A Project Report On

"Online Exam Management System"

Submitted by

Mistry Mohit Arvindbhai

Under the Guidance of

Prof. Dr. Jinal Tailor

In fulfilment for the award of the degree

Of

Master of Computer Applications

(2nd semester)

In

Computer Science Department

S.S Agrawal Institute of Management and Technology

Opp vidya kunj School, Nr Devina Park Soc, Gandevi Rd, Navsari

Affiliated Under

Gujarat Technological University, Ahmedabad

July 2023

S.S Agrawal Institute of Management and Technology

Master of Computer Application Department

2023

CERTIFICATE

This is to certify that the project entitled **Online Exam Management System** has been carried out by **Mistry Mohit Arvindbhai** (228060694025) under my guidance in fulfilment of the degree of Master of Computer Applications (MCA 2nd Semester) of Gujarat Technological University, Ahmedabad during the academic year 2023.

Date:

Guide Name& Signature: Prof. _____

Examiners' Signature:

1._____

2._____

3._____

Prof Jinal Tailor

Head of Department

SSAIMT, Navsari

Preface

Knowledge is power. But Power on its own cannot be a key to success. It needs to be harnessed and channelized properly according to the needs and acquirement of the circumstances. In MCA we are supposed to work on a project, which gives us a required exposure to the professional field, which we will be embracing as soon as we move from our student phase to the professional phase.

Today we have many tools at our disposal, so how to use them judiciously according to the requirements of the clients as well as to convince our client to venture out into Uncharted-Red territory and convince them to adapt to emerging technology is the responsibility of the developer.

A Master of Computer Application (MCA) is a two-year full-time course. It covers various topics associated with the area of computer science. We did a project “Online Exam Management System”. An effort has been to exhaustively deal with every part of the systems developed and at the appropriate position, so That a user can easily generate the reports.

Acknowledgement

Mistry Mohit Arvindbhai a student of MCA (Master of Computer Application) Semester – 2 at S.S Agrawal Institute of Management and Technology, Navsari have completed this project. We would like to convey our heartfelt gratitude to all those people from whom we have got considerable support and encouragement during this project.

We would like to thank our **principal Dr. Sham Sachinwala**, S.S Agrawal Institute of Computer Science for their motivation & constant support to complete this project successfully.

We would like to extend our gratitude to **Prof. Jinal Tailor (Project Guide)** for her valuable guidance and support. Her suggestions and motivation encouraged us throughout the project.

And finally, we would also like to acknowledge the faculty of the Computer Science department for their constant support during the period with them, mounding our future to survive in this highly competitive world.

Mistry Mohit Arvindbhai

Learning during Project Work

It was a wonderful learning experience for me while working on this project. This project took me through the various phases of project development and gave me real insight into the project. The joy of working and the thrill involved while tackling the various problems and challenges gave me a feel of the developer's industry.

There are numerous things that we have learned, while working on the project in the IT industry. The project has helped us in perfecting the concepts which we have learned in the past years.

The project gave us a better understanding that there is so much more to learn from thesurrounding environments of the colleagues and the power of the technological languages.

The project helped us in organizing the time effectively, and working with teammates and generating substantial output of the efforts. It has prepared us for analysing and programming for industrial problems and to work on large projects in the future.

Even though we know so much about the technology, still there is so much more to learn from it. The most important thing that we have learned is to never stop learning.

Abstract

This project report on "**Online Exam Management System**". During the developing of this project, we explore new ideas and functionality. In this project we have basically 3 modules. These three modules include Admin, Faculty and Student.

The admin module profile, add faculty, add students, add Subjects, add course and notification. The faculty module can add subjects and conduct exam and also see their students information and see their student results. The student module profile, students can give exam on exam time and see future exam and students can see their exam marks as well.

This document shows details and various diagrams like use case diagram, class diagram, interaction diagram, activity diagram and also add the data dictionary and screen shorts are define in this document.

INDEX

1. Introduction

1.1 Existing System

- Students around the world face various challenges when it comes to taking exams. They often struggle to find the right study materials and resources, which can lead to frustration and poor academic performance. To address this issue, we have developed an online exam management system.

- Our system is designed to streamline the exam process and eliminate the traditional paper-based methods of conducting exams. By automating the exam management process, our system reduces the workload on administrators and teachers, saving them valuable time and resources. Additionally, by reducing the reliance on paper-based systems, our solution helps institutions reduce their environmental impact.

- Our online exam management system offers a many features, including automatic exam scheduling, grading, and reporting. It also provides access to a vast library of study resources, such as practice exams. With our system, students can easily access all the Exam practices they need to prepare for exams, saving them time and effort.

- Overall, our online exam management system represents a major step forward in modernizing the way students and institutions approach exam management. We believe it will provide significant benefits to students, faculty, and administrators alike, making the entire process more efficient, effective, and convenient.

1.2 Needs for the New System

- In this systems student can give exam easily.
- User authentication and authorization: The system should have a secure login feature that requires students to enter their unique username and password before they can access an exam.
- Exam creation and management: The system should allow educators or administrators to create exams with various question types, including multiple-choice, true/false, essay questions, and more. They should also be able to set the time limit for each exam and manage the schedule of exams.
- Exam-taking interface: The system should provide students with an easy-to-use interface for taking exams. This should include features such as navigation buttons, saving and submitting answers, and a timer that tracks the remaining time.
- Reporting and analytics: The system should generate reports and analytics that provide insights into student performance, including overall class performance, individual student performance, and question-level analysis.

By fulfilling these needs, a new online exam management system can effectively streamline the exam process, improve security, increase accessibility, and enhance the overall learning experience for both educators and students.

1.3 Objective of the New System

- This system is developed for Exam.
- All users have access right as per their role.

- The main objective is taking exam to student's easily.
- Admin can add faculty, student, Subjects and course.
- Faculty can add MCQ according to their course.
- Student can see that Whats up coming exam coming. And when that time near then student can give exam etc…
- Create a database for easy to retrieval, storage and maintenance of student record.

1.4 Problem Definition

The Online Exam Management System currently lacks the feature of allowing instructors to upload course materials for students. This presents a number of challenges for both students and faculty, including:

- **Inability to access necessary materials:** Without the ability to upload course materials, students may miss out on important information necessary for success in their courses.
- **Inefficient communication:** Faculty may have to resort to other means of sharing materials, such as email or in-person handouts, which can be time-consuming and inefficient.
- **Limited flexibility:** Students may not be able to access required materials outside of class time, limiting their ability to study and learn at their own pace.
- **Increased workload:** Faculty members may need to spend extra time organizing and distributing materials, which can detract from their ability to focus on other aspects of their teaching.

By adding a feature to allow for easy and efficient uploading of course materials, the Online Exam Management System can help alleviate these issues and create a more streamlined and effective learning experience for all involved.

1.5 Core Components

1.5.1 Scope

- The proposed software product is the Online Exam Management System.
- The system main purpose of this system Students can give exam easily.
- Managing all records of Admin, Faculty

1.5.2 Modules

- There are 3 panel of our system as below: -

1. Admin
2. Faculty
3. Student

Details Functionality of the above listed modules

1. **Admin**
    - Admin can Login.
    - Admin can Insert Update & Delete Faculty and Student.
    - Adding the Subjects based on their course
    - Also, admin can update Subjects and Delete.
    - Admin can give approve user so that faculty and student can login.
2. **Faculty**
    - Faculty can Login using the email id and password.
    - Faculty can add question according to their course
    - Faculty can see overview of total question
    - Faculty can see overview of total students whose give exam.
    - Faculty can see all students marks according to schedule exams
    - Faculty can schedule or conduct exam for future event and all students which belong to their course they can get notified on dashboard.
3. **Student**
    - Student can Login using the email id and password.
    - Student can view their marks.
    - Student can join exam when it's start.
    - It's shows scheduled exam.
    - Student can see their growth of marks in a format of graph.

1.6 Project Profile

Project Profile:

Project Name:	**Online Exam Management System**
Developers Name:	Mistry Mohit Arvindbhai
Team strength:	1
Front end:	React JS
Frame Back end Work:	Laravel(DB:MYSQL)
Operating System:	Windows 11
Project Duration:	4 months
Submitted to:	S.S Agrawal Institute of Management and Technology
Internal guide:	Prof. Jinal Tailor

Software Used:	Visual studio code, Xampp or Wampp, MS Word, Edrawmax, Paint.
Head of Departments:	Prof. Jinal Tailor

1.7 Advantages and Limitations of the Proposed System

- Advantages of This System as below: -
  - It is very efficient way to taking exam online.
  - It save stationery cost as well as it's saved time to faculty also.
  - Possible for some time cutting in the Educational Process.
  - Free and Easy to Understand.
  - Time saving for students.
  - If faculty creates all types of question then next year they do not make again and again that question they can simply conduct exam for upcoming students
  - Due to its convenience and flexibility, Faculty can conduct exam anytime anywhere.
  - Student can't depend on anyone for practice and preparation for exam.

- Limitations of this systems: -
  - Network problem can affect the system and specially when students give exam.
  - Internet problem.

2. Requirement Determination & Analysis

2.1 Requirement Determination

2.1.1 Hardware Requirement

This phase of the software development process deals with a brief study of different hardware used in the computerized system. This is a list of hardware materials used during the making and the use of the proposed system. As the new system to be made into a computerized functional system required of a computer is must. All the hardware needed here are generally the basic configuration of an office computer.

- **Minimum Hardware Requirement:**

To run the application software of the system in the computer the minimum configuration required is as below:

Processor:	Intel Core i3
RAM:	4 GB DDR4 or Higher RAM
Hard Disk:	5 GB Free Space
Monitor:	1024 DPI or LCD
Keyboard:	Multimedia or 104 Keys
Mouse:	Optical or Scroll
Printer:	Laser Printer or Inkjet printer or Dot Matrix
Backup:	Git hub Or External Hard Disk Or Pen drive
Motherboard:	Intel

2.1.2 Software Requirement

This phase of the software development process deals with a brief study of different Software used in the computerized system. This is a list of Software's used during the making and the use of the proposed system. As the new system to be made into a computerized functional system required of a computer is must. All the Software needed here are generally the basic configuration of an office computer.

To run the application following minimum software must be required:

Front end:	React JS
Back end Framework:	**Laravel (DB: MY SQL)**
Web Server:	**Xampp or Wampp**
Operating System:	**Windows 10 or Window 11**
Photo Editor:	**Photoshop, paint, Edrawmax**
Browser:	**Chrome, Firefox**
Documents:	**MS Word**
Presentation:	**MS Power point**

2.2 Targeted Users

1. **Admin**
   - Admin can Login.
   - Admin can Insert Update & Delete Faculty and Student.
   - Adding the Subjects based on their course
   - Also, admin can update Subjects and Delete..
2. **Faculty**
   - Faculty can Login using the email id and password.
   - Faculty can add question according to their course
   - Faculty can see overview of total question
   - Faculty can see overview of total students whose give exam.
   - Faculty can see all students marks according to schedule exams
   - Faculty can schedule or conduct exam for future event and all students which belong to their course they can get notified on dashboard.
3. **Student**
   - Student can Login using the email id and password.
   - Student can view their marks.
   - Student can join exam when it’s start.
   - It’s shows scheduled exam.

3. System Design

A Use case diagram is a dynamic or behavior diagram in UML. Use case diagrams model the functionality of a system using actors and use cases. Use cases are a set of actions, services, and functions that the system needs to perform. In this context, a "system" is something developed or operated, such as a website. The "actors" are people or entities operating under defined roles within the system.

Basic use case diagram symbols and Notations:

Use case

- Draw use cases using ovals. Labels the ovals with verbs that represent the system's functions.

Actors

- Actors are the users of a system. When one system is the actor of another system, label the actor system with the actor stereotype.

Actor

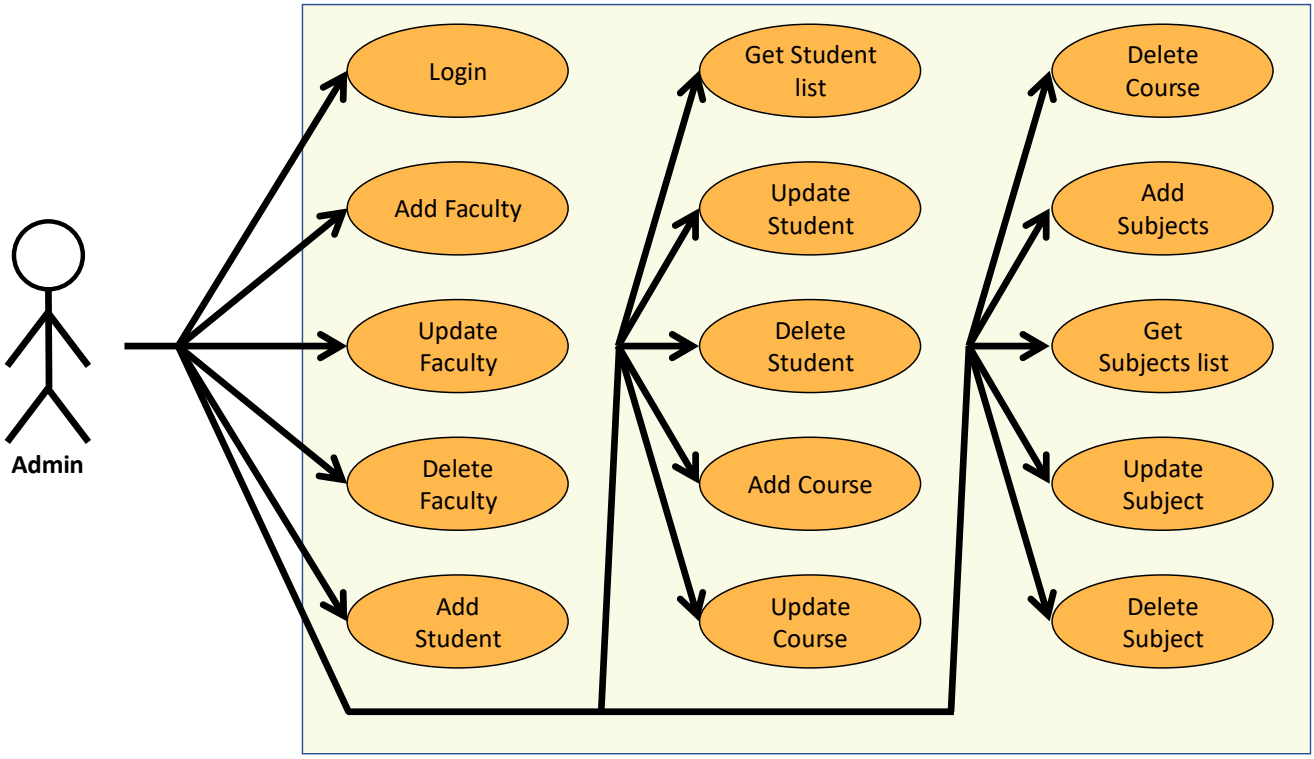
Relationships

- Illustrate the relationship between an actor and a use case with a simple line. For relationships among use cases, use arrows labeled either "users" or "extends". A "users" relationship indicates that one use case is needed by another in order to perform a task. An "extends" relationship indicates alternative options under a certain use case.

