

COLLEGE EXAMINATION SITE

A Major Project Report

*Submitted in partial fulfillment of
the requirements for the award of the degree of*

Bachelor of Technology in Computer Science and Engineering

Submitted by

Beulah P (16SS1A0541)
Nikhitha P (16SS1A0540)
Kavya T (16SS1A0556)
Sirisha K (16SS1A0522)

Under the guidance of

Sri. Joshi Shripad



Department of Computer Science and Engineering

JNTUH College of Engineering Sultanpur

Sultanpur(V),Pulkal(M),Sangareddy district,Telangana-502273

May 2020

JNTUH COLLEGE OF ENGINEERING SULTANPUR

Sultanpur(V),Pulkal(M),Sangareddy-502273 ,Telangana



Department of Computer Science and Engineering

Certificate

This is to certify that the Major Project report work entitled “COLLEGE EXAMINATION SITE ” is a bonafide work carried out by a team consisting of P. BEULAH bearing Roll no.16SS1A0541, P.NIKHITHA bearing Roll no.16SS1A0540, T. KAVYA bearing Roll no.16SS1A0556, K.SIRISHA bearing Roll no.16SS1A0522, in partial fulfillment of the requirements for the degree of BACHELOR OF TECHNOLOGY in COMPUTER SCIENCE AND ENGINEERING discipline to Jawaharlal Nehru Technological University Hyderabad College of Engineering Sultanpur during the academic year 2019- 2020.

The results embodied in this report have not been submitted to any other University or Institution for the award of any degree or diploma.

Guide

Sri. JOSHI SHRIPAD
Associate Professor

Head of the Department

Sri. JOSHI SHRIPAD
Associate Professor

EXTERNAL EXAMINER

Declaration

We hereby declare that the Major Project entitled “**COLLEGE EXAMINATION SITE**” is a bonafide work carried out by a team consisting of P. BEULAH bearing Roll no.16SS1A0541, P. NIKHITHA bearing Roll no.16SS1A0540, T. KAVYA bearing Roll no.16SS1A0556, K. SIRISHA bearing Roll no. 16SS1A0522, in partial fulfillment of the requirements for the degree of Bachelor of Technology in Computer Science and Engineering discipline to Jawaharlal Nehru Technological University Hyderabad College of Engineering Sultanpur during the academic year 2019-2020. The results embodied in this report have not been submitted to any other University or Institution for the award of any degree or diploma.

Beulah P (16SS1A0541)

Nikhitha P (16SS1A0540)

Kavya T (16SS1A0556)

Sirisha K (16SS1A0522)

Acknowledgment

We wish to take this opportunity to express our deep gratitude to all those who helped us in various ways during our Major Project report work. It is our pleasure to acknowledge the help of all those individuals who were responsible for foreseeing the successful completion of our Major Project report.

We express our sincere gratitude to **Prof. B. Balu Naik, Principal, JNTUHCES** for his support during the course period.

We sincerely thank **Prof. V. Venkateshwara Reddy, Vice Principal, JNTUHCES** for his kind help and co-operation.

We are thankful to our Guide **Sri. Joshi Shripad , Associate Professor and Head of the Department**, Computer Science and Engineering, for his effective suggestions during the course period.

We are thankful to **Dr. B. V. Ram Naresh Yadav, Associate Professor** for his support and guidance in the completion of our Major Project.

We are thankful to our Co-Guide **Mrs. G. Priyanka Jeeva Karunya, Assistant Professor(C)** for her support and encouragement throughout the course period.

Finally, we express our gratitude with great admiration and respect to our faculty for their moral support and encouragement throughout the course.

Beulah P (16SS1A0541)
Nikhitha P (16SS1A0540)
Kavya T (16SS1A0556)
Sirisha K (16SS1A0522)

Contents

Certificate	i
Declaration	ii
Acnowledgement	iii
Abstract	viii
1 INTRODUCTION	1
1.1 Project Overview	1
1.2 Purpose	1
1.3 Existing System	2
1.4 Proposed System	2
1.5 Scope	3
2 LITERATURE SURVEY	4

2.1	PHP	4
2.2	JavaScript	7
2.2.1	Materialize.JS	8
2.3	HTML	11
2.4	CSS	12
2.4.1	Materialize.CSS	12
2.5	MySql	15
2.6	Apache Server	16
2.7	XAMPP	16
3	ANALYSIS	17
3.1	Security	17
3.2	Reliability	17
3.3	Availability	18
3.4	Constraints	18
3.5	Portability	18
3.6	Performance	18
3.7	System Architecture	19
3.8	Feasibility Study	19

3.9	Hardware and Software Requirements	20
3.10	Objectives	20
3.11	System Activities	22
4	DESIGN	25
4.1	Use Case Diagram	26
4.2	Activity Diagram	27
4.3	Class Diagram	28
4.4	Flow Chart Diagram	29
5	IMPLEMENTATION	30
5.1	Code	30
5.1.1	index.php	30
5.1.2	insertquestions.php	32
5.1.3	getquestions.php	35
5.1.4	showresult.php	38
5.2	Testing	40
5.2.1	Functionality Testing	41
5.2.2	Usability Testing	42

5.2.3	Interface Testing	43
5.2.4	Security Testing	43
6	SCREENSHOTS	44
6.1	Admin Module	44
6.2	Faculty Module	49
6.3	Student Module	53
6.4	Database	60
	CONCLUSION	64
	REFERENCES	65

Abstract

Our project is a website intended to conduct examinations to students. It stores all the questions in the database in separate modules divided according to the subject, topic and difficulty of questions i.e. easy, medium, hard. Input for number of questions and topics must be given. Question paper will be generated according to the input. It picks questions randomly from the database by including the combination of all levels of difficulty. Questions once generated in one question paper will not be repeated in the other. Results will be evaluated.

List of Figures

2.1	PHP evolution	4
2.2	JavaScript evolution	7
2.3	HTML evolution	11
4.1	Use Case diagram of college examination site	26
4.2	Activity diagram of college examination site	27
4.3	Class diagram of college examination site	28
4.4	Flow Chart diagram of college examination site	29
6.1	Admin panel to login	44
6.2	Admin login	45
6.3	Existing faculty list	45
6.4	page to add faculty	46
6.5	page to add faculty	46
6.6	Giving faculty details	47

6.7	Giving faculty details	47
6.8	showing newly added faculty	48
6.9	Faculty main page	49
6.10	Creating exam schema	50
6.11	Giving details of exam schema/paper	50
6.12	page to enter exam details	51
6.13	page to enter no.of questions,exam name in-order to enter questions into database	51
6.14	Inserting questions into database	52
6.15	Message showing that questions are successfully inserted	52
6.16	user(student/faculty) login page	53
6.17	student action page	54
6.18	page for selecting exam	54
6.19	selecting exam	55
6.20	Exam with timer	55
6.21	submitting the exam	56
6.22	showing results at the end of exam	56
6.23	view results page	57

6.24 Student registration page	57
6.25 Entering student details	58
6.26 message showing successful registration	58
6.27 message if question paper does not exist	59
6.28 online exam database	60
6.29 Admin details table	61
6.30 multiple choices(options) table	61
6.31 list of question papers table	62
6.32 list of questions table	62
6.33 Faculty and student details table	63
6.34 students results table	63

Chapter 1

INTRODUCTION

1.1 Project Overview

This project assesses students by conducting online objective tests. The tests would be highly customizable. This project will enable educational institutions to conduct test and have automated checking of answers based on the response by the candidates.

The project allows faculties to create their own tests. It would enable educational institutes to perform tests, quiz. It asks faculty to create his/her set of questions. Further the tests are associated[8] with specific groups so that only associated students can appear for the test. The result of the response would be available to the faculty of the question set. This project would be helpful for creating practice tests, say for educational institutes.

1.2 Purpose

Responses by the candidates will be checked automatically and instantly.

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

- Can generate various reports almost instantly when and where required.

1.3 Existing System

Existing system is manual entry and keeping the details of students who are registered already. And it is very difficult for each student to come to exam center. It is very difficult for students from far distance to reach the exam center. This system is required to prepare registration/application form, question paper for the students and required to print a lot of number manually. To calculate how many students registered, and verification of details of these students in a month by hand is very difficult. This requires quite a lot of time and wastage of money as it requires quite lot of manpower to do that. Another factor that takes into account is the possibility of errors. The limitation of existing system is that it is not all personalized. It cannot be used for personal and quick reference. Even the other staff members can make quick entries if the responsible person is not present.

- Time Consuming for creating question paper
- Time to check right and wrong answers
- Calculation of Marks
- Human error
- Limitation of no of student can give examination at a time
- Require teacher to monitor exam centre
- Students need to come to exam centre for giving test

1.4 Proposed System

The modern computerized system is developed with the aim to overcome the drawbacks of existing manual system. The proposed system has got many advantages. People from different parts of the world can register very easily. The new system is more personalized. It is maze in such a manner that all the new users can understand all the

options in it very easily. It is made in a quick and easy referential manner. Access to all important matters are not always locked and can be opened easily at the time of urgency. The advantages of proposed system are that security is maintained in the new system. Securities for all important data are maintained confidentially. As it is easily understandable and user friendly, quick entries can be made in this system.

- Provides complete online web based solution, including student registration, giving tests, storing of results.
- Complete web based administration, administrator can manage examination and question bank from web interface.
- No geographical boundary.
- Student can give examination from anywhere of the world by 24X7.
- 100% accuracy in result calculation.
- Randomization of question set.

1.5 Scope

This project would be very useful for educational institutes where regular evaluation of students is required. Further it can also be useful for anyone who requires feedback based on objective type responses.

Chapter 2

LITERATURE SURVEY

2.1 PHP

Originally PHP stood for “Personal Home Page”. PHP is an ”HTML-embedded scripting[1] language” primarily used for dynamic Web applications. The first work was done by Rasmus Lerdorf and dates back to about 1994. Lerdorf originally used it for his personal page – specifically to track visitors. Soon, a lot of additional functions were added. It took a few years, though, for it to become a language and not just a set of tools.

The code was released in the mid-1990s. Israel’s Andi Gutmans and Zeev

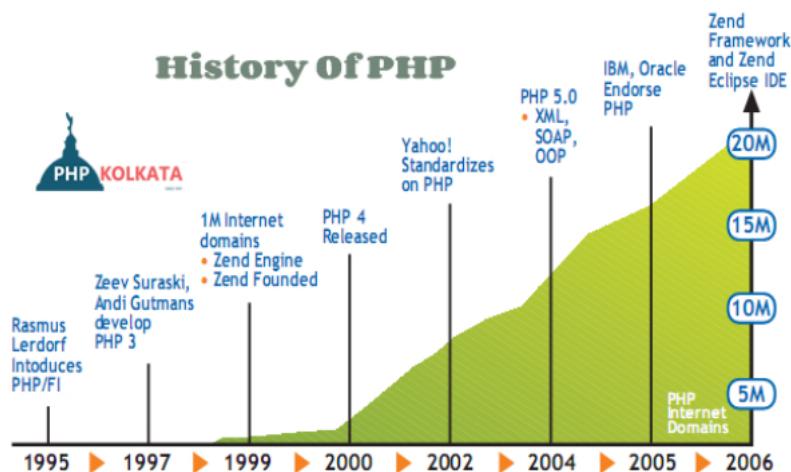


Figure 2.1: PHP evolution

Suraski did a major overhaul in 1997 with the goal of using PHP to run an e-Commerce site. Their version, termed PHP 3.0, had more of the features we expect and see today. The same duo later created Version 4.0. 2004 saw another major revision, PHP 5.0. Although there has not yet been a 6.0, there have been significant improvements since 5.0, including the removal of several[4] things that had caused instability or potential security breaches.

Fig 2.1 shows the evolution of php.

What is PHP?

- PHP is an acronym for "PHP: Hypertext Pre-processor."
- PHP is a widely-used, open source scripting language.
- PHP scripts are executed on the server.
- PHP is free to download and use.

What is a PHP File?

- PHP files can contain text, HTML, CSS, JavaScript, and PHP code.
- PHP code is executed on the server, and the result is returned to the browser as plain HTML.
- PHP files have extension ".php"

What Can PHP Do?

- PHP can generate dynamic page content.
- PHP can create, open, read, write, delete, and close files on the server.
- PHP can collect form data

- PHP can send and receive cookies.
- PHP can add, delete, modify data in your database.
- PHP can be used to control user-access.
- PHP can encrypt data.

Why PHP?

- PHP runs on various platforms (Windows, Linux, Unix, Mac OS X, etc.)
- PHP is compatible with almost all servers used today (Apache, IIS, etc.)
- PHP supports a wide range of databases.
- PHP is easy to learn and runs efficiently on the server side
- PHP is free.

2.2 JavaScript

JavaScript is a high-level programming language that conforms to the ECMA script specification. JavaScript is a just-in-time compiled and multi-paradigm and has curly-bracket syntax, dynamic typing, prototype-based object-orientation, and first-class functions. Alongside HTML and CSS, JavaScript is one of the core technologies of the World Wide Web. JavaScript enables interactive web pages and is an essential part of web applications. The vast majority of websites use it for client-side page behavior, and all major web browsers have a dedicated JavaScript engine to execute it.

As a multi-paradigm language, JavaScript supports event-driven, functional,

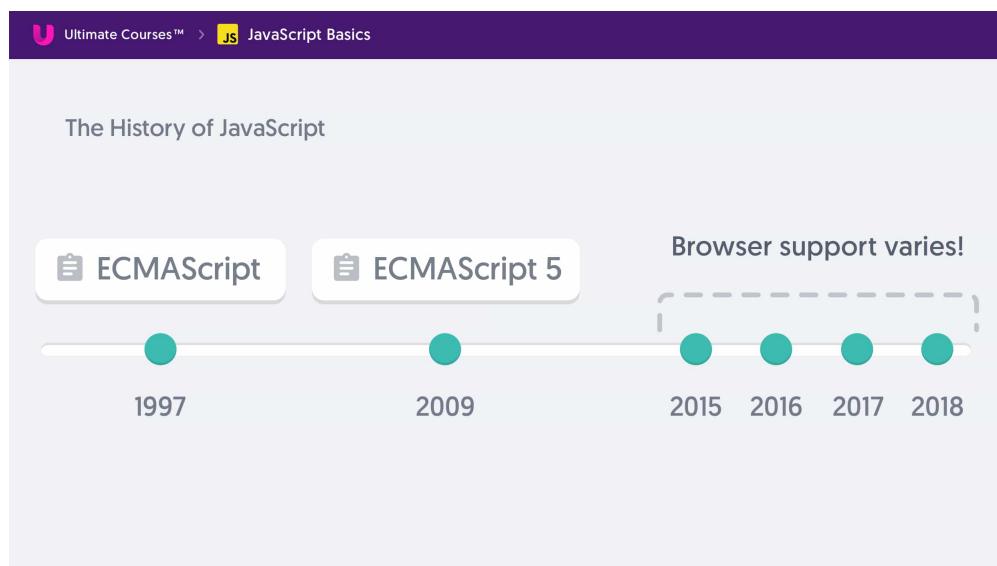


Figure 2.2: JavaScript evolution

and imperative programming styles. It has application programming interfaces (APIs) for working with text, dates, regular expressions, standard data structures, and the Document Object Model (DOM). JavaScript engines were originally used only in web browsers, but they are now embedded in some servers, usually via Node.js. They are also embedded in a variety of applications created with frameworks such as Electron and Cordova.

Fig 2.2 shows the evolution of JavaScript.

- JavaScript is the programming language of HTML and the Web.
- JavaScript is easy to learn.

2.2.1 Materialize.JS

Materialize is a UI component library created with CSS, JavaScript, and HTML. Materialize.js include the following:

Materialize Collapsible

Collapsible are accordion that expands when you click on it. It displays only the relevant content and hides other content. Materialize provides collapsible with various options which can be used easily anywhere.

- To create pop-out collapsible we need to add pop-out class.
- Sometimes we need to open some section by default or using JavaScript, to do this simply add the active class to header.
- The following options are available:
 - **accordion(boolean):** This is used to change the collapsible behavior to expandable instead of the default accordion style.
 - **onOpen(function):** This is callback function triggered when collapsible is opened.
 - **onClose(function):** This is callback function triggered when collapsible is closed.

There are two types of Collapsible available in Materialize:

- **Accordion:** In accordion one section is opened at a time.
- **Expandable:** In expandable more than one section can be opened at the same time.

Materialize Toasts

Toasts are basically used to provide the unobtrusive alerts to the user.

- **Call back:** You can add a callback function when the toast is dismissed.

- **Rounded Toast:** To create rounded toast you can simply pass the third parameter as 'rounded'.

Materialize Tooltips

Tooltips are basically used to provide the extra information about the elements. It is very easy to create Tooltips in Materialize framework.

