuestionsBean;

import java.util.*;

import com.javabean.loginbean;

import java.sql.*;


public class QuestionsDao {

   public static Connection getConnection() throws ClassNotFoundException,
SQLException {

      Connection con = null;

      try {

         Class.forName("com.mysql.cj.jdbc.Driver");

         con =
DriverManager.getConnection("jdbc:mysql://localhost/midsem?useSSL=false",
"root", "shreya*123");

         if (con != null) {

            System.out.println("Connection done");

         }

      } catch (Exception e) {

         System.out.println(e);

      }

      return con;

   }

   public static int insert(QuestionsBean q) {

      int status = 0;

      try {
         Connection con = getConnection();
         PreparedStatement pst = con.prepareStatement("INSERT INTO questions
("+"sem,branch,subject,ques,op1,op2,op3,op4,"+"ans,"+"marks)
VALUES(?,?,?,?,?,?,?,?,?,?)");
         pst.setInt(1, q.getSem());
         pst.setString(2, q.getBranch());
         pst.setString(3, q.getSubject());
         pst.setString(4, q.getQues());
         pst.setString(5, q.getOp1());
         pst.setString(6, q.getOp2());
         pst.setString(7, q.getOp3());
         pst.setString(8, q.getOp4());
         pst.setInt(9, q.getAns());
         pst.setInt(10, q.getMarks());
         status = pst.executeUpdate();
         System.out.println("Record succesfully Inserted !!");
      } catch (Exception e) {
         System.out.println(e);
      }
      return status;
```

```
    }


    public static int edit(QuestionsBean q) throws ClassNotFoundException {
        int status = 0;
        try {
            Connection con = getConnection();
            PreparedStatement pst = con.prepareStatement("update set ques=?, op1=?,
op2=?, op3=?, op4=?, ans=?, marks=?, where id=?");
            pst.setString(1, q.getQues());
            pst.setString(2, q.getOp1());
            pst.setString(3, q.getOp2());
            pst.setString(4, q.getOp3());
            pst.setString(5, q.getOp4());
            pst.setInt(6, q.getAns());
            pst.setInt(7, q.getMarks());
            pst.setInt(8, q.getId());
            status = pst.executeUpdate();
        } catch (SQLException e) {
            System.out.println(e);
        }
        return status;
    }
  public static int delete(QuestionsBean q) throws ClassNotFoundException {
        int status = 0;
        try {
            Connection con = getConnection();
            PreparedStatement pst = con.prepareStatement("delete from questions where
id=?");
            pst.setInt(1, q.getId());
            status = pst.executeUpdate();
        } catch (SQLException e) {
            System.out.println(e);
        }
        return status;
```

```
}


public static List<QuestionsBean> getAllRecords() {
    List<QuestionsBean> list = new ArrayList<QuestionsBean>();
    try {
        Connection con = getConnection();
        PreparedStatement ps = con.prepareStatement("select * from questions");
        ResultSet rs = ps.executeQuery();
        while (rs.next()) {
            QuestionsBean q = new QuestionsBean();
            q.setId(rs.getInt("id"));
            q.setSem(rs.getInt("sem"));
            q.setBranch(rs.getString("branch"));
            q.setSubject(rs.getString("subject"));
            q.setQues(rs.getString("ques"));
```

```
            q.setOp1(rs.getString("op1"));
            q.setOp2(rs.getString("op2"));
            q.setOp3(rs.getString("op3"));
            q.setOp4(rs.getString("op4"));
            q.setAns(rs.getInt("ans"));
            q.setMarks(rs.getInt("marks"));
            list.add(q);
        }
    } catch (Exception e) {
        System.out.println(e);
    }
```