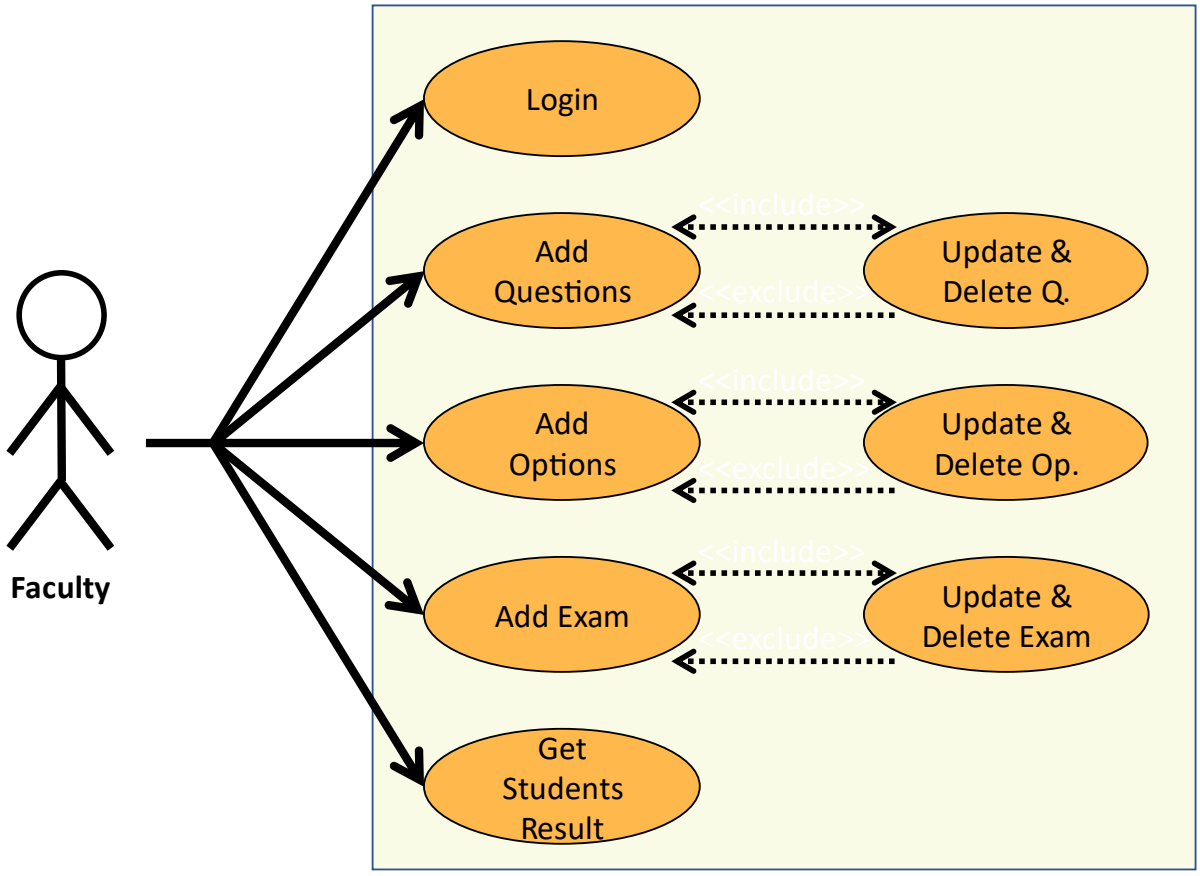
<< Include>>

<<Exclude>>

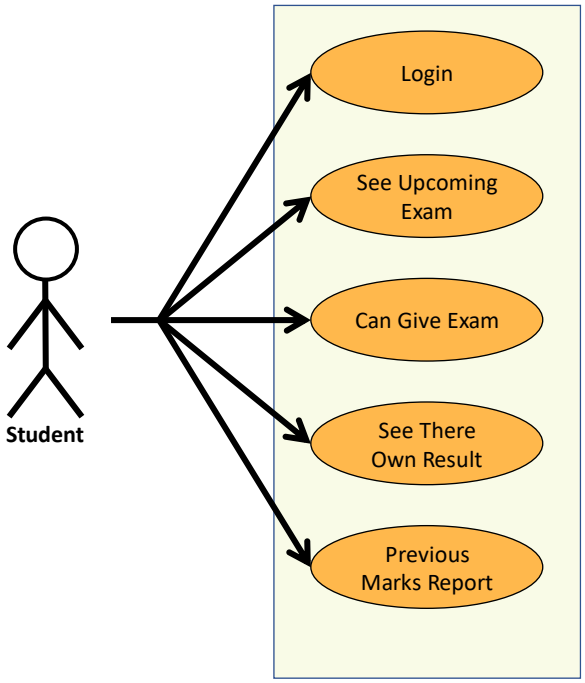
3.1.1 Admin



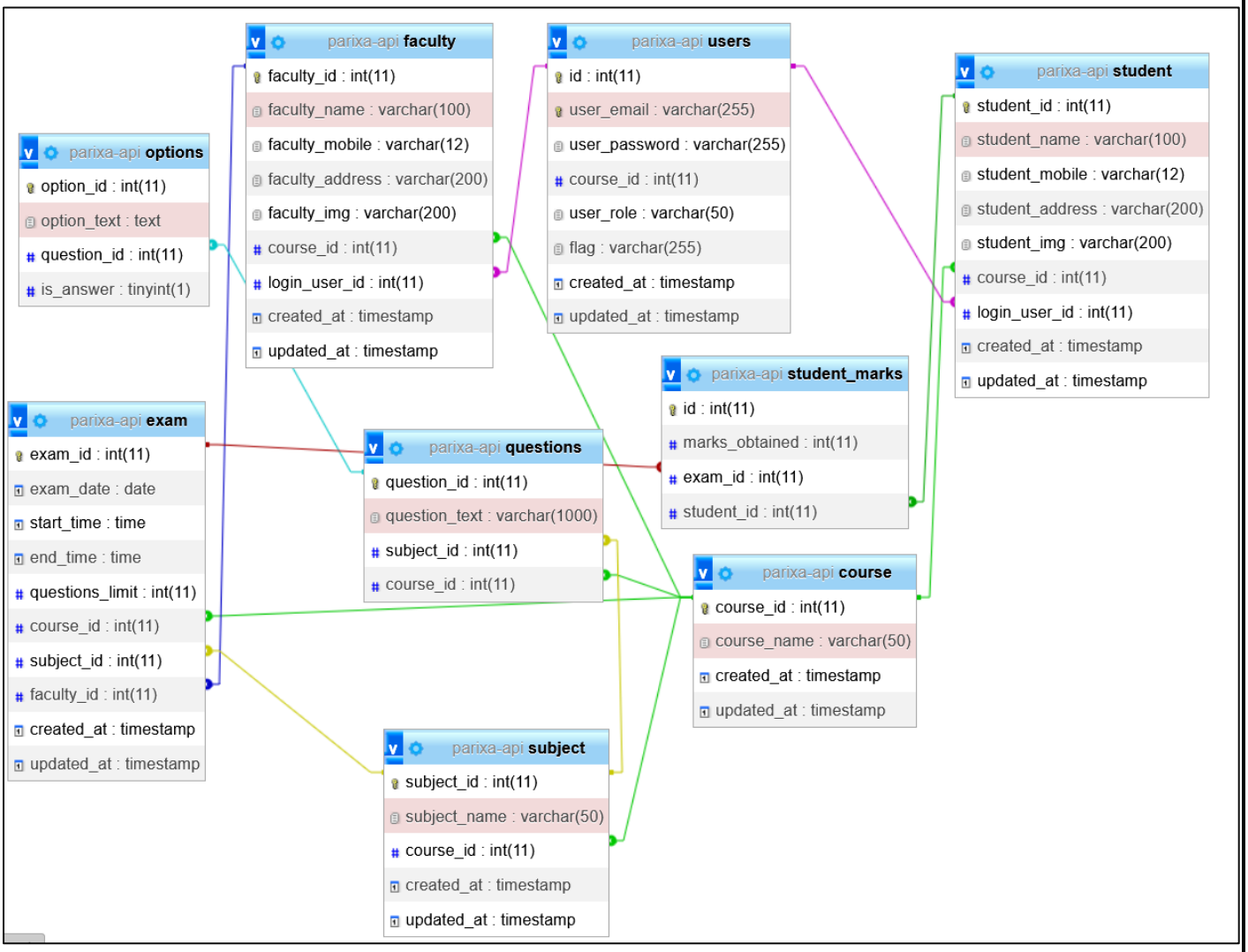
3.1.2 Faculty



3.1.3 Student



3.2 Class Diagram

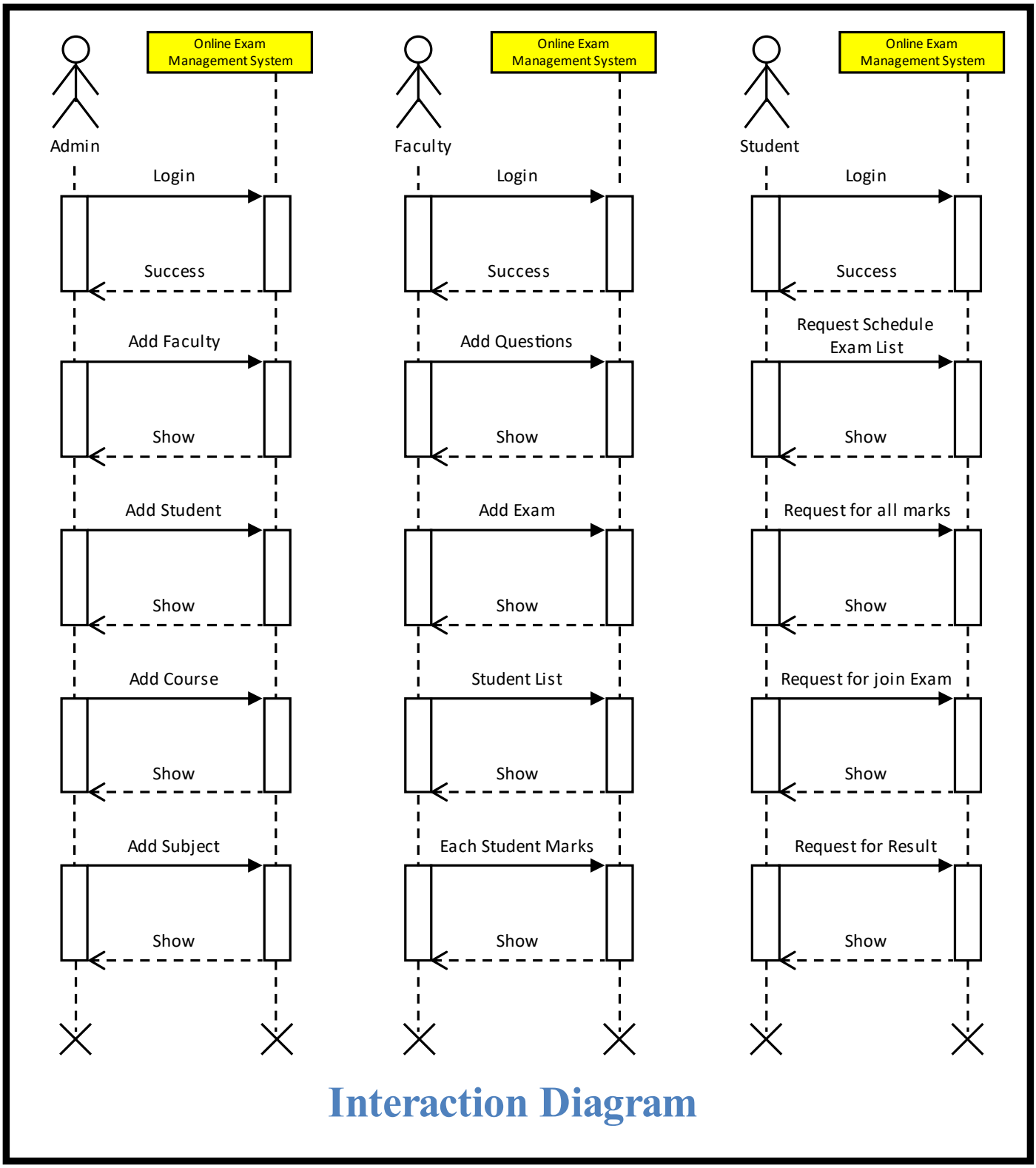


3.3 Interactive Diagram

Interaction diagrams are models that describe how a group of objects collaborate in some behaviour-typically a single use-case. The diagrams show a number of example objects and the messages that arepassed between these objects within the use-case.

Interaction diagrams come in two forms, both present in the UML. The first form is the sequence diagram. In this form objects are shown as vertical lines with the messages as horizontal lines between them. This form was first popularized by Jacobson.

As below the Interaction Diagram For This System :-



Interaction Diagram

3.4 Activity Diagram

An activity diagram visually presents a series of actions or flow of control in a system similar to a flowchart or a data flow diagram. Activity diagrams are often used in business process modeling. They can also describe the steps in a use case diagram. Activities modeled can be sequential and concurrent. In both cases an activity diagram will have a beginning (an initial state) and an end (a final state).

Basic Activity Diagram Notations and Symbols:-

Initial State or Start Point

- A small filled circle followed by an arrow represents the initial action state or the start point for any activity diagram. For activity diagram using swim lanes, make sure the start point is placed in the top left corner of the first column.

Start Point/Initial State

Activity or Action State

- An action state represents the non-interruptible action of objects. You can draw an action state in Smart Draw using a rectangle with rounded corners.

Activity

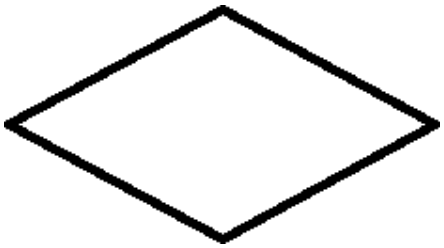
Action Flow

Action flows, also called edges and paths, illustrate the transitions from one action state to another. They are usually drawn with an arrowed line.

Action Flow

Decisions and Branching

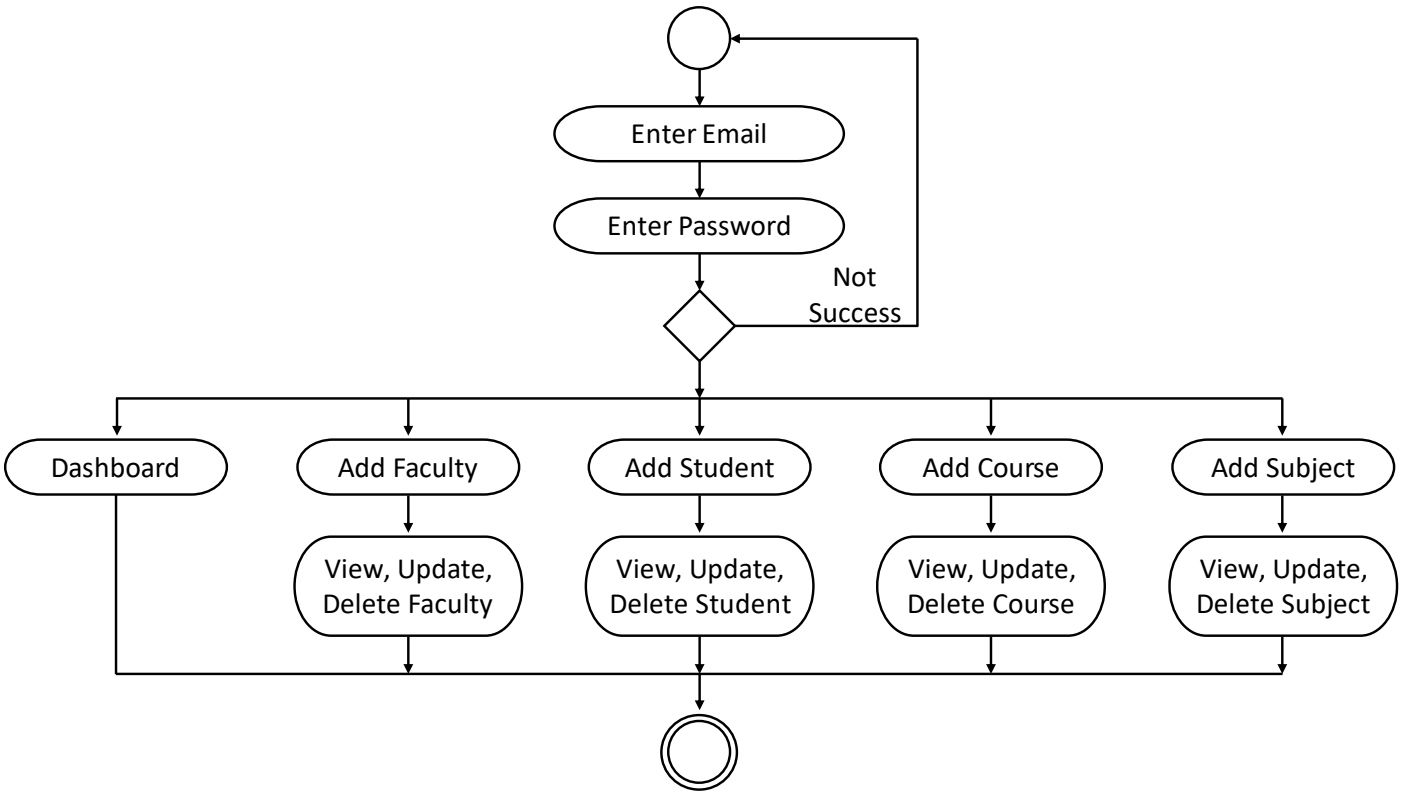
A diamond represents a decision with alternate paths. When an activity requires a decision prior tomoving on to the next activity, add a diamond between the two activities. The outgoing alternates should belabeled with a condition or guard expression. You can also label one of the paths "else."