The following options are available in Tooltips:

- **delay:** Delay time before tooltip appears, Default is 350 ms.
- **tooltip:** Text to display on hover. It can be HTML if you have set html to true.
- **position:** This is used to set the position of the tooltip- it can be top, left, bottom, right. Default position is bottom.
- **html:** This is used to set true to display the html content in tooltip. Default is false.
- **Remove:** We can remove tooltip suing jQuery.

Materialize Dropdown

Materialize provides dropdown which can be embed with the buttons.

The following options are available in dropdown:

- **induration:** The duration of the transition enter in milliseconds. Default time for this is 300ms.
- **outduration:** This is used to transition out in millisecond, Default time for this is 225ms.
- **constrainwidth:** If true, constrainWidth to the size of the dropdown activator. Default is set true.
- **hover:** This is used to open dropdown on hover by default it is false, if you want to open on hover set it true.
- **gutter:** This defines the spacing from the aligned edge. Default value is 0.
- **beloworigin:** This is used to set whether the dropdown will open below the activator, if you want to open dropdown below the activator set it true. Default is set false.

- **alignment:** Defines the edge the menu is aligned to. Default is set ‘left’.
- **stoppropagation:** If true, stops the event propagating from the dropdown origin click handler. Default is set false.

Materialize SideNav

SideNav is one of the main component of any webpage which provides users side navigation. It is very easy to create sidenav in Materialize framework. As it is jQuery plugin so it needs the initialization.

Materialize Css Modal Popup

Materialize Css provides predefined classes to create beautiful modals.

Class modal is used to define the modal, Class modal-content is used to define the content body. Class modal-footer defines the modal footer. Class modal-close is used to add close button in modal.

- **Modal fixed footer:** This class is used to create fixed footer.
- **jQuery Initialization:** we must initialize modal before using it.
- **Open and close modal:** By using this classes we can open and close modal.

2.3 HTML

HTML stands for “Hyper Text Markup Language” and is the standard language used to produce[2] web pages and applications. HTML was first presented by Tim Berners-Lee, the creator of the World Wide Web, in 1989. HTML 1.0 was the first version of

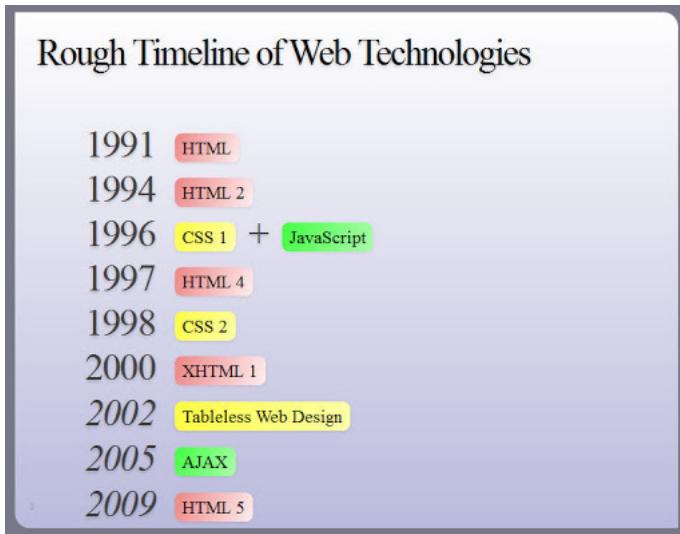


Figure 2.3: HTML evolution

HTML, used from 1989 to 1994. It was a very limited version and included only 20 elements. It didn't support altering of page background.

HTML 2.0 created in 1995, becomes the first official set of standards for HTML , the base standard by which all browsers were measured until HTML 3.2. It was able to support the changing of a page background, text colour, text face, the use of tables and text boxes etc. In January 1997, HTML 3.2 was endorsed by the W3 Consortium and approved of by many, including significant browsers such as Netscape and Microsoft. HTML 3.2 included tables, applets, text flow around images, subscripts and superscripts. HTML 4.01, created in 1999, included cascading style sheets (css) which allowed aspects such as text, colour, font and backgrounds to be easily altered.

HTML 5 is the current version of HTML and can be used to write web applications that still work when you're not connected to the net to tell websites where you are physically located to handle high definition video and to deliver extraordinary graphics. It continues to evolve and is supported by all of the biggest browsers such as Firefox, Chrome, Safari, Internet Explorer, Opera and Edge.

Fig 2.3 shows the evolution of HTML.

2.4 CSS

Cascading Style Sheets (CSS) developed by Hakon Wium Lie ,is a style sheet language used for describing the presentation of a document written in a markup language like HTML. It is a cornerstone technology of the WWW, alongside HTML and JS. It is designed to enable the separation of presentation and content, including layout, colors, and fonts.

2.4.1 Materialize.CSS

Materialize is a UI component library created with CSS, JavaScript, and HTML. Materialize.css include the following:

Materialize Color

Materialize provides rich predefined[5] colors. These colors are based on Material Design base colors. Colors in Materialize are defined with a base color class and an optional class which can be used for lighten or darken.

Materialize Helpers

Materialize Provides inbuilt classes which fulfills the common and frequent tasks such as alignment, float, formatting, hover etc.

The following Helper Classes are available:

1. Alignment Classes

- **Vertical Align:** If you want to align the things vertically just add the class valign-wrapper to the container and valign class to the element you want to align vertically.
- **ii. Text Align:** There are following three classes which can be used to align text horizontally.
 - **left-align :** This class is used to left align the text.
 - **right-align :** This class is used to right align the text.
 - **center-align :** This class is used to center align the text.

2. Float Classes

- Float left
- Float Right

3. Hiding Content Classes

Sometimes we need to hide some content for all devices or some specific devices. Materialize provides inbuilt classes to hide content on several devices.

- **hide:** Hide for all Devices.
- **hide-on-small-only:** Hide for Mobile Only.
- **hide-on-med-only:** Hide for Tablet Only.
- **hide-on-med-and-down:** Hide for Tablet and Below.
- **hide-on-med-and-up:** Hide for Tablet and Above.
- **hide-on-large-only:** Hide for Desktop Only.

4. Formatting Class

Materialize provides Formatting classes to format content.

There are two formatting classes:

- **Truncation:** This class is used to truncate the long text in an ellipsis.
- **Hover:** Materialize provides the hoverable class that adds the box shadow animation.

5. Browser Default Class

Some default styles are overridden in Materialize and sometimes we need default styles instead of overridden, so it provides browser-default class to revert the elements style to their original state.

Materialize Media

Materialize provides classes for displaying media content. Here in this tutorial we are going to explain how you can use media classes in Materialize CSS to manage the media content. Media include Images and Videos. Images include Responsive Images where we add the class responsive-img to the image tag which makes image responsive and Circular Images where we need to add Circle class to the image tag. We can embed the videos in a page using responsive-video and video-container class. In Responsive Embed if, we want to add any youtube videos we can embed videos simply by giving the link which is responsive on all devices. We can make html videos responsive simply adding the class responsive-video.

Materialize Shadow

Materialize Provides Various Shadow classes to add the Shadow effect in the element. The following are the classes available to add Shadows:

- **z-depth-1:** This Adds 1px border and z-depth 1
- **z-depth-2:** This Adds 2px border and z-depth 2
- **z-depth-3:** This Adds 3px border and z-depth 3
- **z-depth-4:** This Adds 4px border and z-depth 4
- **z-depth-5:** This Adds 5px border and z-depth 5
- **z-depth-0:** This is used to remove shadow from the element that have z-depths defaults.

Materialize Table

Materialize Provides Utility Classes to create tables, Using these classes we can create beautiful tables.

The following are utility classes to create tables:

- **Borderless Table:** By default tables in materialize framework are borderless.
- **Bordered Table:** To create bordered table add class bordered to the table tag.

- **Striped Table:** To create striped table add class Striped to the table tag.
- **Highlight Table:** To create highlight table add class Highlight to the table tag.
- **Centered table:** To create centered table add class Centered to the table tag.
- **Responsive Table:** To create responsive table add class Responsive table to the table tag.

Materialize Typography

Roboto 2.0 standard font is used in Materialize framework. Sometimes we do not need the default Roboto font so we need to override/remove the default Roboto font.

2.5 MySql

MySQL was created by a Swedish company, MySQL AB, founded by David Axmark, Allan Larsson and Michael "Monty" Widenius. Original development of MySQL by Widenius and Axmark began in 1994. The first version of MySQL appeared on 23 May 1995. It was initially created for[3] personal usage from mSQL based on the low-level language ISAM, which the creators considered too slow and inflexible. They created a new SQL interface, while keeping the same API as mSQL. By keeping the API consistent with the mSQL system, many developers were able to use MySQL instead of the (proprietary licensed) mSQL antecedent.

MySQL is an open-source relational database management system (RDBMS). Its name is a combination of "My", the name of co-founder Michael Widenius's daughter, and "SQL", the abbreviation for Structured Query Language. 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.

2.6 Apache Server

Originally based on the NCSA HTTPd server, development of Apache began in early 1995 after work on the NCSA code stalled. Apache played a key role in the initial growth[6] of the World Wide Web, quickly overtaking NCSA HTTPd as the dominant HTTP server, and has remained most popular since April 1996. In 2009, it became the first web server software to serve more than 100 million websites. As of April 2020, Netcraft estimated that Apache served 29.12% of the million busiest websites, while Nginx served 25.54%; according to W3Techs, Apache served 39.5% of the top 10 million sites and Nginx served 31.7%.

The Apache HTTP Server, colloquially called Apache, is a free and open-source cross-platform web server software, released under the terms of Apache License 2.0. Apache is developed and maintained by an open community of developers under[3] the auspices of the Apache Software Foundation. The vast majority of Apache HTTP Server instances run on a Linux distribution, but current versions also run on Microsoft Windows and a wide variety of Unix-like systems. Past versions also ran on OpenVMS, NetWare, OS/2 and other operating systems, including ports to mainframes.

2.7 XAMPP

XAMPP is a free and open-source cross-platform web server solution stack package developed by Apache Friends, consisting mainly of the Apache HTTP Server, MariaDB database, and interpreters[3] for scripts written in the PHP and Perl programming languages. Since most actual web server deployments use the same components as XAMPP, it makes transitioning from a local test server to a live server possible. XAMPP is regularly updated to the latest releases of Apache, MariaDB, PHP and Perl. It also comes with a number of other modules including OpenSSL, phpMyAdmin, MediaWiki, Joomla, WordPress and more. Self-contained, multiple instances of XAMPP can exist on a single computer, and any given instance can be copied from one computer to another. XAMPP is offered in both a full and a standard version (Smaller version).

Chapter 3

ANALYSIS

3.1 Security

- Administrator has the highest authority to edit/delete/create database.
- Faculty have the authority to add or delete the questions in the question paper.
- Students can only view their test records.
- Faculty can view all the test records of every student.
- Critical information like password should be transferred in encrypted form.
- Passwords should be stored in encrypted form.

3.2 Reliability

- Data validation and verification needs to be done at every stage of activity.
- Validating user input.
- Use of locking mechanism while updating database like transaction processing.
- Recovering the transaction using rollback.

3.3 Availability

The examination system being an online system should be available anytime.

3.4 Constraints

- The system should be available 24x7.
- The test may be time limited so the candidates appearing will have limited time to answer the test.

3.5 Portability

- The web application will be built using PHP which has support to run on any platform provided the required compilers are available.
- For database either XML or MySQL would be used, that too has extensive support over many popular architectures and operating systems.
- Portability would be limited to the support provided by the respective application vendor on various architectures and operating environments.

3.6 Performance

- The system would be used by multiple users at a time and may grow as time passes.
- The system would need to implement multi-threading to achieve acceptable performance. Further a database connection pool may also be required for assigning faster database connection.

3.7 System Architecture

- The administrator module, faculty and student modules include their part of functions to the Online Examination.
- The administrator adds registered information of the users to the Online Examination system database and edited or deletes it as needed.
- The faculty inserts the questions to the question paper of the Online Examination in subject wise manner.
- The question numbers are automatically generated. The complete subject wise results of students can be viewed by the administrator at any time after the completing the exam.
- The student only have to login to attend the exam and after completing and submitting the exam the results are immediately generated.

3.8 Feasibility Study

1. 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 system and compare them with costs, decisions is made to design and implement the system.

This part of feasibility study gives the top management the economic justification for the new system. This is an important input to the management the management because very often the top management does not like to get confounded by the Various technicalities that bound to be associated with a project of this kind. A simple economic analysis that gives the actual comparison of costs and benefits is much more meaningful in such cases.

In the system, the organization is must be 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.

2. TECHNICAL FEASIBILITY

Technical feasibility centres on the existing manual system of the test management process and to what extent it can support the system. According to feasibility analysis procedure the technical feasibility of the system is analyzed and the technical requirements such as software facilities, procedure, inputs are identified. It is also one of the important phases of the system development activities. The system offers greater levels of user friendliness combined with greater processing speed. Therefore, the cost of maintenance can be reduced. Since processing speed is very high and the work is reduced in the maintenance point of view management convince that the project is operationally feasible.

3. 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.

3.9 Hardware and Software Requirements

Front end: HTML, CSS, JavaScript

Back end: PHP, MySQL

3.10 Objectives

Online Examination System – e-Examination is complete end to end solution to cover all aspects of online examination system.

The basic objective of developing this project is:

- Provides complete web site solution, including member registration, giving tests, storing of results. Complete web based administration.

- The online examination system can automatically add the marks allocated in each question to determine the total mark for the test.
- Using this feature the questions can be selected from the question Bank, and also the appearance of questions in the Random Order.
- Additional to the already available questions in the Question Bank, the Administrator has the ability to add more questions as per the subject.
- The Administrator also has the ability to modify the User-defined Questions and not the already Available Questions in the Question Bank.
- Complete web based system no installation required to run the application in client system.
- In our project faculty can take the test of student, insert the question set, and admin can provide the login for student after student registration.
- Students result will be saved and can be retrieved later on.
- Question must provide answer type like multiple right answer selection and one right answer selection. Test screen must provide options accordingly.
- System must able to calculate the examination marks and display the result quickly after completing the exam.
- System has powerful logical access management in place, each user must be identified by user-id and password authentication policy is applied to secure the examination system.
- The online examination system generates number of reports for administration like, Total students/members registered for examination, no of available examination, test given by the student.

3.11 System Activities

1. Login

- **login as faculty:** By using already stored faculty name and password the individual can log on to the system any time he/she desires as an examiner. Logging is successful only if the input detail is matched with the database, else an error message is displayed.
- **Login as student:** The information of each student will be sorted by the admin upon the registration process, enabling this way the particular student to log on the system without having to undergo the process of registration again. Logging is successful only if the input detail is matched with the database, else an error message is displayed.

2. Admin Activities

Database Management

- Adding Questions: include Multiple choice questions with four options.
- Deleting Questions: include deleting questions from database.

Student Management

- Registering students: include inserting the information of each student to complete the registration process.
- Deleting student: Admin can delete student from database.

Result Management: show all students results.

3. Student Module:

An enhanced interface for student to registration, edit profile, login, select examination, and give exam, view the exam results, view answers of the questions following

modules pages are available for the student.

4. Administrator Aspect

1. Taking backup of the database
2. Editing/Deleting/Creating the database
3. Adding or expelling faculty.

5. Faculty Aspect

- 1.Logging into the system
- 2.Save the automatically generated facility id
- 3.Creating exam schema
- 4.Choose the subject to create a question paper
5. Creating a test
6. Posting questions in the above test
7. Posting multiple options to respective question
8. Marking correct answer within the given options
9. Specifying to allow user defined answers
10. Time limit of the test if any
11. Whether to randomize the questions
12. Whether to randomize the options displayed
13. Log out from the faculty portal

6. Student Aspect:

1. Requesting registration
2. Logging into the system
3. Selecting the test
4. Appearing for the examination
5. Submitting the result at the end of the examination
6. Reviewing the results

7. Analysis:

1. Authenticating users based on user-id and password
2. Keeping session track of user activity
3. Recording candidates' responses to every question
4. Checking whether the given response is correct or not

Chapter 4

DESIGN

Design is the abstraction of a solution it is a general description of the solution to a problem without the details. Design is view patterns seen in the analysis phase to be a pattern in a design phase. After design phase we can reduce the time required to create the implementation.

A UML diagram is a diagram based on the UML (Unified Modeling Language) with the purpose of visually representing a system along with its main actors, roles, actions, artifacts or classes, in order to better understand, alter, maintain, or document information about the system.

What is UML?

UML is an acronym that stands for Unified Modelling Language. Simply put, UML is[7] a modern approach to modelling and documenting software. In fact, it's one of the most popular business process modelling techniques.

It is based on diagrammatic representations of software components. As the old proverb says: “a picture is worth a thousand words”. By using visual representations, we are able to better understand possible flaws or errors in software or business processes.

Building Blocks of the UML: The vocabulary of the UML encompasses three kinds of building blocks.

- **Things:** Things are the abstractions that are first-class citizens in a model
- **Relationships:** ; relationships tie these things together

- **Diagrams:** diagrams group interesting collections of things

4.1 Use Case Diagram

Use case diagrams are a set of use cases, actors, and their relationships. They represent the use case view of a system.