```
        return list;
    }


   public static List<QuestionsBean> SearchRecords(QuestionsBean q1) {
        List<QuestionsBean> list = new ArrayList<QuestionsBean>();
        try {
            Connection con = getConnection();
            PreparedStatement pst = con.prepareStatement("select * from questions where
sem=? and branch=? and subject=?");
            pst.setInt(1, q1.getSem());
            pst.setString(2, q1.getBranch());
            pst.setString(3, q1.getSubject());
            ResultSet rs = pst.executeQuery();
            while (rs.next()) {
                QuestionsBean q = new QuestionsBean();
                q.setId(rs.getInt("id"));
                q.setSem(rs.getInt("sem"));
                q.setBranch(rs.getString("branch"));
                q.setSubject(rs.getString("subject"));
                q.setQues(rs.getString("ques"));
                q.setOp1(rs.getString("op1"));
                q.setOp2(rs.getString("op2"));
                q.setOp3(rs.getString("op3"));
                q.setOp4(rs.getString("op4"));
                q.setAns(rs.getInt("ans"));
                q.setMarks(rs.getInt("marks"));
                list.add(q);
            }
        } catch (Exception e) {
            System.out.println(e);
        }
        return list;
    }
}
```

Chapter 6

Testing Process

6.1 Testing Methodology

1. Functionality Testing:

Test Forms are working as expected. This will include-

- Scripting checks on the form are working as expected. For example- if a user does not fill a mandatory field in a form an error message is shown.
- Check default values are being populated
- Once submitted, the data in the forms is submitted to a live database or is linked to a working email address
- Forms are optimally formatted for better readability Test HTML and CSS to ensure that search engines can crawl your site easily. This will include
- Checking for Syntax Errors
- Readable Color Schemas
- Standard Compliance. Ensure standards such W3C, OASIS, IETF, ISO, ECMA, or WS-I are followed.

2. Usability testing:

Usability Testing has now become a vital part of any web based project. It can be carried out by testers like you or a small focus group similar to the target audience of the web application.

Test the site Navigation**:**

- Menus, buttons or Links to different pages on your site should be easily visible and consistent on all webpage.

Test the Content**:**

- Content should be legible with no spelling or grammatical errors.
- Images if present should contain an "alt" text

3. Interface Testing:

Three areas to be tested here are - Application, Web and Database Server

- Application**:** Test requests are sent correctly to the Database and output at the client side is displayed correctly. Errors if any must be caught by the application and must be only shown to the administrator and not the end user.