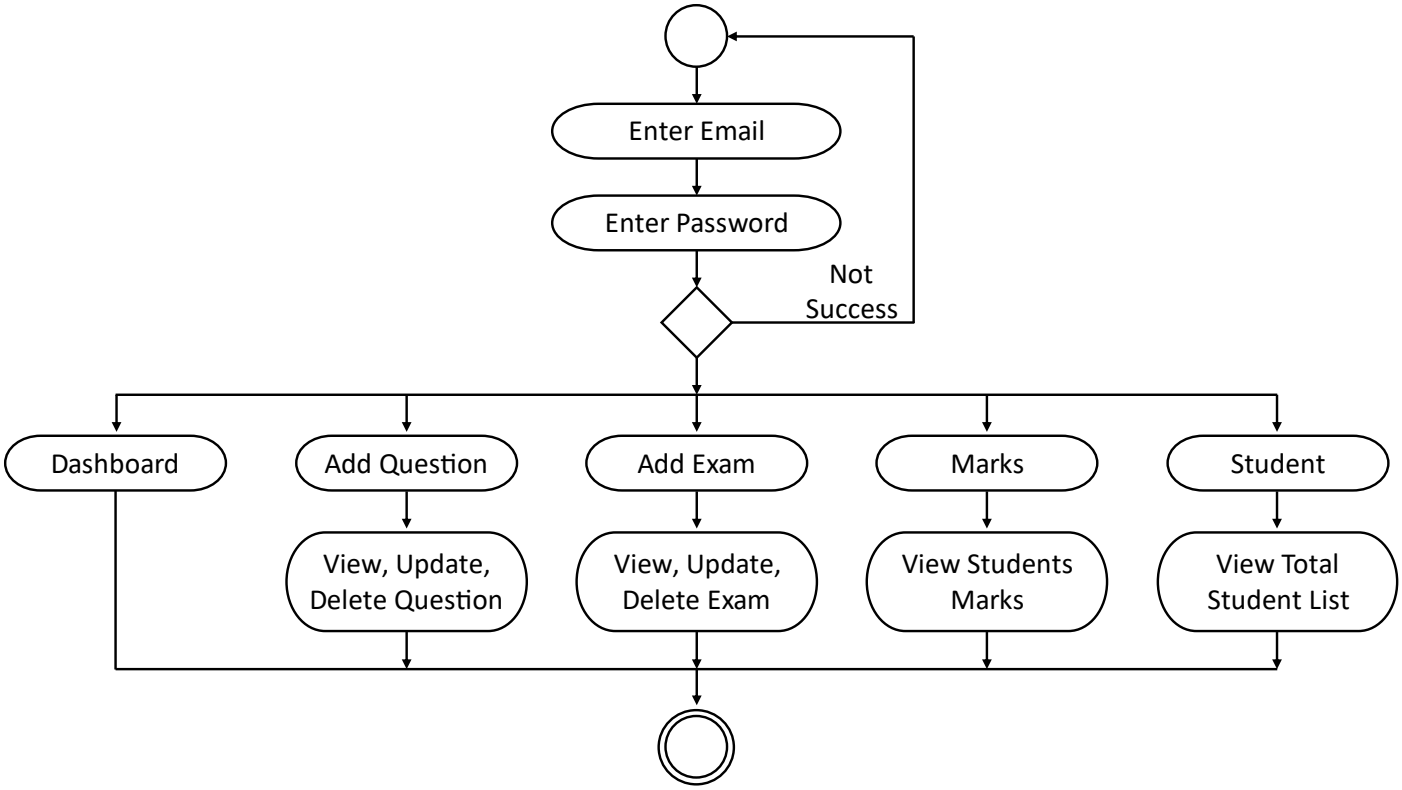
Decision Symbol

As Below the Interaction Diagram for this System: -

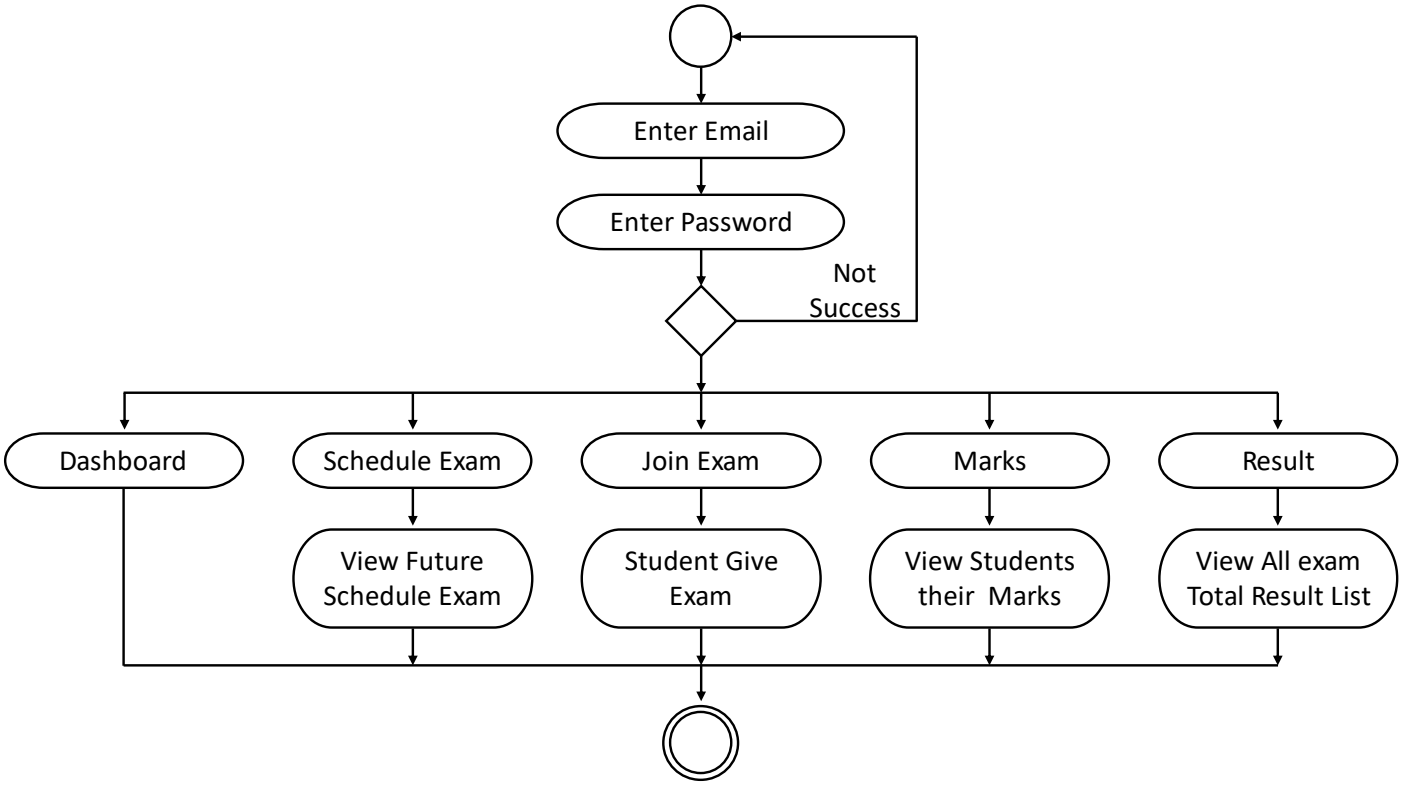
3.4.1 Admin



3.4.2 Faculty



3.4.3 Students



3.5 Data Dictionary

- **Table structure: -**

1. **Table name: -** users

Primary key: - id

Foreign key : user_email

Description: - To store the All users detail.

Sr no.	FEILDS		DATA_TYPE	DESCRIPTION
1.	id		int(11)	Users id store here.
2.	user_email		varchar (255)	Users store email.
3.	user_password		varchra(255)	Users store password.
4.	course_id		int(11)	Users course id store here
5.	user_role		varchar (50)	Users role store data.
6.	flag		Varchar (255)	Users Access Rights.
7.	created_at		timestamp	Users create timestamp.
8.	updated_at		timestamp	Users update timestamp.

id	user_email	user_password	course_id	user_role	flag	created_at	updated_at
63	admin@admin.com	e10adc3949ba59abbe56e057f20f883e	0	admin	1	2023-05-26 20:20:30	2023-05-26 20:22:00
64	veena.behl@hotmail.com	e10adc3949ba59abbe56e057f20f883e	17	student	1	2023-05-26 20:49:58	2023-06-07 08:05:41
65	udara@yahoo.co.in	e10adc3949ba59abbe56e057f20f883e	17	student	1	2023-05-26 20:51:48	2023-06-07 08:06:02
66	chameli.iyer@yahoo.co.in	e10adc3949ba59abbe56e057f20f883e	18	student	1	2023-05-26 20:54:09	2023-06-07 08:06:20
67	lakshmi.rajan@sood.in	e10adc3949ba59abbe56e057f20f883e	18	student	1	2023-05-26 20:56:04	2023-06-07 08:06:26
68	mohit@gmail.com	e10adc3949ba59abbe56e057f20f883e	17	faculty	1	2023-05-26 20:57:46	2023-06-07 08:07:12
69	haresh@gmail.com	e10adc3949ba59abbe56e057f20f883e	18	faculty	1	2023-05-26 20:58:53	2023-06-08 15:11:39
70	abc@gmail.com	e10adc3949ba59abbe56e057f20f883e	21	student	1	2023-06-01 11:48:53	2023-06-08 17:59:37
80	nilesh@gmail.com	e10adc3949ba59abbe56e057f20f883e	18	faculty	2	2023-06-08 21:57:23	2023-06-08 21:57:23
81	kamlesh@gmail.com	e10adc3949ba59abbe56e057f20f883e	18	student	2	2023-06-08 21:58:23	2023-06-08 21:58:23

2. **Table name:** - faculty

Primary key: - faculty_id

Foreign key : course_id, login_user_id

Description: - To store the All Faculty detail.

Sr no.	FEILDS		DATA_TYPE	DESCRIPTION
1.	faculty_id		int(11)	Faculty id store here.
2.	faculty_name		varchar (100)	Faculty store Name.
3.	faculty_mobile		varchra(12)	Faculty store Mobile Number.

4.	faculty_address		varchar (200)	Faculty Address store data.
5.	course_id		int(11)	Faculty which belong to course
6.	login_user_id		int(11)	Faculty Login Id Store here
7.	created_at		timestamp	Faculty create timestamp.
8.	updated_at		timestamp	Faculty update timestamp.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	**faculty_id**	int(11)			No	*None*		AUTO_INCREMENT
2	**faculty_name**	varchar(100)	utf8mb4_general_ci		No	*None*		
3	**faculty_mobile**	varchar(12)	utf8mb4_general_ci		No	*None*		
4	**faculty_address**	varchar(200)	utf8mb4_general_ci		No	*None*		
5	**faculty_img**	varchar(200)	utf8mb4_general_ci		Yes	*NULL*		
6	**course_id**	int(11)			No	*None*		
7	**login_user_id**	int(11)			No	*None*		
8	**created_at**	timestamp			No	current_timestamp()		
9	**updated_at**	timestamp			No	current_timestamp()		ON UPDATE CURRENT_TIMESTAMP()

3. **Table name: -** student

   Primary key: - student_id

   Foreign key : course_id, login_user_id

   Description: - To store the All Student detail.

Sr no.	FEILDS		DATA_TYPE	DESCRIPTION
1.	student_id		int(11)	Student id store here.
2.	student_name		varchar (100)	Student store Name.
3.	student_mobile		varchra(12)	Student store Mobile Number.
4.	student_address		varchar (200)	Student Address store data.
5.	course_id		int(11)	Student which belong to course
6.	login_user_id		int(11)	Student Login Id Store here
7.	created_at		timestamp	Student create timestamp.

8.	updated_at		timestamp	Student update timestamp.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	**student_id**	int(11)			No	*None*		AUTO_INCREMENT
2	**student_name**	varchar(100)	utf8mb4_general_ci		No	*None*		
3	**student_mobile**	varchar(12)	utf8mb4_general_ci		No	*None*		
4	**student_address**	varchar(200)	utf8mb4_general_ci		No	*None*		
5	**student_img**	varchar(200)	utf8mb4_general_ci		Yes	*NULL*		
6	**course_id**	int(11)			No	*None*		
7	**login_user_id**	int(11)			No	*None*		
8	**created_at**	timestamp			No	current_timestamp()		
9	**updated_at**	timestamp			No	current_timestamp()		ON UPDATE CURRENT_TIMESTAMP()

4. **Table name:** - course

   Primary key: - course_id

   Description: - To store the All course detail.

Sr no.	FEILDS		DATA_TYPE	DESCRIPTION
1.	course_id		int(11)	Course id store here.
2.	course_name		varchar (50)	Course store Name.
3.	created_at		timestamp	Course create timestamp.
4.	updated_at		timestamp	Course update timestamp.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	**course_id**	int(11)			No	*None*		AUTO_INCREMENT
2	**course_name**	varchar(50)	utf8mb4_general_ci		No	*None*		
3	**created_at**	timestamp			No	current_timestamp()		
4	**updated_at**	timestamp			No	current_timestamp()		ON UPDATE CURRENT_TIMESTAMP()

5. **Table name:** - subject

   Primary key: - subject_id

   Foreign key:- course_id

   Description: - To store the All subject detail.

Sr no.	FEILDS		DATA_TYPE	DESCRIPTION
1.	subject_id		int(11)	Subject id store here.
2.	subject_name		varchar (50)	Subject store Name.
3.	course_id		int(11)	Store course Id
4.	created_at		timestamp	Subject create timestamp.
5.	updated_at		timestamp	Subject update timestamp.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	**subject_id**	int(11)			No	*None*		AUTO_INCREMENT
2	**subject_name**	varchar(50)	utf8mb4_general_ci		No	*None*		
3	**course_id**	int(11)			No	*None*		
4	**created_at**	timestamp			No	current_timestamp()		
5	**updated_at**	timestamp			No	current_timestamp()		ON UPDATE CURRENT_TIMESTAMP()

6. **Table name: -** exam

   Primary key: - exam_id

   Foreign key:- course_id, subject_id, faculty_id

   Description: - To store the All Schedule Exam detail.

Sr no.	FEILDS		DATA_TYPE	DESCRIPTION
1.	exam_id		int(11)	Store Exam Id
2.	exam_date		date	Store Exam Date
3.	start_time		time	Store start exam time
4.	end_time		time	Store end exam time
5.	questions_limit		int(11)	Store number of questions
6.	course_id		int(11)	Store course id
7.	subject_id		int(11)	Store subject id
8.	faculty_id		int(11)	Store faculty id
9.	created_at		timestamp	Exam create timestamp
10.	updated_at		timestamp	Exam update timestamp

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	**exam_id**	int(11)			No	*None*		AUTO_INCREMENT
2	**exam_date**	date			No	*None*		
3	**start_time**	time			No	*None*		
4	**end_time**	time			No	*None*		
5	**questions_limit**	int(11)			No	*None*		
6	**course_id**	int(11)			No	*None*		
7	**subject_id**	int(11)			No	*None*		
8	**faculty_id**	int(11)			No	*None*		
9	**created_at**	timestamp			No	current_timestamp()		
10	**updated_at**	timestamp			No	current_timestamp()		ON UPDATE CURRENT_TIMESTAMP()

7. **Table name:** - options

   Primary key: - option_id

   Foreign key:- question_id

   Description: - To store the All Options of questions detail.

Sr no.	FEILDS		DATA_TYPE	DESCRIPTION
1.	option_id		int(11)	Store option Id
2.	option_text		text	Store option Date
3.	question_id		int(11)	Store question id
4.	is_answer		Tinyint(1)	Store answer

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	**option_id**	int(11)			No	*None*		AUTO_INCREMENT
2	**option_text**	text	utf8mb4_general_ci		No	*None*		
3	**question_id**	int(11)			No	*None*		
4	**is_answer**	tinyint(1)			No	0		

8. **Table name:** - questions

   Primary key: - question_id

   Foreign key:- subject_id, course_id

Description: - To store the All questions detail.

Sr no.	FEILDS		DATA_TYPE	DESCRIPTION
1.	question_id		int(11)	Store question Id
2.	question_text		varchar(1000)	Store question text
3.	subject_id		int(11)	Store subject id
4.	course_id		int(11)	Store course id

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	**question_id**	int(11)			No	*None*		AUTO_INCREMENT
2	**question_text**	varchar(1000)	utf8mb4_general_ci		No	*None*		
3	**subject_id**	int(11)			No	*None*		
4	**course_id**	int(11)			No	*None*		

9. **Table name: -** student_marks

Primary key: - id

Foreign key:- exam_id, student_id

Description: - To store the marks of student detail.

Sr no.	FEILDS		DATA_TYPE	DESCRIPTION
1.	id		int(11)	Store mark Id
2.	marks_obtained		int(11)	Store marks
3.	exam_id		int(11)	Store exam id
4.	student_id		int(11)	Store student id

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	**id**	int(11)			No	*None*		AUTO_INCREMENT
2	**marks_obtained**	int(11)			No	*None*		
3	**exam_id**	int(11)			No	*None*		
4	**student_id**	int(11)			No	*None*		

4. Development

Coding Standards

Different modules specified in the design document are coded in the Coding phase according to the module specification. The main goal of the coding phase is to code from the design document prepared after the design phase through a high-level language and then to unit test this code.

Good software development organizations want their programmers to maintain to some well-defined and standard style of coding called coding standards. They usually make their own coding standards and guidelines depending on what suits their organization best and based on the types of software they develop. It is very important for the programmers to maintain the coding standards otherwise the code will be rejected during code review.

Purpose of Having Coding Standards:

- A coding standard gives a uniform appearance to the codes written by different engineers.
- It improves readability, and maintainability of the code and it reduces complexity also.
- It helps in code reuse and helps to detect error easily.
- It promotes sound programming practices and increases efficiency of the programmers.

Some of the coding standards are given below:

1. **Limited use of globals:**
   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

3. **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.

4. **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.

5. **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.

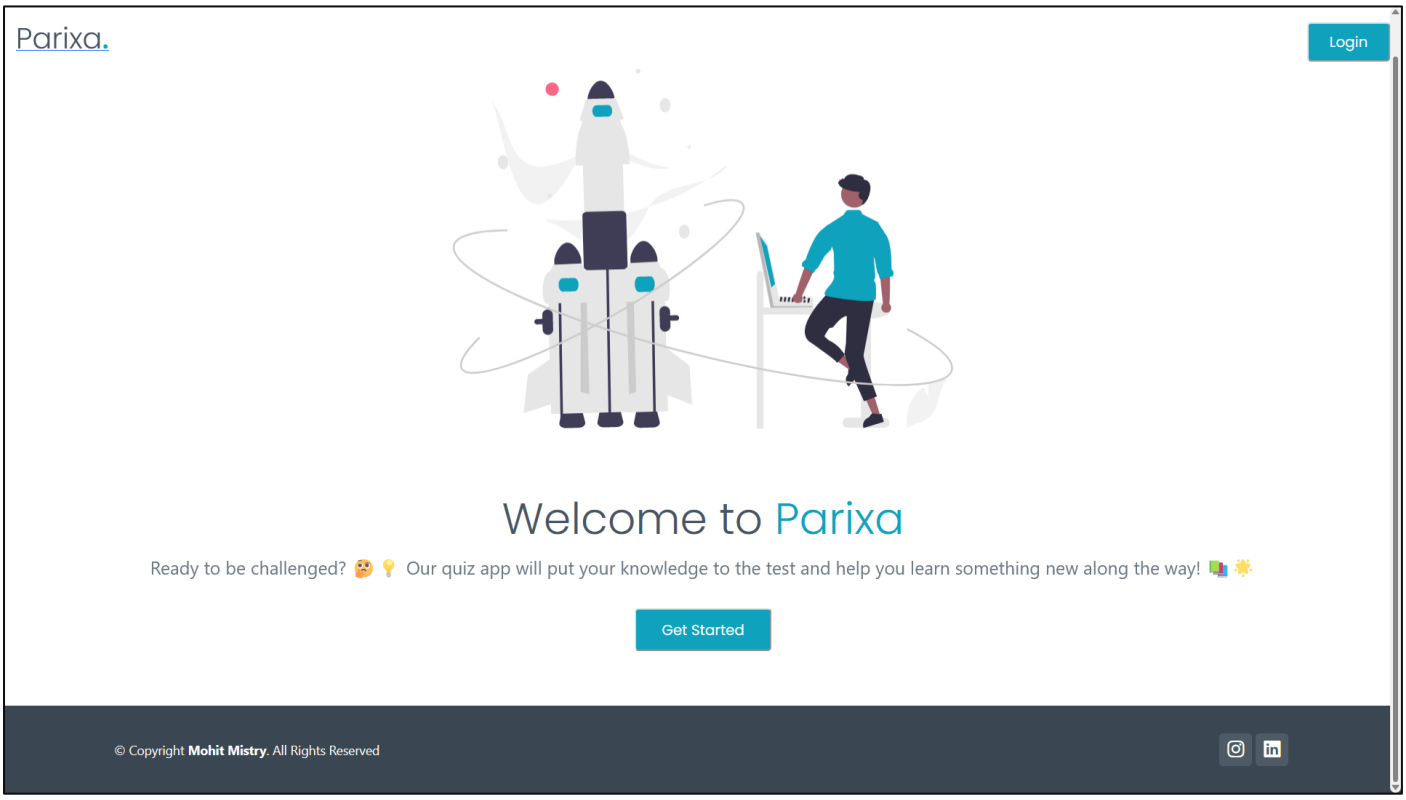
Advantages of Coding Guidelines:

- Coding guidelines increase the efficiency of the software and reduces the development time.
- Coding guidelines help in detecting errors in the early phases, so it helps to reduce the extra cost incurred by the software project.
- If coding guidelines are maintained properly, then the software code increases readability and understandability thus it reduces the complexity of the code.