A use case represents a particular functionality of a system. Hence, use case diagram is used to describe the relationships among the functionalities and their internal/external controllers. These controllers are known as actors. In this project, faculty and student are the actors.

Fig 4.1 shows the Use Case diagram of college examination site.

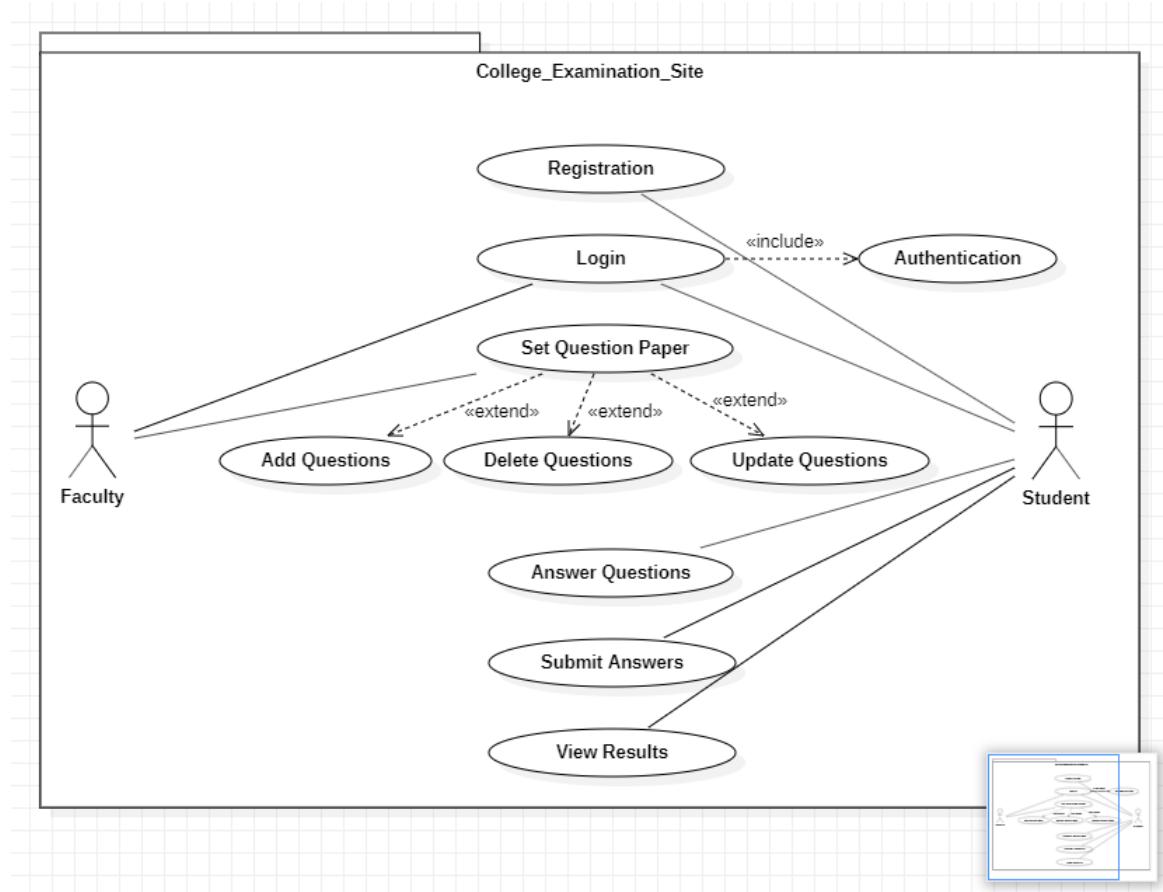


Figure 4.1: Use Case diagram of college examination site

4.2 Activity Diagram

Activity diagrams are used to document workflows in a system, from the business level down to the operational level. The general purpose of Activity diagrams is to focus on flows driven by internal processing vs. external events.

Activities are nothing but the functions of a system. Numbers of activity diagrams are prepared to capture the entire flow in a system.

Fig 4.2 shows the Activity diagram of college examination site.

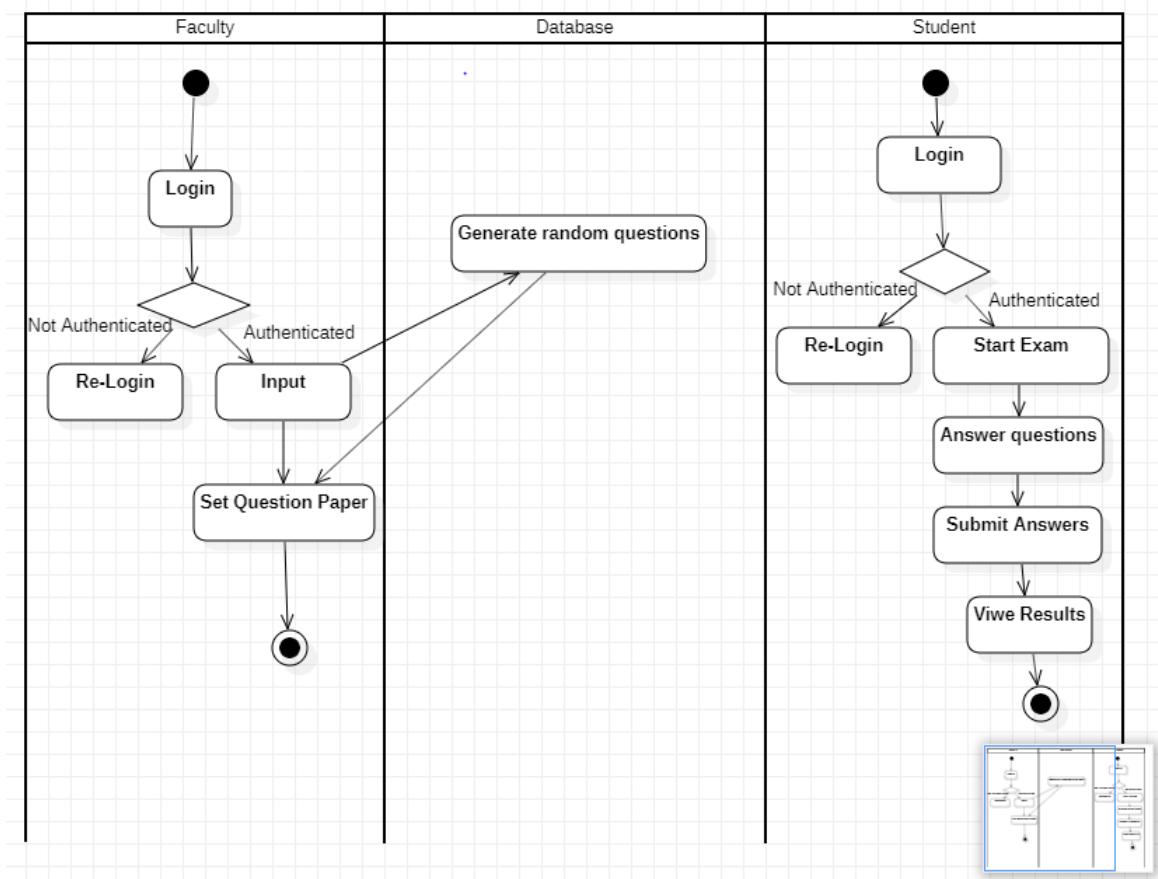


Figure 4.2: Activity diagram of college examination site

4.3 Class Diagram

Class diagram is a static diagram. It represents the static view of an application. Class diagram is not only used for visualizing, describing, and documenting different aspects of a system but also for constructing executable code of the software application.

Class diagram describes the attributes and operations of a class and also the constraints imposed on the system. The class diagrams are widely used in the modelling of object-oriented systems because they are the only UML diagrams, which can be mapped directly with object-oriented languages. Class diagram shows a collection of classes, interfaces, associations, collaborations, and constraints. It is also known as a structural diagram.

Fig 4.3 shows the Class diagram of college examination site.

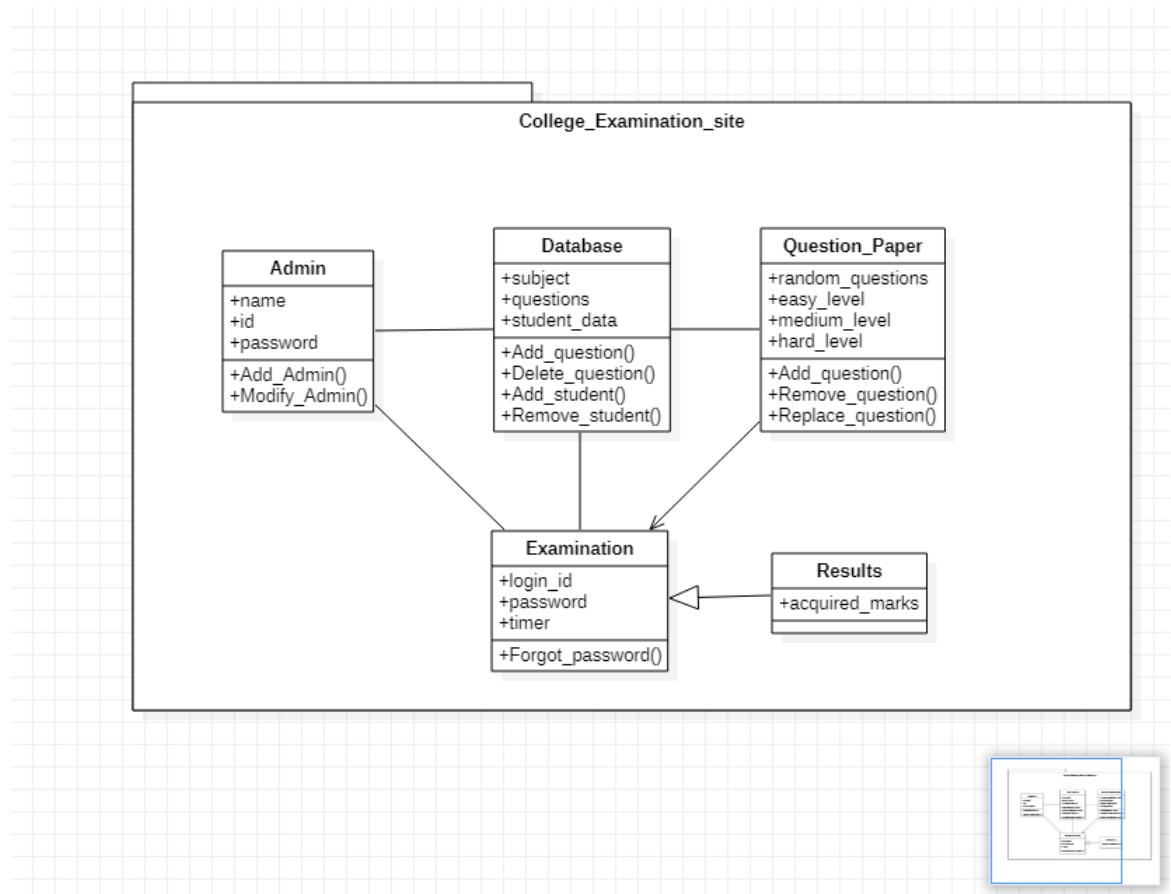


Figure 4.3: Class diagram of college examination site

4.4 Flow Chart Diagram

A flowchart is a type of diagram that represents a workflow or process. A flowchart can also be defined as a diagrammatic representation of an algorithm, a step-by-step approach to solving a task.

The flowchart shows[3] the steps as boxes of various kinds, and their order by connecting the boxes with arrows. This diagrammatic representation illustrates a solution model to a given problem. Flowcharts are used in analyzing, designing, documenting or managing a process or program in various fields.

Fig 4.4 shows the Flow chart diagram of college examination site.

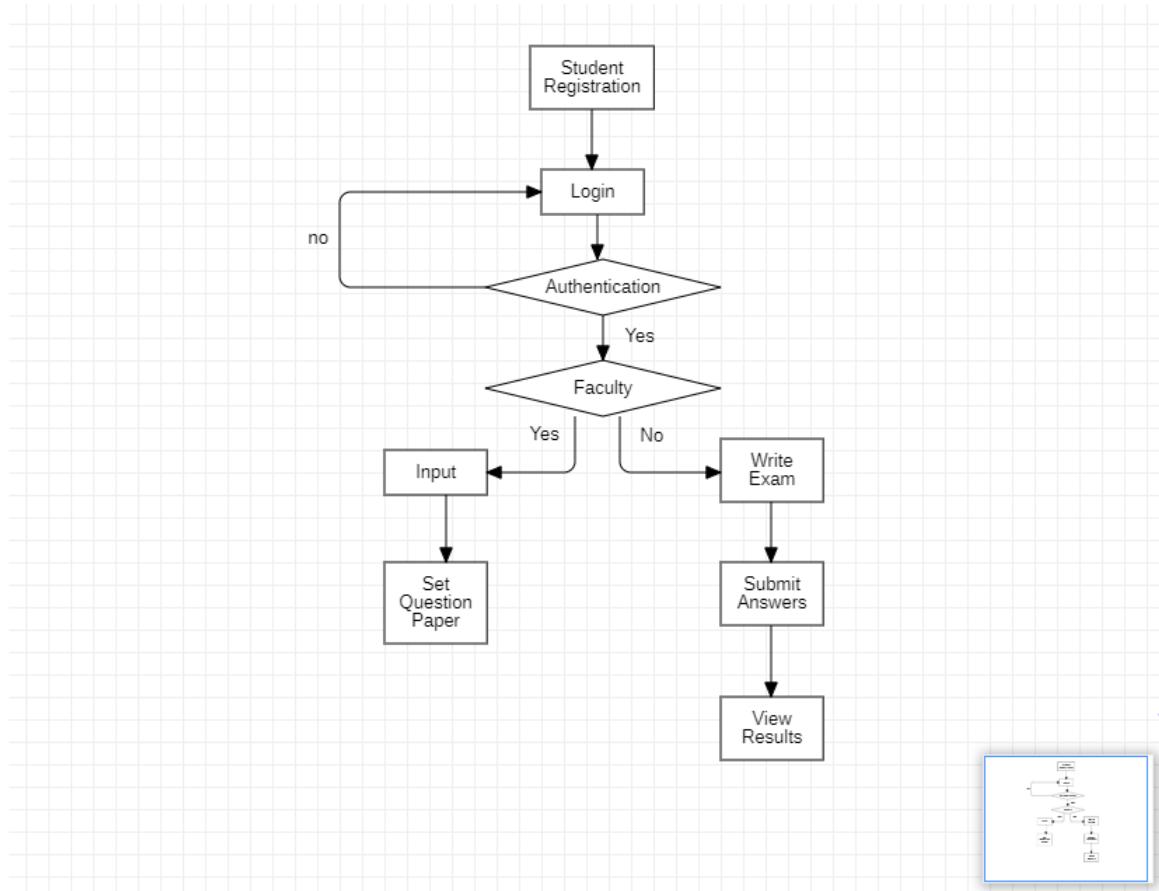


Figure 4.4: Flow Chart diagram of college examination site

Chapter 5

IMPLEMENTATION

This website helps to conduct examinations to students. It stores all the questions in the database in separate modules divided according to the subject, topic and difficulty of questions i.e. easy, medium, hard. Input for number of questions and topics must be given. Question paper will be generated according to the input. It picks questions randomly from the database by including the combination of all levels of difficulty. Questions once generated in one question paper will not be repeated in the other. Results will be evaluated.

Admin handles the database and faculty.