- Web Server: Test Web server is handling all application requests without any service denial.
- Database Server**:** Make sure queries sent to the database give expected results.

- Test system response when connection between the three layers (Application, Web and Database) cannot be established and appropriate message is shown to the end user.

4. Database Testing:

Database is one critical component of your web application and stress must be laid to test it thoroughly. Testing activities will include-

- Test if any errors are shown while executing queries
- Data Integrity is maintained while creating, updating or deleting data in database.
- Check response time of queries and fine tune them if necessary.
- Test data retrieved from your database is shown accurately in your web application

5. Compatibility testing:

Compatibility tests ensure that your web application displays correctly across different devices. This would include-

Browser Compatibility Test**:** Same website in different browsers will display differently. You need to test if your web application is being displayed correctly across browsers, JavaScript, AJAX and authentication is working fine. You may also check for Mobile Browser Compatibility.

The rendering of web elements like buttons, text fields etc. changes with change in Operating System**.** Make sure your website works fine for various combinations of Operating systems such as Windows, Linux, Mac and Browsers such as Firefox, Internet Explorer, Safari etc.

6. Performance Testing:

This will ensure your site works under all loads. Software testing activities will include but not limited to -

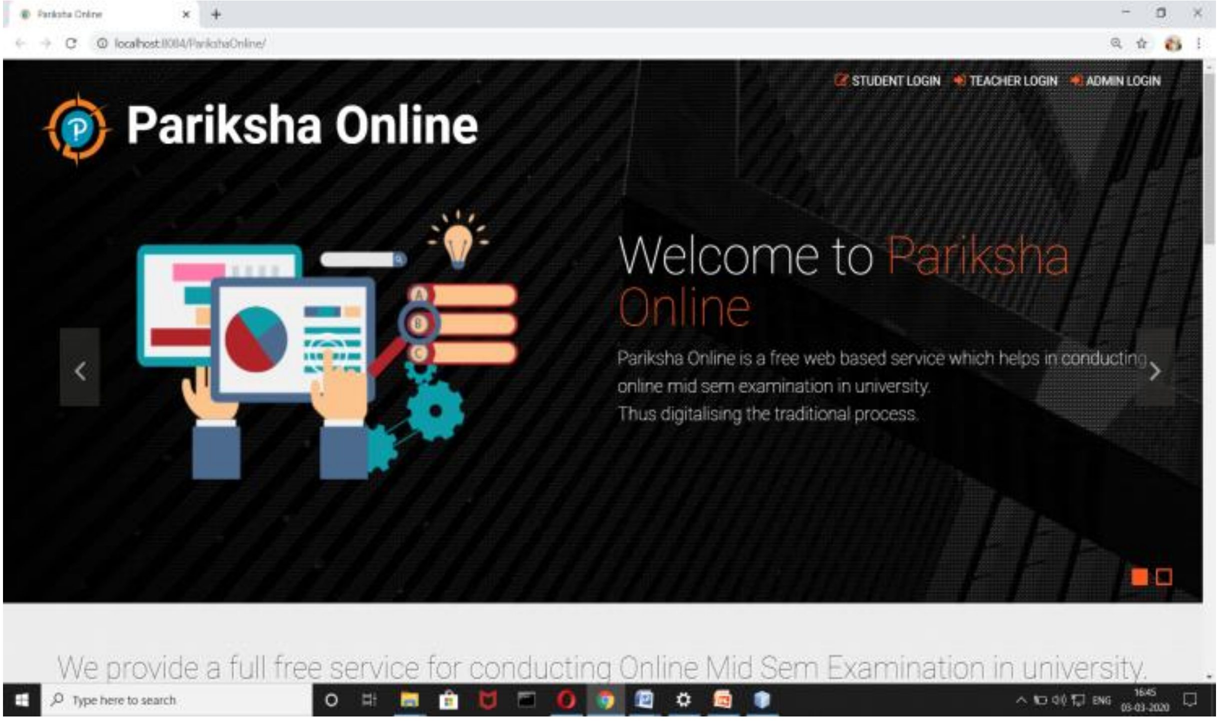
- Website application response times at different connection speeds
- Load test your web application to determine its behavior under normal and peak loads
- Test if a crash occurs due to peak load, how does the site recover from such an event

6.2 Test Cases and Test Steps

Chapter 7

Results

Results:



Welcome to Pariksha Online

Pariksha Online is a free web based service which helps in conducting online mid sem examination in university. Thus digitalising the traditional process.

We provide a full free service for conducting Online Mid Sem Examination in university.

Fig.4: Welcome Page

This is the first page which contains various login options for three actors: admin, student, teacher.

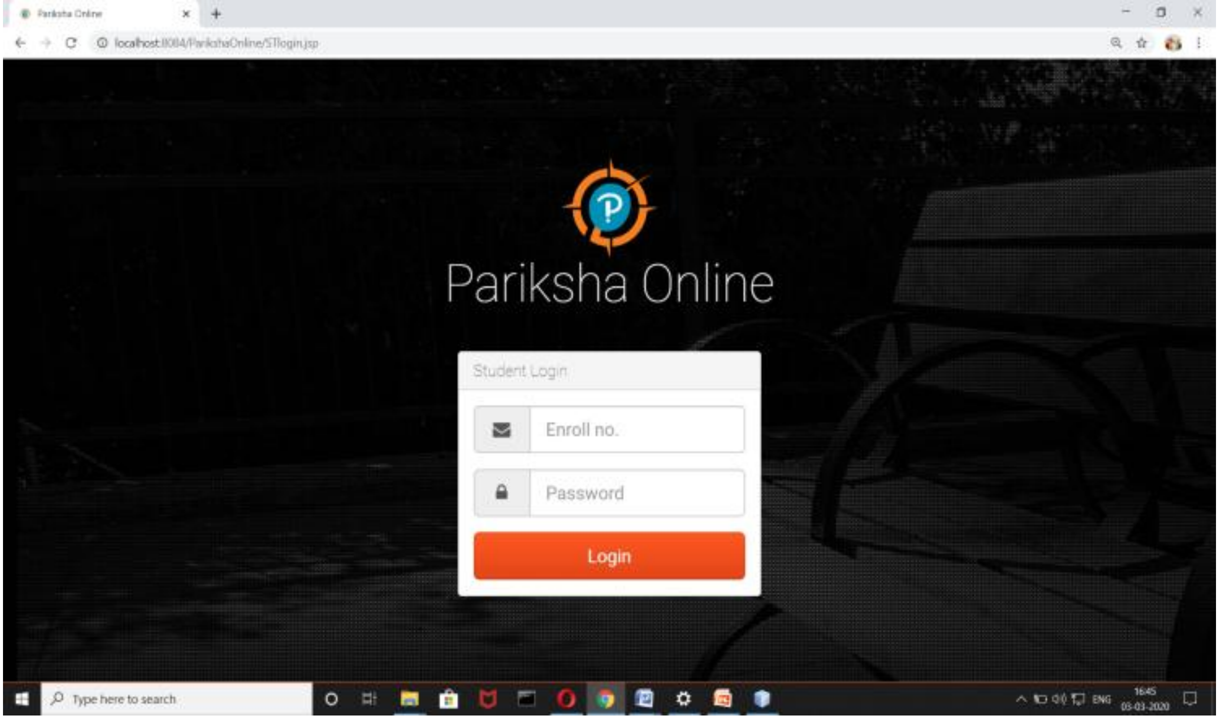
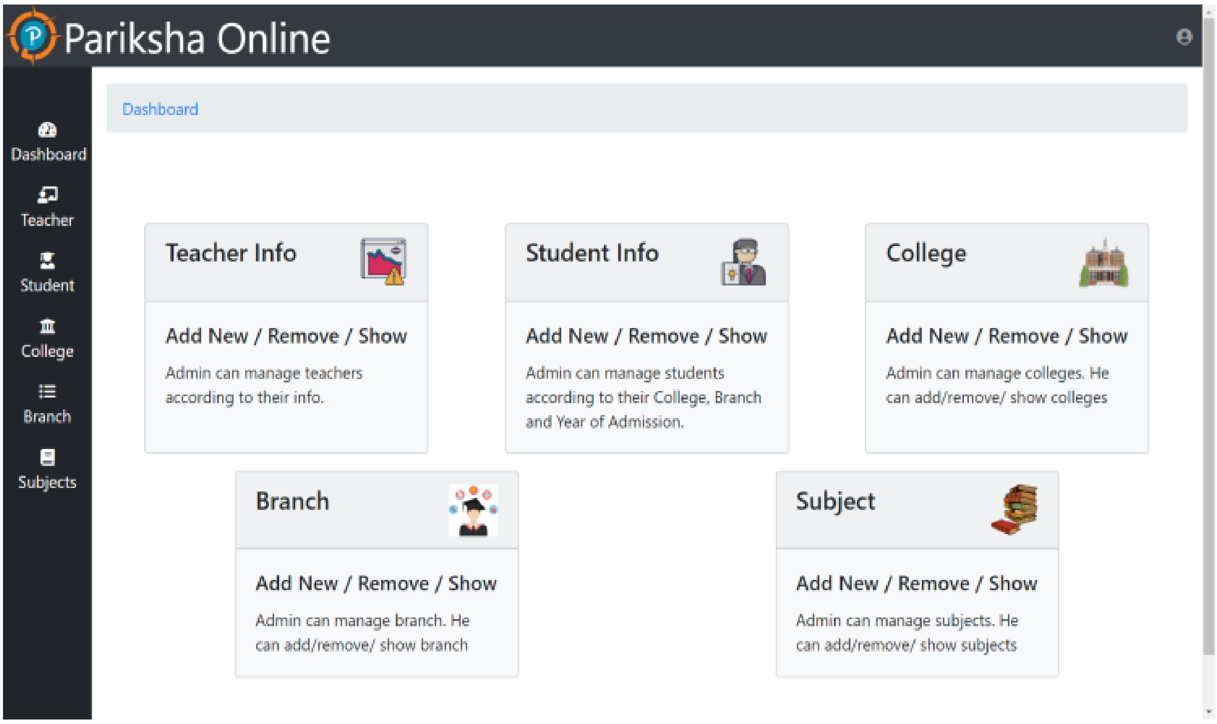


Fig.5: Login Page

This is the login page using correct userId and password an actor can login.



Teacher Info

Add New / Remove / Show

Admin can manage teachers according to their info.

Student Info

Add New / Remove / Show

Admin can manage students according to their College, Branch and Year of Admission.

College

Add New / Remove / Show