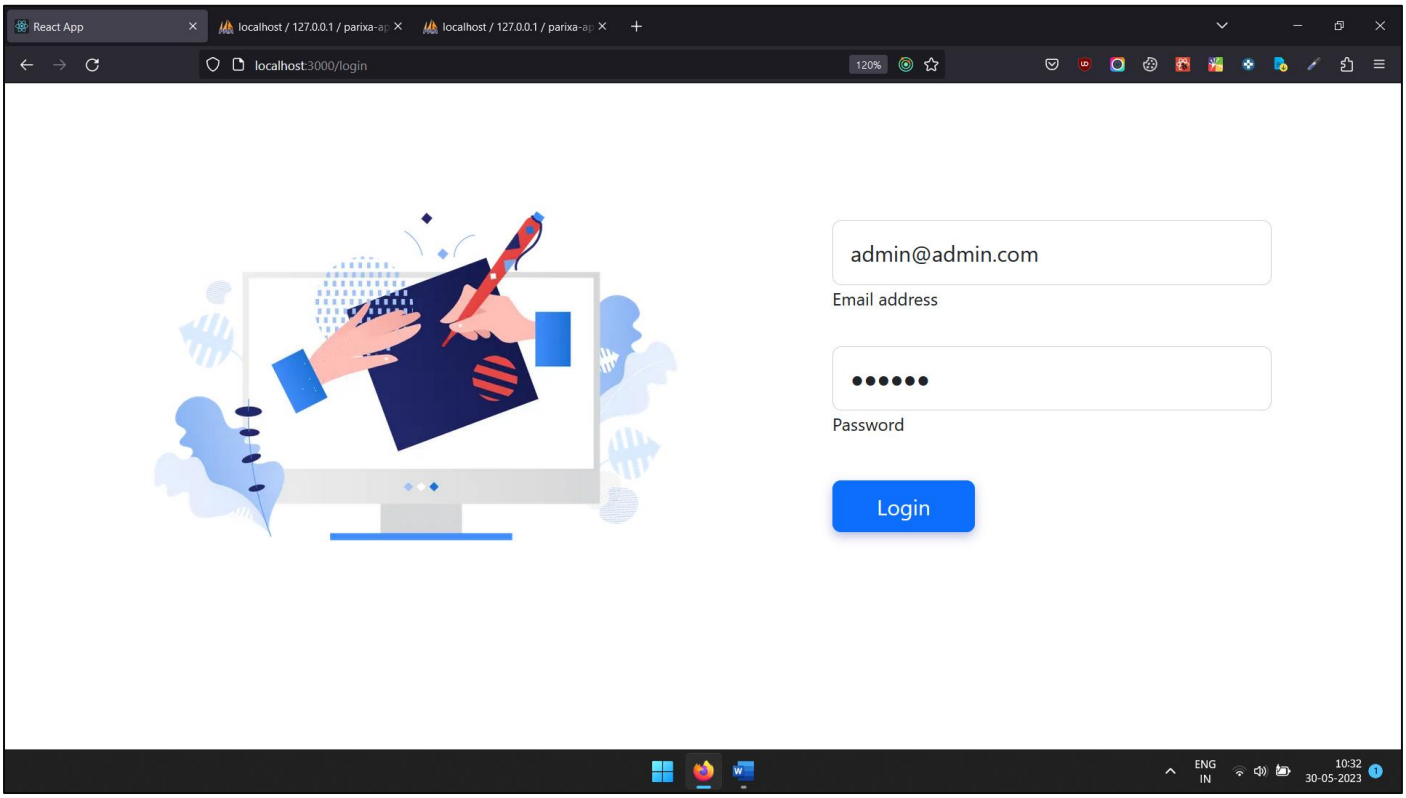
It reduces the hidden cost for developing the software.