5.1 Code

5.1.1 index.php

```
<html>
<head>
<title>Online Examination</title>
<link rel="stylesheet" href="style.css">
</head>
<body>
<div class="login-page">
```

```

<div class="form" >

<!-- Registration Form -->

<form method="POST" action="registration.php"
class="registration-form">
<input id="first_name" type="text" name="first_name" placeholder="First Name" required/>
<input id="last_name" type="text" name="last_name" placeholder="Last Name" required/>
<input id="sid" type="text" name="student_id" placeholder="User Id" required/>
<input id="mail" type="text" name="email" placeholder="Email Id" required/>
<input id="type" type="text" name="type" placeholder="Type" value="student" readonly>
<input id="type" type="text" name="role" placeholder="Ex: 1-1" required>
<input id="pwd" type="password" name="pwd" placeholder="Password" required/>
<button type="submit" name="submit">Register </button><p class="message">Already
registered? <a href="#">Login </a></p>
</form>

<!-- Login Form -->

<form method="POST" class="login-form" action="login.php">
<input id="sid" type="text" name="student_id" placeholder="User Id" required/>
<input id="pwd" type="password" name="password" placeholder="Password" required/>
<button type="submit" name="submit">Login </button>
<p class="message">Not registered? <a href="#">Register </a></p>
</form>
</div>
</div>

<!-- jquery for animation-->

<script src='https://code.jquery.com/jquery-3.4.1.min.js'>
</script>
<script>
$('.message a').click(function(){
  $('form').animate({height: "toggle", opacity: "toggle"}, "slow");
});

```

```
</script>
</body>
</html>
```

5.1.2 insertquestions.php

```
<?
// Database configuration
require('config.php');

// start session
session_start();

if ($_SESSION['type'] != 'faculty') {
    header('Location:logout.php');
}

?>
<!doctype html>
<html>
<head>
<link type="text/css" rel="stylesheet" href =
"./admin/css/materialize.min.css" media="screen,projection" />
<script type="text/javascript" src="./admin/js/jquery.js"></script>
</head>
<body>
<form action="insertquestions1.php" method="POST" class="container">
<?php
if (isset($_POST['submit'])) {
    $no_of_questions = $_POST['questions'];
    $exam_name = $_POST['exam_name'];
    $subject = $_POST['subject'];
}
?>
<div class="row">
<div class="input-field col s4">
    <input id="no_of_questions" class="validate" type="text" value="<?php echo
$no_of_questions; ?>" name="no_of_questions" readonly>
```

```

<label for="no_of_questions">No Of Questions</label>
</div>
<div class="input-field col s4">
<input id="exam_name" type="text" class="validate" value=<?php echo
$exam_name; ?>" name="exam_name" readonly><br>
<label for="exam_name">Exam Name</label>
</div>
<div class="input-field col s4">
<input id="subject" type="text" class="validate" value=<?php echo $subject;
?>" name="subject" readonly><br>
<label for="subject">Subject</label>
</div>
</div>
<?php

// Enter the question details like question, options , answer, topic ,level of question
(i.e easy, medium, difficulty)
for ($i = 0; $i <(int) $no_of_questions; $i++)
?>
<div class="row">Question: <?php echo $i + 1; ?>
<div class="input-field">
<input type="text" name="question<?php echo $i; ?>" placeholder="Question"
required/>

</div>
<div class="input-field">
<input type="text" name="op1<?php echo $i; ?>" placeholder="Option 1"
required>
</div>
<div class="input-field">
<input type="text" name="op2<?php echo $i; ?>" placeholder="Option 2"
required>
</div>
<div class="input-field">
<input type="text" name="op3<?php echo $i; ?>" placeholder="Option 3"
required>

```

```

</div>
<div class="input-field">
<input type="text" name="op4<?php echo $i; ?>" placeholder=" Option 4"
required>
</div>
<select name="option<?php echo $i; ?>" class="browser-default" required>
<option value="0">Select Option</option>
<option value="1">A</option>
<option value="2">B</option>
<option value="3">C</option>
<option value="4">D</option>
</select>
<div class="input-field">
<input type="text" name="topic<?php echo $i; ?>" placeholder="Topic" re-
quired/>
</div>
<select name="qust_type<?php echo $i; ?>" class="browser-default" required>
<option value="0">Select Option</option>
<option value="easy">Easy</option>
<option value="medium">Medium</option>
<option value="difficulty">Hard</option>
</select>
</div>
<?php

}

?>
<br>
<button class="btn" name="submit">submit</button>
</form>
<script type="text/javascript" src=".//admin/js/materialize.min.js"></script>
</body>
</html>
```

5.1.3 getquestions.php

```
<?php
// Database configuration
require('config.php');

// start session
session_start();

// To check student is loggedin or not
if ($_SESSION['type'] != 'student') {
    header('Location:logout.php');
}
if (isset($_POST['submit'])) {
    $exam_name = $_POST['exam_name'];

    // select exam schema details based on schema name
    $schema = mysqli_query($conn, "SELECT * FROM pattern WHERE
exam_name='$exam_name'");
    $limit = mysqli_fetch_array($schema);

    // generate Random question paper based on schema
    $query = mysqli_query($conn, "SELECT q.* ,o.* FROM questions as q JOIN options
as o ON q.qid=o.qid AND q.exam_name='$limit[exam_name]' ORDER BY RAND()
LIMIT $limit[no_of_questions]");
    if(mysqli_num_rows($query)<1 && $_SESSION['type']=='student')
    {
        echo "<script>alert('Paper was not available');
window.location.href='facpage.php';</script>";
    }
    else if(mysqli_num_rows($query)<1 && $_SESSION['type']=='faculty')
    { echo "<script>
        alert('Paper was not available');window.location.href='create.php';</script>";
    }
    ?>
<html>
<head>
<link type="text/css" rel="stylesheet"
```

```

    href=”./admin/css/materialize.min.css” media=“screen,projection” />
<script type=“text/javascript” src=”./admin/js/jquery.js”></script>
</head>
<body>
<div class=“container”>

<!-- Show exam time -->
<div id=“quiz-time-left”></div>

<!-- Showresult Form -->
<form action=“showresult.php?q=<?php echo mysqli_num_rows($query);
?>&e=<?php echo $exam_name; ?>” method=“POST” name=“quiz” id=“quiz_form”>
<input type=“hidden” name=“val” value=“10”>
<?php
$i = 0;

// Display question paper
while ($row = mysqli_fetch_array($query)) {
?>
<p id=“display”>Question<?php echo $i + 1; ?><br><?php echo $row[‘question’];
?><input type=“hidden” name=“number<?php echo $i; ?>” value=“<?php
echo $row[‘qid’]; ?>” /></p>
<p>
<label>
<input name=“answer<?php echo $row[‘qid’]; ?>” value=“1” type=“radio”>
<span><?php echo $row[‘op1’]; ?></span>
</label>
<label>
<input name=“answer<?php echo $row[‘qid’]; ?>” value=“2” type=“radio”>
<span><?php echo $row[‘op2’]; ?></span>
</label>
<label>
<input name=“answer<?php echo $row[‘qid’]; ?>” value=“3” type=“radio”>
<span><?php echo $row[‘op3’]; ?></span>
</label>
<label>
<input name=“answer<?php echo $row[‘qid’]; ?>” value=“4” type=“radio”>
<span><?php echo $row[‘op4’]; ?></span>

```

```

</label>
</p>
<?php
$i++;
}
?>
<button class="btn" name="submit1">Submit</button>
</form>
</div>
<?php
}
?>

<!-- Exam time Count logic-->
<script>
var max_time = <?php echo $limit['minutes']; ?>;
var c_seconds = 0;
var total_seconds = 60 * max_time;
max_time = parseInt(total_seconds / 60);
c_seconds = parseInt(total_seconds % 60);
document.getElementById("quiz-time-left").innerHTML = 'Time Left: ' + max_time
+ ' minutes ' + c_seconds + ' seconds';
function init() {
    document.getElementById("quiz-time-left").innerHTML = 'Time Left: ' +
max_time + ' minutes ' + c_seconds + ' seconds';
    setTimeout("CheckTime()", 999);
}
function CheckTime() {
    document.getElementById("quiz-time-left").innerHTML = 'Time Left: ' +
max_time + ' minutes ' + c_seconds + ' seconds';
if (total_seconds <= 0) {
    setTimeout('document.quiz.submit()', 1);
} else {
    total_seconds = total_seconds - 1;
    max_time = parseInt(total_seconds / 60);
    c_seconds = parseInt(total_seconds % 60);
    setTimeout("CheckTime()", 999);
}

```

```

    }
init();
</script>
<script type="text/javascript" src="./admin/js/materialize.min.js"></script>
</body>
</html>

```

5.1.4 showresult.php

```

<?php

// Database configuration
require('config.php');

// start session
session_start();
if (isset($_POST['val'])) {

// get number of question value from URL
    $no_of_questions = $_GET['q'];

// get exam name value from URL
    $exam_name = $_GET['e'];
    $count = 0;
    $wrong = 0;

// loop questions based on number of questions
    for ($i = 0; $i < $no_of_questions; $i++) {
        $number = $_POST['number' . (string) $i];
        if (isset($_POST['answer' . $number])) {

// selected option
        $ans = $_POST['answer' . $number];

// get question options based on question id
        $query = mysqli_query($conn, "SELECT * FROM options WHERE qid =

```

```

'$number"') or die(mysqli_error($conn));
while ($rows = mysqli_fetch_array($query)) {

// check selected option is correct or not

// if option is correct increment the count value
if ($rows['answer'] == $ans) {
    $count++;
}

// if option is wrong increment the wrong value
else
    $wrong++;
}

}

} else {
    $wrong++;
}

}

// Insert the final result into database with specified fields
mysqli_query($conn, "INSERT INTO results(student_id,no_of_questions,exam_name,
correct,wrong,marks)VALUES('$_SESSION[std_id]',$no_of_questions,
'$exam_name',$count,$wrong,$count") or die(mysqli_error($conn));
}

?>
<html>
<head>
<link type="text/css" rel="stylesheet" href=".//admin/css/materialize.min.css"
media="screen,projection" />
<script type="text/javascript" src=".//admin/js/jquery.js"></script>
</head>
<body class="container center" style="padding-top:14%;">

<!-- Display the Result -->
<table class="striped" border="1" width="100%" style="text-align:center;">
<thead>
<tr>

```

```

<th>Number of Questions</th>
<th>Correct Questions</th>
<th>Wrong Questions</th>
<th>Total Score</th>
</tr>
</thead>
<tbody>
<tr>
<td><?php echo $count + $wrong ?></td>
<td><?php echo $count; ?></td>
<td><?php echo $wrong; ?></td>
<td><?php echo $count; ?></td>
</tr>
</tbody>
</table>
<button class="btn" style="margin-top:10px;" onclick="closed()>Close</button>
<script>
function closed() {
    window.location.href = 'action.php';
    console.log('clicked');
}
</script>
<script type="text/javascript" src=".//admin/js/materialize.min.js"></script>
</body>
</html>

```

5.2 Testing

We all have to agree that in today's ever-changing and competitive world, the internet has become an integral part of our lives. Most of us make our decisions by searching the information on the internet these days, hence hosting a website is no[9] longer optional but mandatory for all kind of businesses. It is the first step in becoming and staying relevant in the market.

Just having a website is not enough. An organization is needed to develop a website that is informative, accessible and user-friendly. To maintain all these qualities,

the website should be well tested, and this process of testing a website is known as web testing.

What Is Web Testing?

Web testing is a software testing practice to test websites or web applications for potential bugs. It's a complete testing of web-based applications before making live. A web-based system needs to be checked completely from end-to-end before it goes live for end users. By performing website testing, an organization can make sure that the web-based system is functioning properly and can be accepted by real-time users. The UI design and functionality are the captains of website testing.

Web Testing Checklists

- 1) Functionality testing
- 2) Usability testing
- 3) Interface testing
- 4) Security testing

5.2.1 Functionality Testing

Test for – all the links in web pages, database connection, forms used for submitting or getting information from the user in the web pages.

Check all the links:

- Test the outgoing links from all the pages to the specific domain under test.
- Test all internal links.
- Test links jumping on the same pages.
- Test links used to send email to admin or other users from web pages.
- Test to check if there are any orphan pages.
- Finally, link checking includes, check for broken links in all the above-mentioned links.

Test forms on all pages:

Forms are an integral part of any website. Forms are used for receiving information from users and to interact with them. So what should be checked in these forms?

- First, check all the validations on each field.
- Check for default values of the fields.
- Wrong inputs in the forms to the fields in the forms.
- Options to create forms if any, form delete, view or modify the forms.

Validate your HTML/CSS:

If you are optimizing your site for Search engines then HTML/CSS validation is the most important one. Mainly validate the site for HTML syntax errors.

Database Testing:

Data consistency is also very important in a web application. Check for data integrity and errors while you edit, delete, modify the forms or do any DB related functionality. Check if all the database queries are executing correctly, data is retrieved and also updated correctly.

5.2.2 Usability Testing

Usability testing is the process by which the human-computer interaction characteristics of a system are measured, and weaknesses are identified for correction. Usability Testing includes the following:

- The website should be easy to use.
- The instructions provided should be very clear.
- Check if the instructions provided are perfect to satisfy its purpose.

- The main menu should be provided on each page.
- It should be consistent enough.

5.2.3 Interface Testing

In web testing, the server-side interface should be tested. This is done by verifying that communication is done properly. Compatibility of the server with software, hardware, network, and the database should be tested.

The main interfaces are:

- Web server and application server interface.
- Application server and Database server interface.

Check if all the interactions between these servers are executed and errors are handled properly. If the database or web server returns an error message for any query by application server then the application server should catch and display these error messages appropriately to the users.

5.2.4 Security Testing

The primary reason for testing the security of a web is to identify potential vulnerabilities and subsequently repair them.

Following are some of the test cases for web security testing:

- Test by pasting the internal URL directly into the browser address bar without login. Internal pages should not open.
- Try some invalid inputs in input fields like login username, password, input text boxes, etc. Check the system's reaction to all invalid inputs.