Admin can manage colleges. He can add/remove/ show colleges

Branch

Add New / Remove / Show

Admin can manage branch. He can add/remove/ show branch

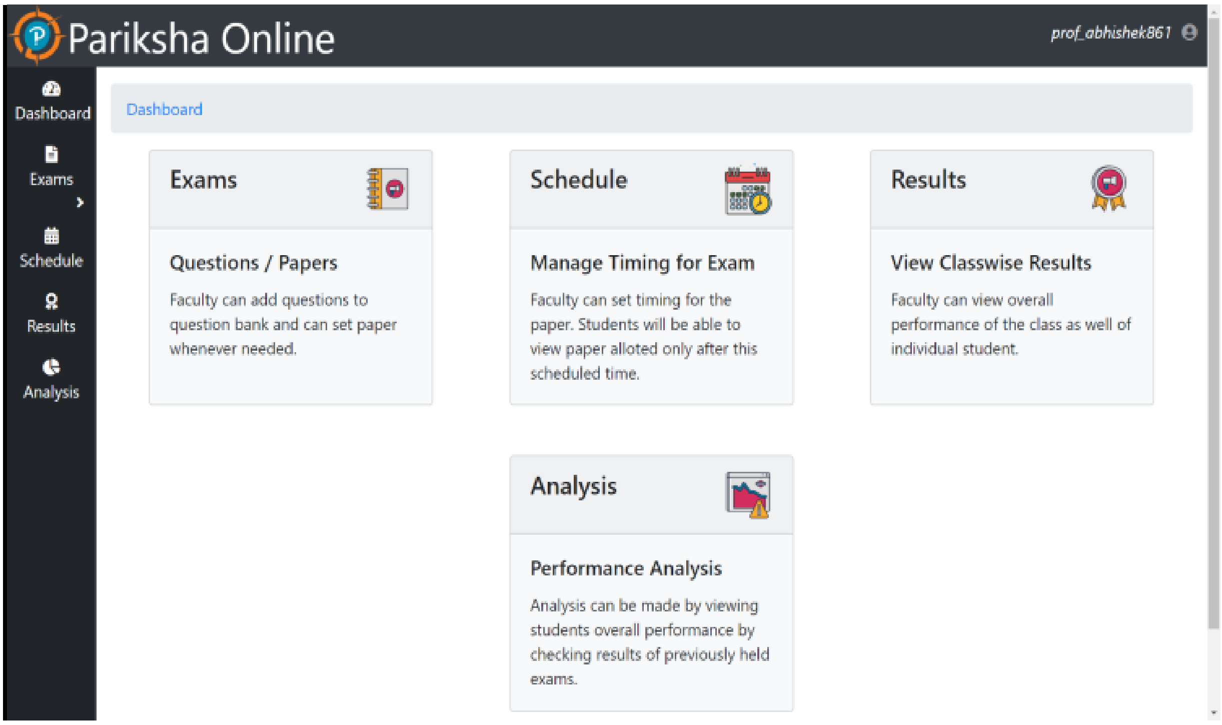
Subject

Add New / Remove / Show

Admin can manage subjects. He can add/remove/ show subjects

Fig.6: Dashboard (Admin module)

Dashboard shows all the functions that include in the admin module



Exams

Questions / Papers

Faculty can add questions to question bank and can set paper whenever needed.

Schedule

Manage Timing for Exam

Faculty can set timing for the paper. Students will be able to view paper alloted only after this scheduled time.

Results

View Classwise Results

Faculty can view overall performance of the class as well of individual student.

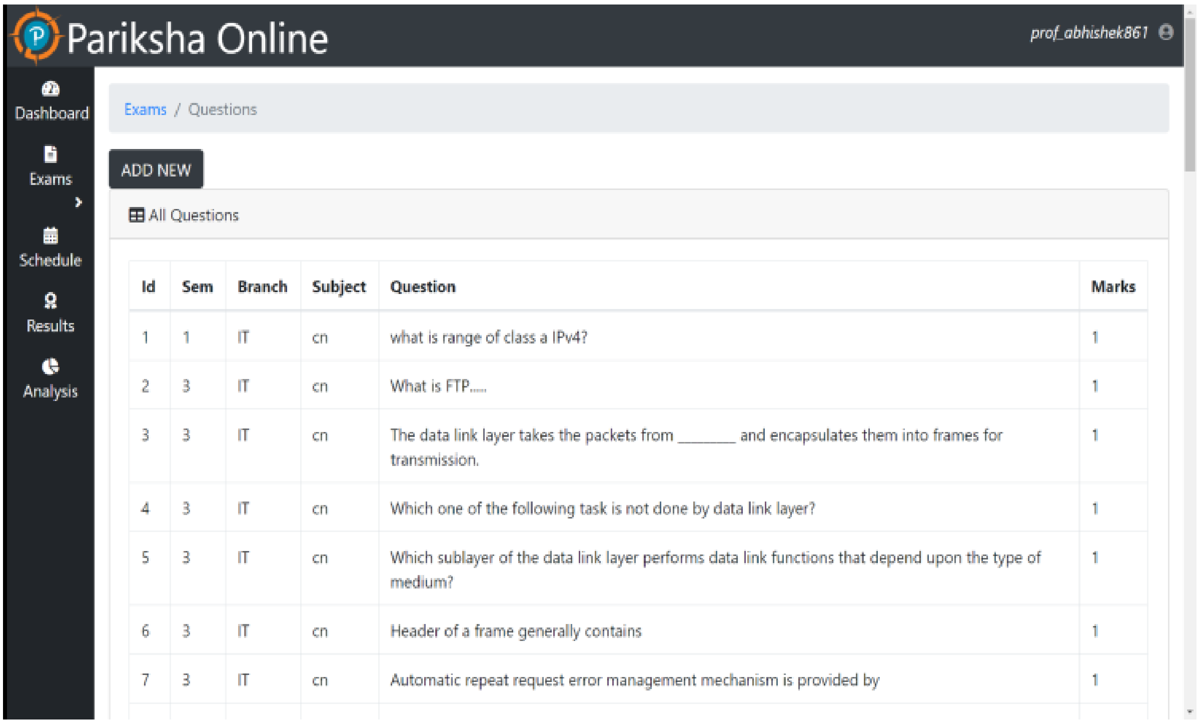
Analysis

Performance Analysis

Analysis can be made by viewing students overall performance by checking results of previously held exams.

Fig.7: Dashboard (Faculty module)

Dashboard shows all the functions that include in the faculty module.



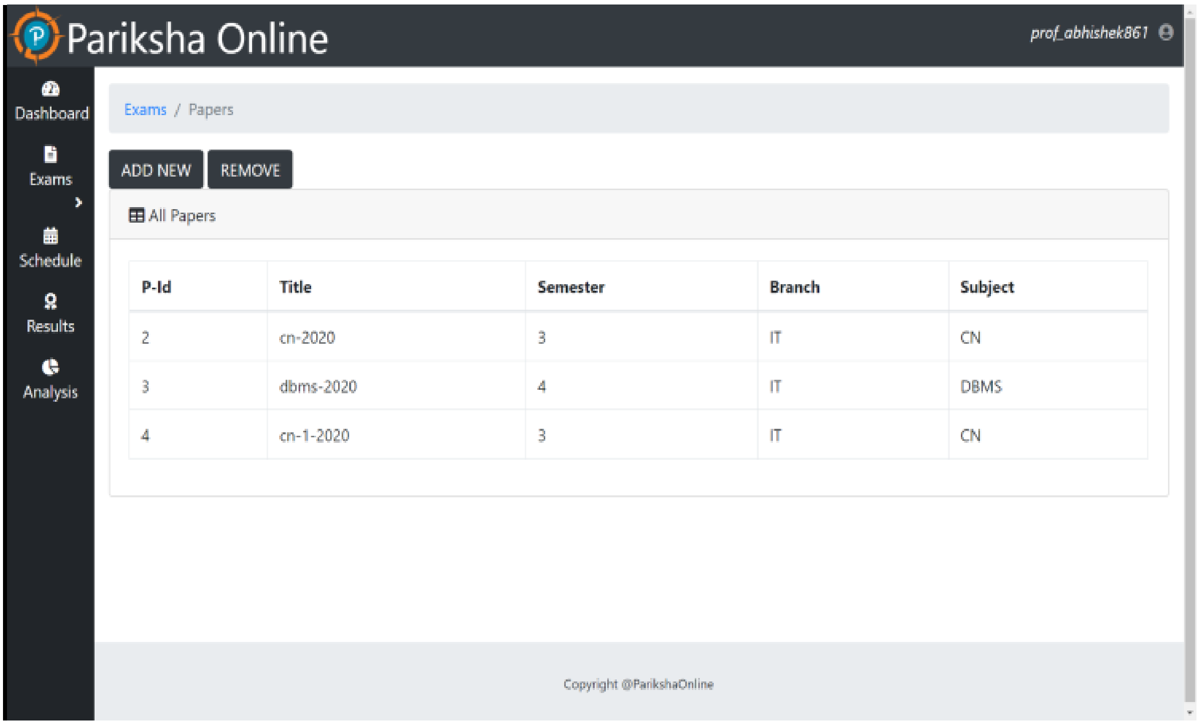
Exams / Questions

All Questions

Id	Sem	Branch	Subject	Question	Marks
1	1	IT	cn	what is range of class a IPv4?	1
2	3	IT	cn	What is FTP.....	1
3	3	IT	cn	The data link layer takes the packets from _______ and encapsulates them into frames for transmission.	1
4	3	IT	cn	Which one of the following task is not done by data link layer?	1
5	3	IT	cn	Which sublayer of the data link layer performs data link functions that depend upon the type of medium?	1
6	3	IT	cn	Header of a frame generally contains	1
7	3	IT	cn	Automatic repeat request error management mechanism is provided by	1

Fig.8: Question page (Faculty module)

A teacher can add a new question in the question bank and can also see the list of all the questions that are stored in the question bank.



Exams / Papers

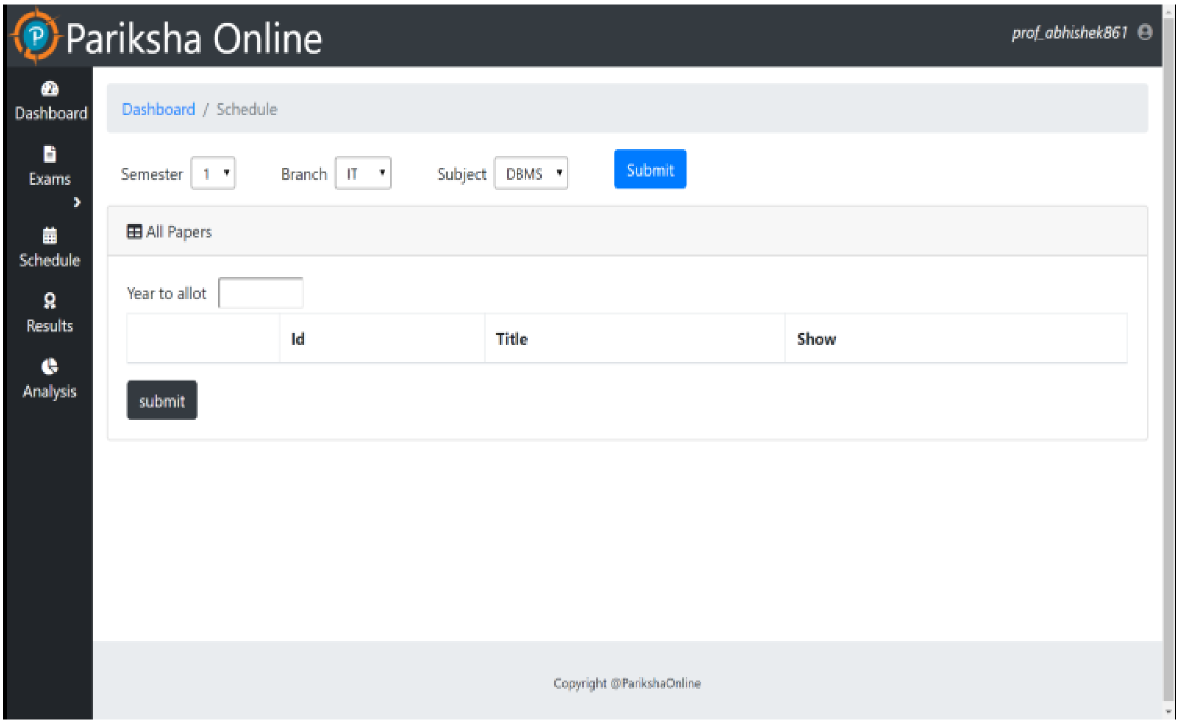
All Papers

P-Id	Title	Semester	Branch	Subject
2	cn-2020	3	IT	CN
3	dbms-2020	4	IT	DBMS
4	cn-1-2020	3	IT	CN

Copyright @ParikshaOnline

Fig.9: Paper page (Faculty module)

A teacher can view the list of all the papers that are already set and can also set a new paper of a particular subject and for a particular semester using the questions from the question bank.



Semester 1 Branch IT Subject DBMS Submit

All Papers

Year to allot

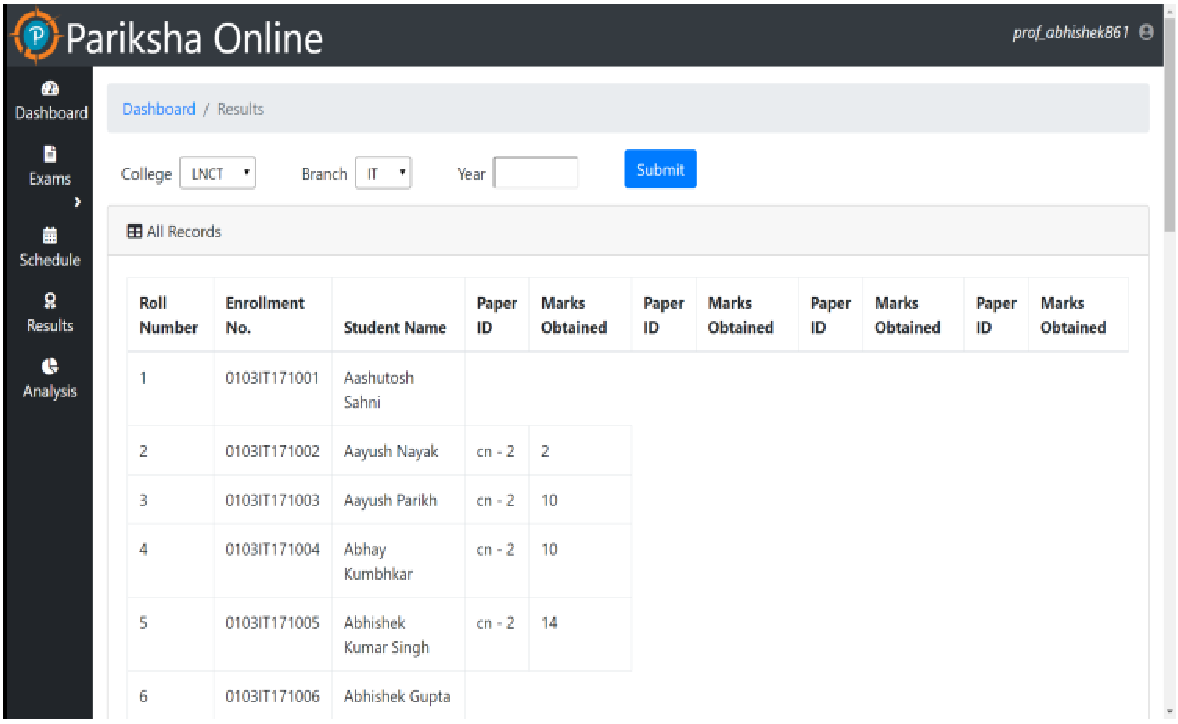
	Id	Title	Show

submit

Copyright @ParikshaOnline

Fig.10: Schedule page (Faculty module)

A teacher can schedule the paper of any particular subject for the students of a particular branch and semester.



College LNCT Branch IT Year Submit

All Records

Roll Number	Enrollment No.	Student Name	Paper ID	Marks Obtained	Paper ID	Marks Obtained	Paper ID	Marks Obtained	Paper ID	Marks Obtained
1	0103IT171001	Aashutosh Sahni								
2	0103IT171002	Aayush Nayak	cn - 2	2						