4.2 Screen Shorts

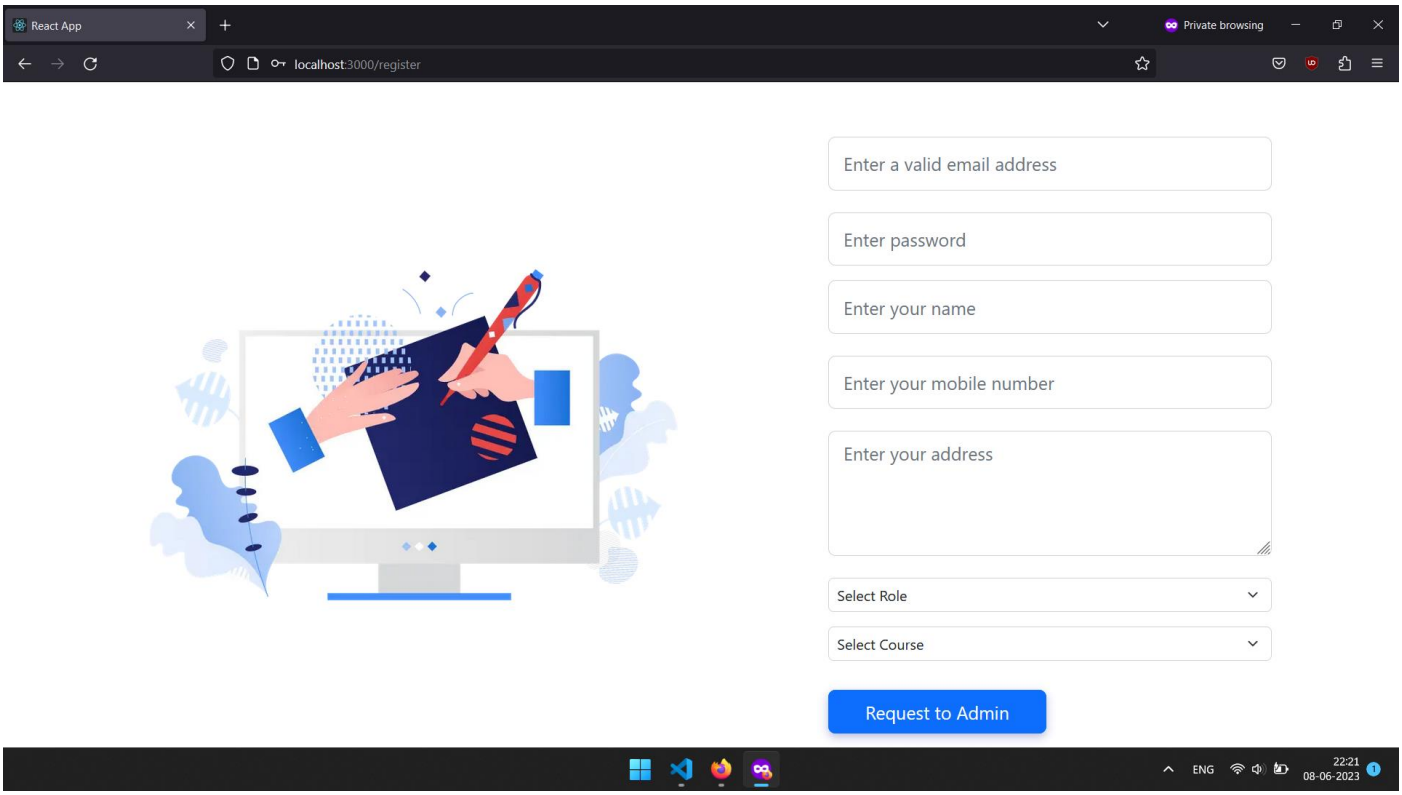
1.Admin Module: -



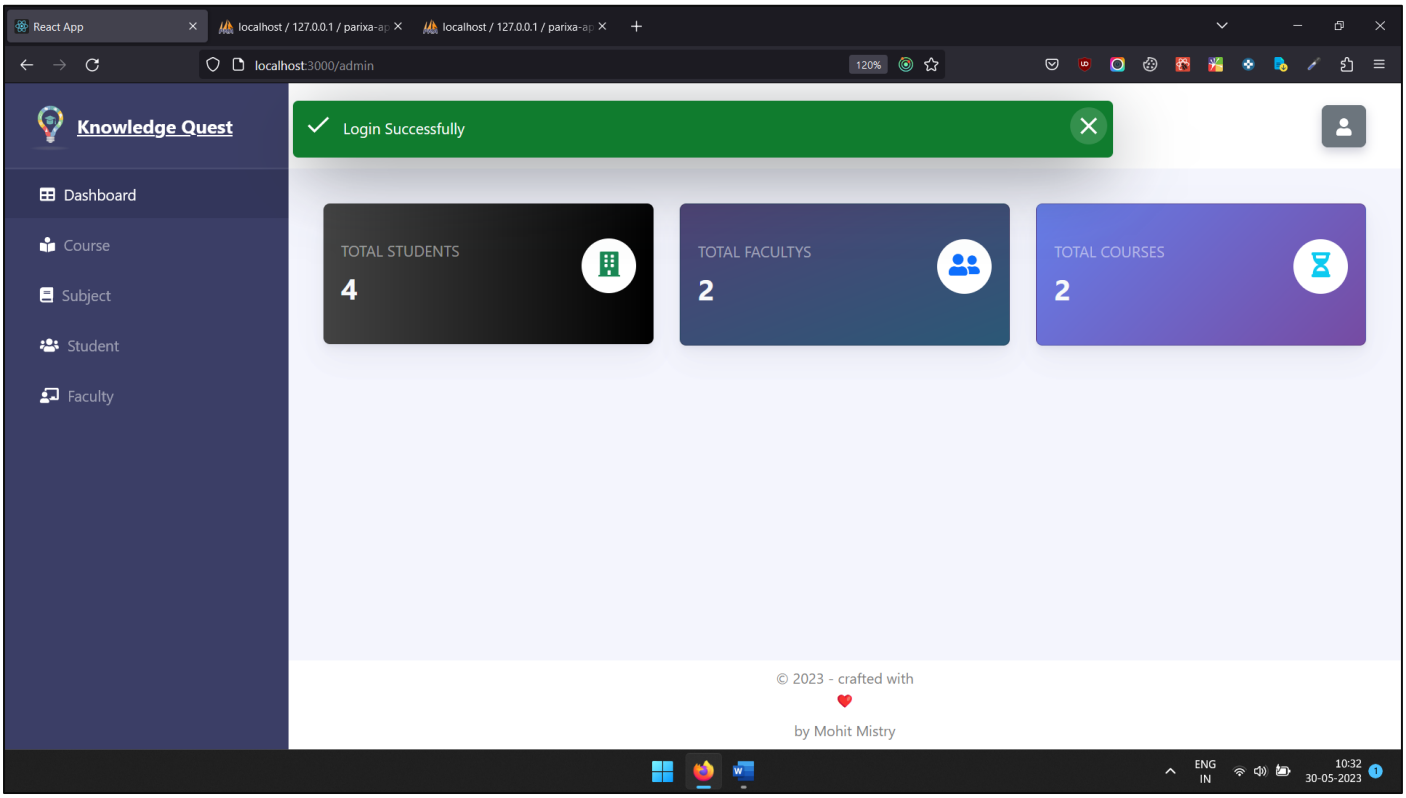
Admin Login Page



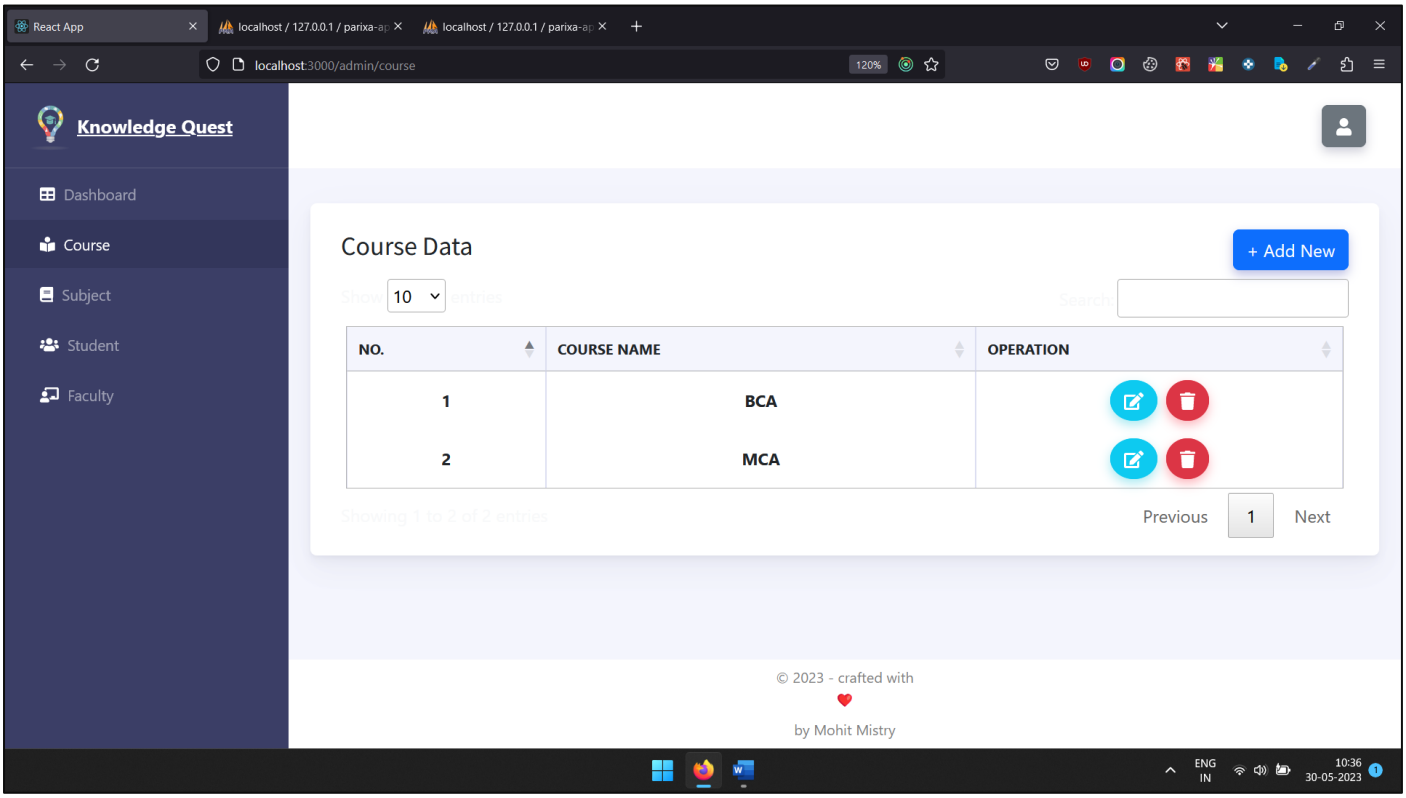
Student and Faculty's register page



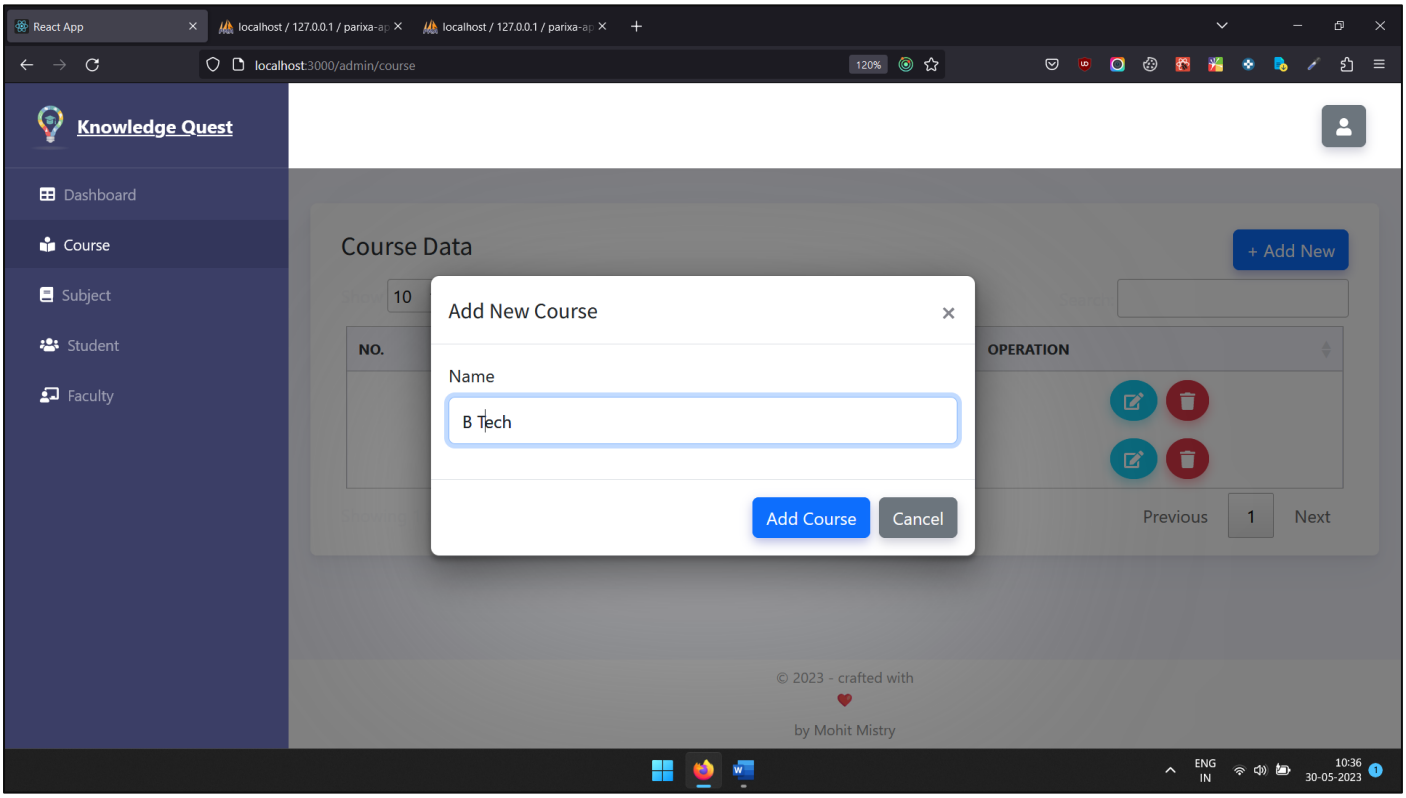
After Login Admin Dashboard



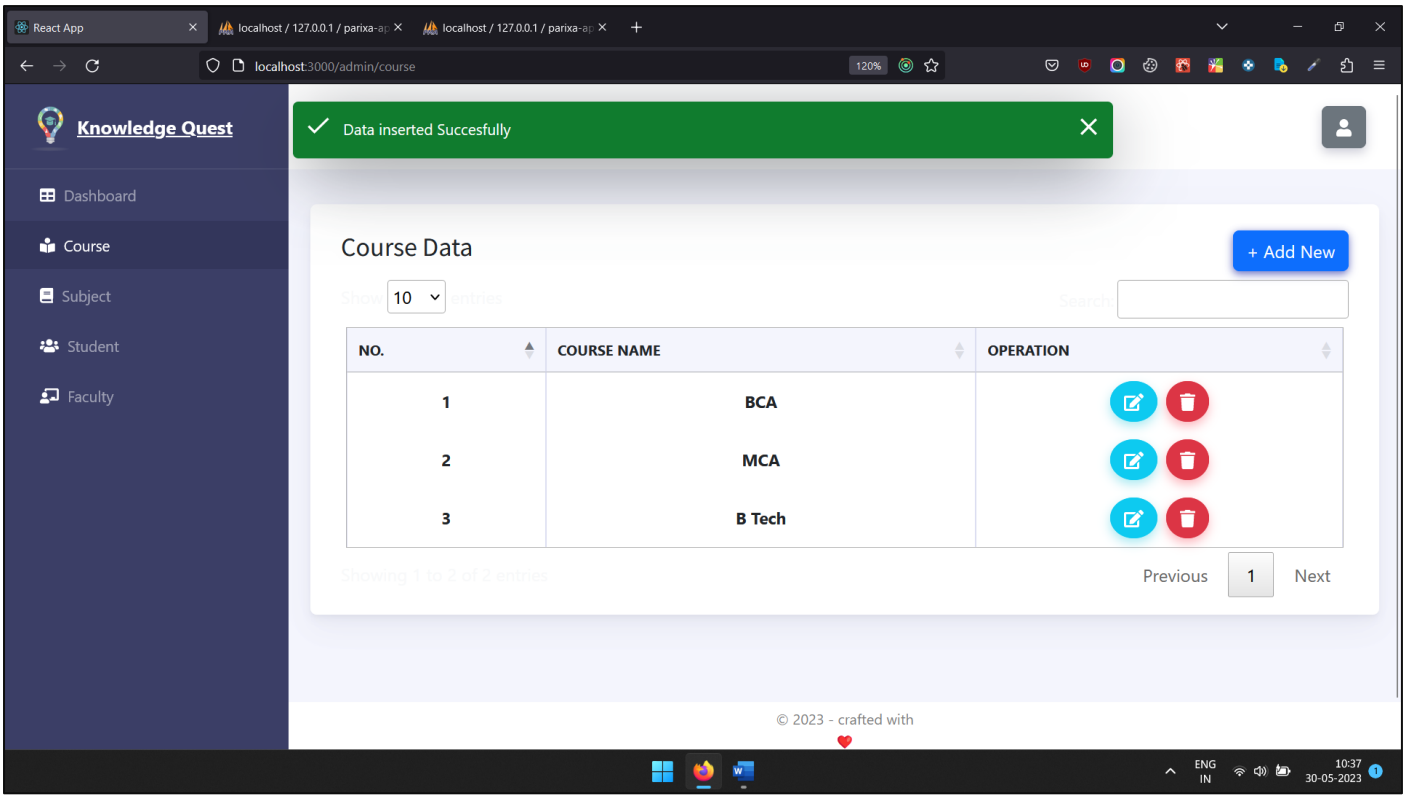
Course List



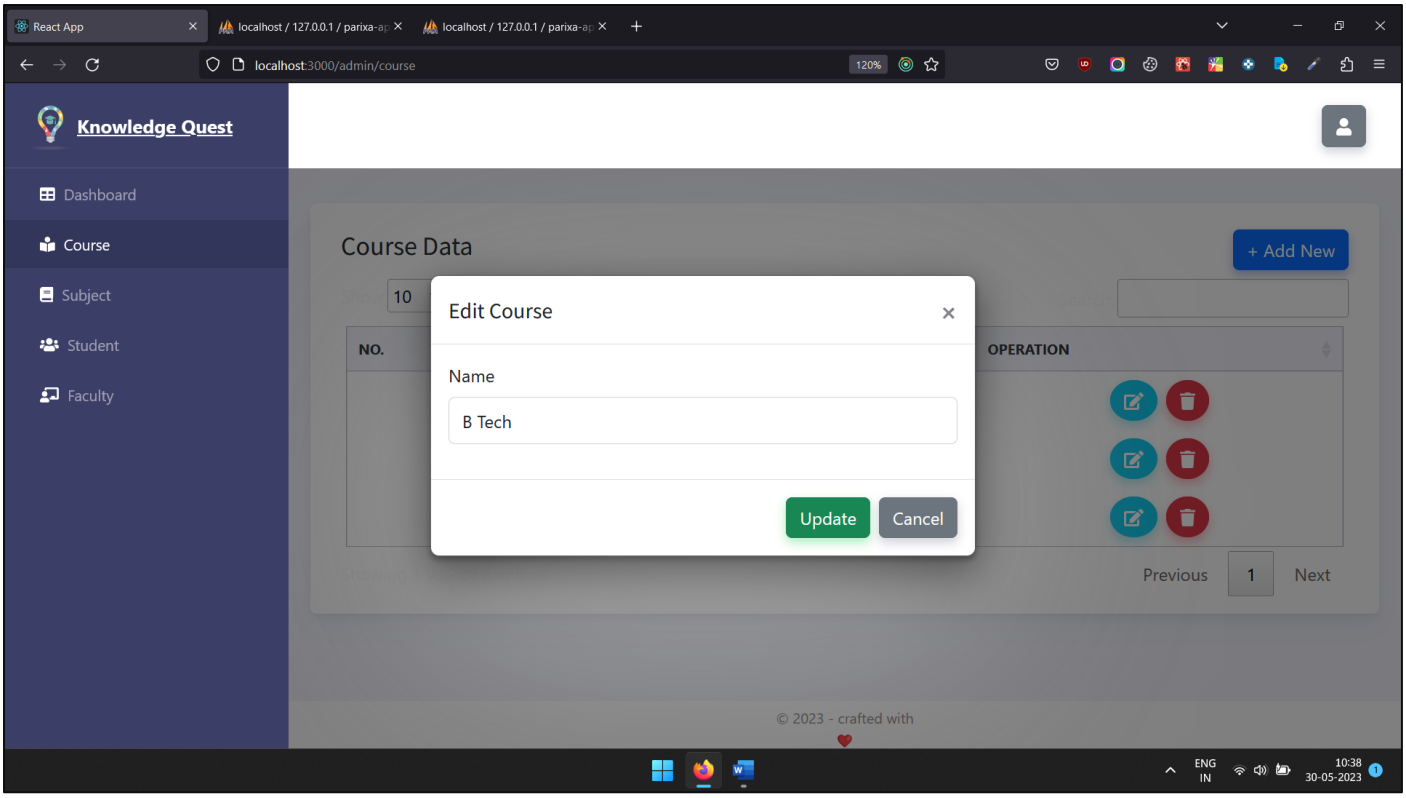
Add New Course