Chapter 6

SCREENSHOTS

6.1 Admin Module

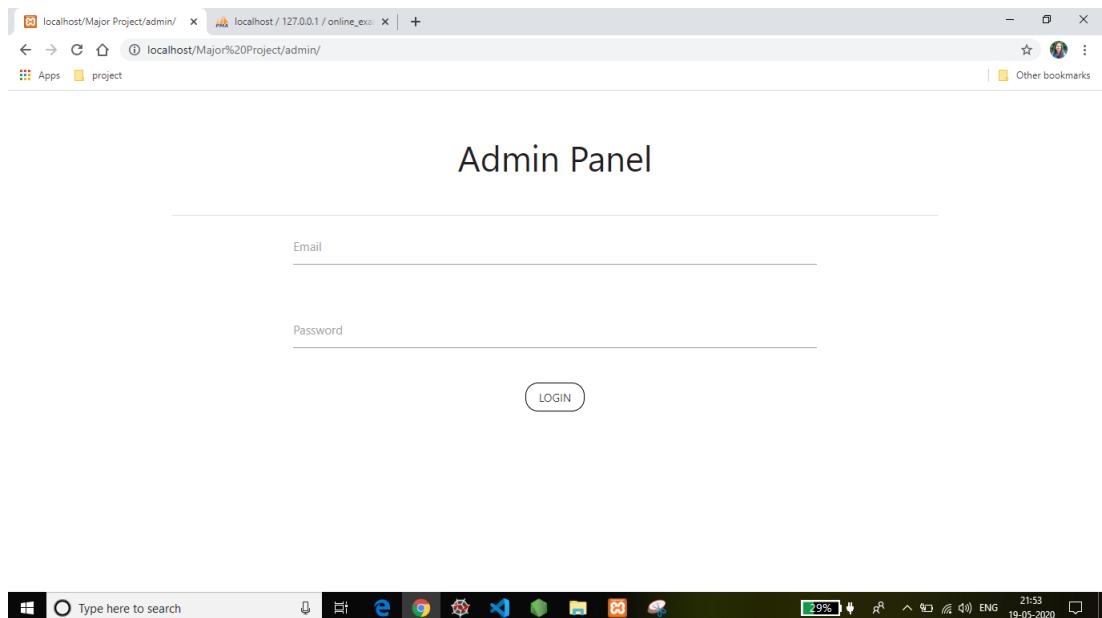


Figure 6.1: Admin panel to login

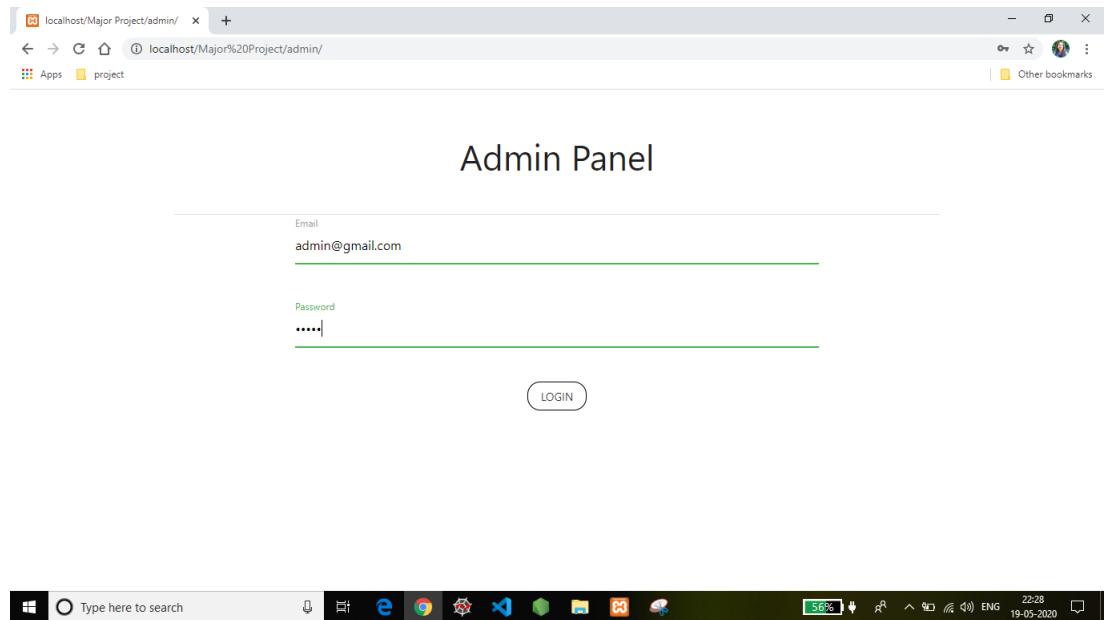


Figure 6.2: Admin login

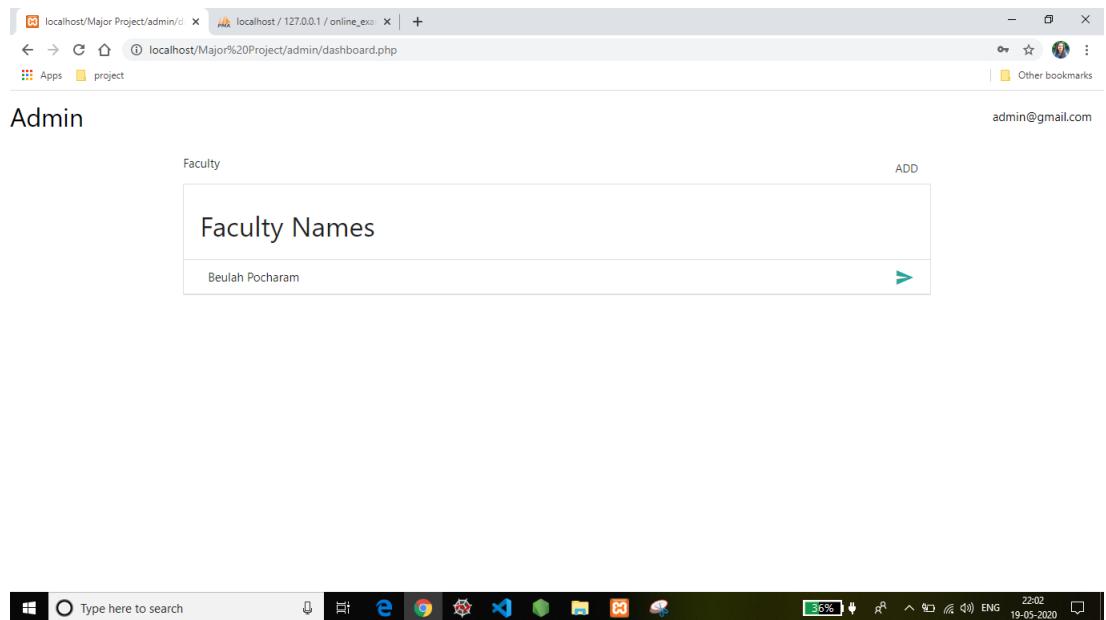


Figure 6.3: Existing faculty list

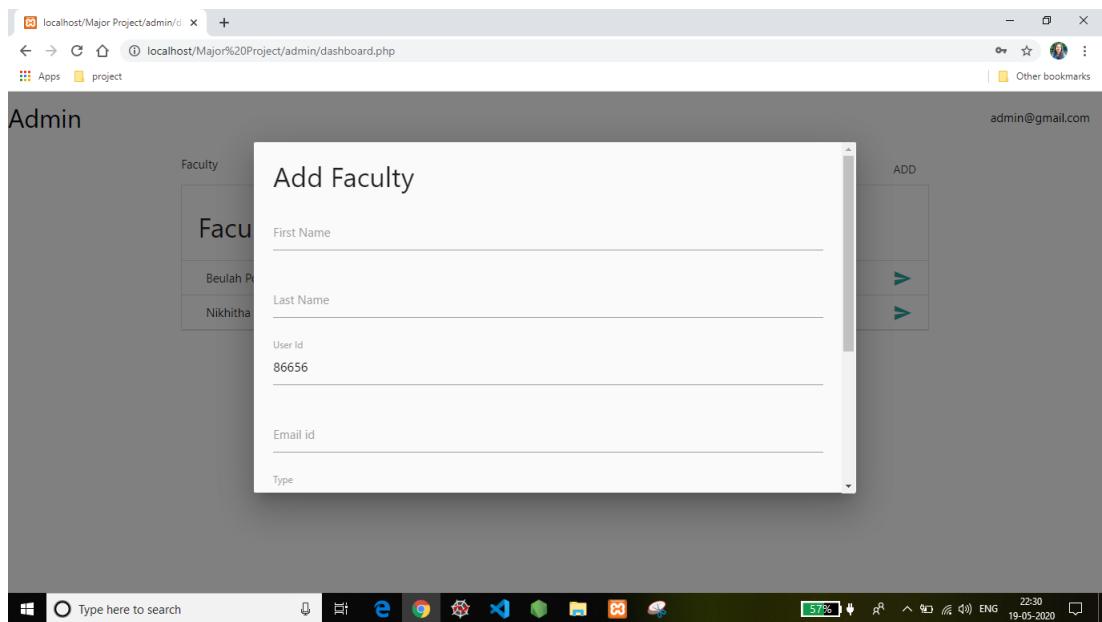


Figure 6.4: page to add faculty

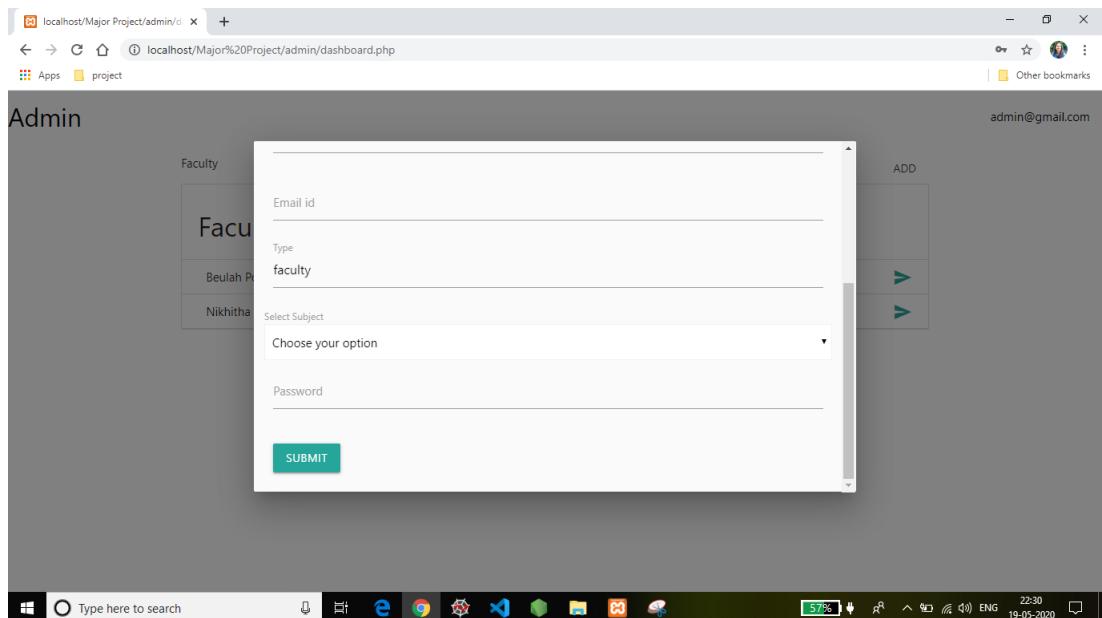


Figure 6.5: page to add faculty

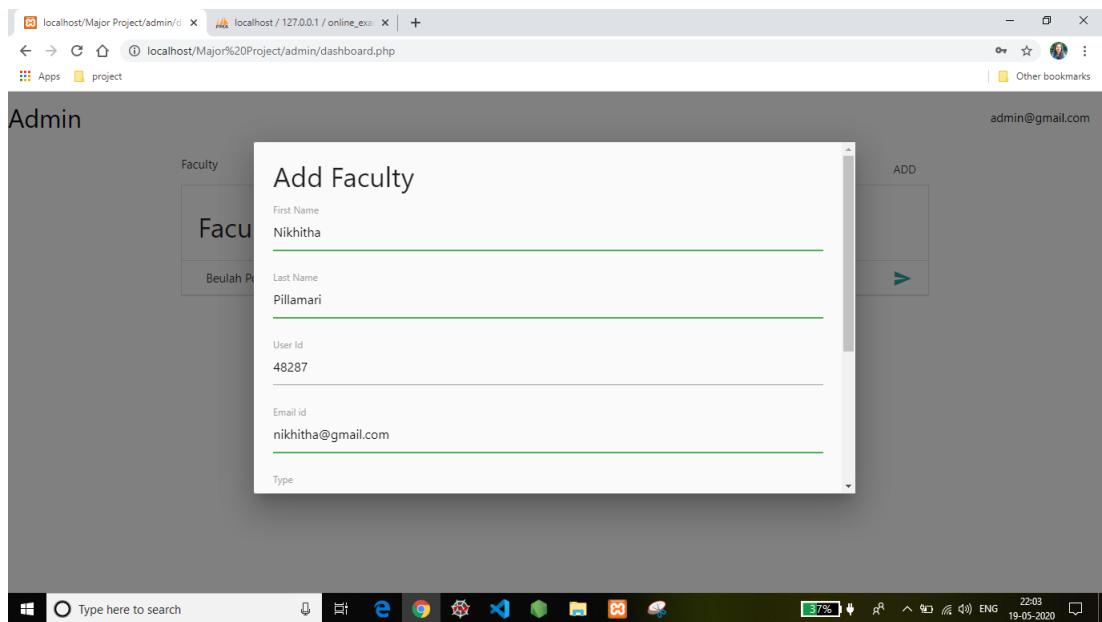


Figure 6.6: Giving faculty details

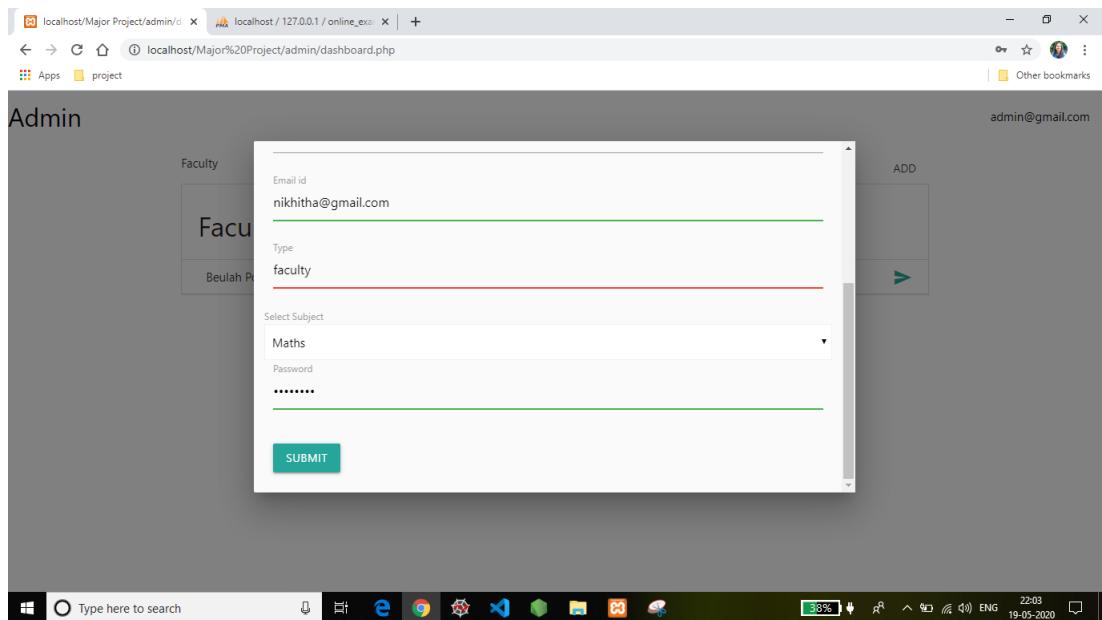


Figure 6.7: Giving faculty details