3	0103IT171003	Aayush Parikh	cn - 2	10						
4	0103IT171004	Abhay Kumbhkar	cn - 2	10						
5	0103IT171005	Abhishek Kumar Singh	cn - 2	14						
6	0103IT171006	Abhishek Gupta								

Fig.11: Result page (Faculty module)

Here a teacher can view the results of the students of a particular college branch and year.

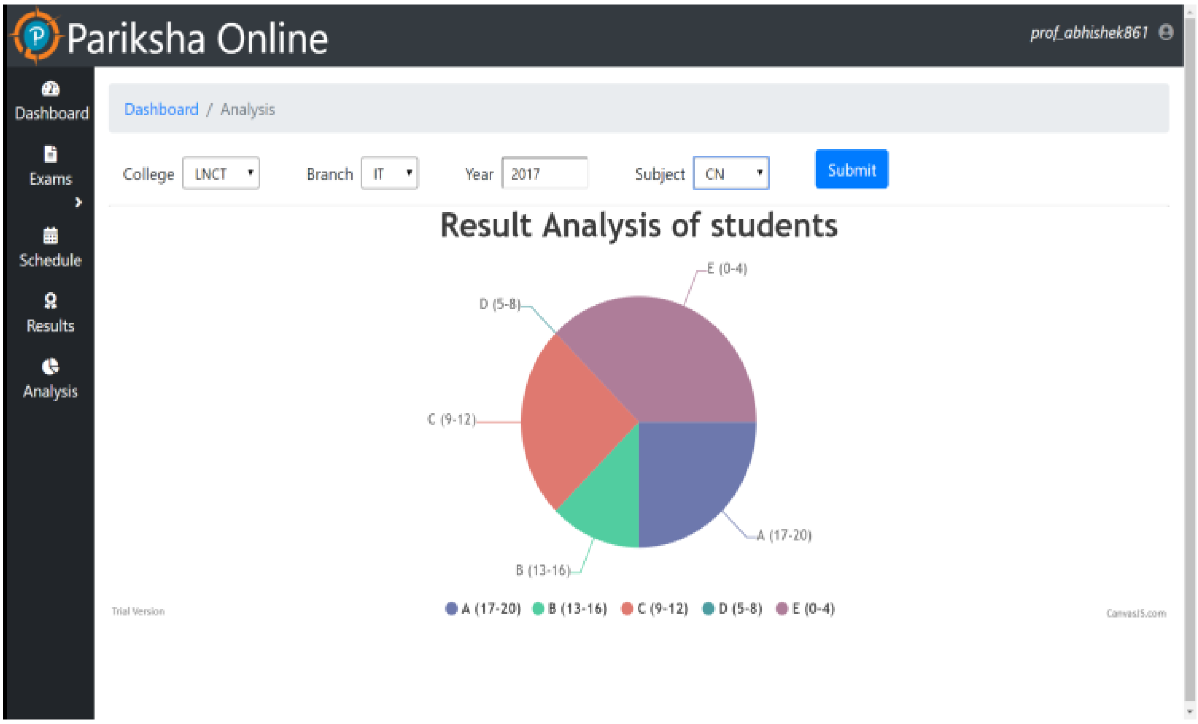
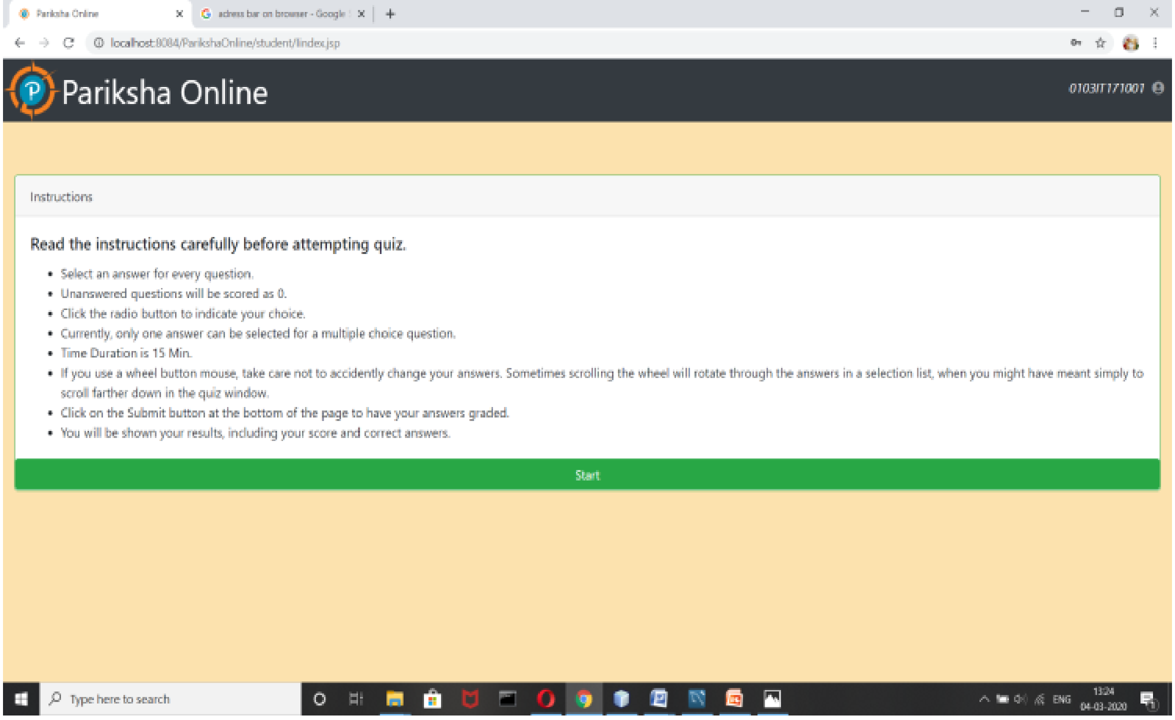


Fig.12: Analysis page (Faculty module)

This will give the analyzed report of the performance of students.



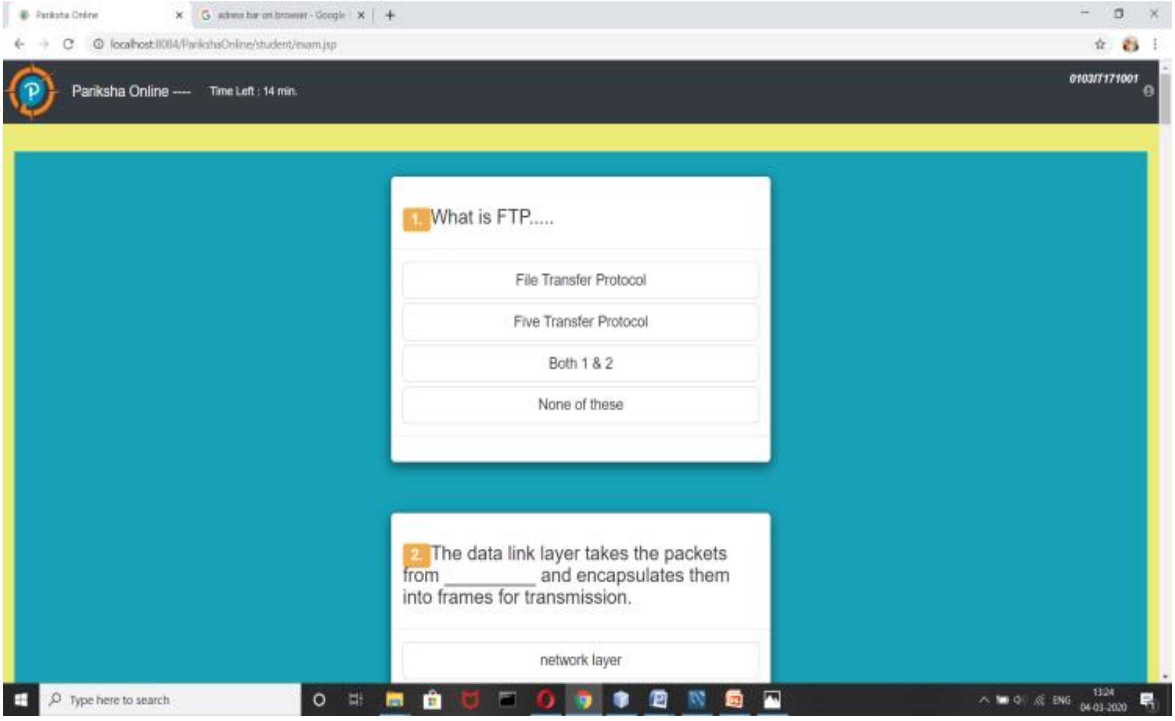
Instructions

Read the instructions carefully before attempting quiz.

- Select an answer for every question.
- Unanswered questions will be scored as 0.
- Click the radio button to indicate your choice.
- Currently, only one answer can be selected for a multiple choice question.
- Time Duration is 15 Min.
- If you use a wheel button mouse, take care not to accidently change your answers. Sometimes scrolling the wheel will rotate through the answers in a selection list, when you might have meant simply to scroll farther down in the quiz window.
- Click on the Submit button at the bottom of the page to have your answers graded.
- You will be shown your results, including your score and correct answers.

Fig.13: Instruction Page (Student Module)

Before the start of exam various instructions are displayed to students.



1. What is FTP.....

File Transfer Protocol

Five Transfer Protocol

Both 1 & 2

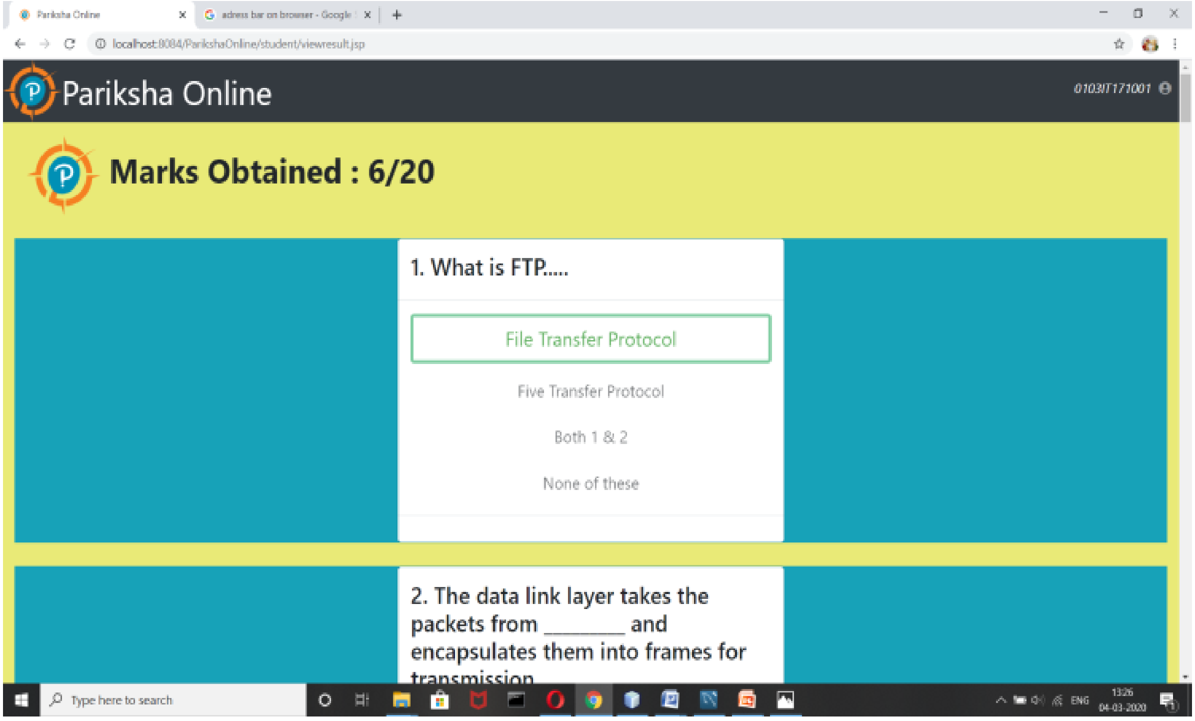
None of these

2. The data link layer takes the packets from ________ and encapsulates them into frames for transmission.

network layer

Fig.14: Mid Sem Page (Student Module)

This page contains the paper with 20 min. timer and a submit button to end the exam.


Pariksha Online
adress bar on browser - Google
localhost:8084/ParikshaOnline/student/viewresult.jsp
Pariksha Online
0103IT171001
Type here to search
ENG
13:26
04-03-2020

Marks Obtained : 6/20

1. What is FTP.....

File Transfer Protocol

Five Transfer Protocol

Both 1 & 2

None of these

2. The data link layer takes the packets from ________ and encapsulates them into frames for transmission

Fig.15: Result Page (Student Module)

This page displays the marks scored by understudies along with the correct answers.

Chapter 8

Conclusions & Future Enhancement

8.1 Conclusion

Pariksha Online is an online mid semester examination website for universities to conduct the online exam of large number of students simultaneously. It can fulfill the needs of university like adding or removing students and faculties, categorizing them on the basis of their semester, year, branch, and college. Creating and scheduling different papers for students from different branches and year and then analyzing the results and performance of the students by categorizing them on the basis of their semester, year, branch, and college. Online assessment over cloud is a relocation of distributed computing innovation in the field of online assessment which gives all the important framework, equipment and programming figuring assets participating in online examination management.