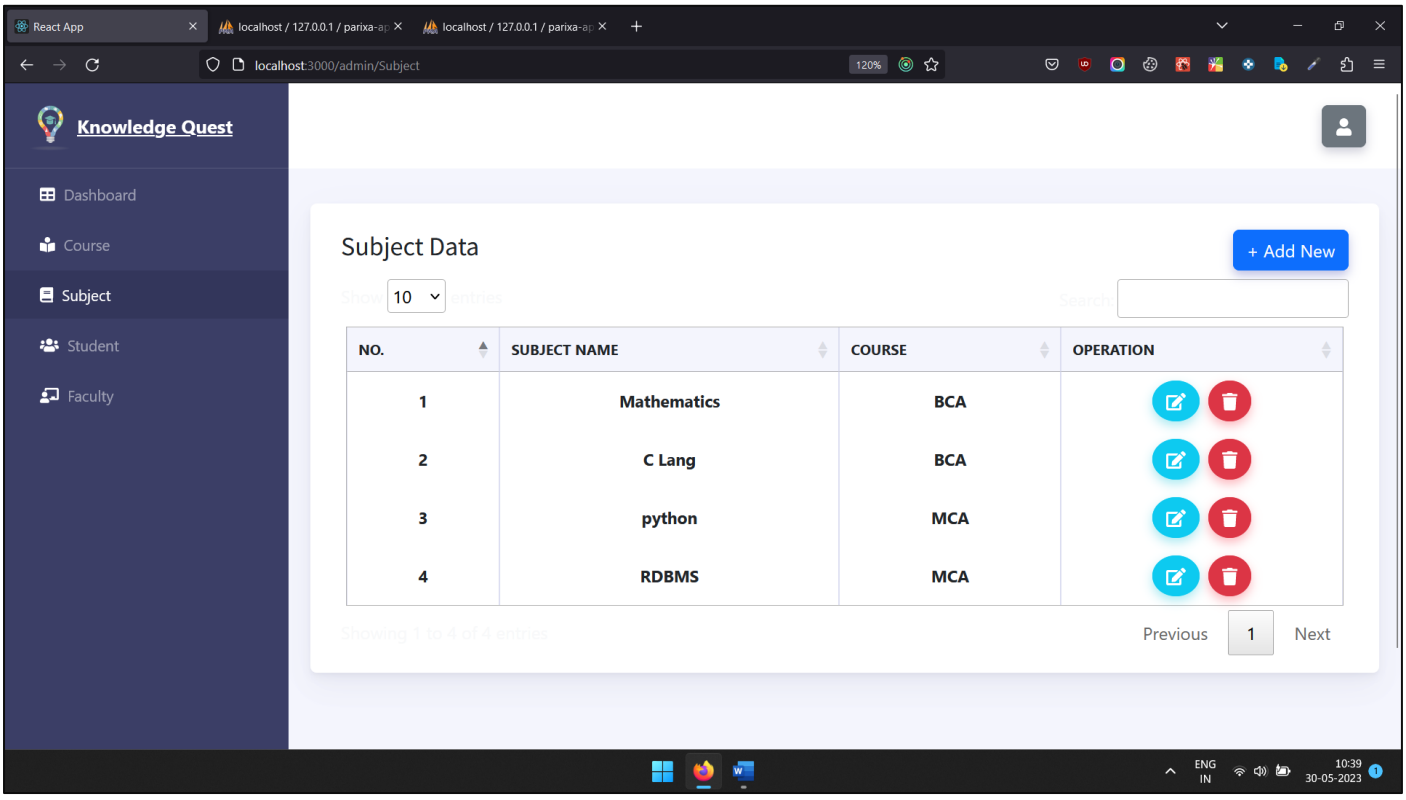
After Added Course



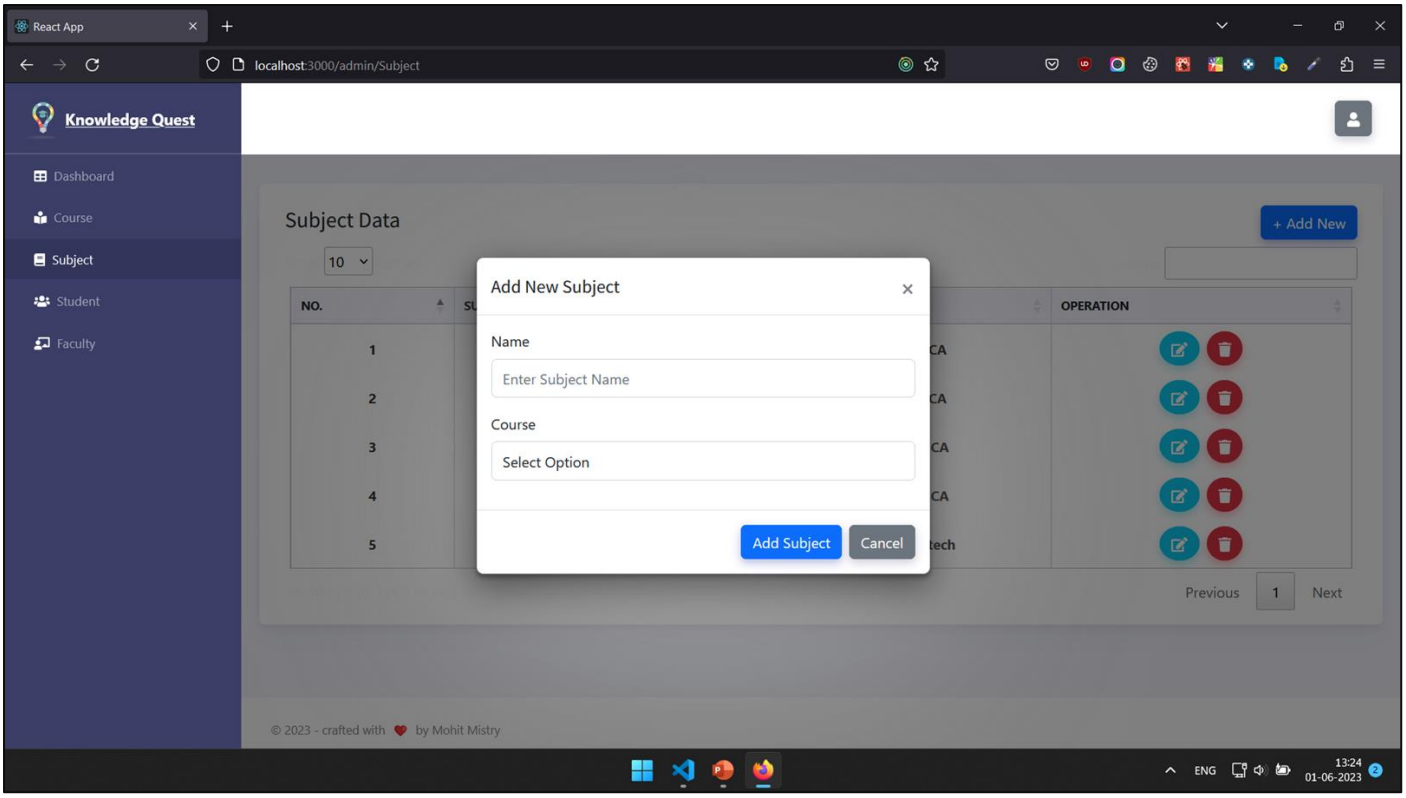
Admin Update Course



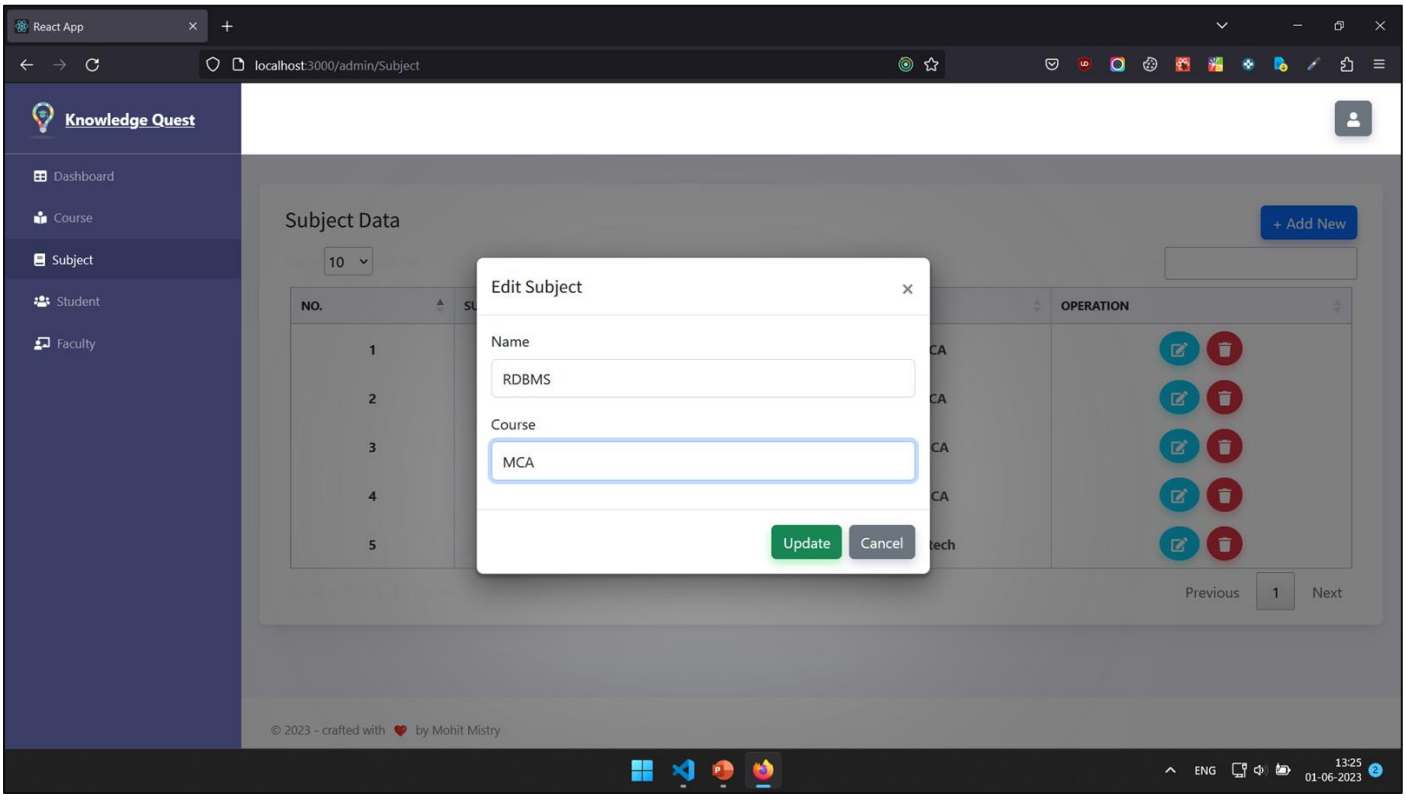
Subjects List



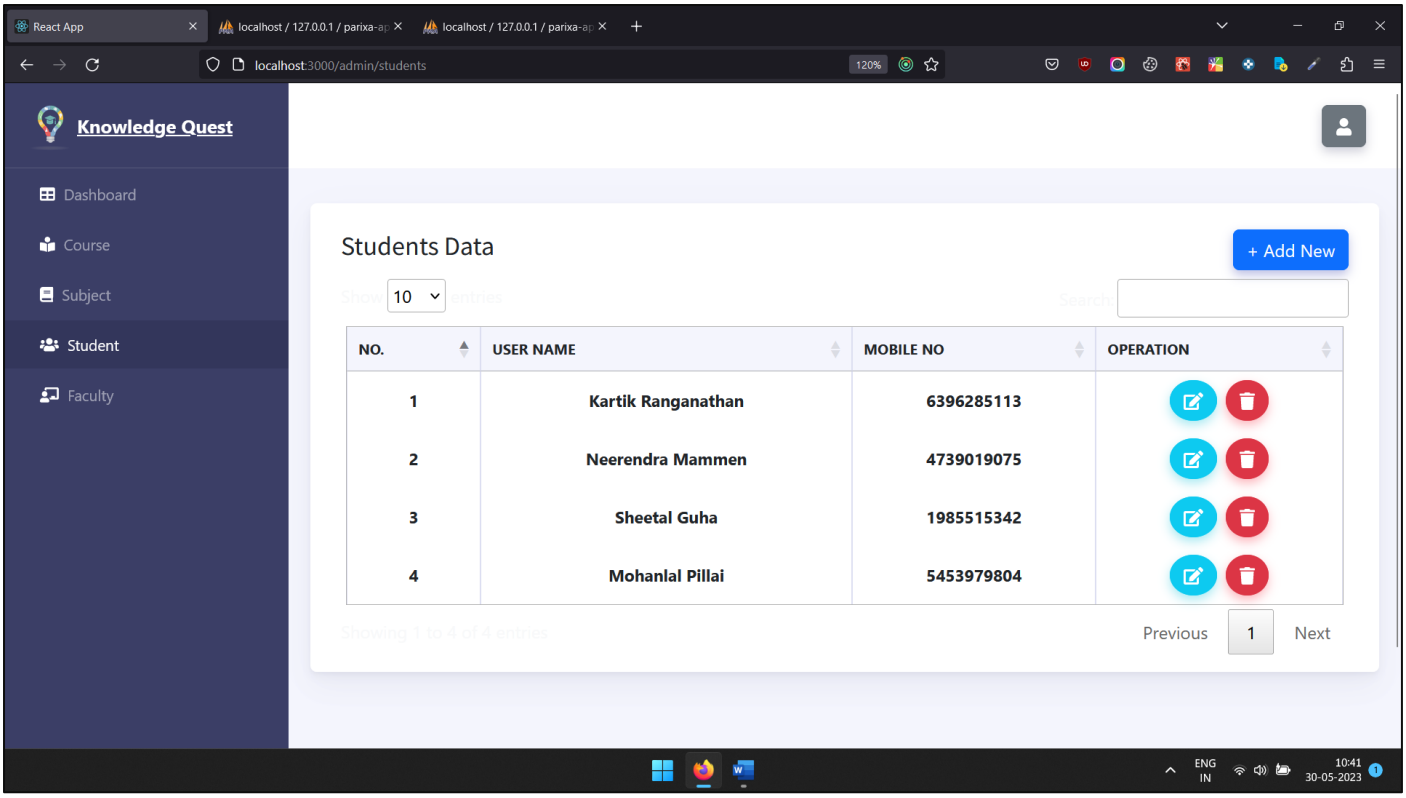
Add New Subject



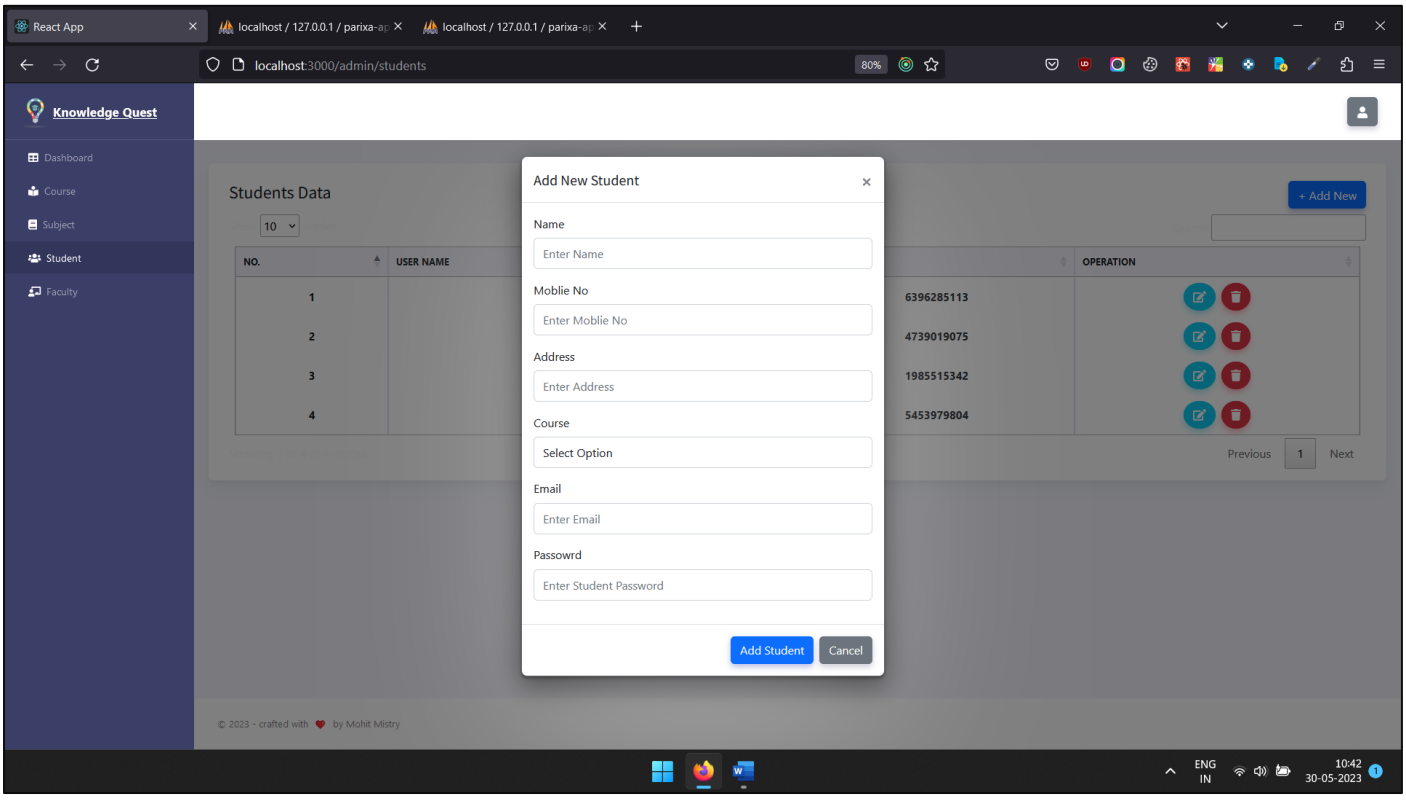
Update Subject



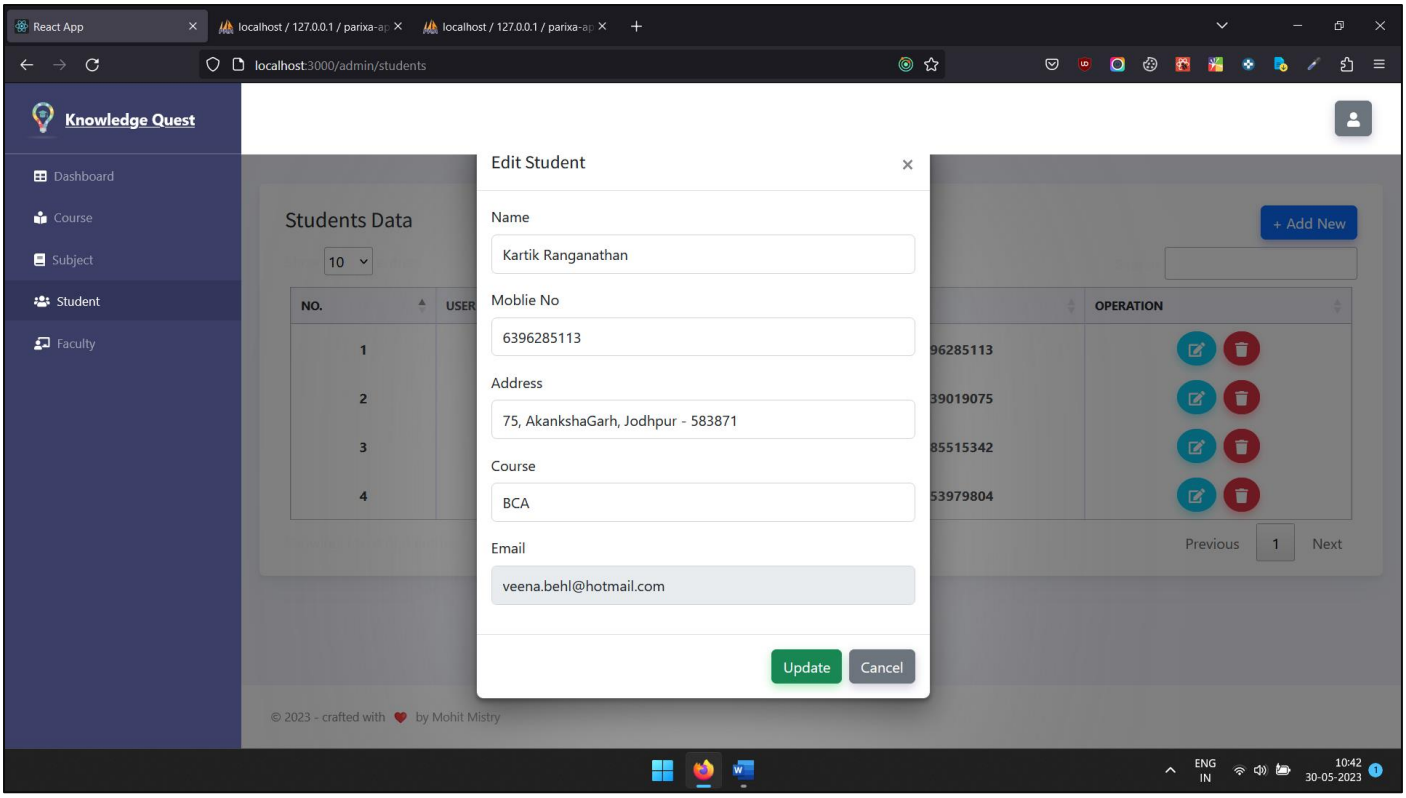
Student List



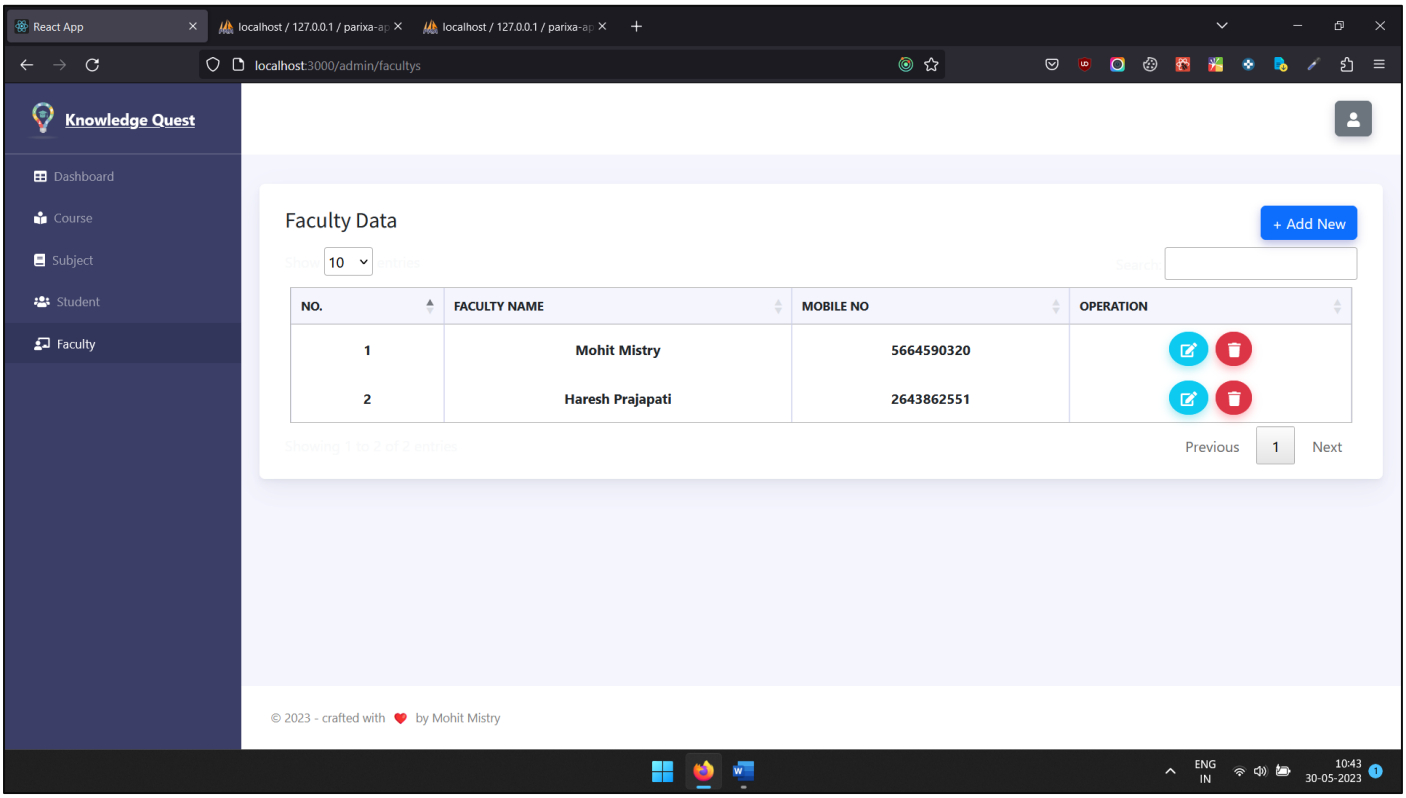