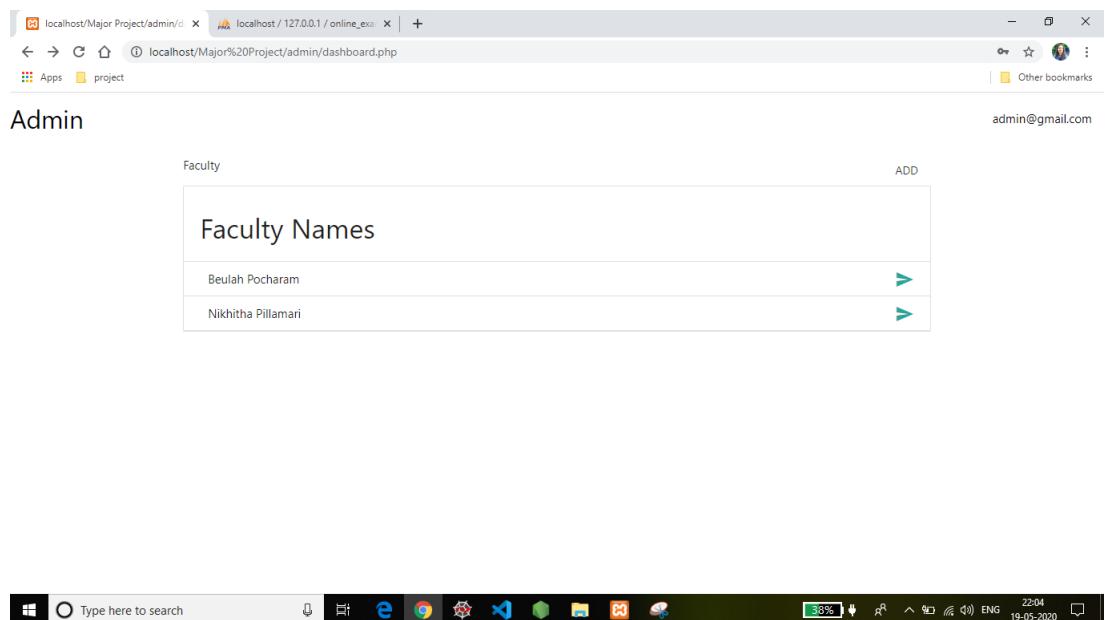


Figure 6.8: showing newly added faculty

6.2 Faculty Module

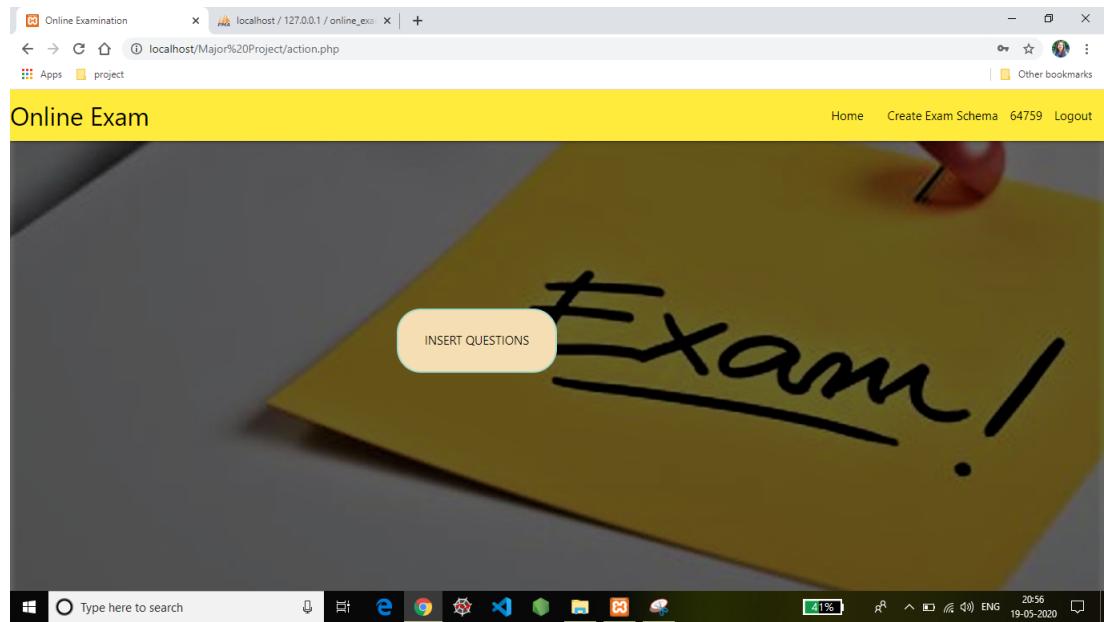


Figure 6.9: Faculty main page

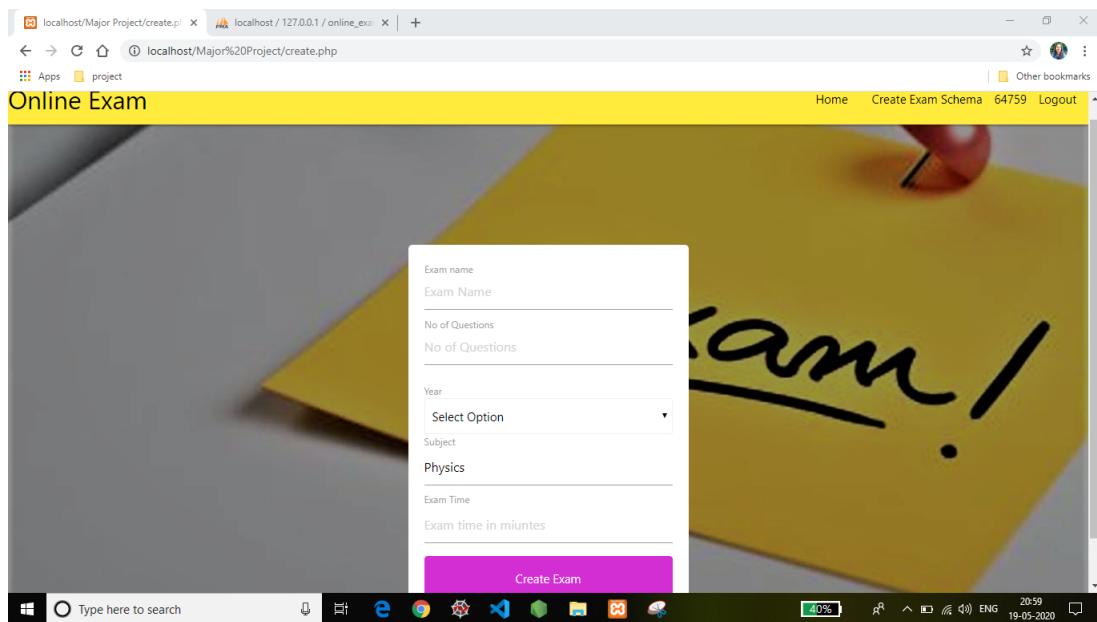


Figure 6.10: Creating exam schema

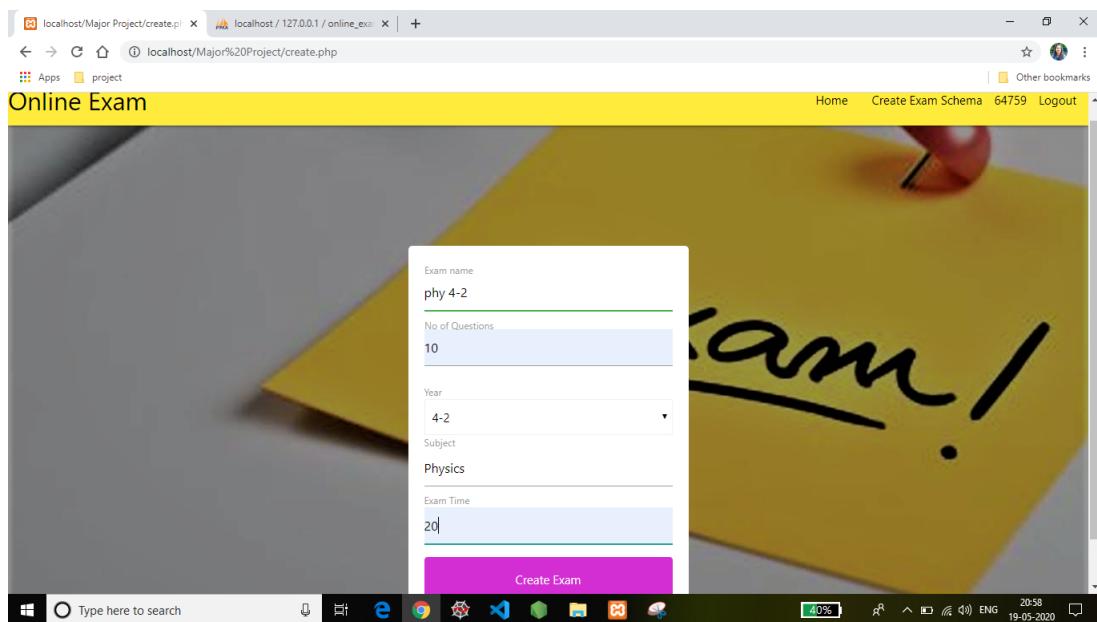


Figure 6.11: Giving details of exam schema/paper

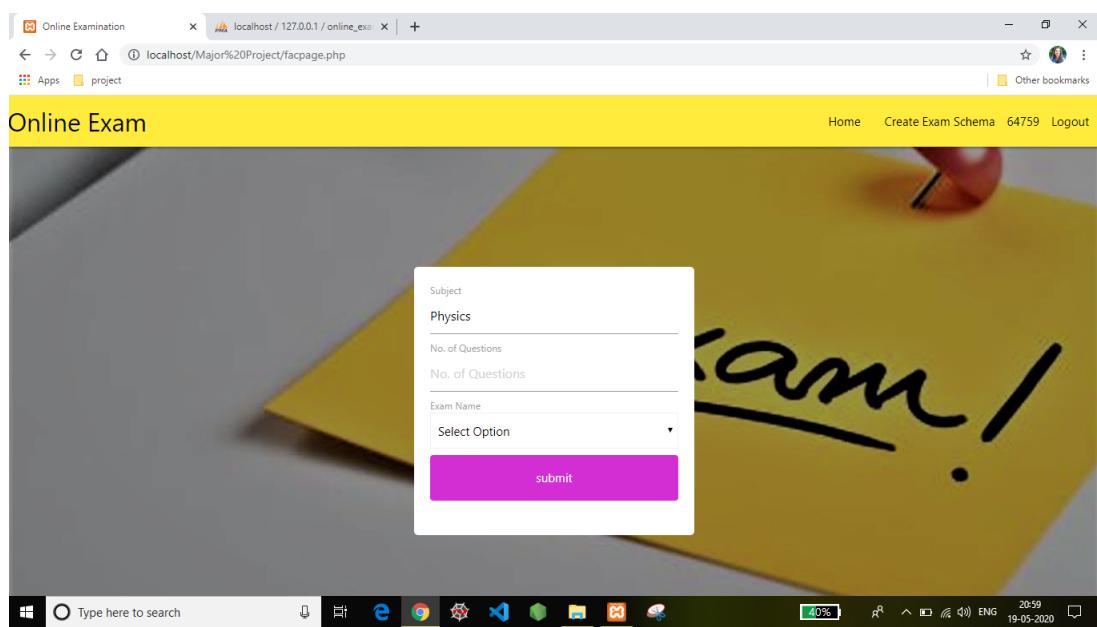


Figure 6.12: page to enter exam details

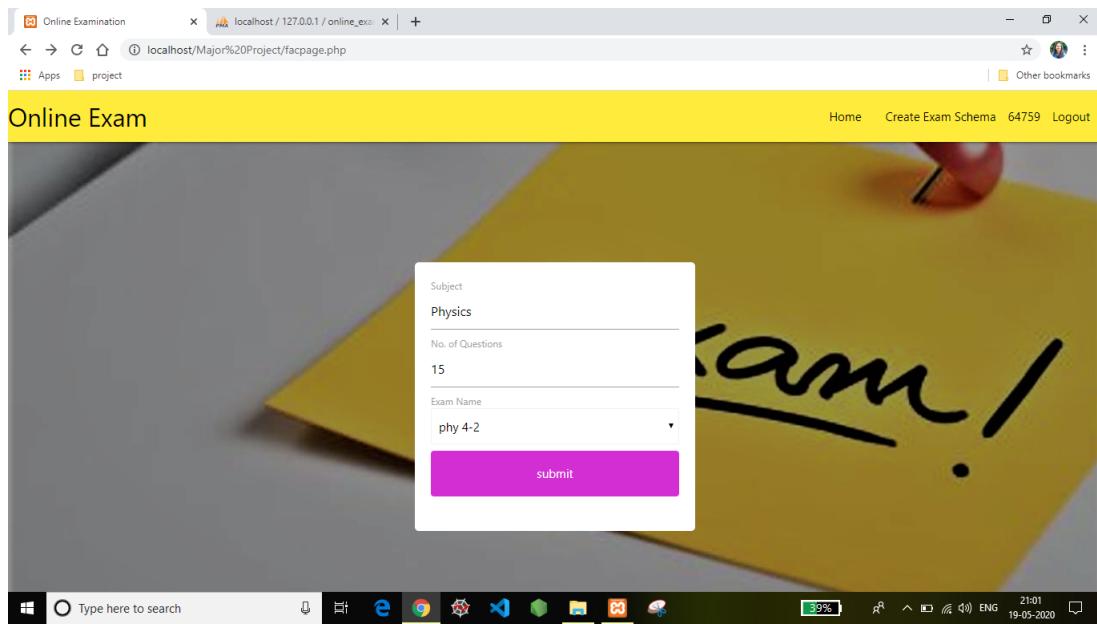


Figure 6.13: page to enter no.of questions,exam name in-order to enter questions into database