Cloud computing is buzzword for today. Most of the applications and services are utilizing it to improve efficiency and reduce cost of managing the services and applications. Online exam is perfectly suited for hosting in cloud environment. It is common that during actual examination process more number of students accesses the system and system utilizes maximum processor and transactions during concurrent examination timing.

Online examination over cloud is a migration of cloud computing technology in the field of online examination, which provides all the necessary infrastructure, hardware and software computing resources engaging in online examination management.

Cloud environment enables maximum utilization of resources during actual online exam time and reduces or scale down the resource utilization when examination is completed. So any number of applicants can appear for exam with ease and cost associated with infrastructure is minimal. Online examination would also provide you advantage of accurate results and analytics and other administration paperwork of manual examination management gets reduced drastically.

8.2 Future Enhancements

This project can be further enhanced to fulfill the needs of other universities as well as it can be improved to meet the demand of written exam conduction. Some sort of security features can also be added to restrict the user to open other tabs in brower while giving exam along with activation of the webcam for invigilating the student.

Chapter 9

References

References

[01] Muna R. Hameed, Firas. A. Abdullatif. “Online Examination System” in
ISSN (Online) 2393-8021
ISSN (Print) 2394-1588

[02] “Web Based Online Examination System” GRD Journals- Global Research and Development Journal for Engineering | Volume 2 | Issue 5 | April 2017 ISSN: 2455-5703

[03] “Partial Public Cloud for Online Exam” by Kavita S. Oza, Snehal R. Patil, Rajanish K.Kamat
IJRCSIT, ISSN No.: 2319-5010, Vol. 2, Issue 2, February 2014

[04] https://www.onlineexambuilder.com/

[05] https://www.thinkexam.com/

[06] https://www.iitms.co.in/university-erp/university-examination-system/

APPENDICES

A. Details of software:

NetBeans:

NetBeans is an integrated development environment (IDE) for Java. NetBeans allows applications to be developed from a set of modular software components called modules. NetBeans runs on Windows, macOS, Linux and Solaris. In addition to Java development, it has extensions for other languages like PHP, C, C++, HTML5 and JavaScript. The NetBeans IDE Bundle for Web & Java EE provides complete tools for all the latest Java EE 6 standards, including the new Java EE 6 Web Profile, Enterprise Java Beans (EJBs), servlets, Java Persistence API, web services, and annotations. NetBeans also supports the JSF 2.0 (Facelets), JavaServer Pages (JSP), Hibernate, Spring, and Struts frameworks, and the Java EE 5 and J2EE 1.4 platforms. It includes GlassFish and Apache Tomcat.

Apache Tomcat Server:

Apache Tomcat (also referred to as Tomcat Server) implements several Java EE specifications including Java Servlet, JavaServer Pages (JSP), Java EL, and WebSocket, and provides a "pure Java" HTTP web server environment in which Java code can run.

Tomcat is developed and maintained by an open community of developers under the auspices of the Apache Software Foundation, released under the Apache License 2.0 license, and is open-source software.

MySQL :

It is an open-source relational database management system (RDBMS). MySQL is free and open-source software under the terms of the GNU General Public License, and is also available under a variety of proprietary licenses. MySQL was owned and sponsored by the Swedish company MySQL AB, which was bought by Sun Microsystems (now Oracle Corporation). In 2010, when Oracle acquired Sun, Widenius forked the open-source MySQL project to create MariaDB. MySQL is used by many database-driven web applications, including Drupal, Joomla, phpBB, and WordPress. MySQL is also used by many popular websites, including Facebook, Flickr, MediaWiki, Twitter, and YouTube.

AWS:

Amazon Web Services (AWS) is the world’s most comprehensive and broadly adopted cloud platform, offering over 165 fully featured services from data centers globally. Millions of customers —including the fastest-growing startups, largest enterprises, and leading government agencies—trust AWS to power their infrastructure, become more agile, and lower costs. Below are the services used in hosting this project:

- Amazon Elastic Compute Cloud

Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides resizable computing capacity—literally, servers in Amazon's data centers—that you use to build and host your software systems.

- Amazon Relational Database Service

Amazon Relational Database Service (Amazon RDS) is a web service that makes it easier to set up, operate, and scale a relational database in the cloud. It provides cost-efficient, resizable capacity for an industry-standard relational database and manages common database administration tasks.

- Elastic Load Balancing

Elastic Load Balancing automatically distributes your incoming application traffic across multiple targets, such as EC2 instances. It monitors the health of registered targets and routes traffic only to the healthy targets. Elastic Load Balancing supports three types of load balancers: Application Load Balancers, Network Load Balancers, and Classic Load Balancers..