Add New Student



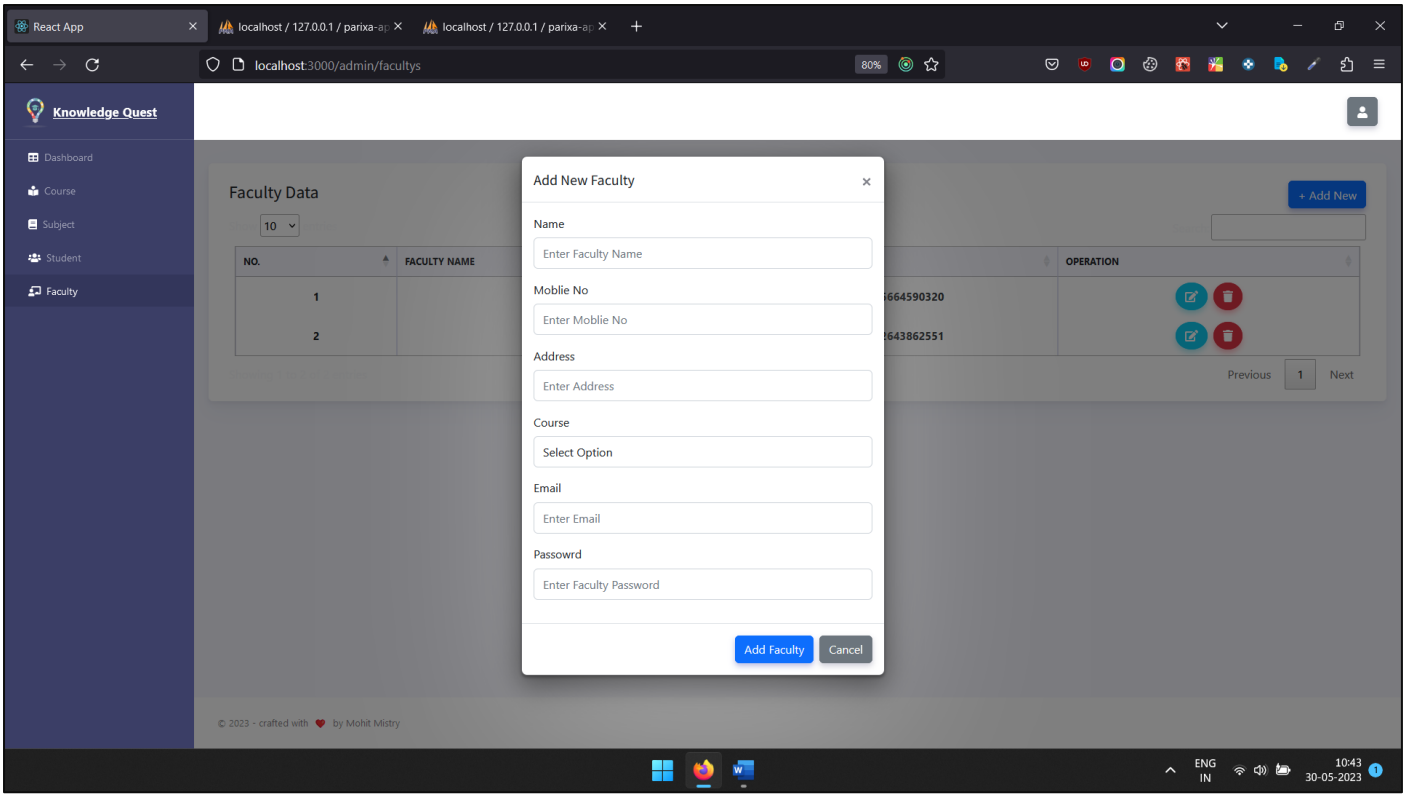
Update Student Data



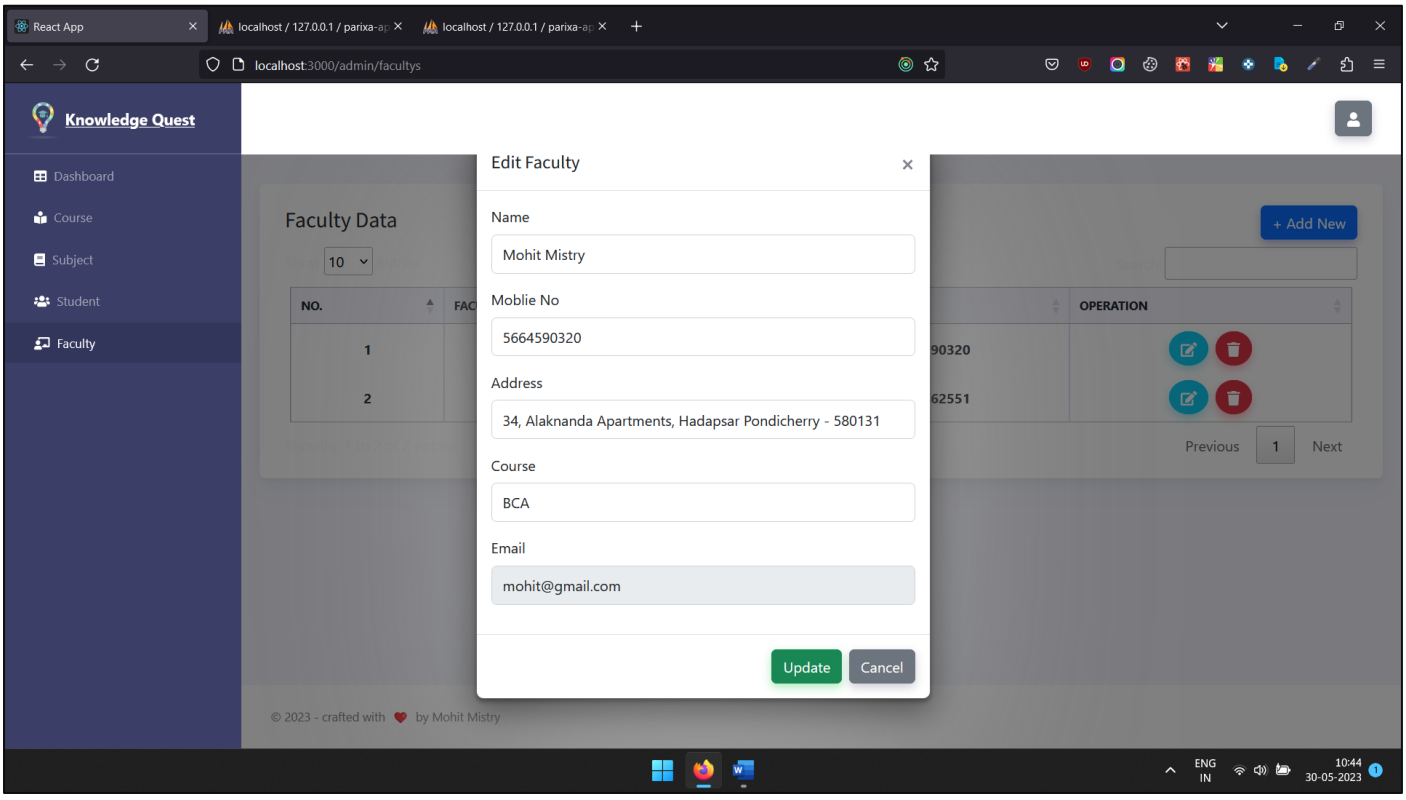
Faculty List



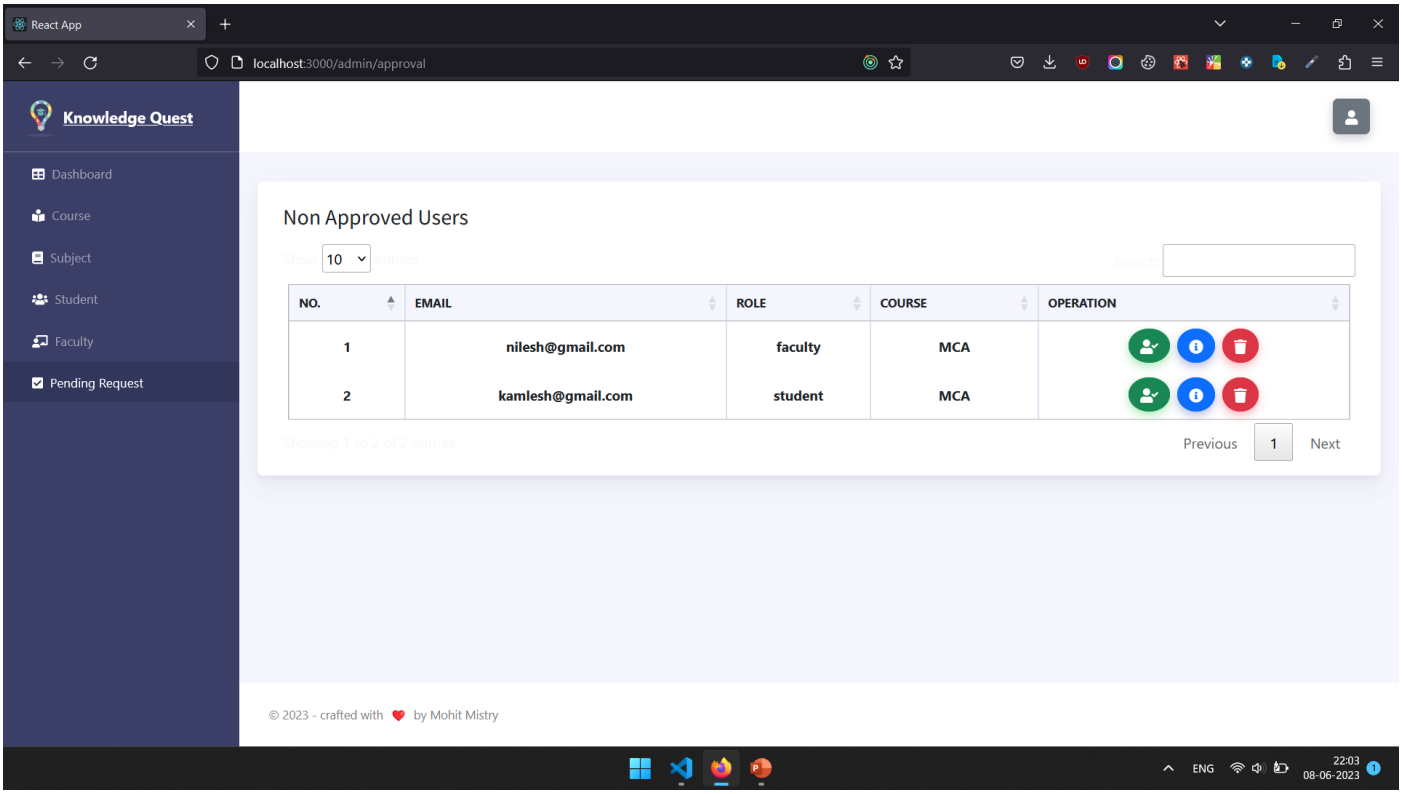
Add New Faculty



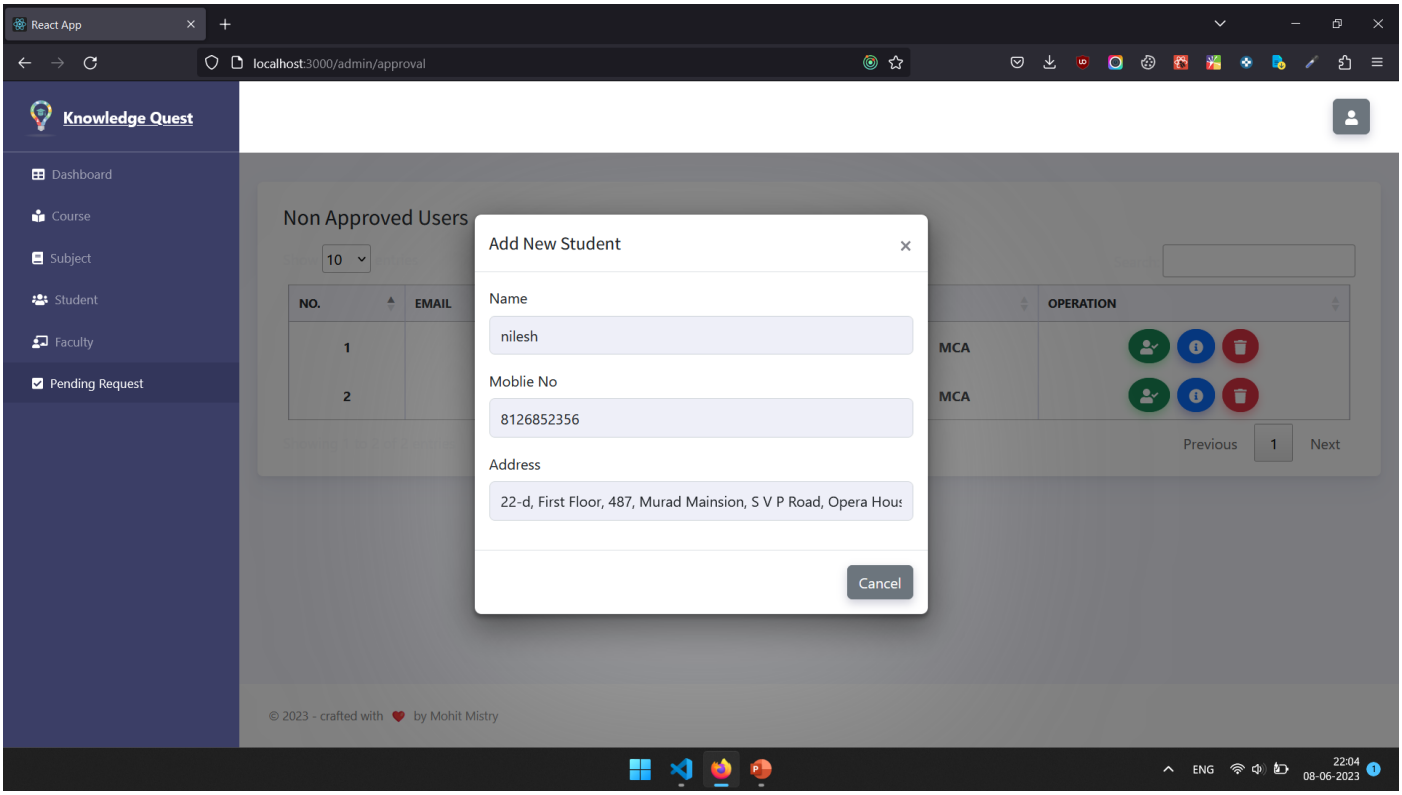
Update Faculty



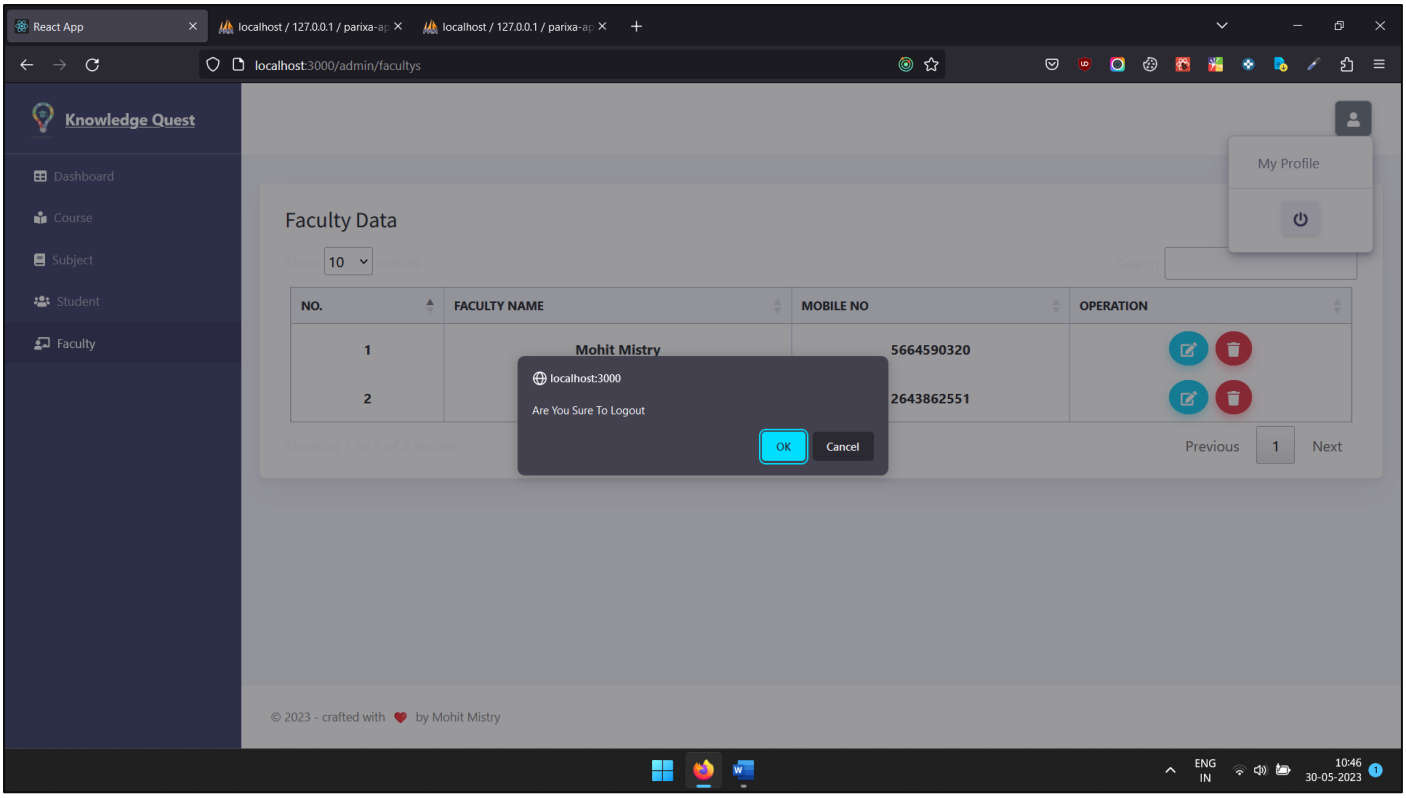
Admin can approve user



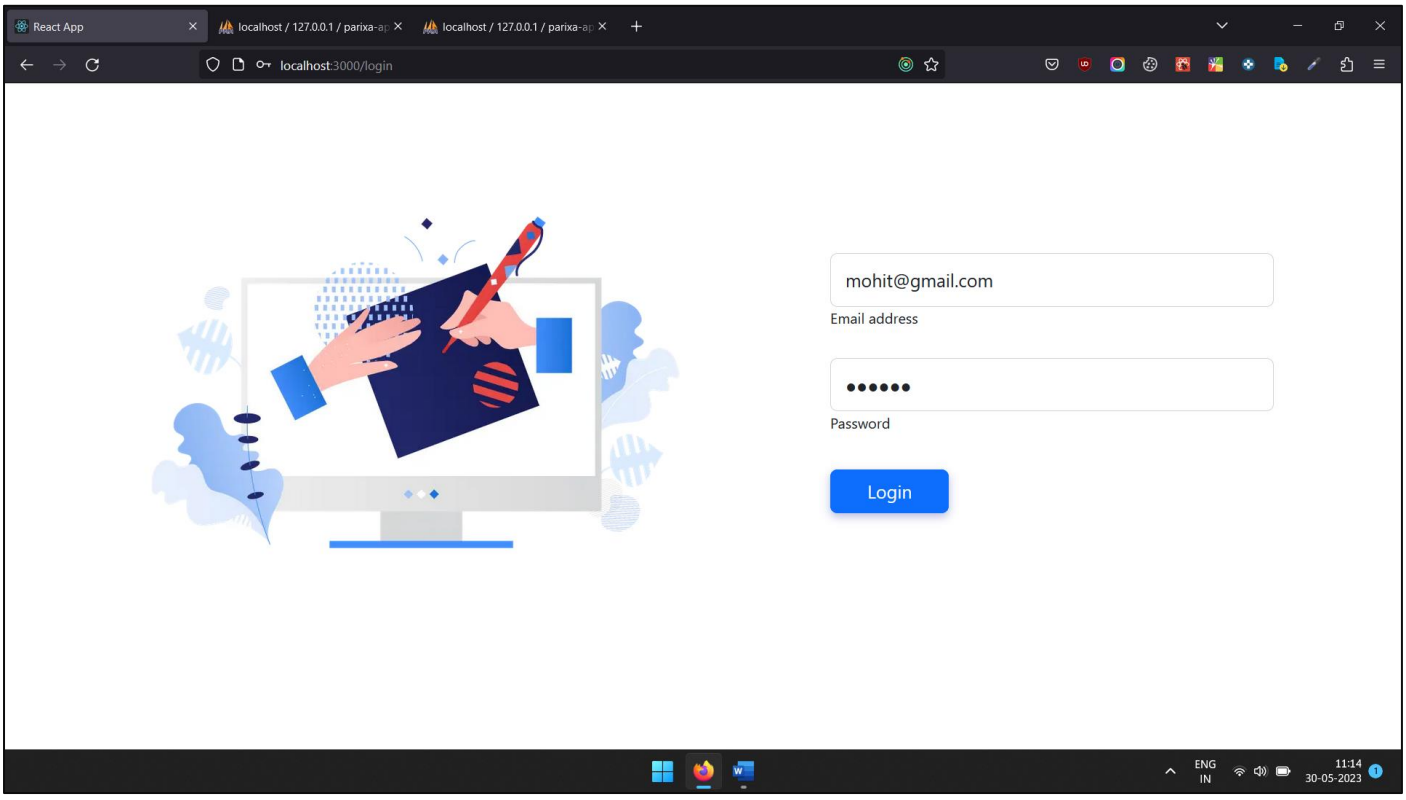
Admin can see information about request user