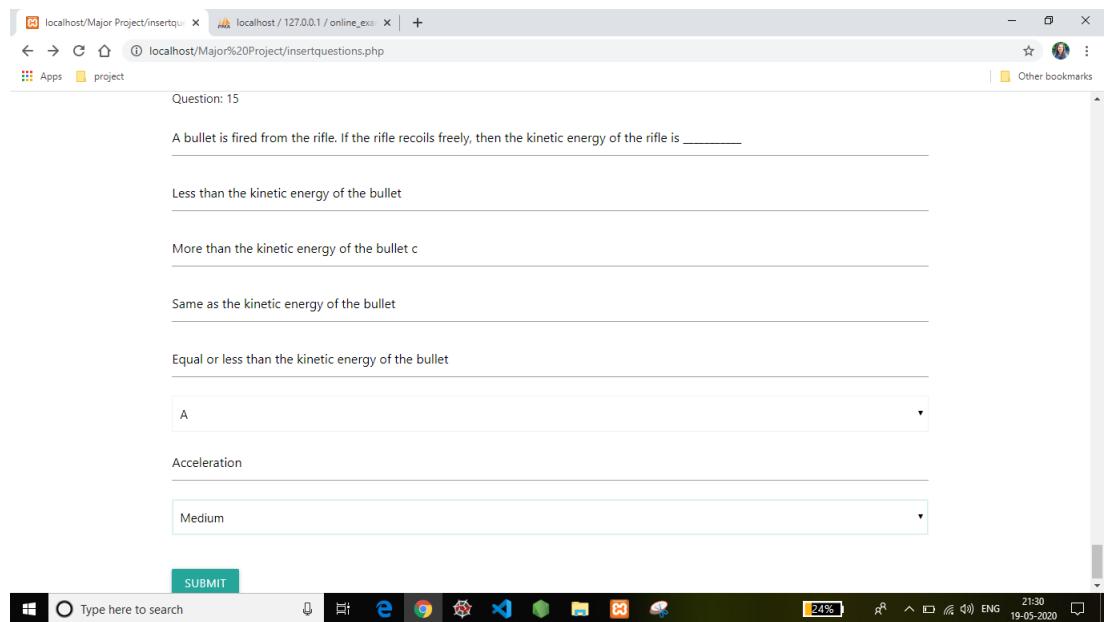


Figure 6.14: Inserting questions into database

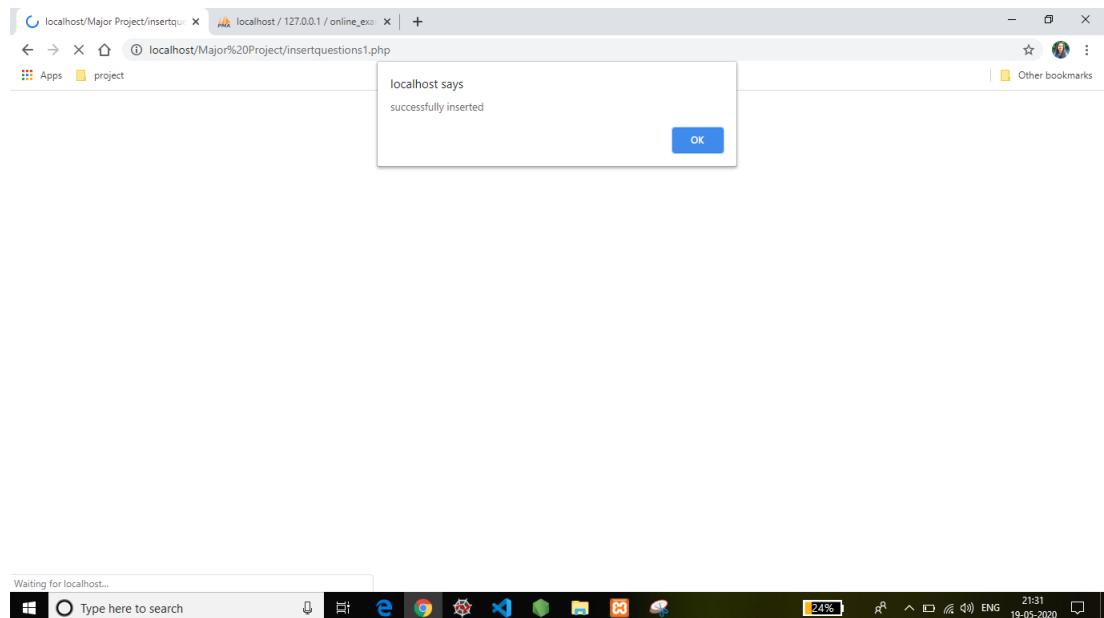


Figure 6.15: Message showing that questions are successfully inserted

6.3 Student Module

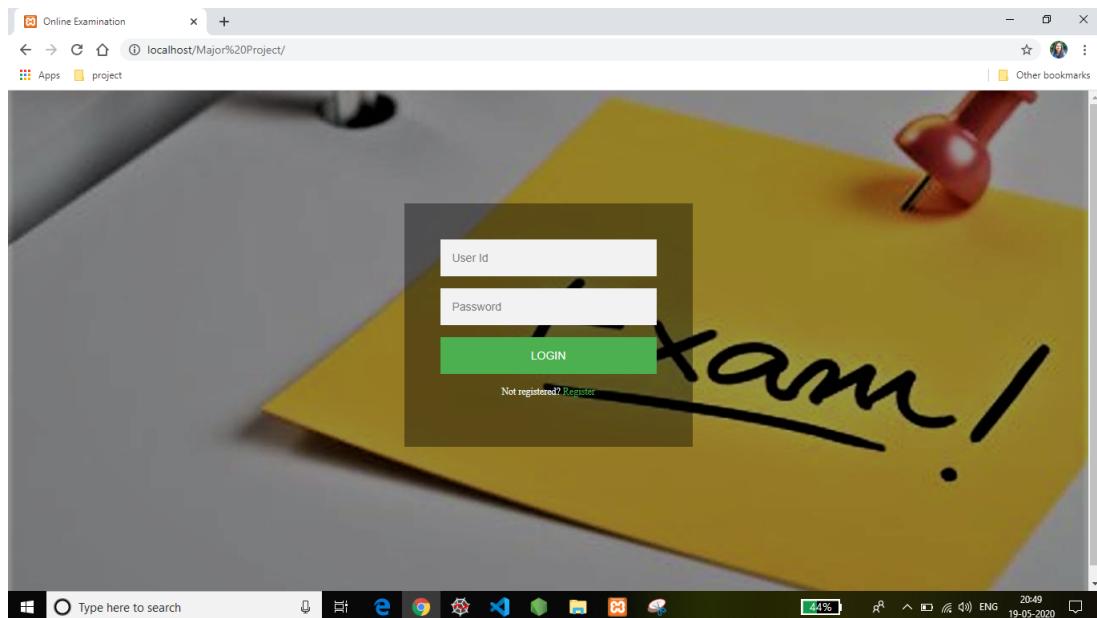


Figure 6.16: user(student/faculty) login page

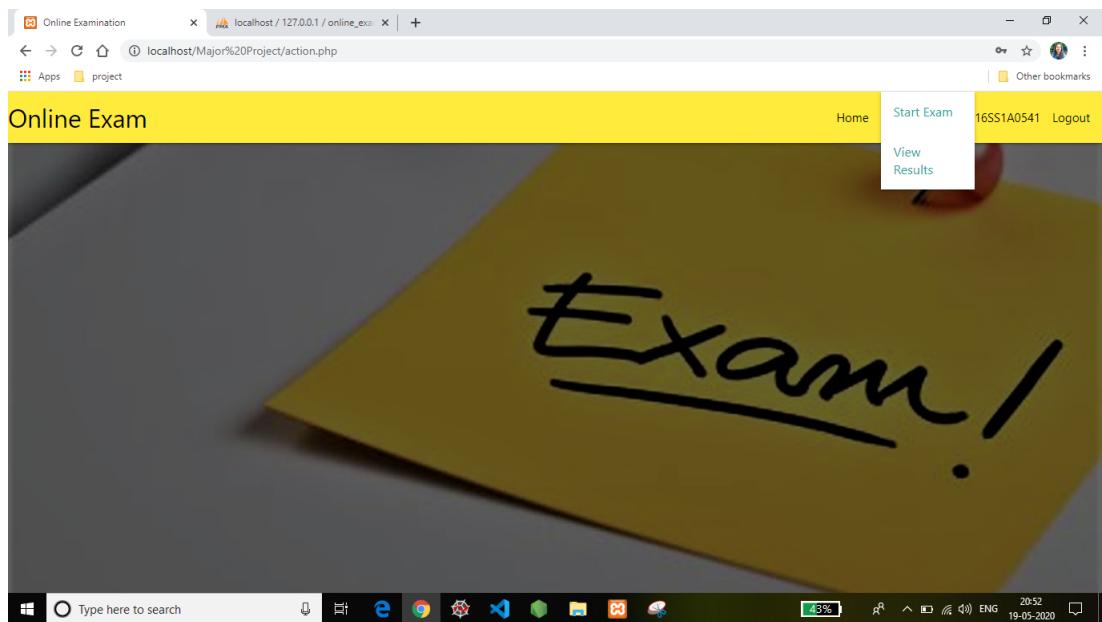


Figure 6.17: student action page

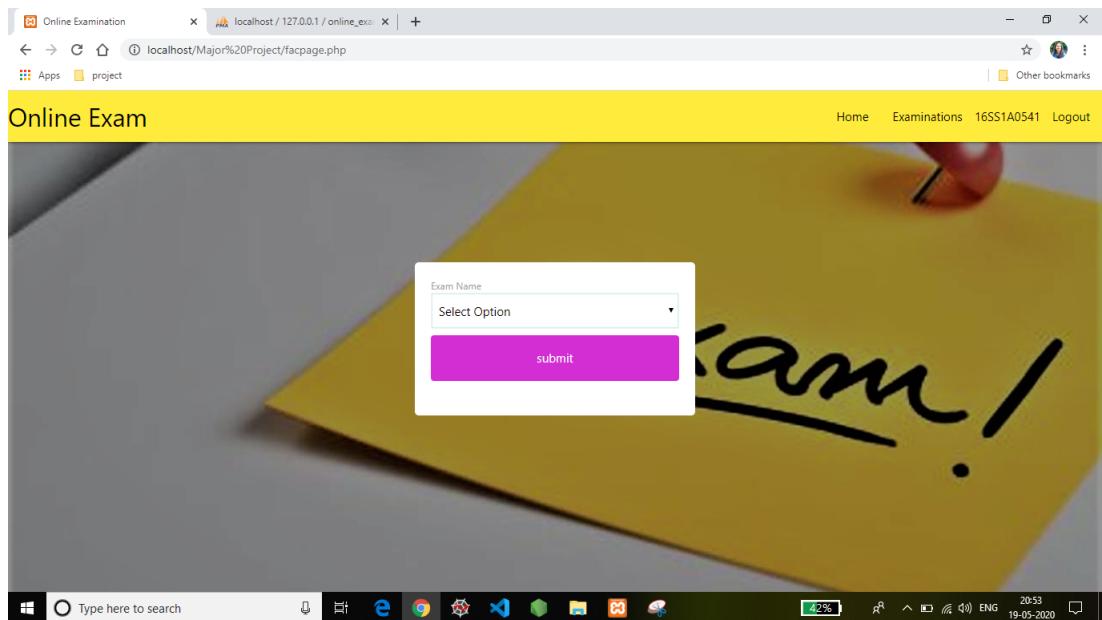


Figure 6.18: page for selecting exam

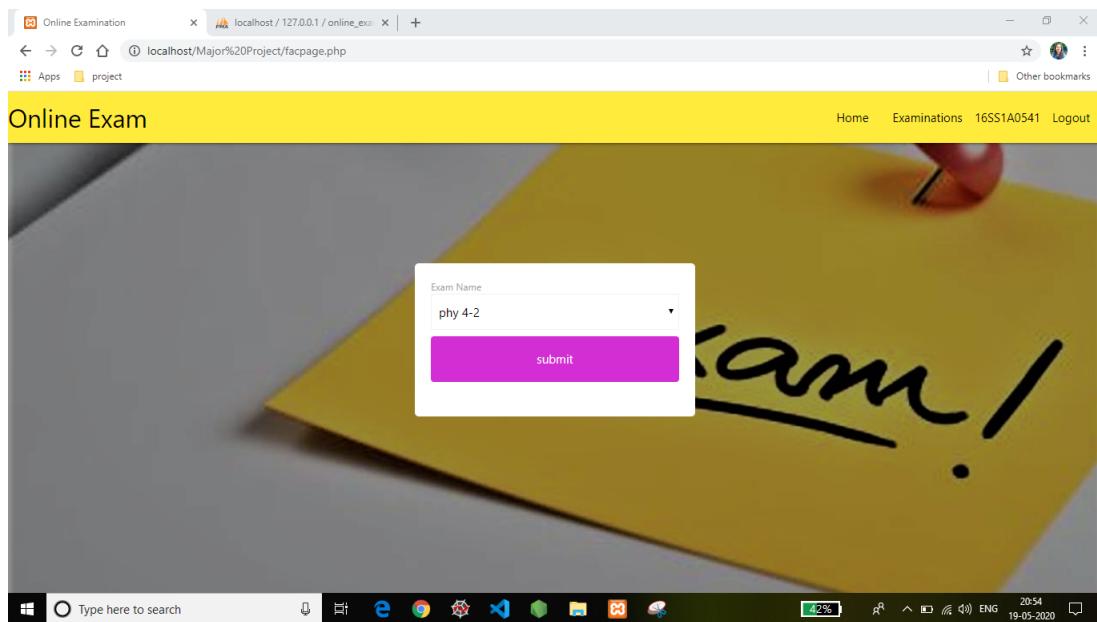


Figure 6.19: selecting exam

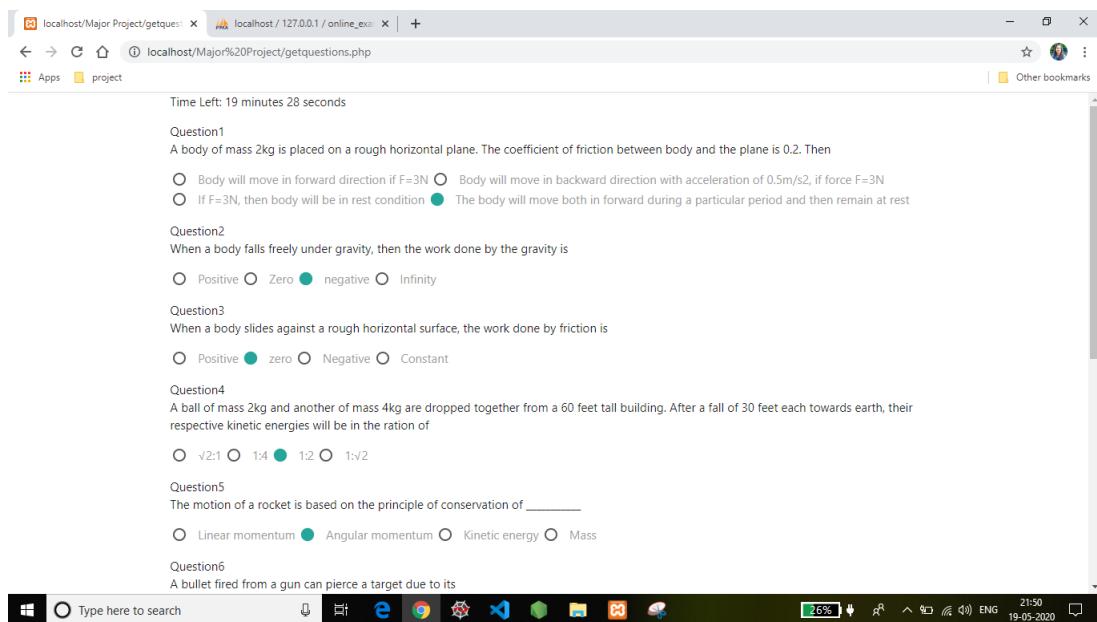


Figure 6.20: Exam with timer

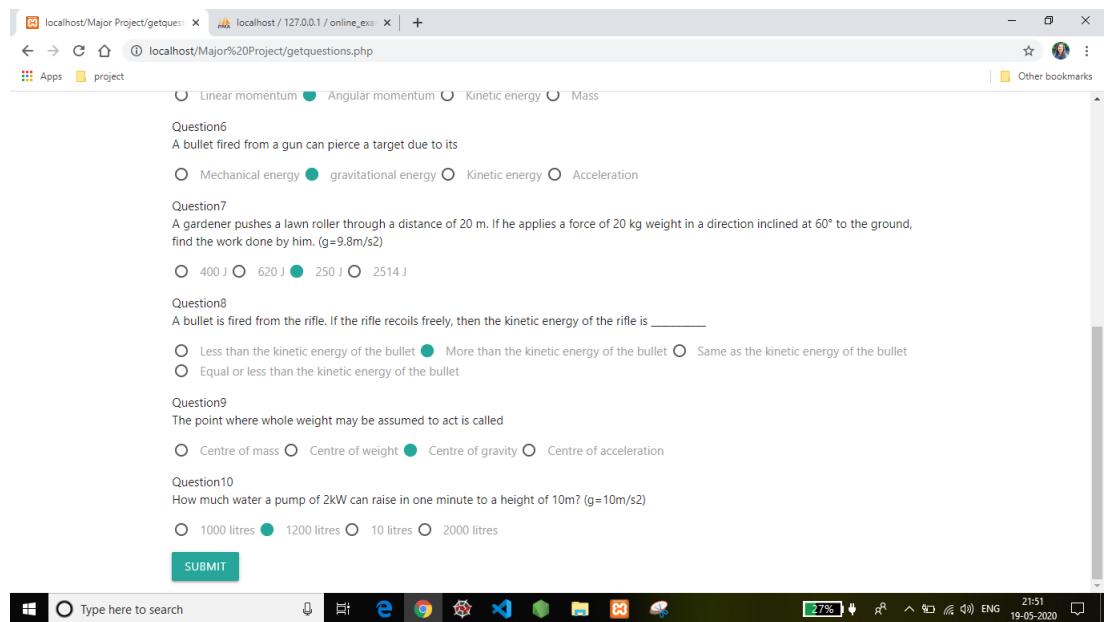


Figure 6.21: submitting the exam

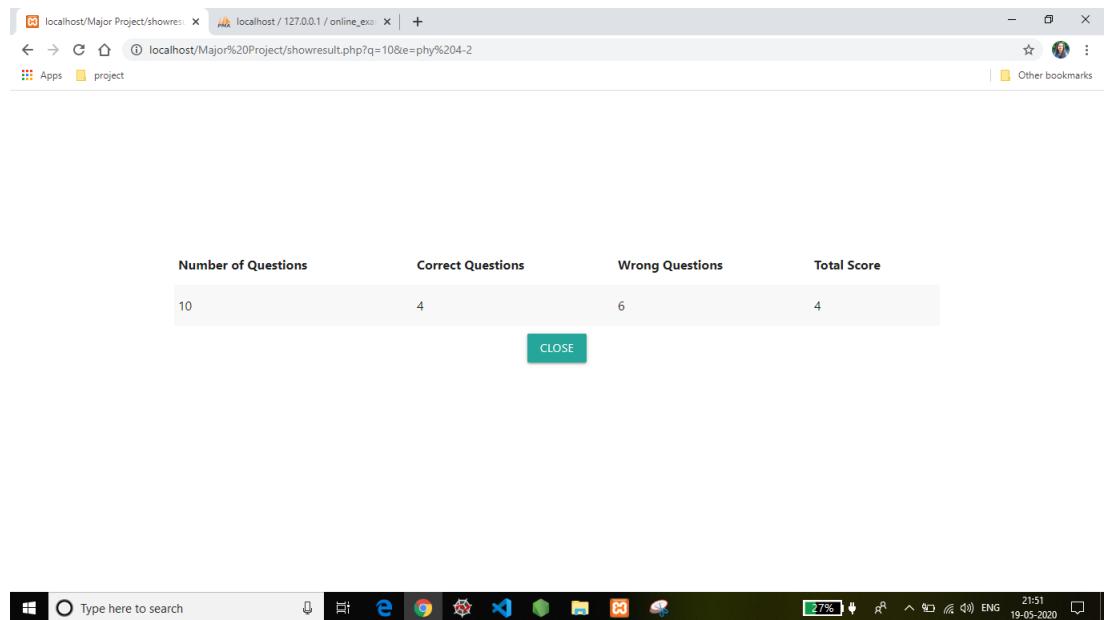


Figure 6.22: showing results at the end of exam

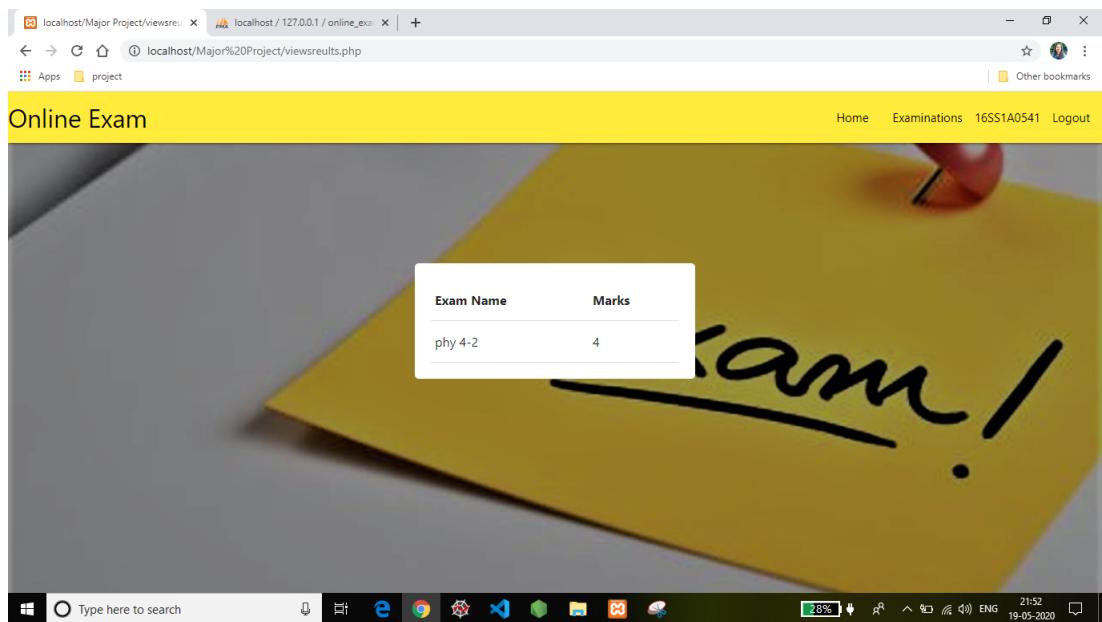


Figure 6.23: view results page

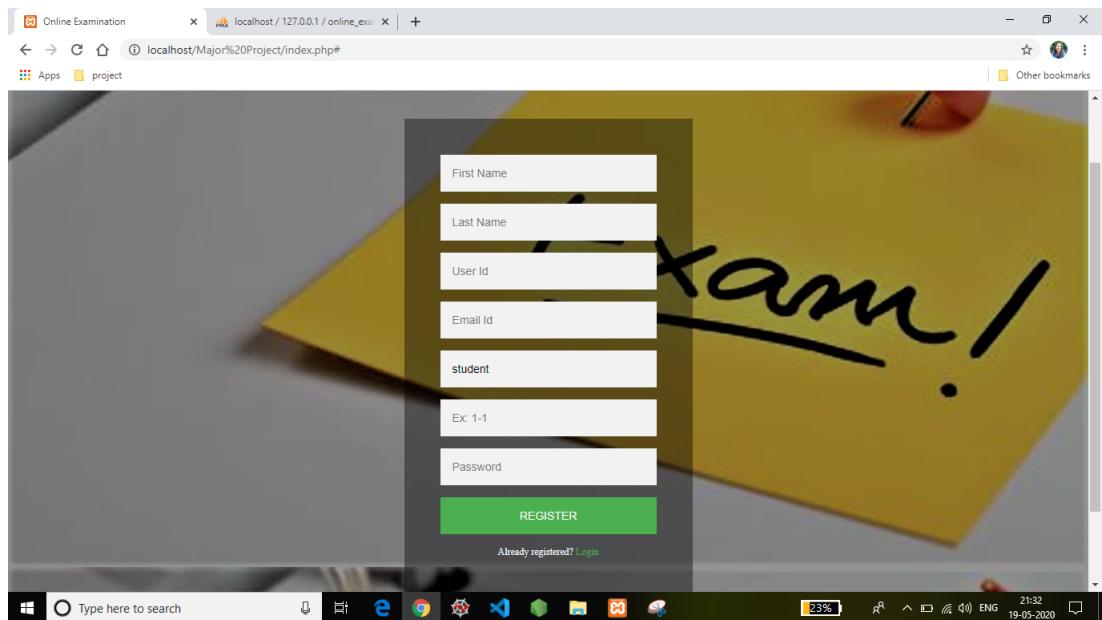


Figure 6.24: Student registration page

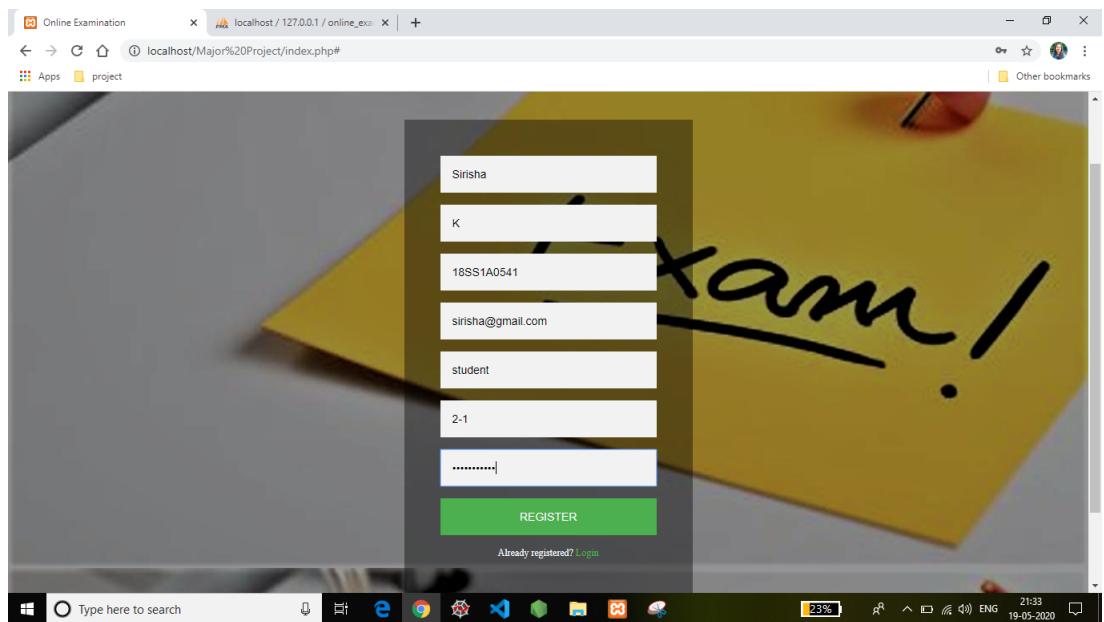


Figure 6.25: Entering student details

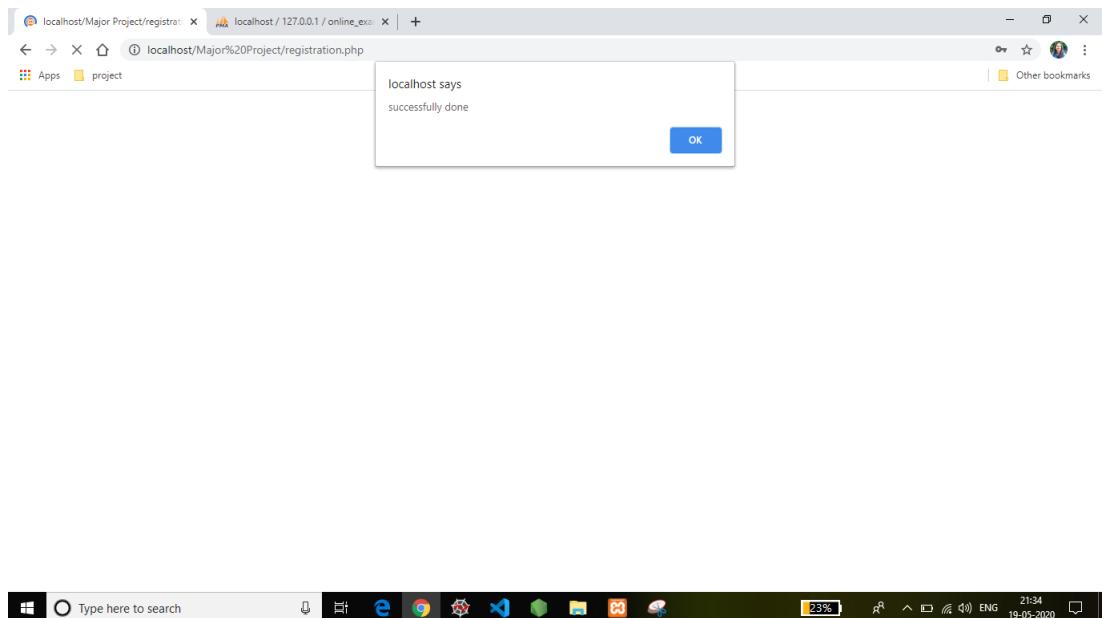


Figure 6.26: message showing successful registration

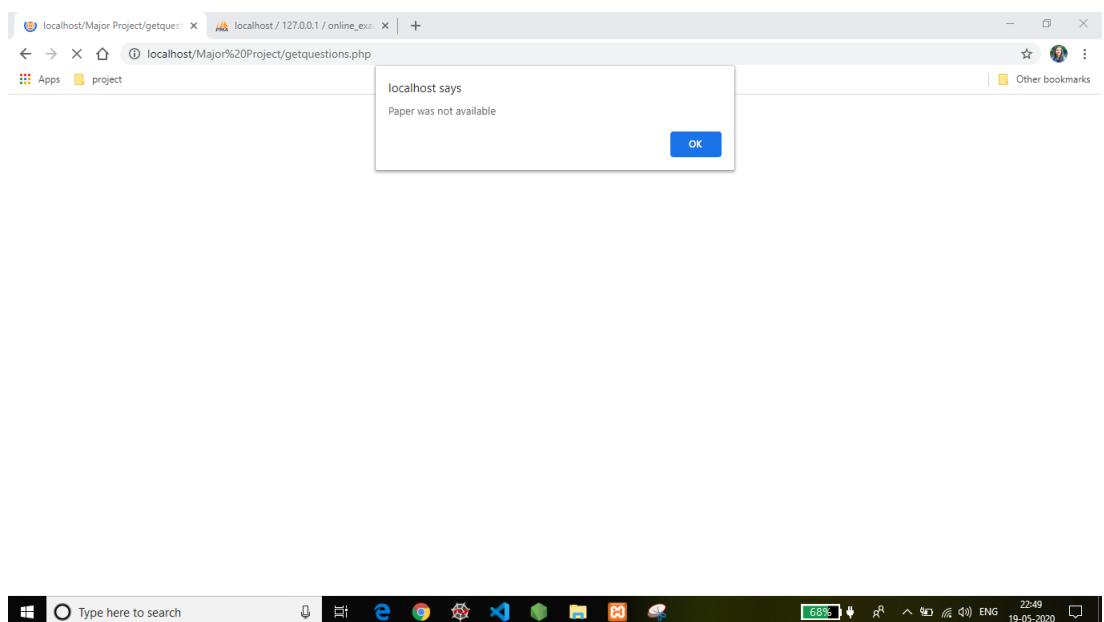


Figure 6.27: message if question paper does not exist

6.4 Database

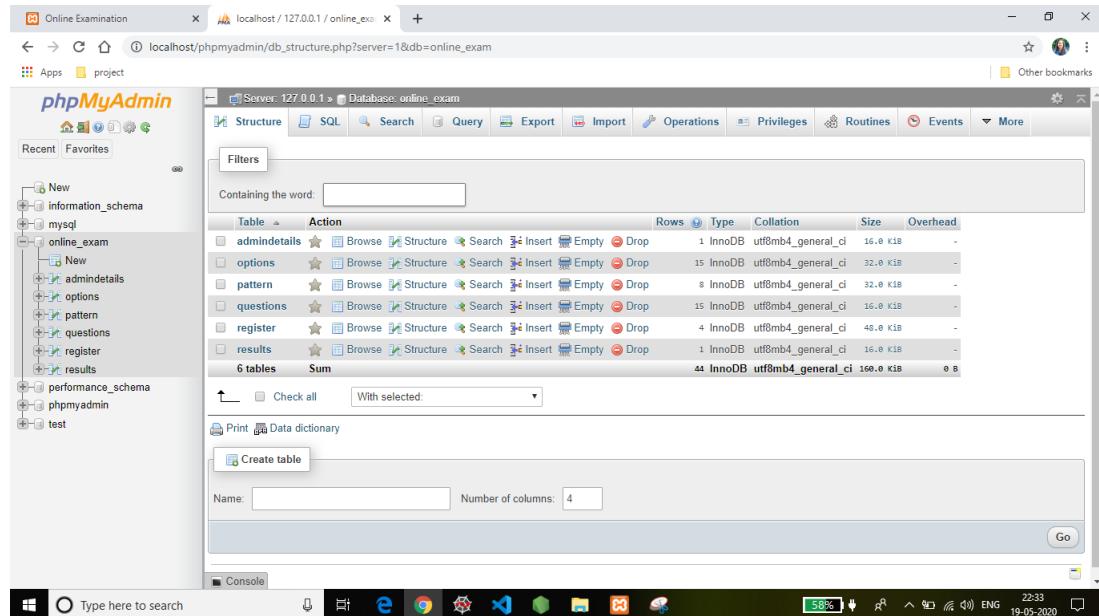


Figure 6.28: online exam database

Showing rows 0 - 0 (1 total, Query took 0.0011 seconds.)

```
SELECT * FROM `admindetails`
```

	Edit	Copy	Delete	id	email	password	first_name	last_name	phone
	Edit	Copy	Delete	2512468	admin@gmail.com	admin	Befalah	Pocharam	9876543210

Figure 6.29: Admin details table

Showing rows 0 - 14 (15 total, Query took 0.0011 seconds.)

```
SELECT * FROM `options`
```

	Edit	Copy	Delete	id	qid	op1	op2	op3	op4	answer
	Edit	Copy	Delete	48	73709	Positive	Zero	negative	Infinity	1
	Edit	Copy	Delete	49	60669	Positive	zero	Negative	Constant	3
	Edit	Copy	Delete	50	50512	400 J	620 J	250 J	2514 J	2
	Edit	Copy	Delete	51	79931	50J	75J	100J	200J	3
	Edit	Copy	Delete	52	39223	Mechanical energy	gravitational energy	Kinetic energy	Acceleration	3
	Edit	Copy	Delete	53	47855	Moves along a straight line path	Moves in zig-zag path	Is thrown away	Rotates about its centre of mass	4
	Edit	Copy	Delete	54	73477	Centre of mass	Centre of weight	Centre of gravity	Centre of acceleration	3
	Edit	Copy	Delete	55	57439	Rotation body	Stationary body	Rigid body	A body in equilibrium	3
	Edit	Copy	Delete	56	50035	$\sqrt{2}$ 1	1.4	1.2	1. $\sqrt{2}$	3
	Edit	Copy	Delete	57	13658	1000 litres	1200 litres	10 litres	2000 litres	2
	Edit	Copy	Delete	58	35494	Body will move in forward direction if F=3N	Body will move in backward direction if F=3N	If F=3N, then body will be in rest condition	The body will move both in forward during a partic...	4
								1.5kq	0.5kq	3

Figure 6.30: multiple choices(options) table

Showing rows 0 - 11 (12 total). Query took 0.0009 seconds.

```
SELECT * FROM `pattern`
```

	id	exam_name	year	no_of_questions	subjects	minutes	faculty_id
Edit	6	phy 1-1	1-1	10	Physics	20	64759
Edit	7	phy 1-2	1-2	10	Physics	20	64759
Edit	8	phy 2-1	2-1	10	Physics	20	64759
Edit	9	phy 2-2	2-2	10	Physics	20	64759
Edit	10	phy 3-1	3-1	10	Physics	20	64759
Edit	11	phy 3-2	3-2	10	Physics	20	64759
Edit	12	phy 4-1	4-1	10	Physics	20	64759
Edit	13	phy 4-2	4-2	10	Physics	20	64759
Edit	14	math 1-1	1-1	10	Maths	30	48287
Edit	15	math 1-2	1-2	10	Maths	30	48287
Edit	16	math 2-1	2-1	10	Maths	30	48287
Edit	17	math 2-2	2-2	10	Maths	30	48287

Figure 6.31: list of question papers table

Showing rows 0 - 14 (15 total). Query took 0.0010 seconds.

```
SELECT * FROM `questions`
```

	id	qid	question	sub	topic	question_type	exam_name
Edit	69	73709	When a body falls freely under gravity, then the w...	Physics	work, power, energy	easy	phy 4-2
Edit	70	60669	When a body slides against a rough horizontal surf...	Physics	Work, power, energy	easy	phy 4-2
Edit	71	50512	A gardener pushes a lawn roller through a distance...	Physics	WPE	medium	phy 4-2
Edit	72	79931	A person is holding a bucket by applying a force o...	Physics	WPE	medium	phy 4-2