- AWS Auto Scaling

With AWS Auto Scaling, you can quickly set up dynamic and predictive scaling for your scalable AWS resources. It uses Amazon EC2 Auto Scaling to scale your EC2 instances and Application Auto Scaling to scale resources from other services.

B. Steps to run the project :

1. Import all the files in any web IDE like eclipse/ NetBeans.
2. Import the database in MySQL database.
3. Open the project in IDE.
4. Change the username and password in the database javaDao files.
5. Javascripts should be enabled in the browser.
6. Start the Tomcat Server or GlassFish Server.

7. Build the project .
8. Run the Project.

❖ Administrative Aspect :
    - Adding or expelling faculty.
    - Adding or expelling students.
    - Providing them with UserId and Password.

❖ Faculty Aspect :
    - Create/Edit/Delete questions.
    - Create/Edit/Delete paper.
    - Scheduling the paper.
    - Viewing the results.
    - Analyzing the performance.

❖ Student Aspect :
    - Appearing for the exam.
    - Viewing the result of exam.

Hoisting over AWS Cloud Platform

To meet with the above mentioned problems as discussed in problem statement, the webapp is hoisted over the cloud using services provided by Amazon Web Services (AWS). Below given a brief description of the hoisting process:

a. Start by launching an Amazon machine from EC2 service, by specifying appropriate security group along with specifying inbound and outbound rules.
b. Load the private key (.ppk file) into Pageant.
c. Using the private ip provided by EC2 login in the machine using putty application, login as ec2-user.
d. Transform the database into a dump file and using WinSCP copy the dump into ec2 machine.
e. Create a tomcat server (to support hoisting process) and maria-db server (to manage database) over the machine along with importing dump at desired location. Following are the various commands:

```
yum install tomcat*
yum install Maria-db server
systemctl start maria-db
mysql_secure_installation
```

```
systemctl restart mariadb
mysql –u root –p
mysql> create database midsem;
mysql> exit;
mysql –u root –p midsem </home/ec2-user/dump
systemctl restart tomcat
```

f. At any browser enter the ip of ec2 machine, and you will be directed to tomcat server homepage.

g. Upload the .war file (project deployable file) there and within seconds you will be redirected to website.

h. Using the complete address from the address bar we can login to website from any mobile/computer device and from any place.

Distinguishing feature of Cloud:

AWS provides various services which can easily manage the load on website, depending on the traffic. Hence resolves all the problems of server down. Also AWS provides hourly charges scheme, so no extra charges are applied except for peak hours.