Edit	73	39223	A bullet fired from a gun can pierce a target due ...	Physics	WPE	easy	phy 4-2
Edit	74	47855	A hammer is resting on a plane surface. If force P...	Physics	Force & Inertia	easy	phy 4-2
Edit	75	73477	The point where whole weight may be assumed to act...	Physics	Gravitation	easy	phy 4-2
Edit	76	57439	Which of the following retain their relative posit...	Physics	Gravitation	medium	phy 4-2
Edit	77	50035	A ball of mass 2kg and another of mass 4kg are dro...	Physics	Gravitation	difficulty	phy 4-2
Edit	78	13658	How much water a pump of 2kW can raise in one minu...	Physics	Gravitation	medium	phy 4-2
Edit	79	35494	A body of mass 2kg is placed on a rough horizontal...	Physics	Gravitation	medium	phy 4-2
Edit	80	44030	A gun fires a bullet of mass 50g with a velocity o...	Physics	Acceleration	difficulty	phy 4-2
Edit	81	25897	Rocket engines lift a rocket from the earth surfa...	Physics	Acceleration	medium	phy 4-2

Figure 6.32: list of questions table

The screenshot shows the phpMyAdmin interface for the 'online_exam' database. The left sidebar lists databases and tables, with 'register' selected. The main area displays the 'register' table results. The table has columns: id, first_name, last_name, student_id, type, email, pwd, role, and year. The data shows 14 entries, including Beulah Pocharam (faculty), Nancy P (student), Sirisha K (student), Nikhitha Pillamari (faculty), Poja T (student), and Bhargavi S (student).

	id	first_name	last_name	student_id	type	email	pwd	role	year
1	9	Beulah	Pocharam	64759	faculty	beulah@gmail.com	beulah	Physics	
2	10	Nancy	P	16SS1A0541	student	nancy@gmail.com	nancy@123	4-2	
3	11	Sirisha	K	18SS1A0541	student	sirisha@gmail.com	sirisha@123	2-1	
4	12	Nikhitha	Pillamari	48287	faculty	nikhitha@gmail.com	nikhitha	Maths	
5	13	Poja	T	19SS1A0526	student	poja@gmail.com	poja@123	1-1	
6	14	Bhargavi	S	17SS1A0536	student	bhargavi@gmail.com	bhargavi@123	3-1	

Figure 6.33: Faculty and student details table

The screenshot shows the phpMyAdmin interface for the 'online_exam' database. The left sidebar lists databases and tables, with 'results' selected. The main area displays the 'results' table results. The table has columns: id, student_id, no_of_questions, exam_name, correct, wrong, and marks. The data shows one entry for student_id 16SS1A0541, who took the phy exam with 10 questions, 4 correct, and 6 wrong, resulting in 4 marks.

	id	student_id	no_of_questions	exam_name	correct	wrong	marks
1	3422	16SS1A0541	10	phy	4	6	4

Figure 6.34: students results table

CONCLUSION

This Web Application provides facility to take online examination worldwide. It saves the time as it allows number of students to give the exam at a time, and displays the results of the exam when exam is completed. So no need to wait for the result. It is automatically done by the server. Admin can manage the database and Faculty has a privilege to create, modify and delete the test papers and its particular questions. User can register to exam system, login and give the exam with his specific id, and can see the results as well.

The website is designed in such a way that future modifications can be done easily. The following conclusions can be deduced from the development of the project. Automation of the entire system improves the efficiency. It provides a friendly graphical user interface which proves to be better when compared to the existing system. It gives appropriate access to the authorized users depending on their permissions. It effectively overcomes the delay in communications. Updating of information becomes so easier. System security, data security and reliability are the striking features. The System has adequate scope for modification in future if it is necessary.

REFERENCES

[1]*https://www.php.net*

[2]*https://www.yourhtmlsource.com*

[3]*https://en.wikipedia.org*

[4]*https://www.w3schools.com*

[5]*https://www.materializecss.com*

[6]*http://tomcat.apache.org*

[7]*https://www.uml.org*

[8]*https://ajprofessionals.googlepages.com*

[9]*https://www.softwaretestinghelp.com*