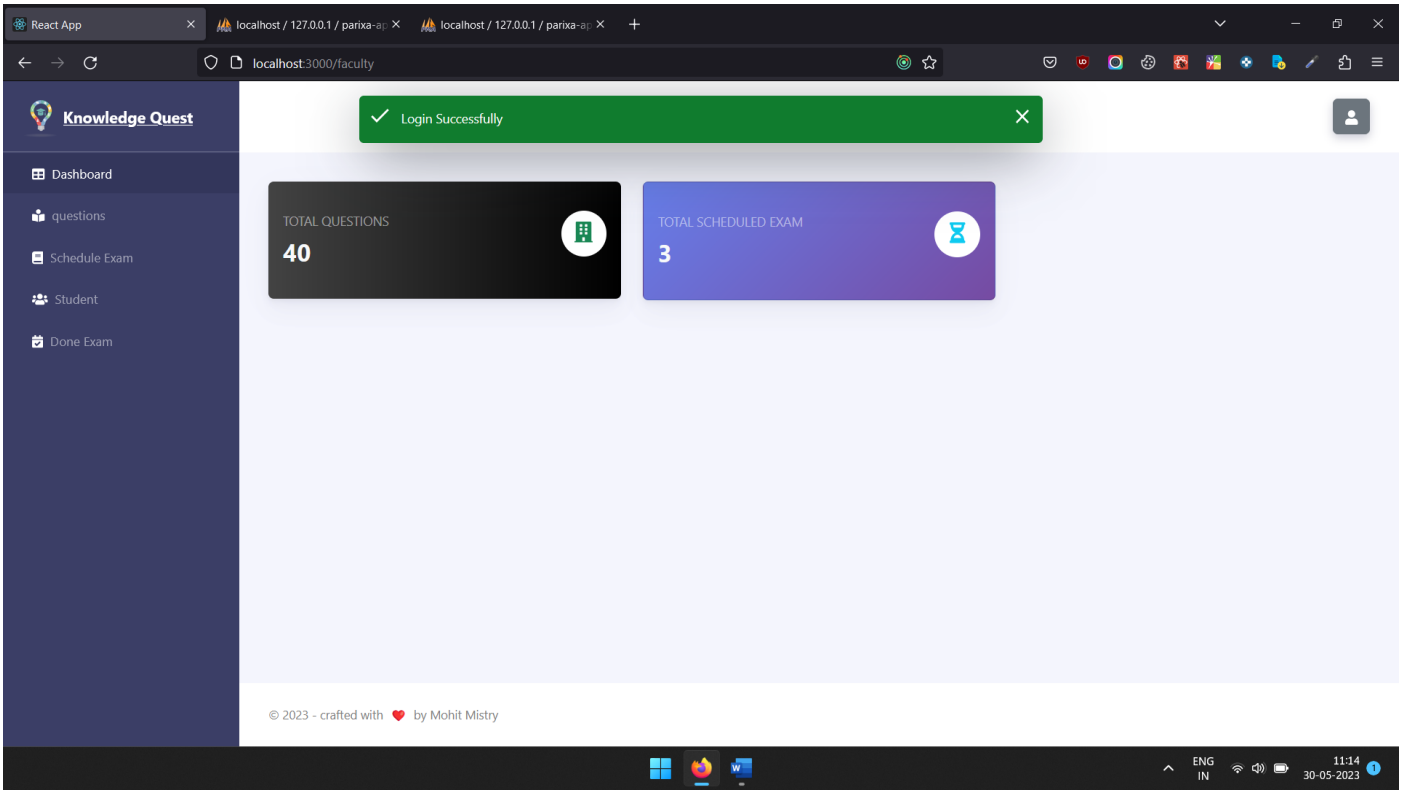
Logout Confirmation



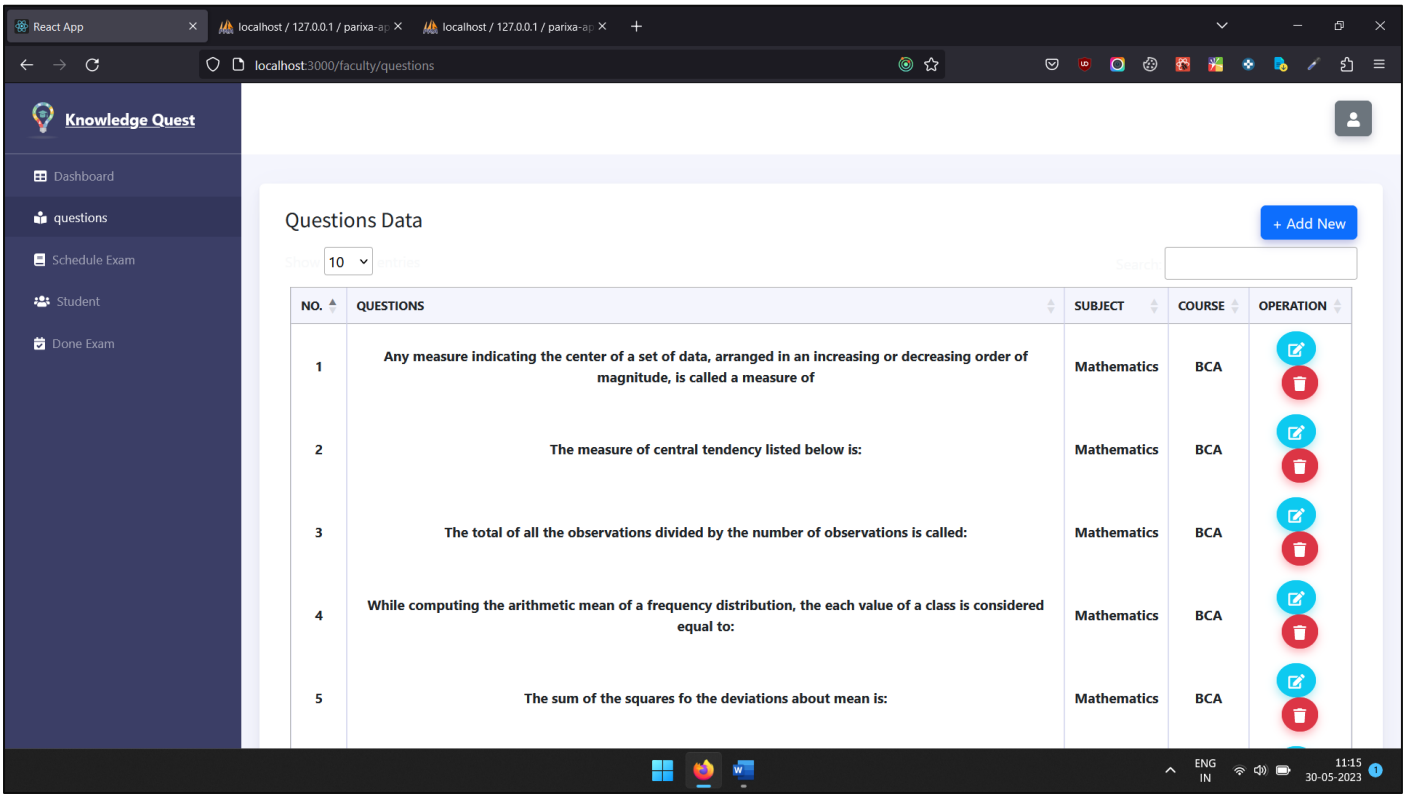
Faculty Login



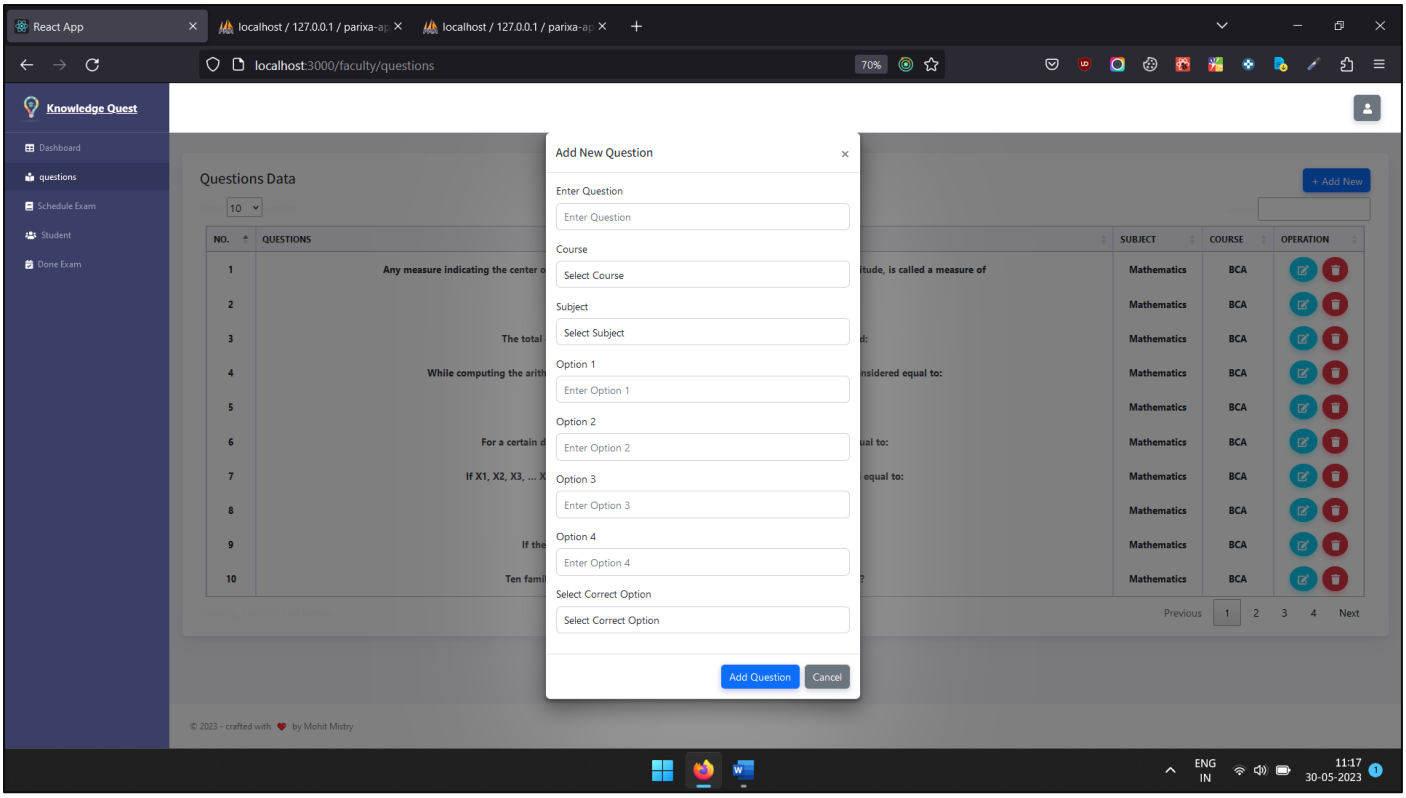
Faculty Dashboard



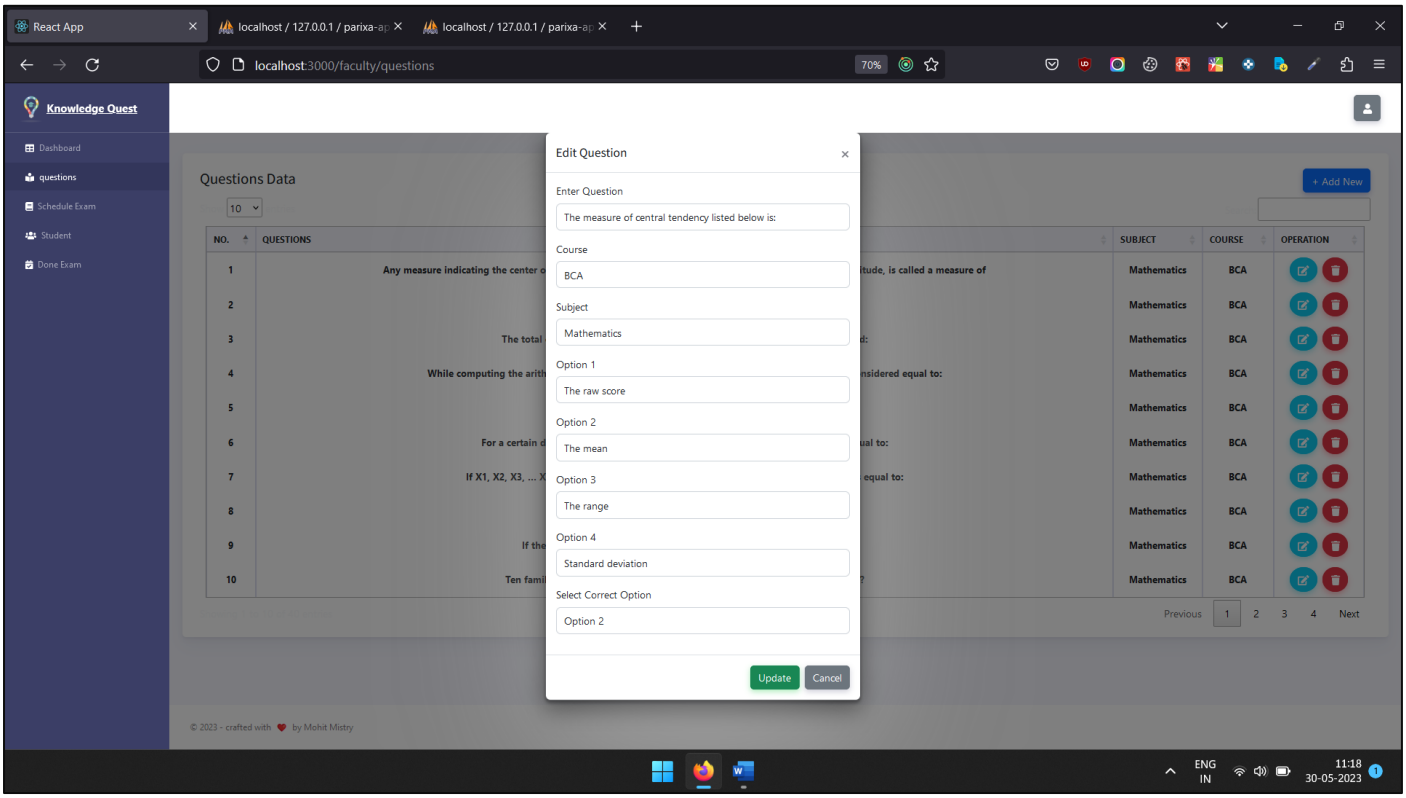
List of questions



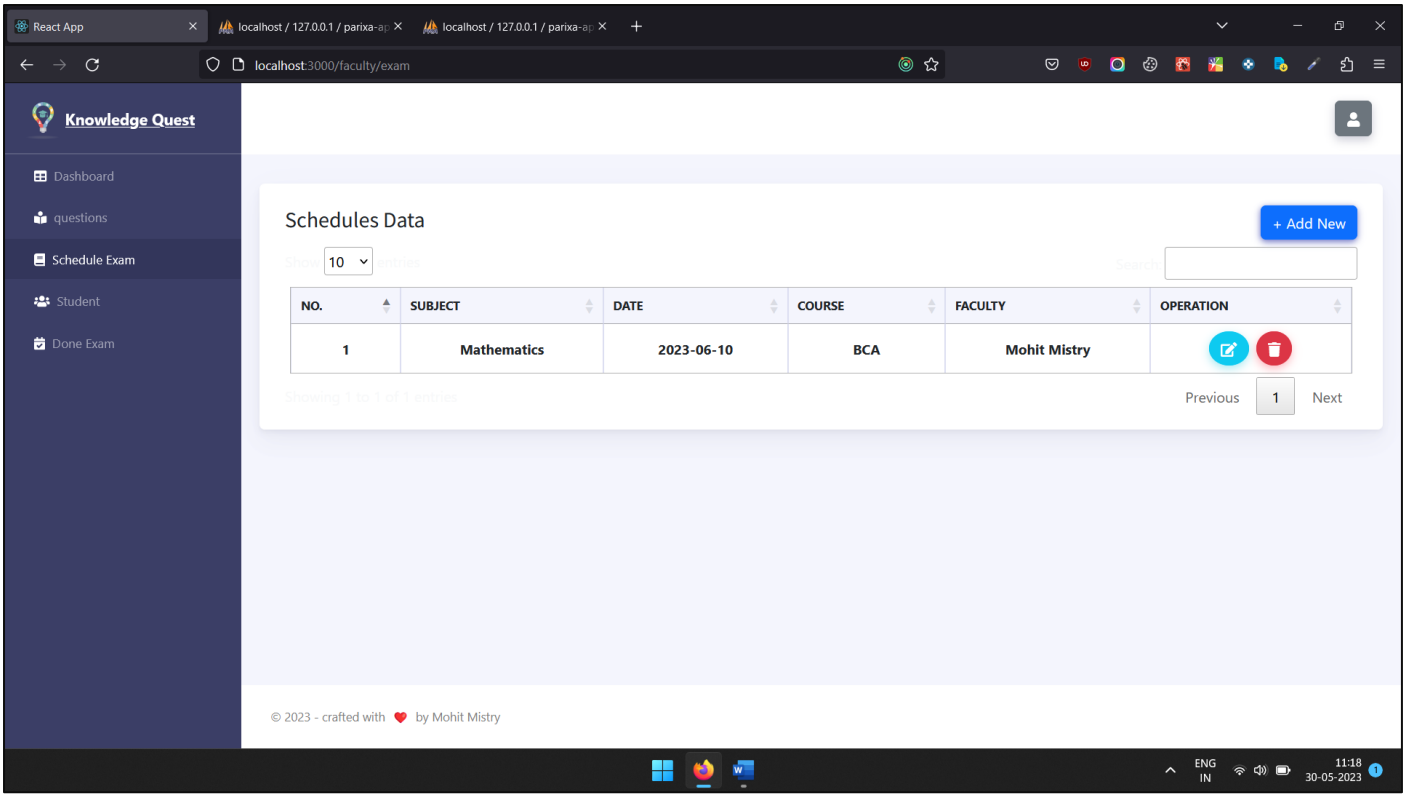
Add New Question



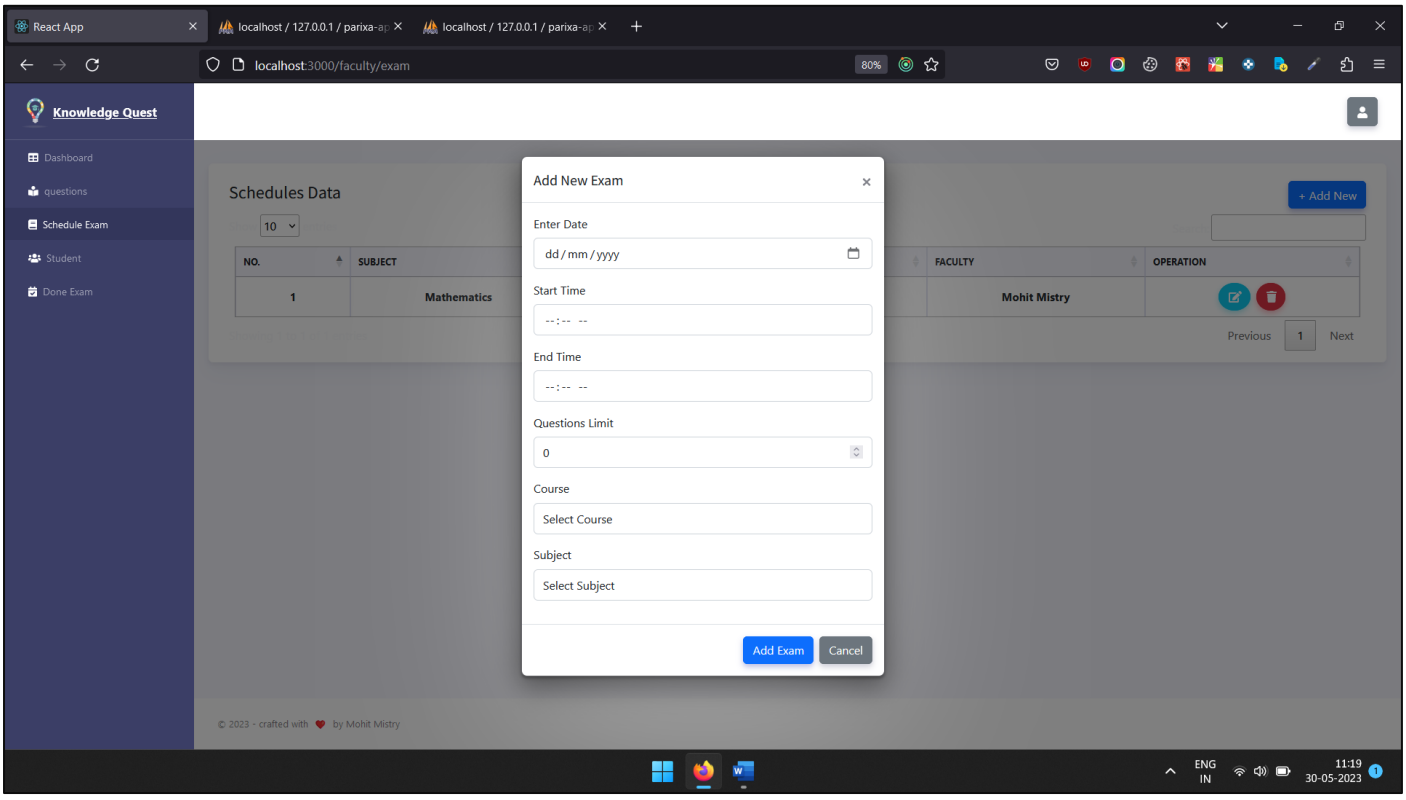
Update Question



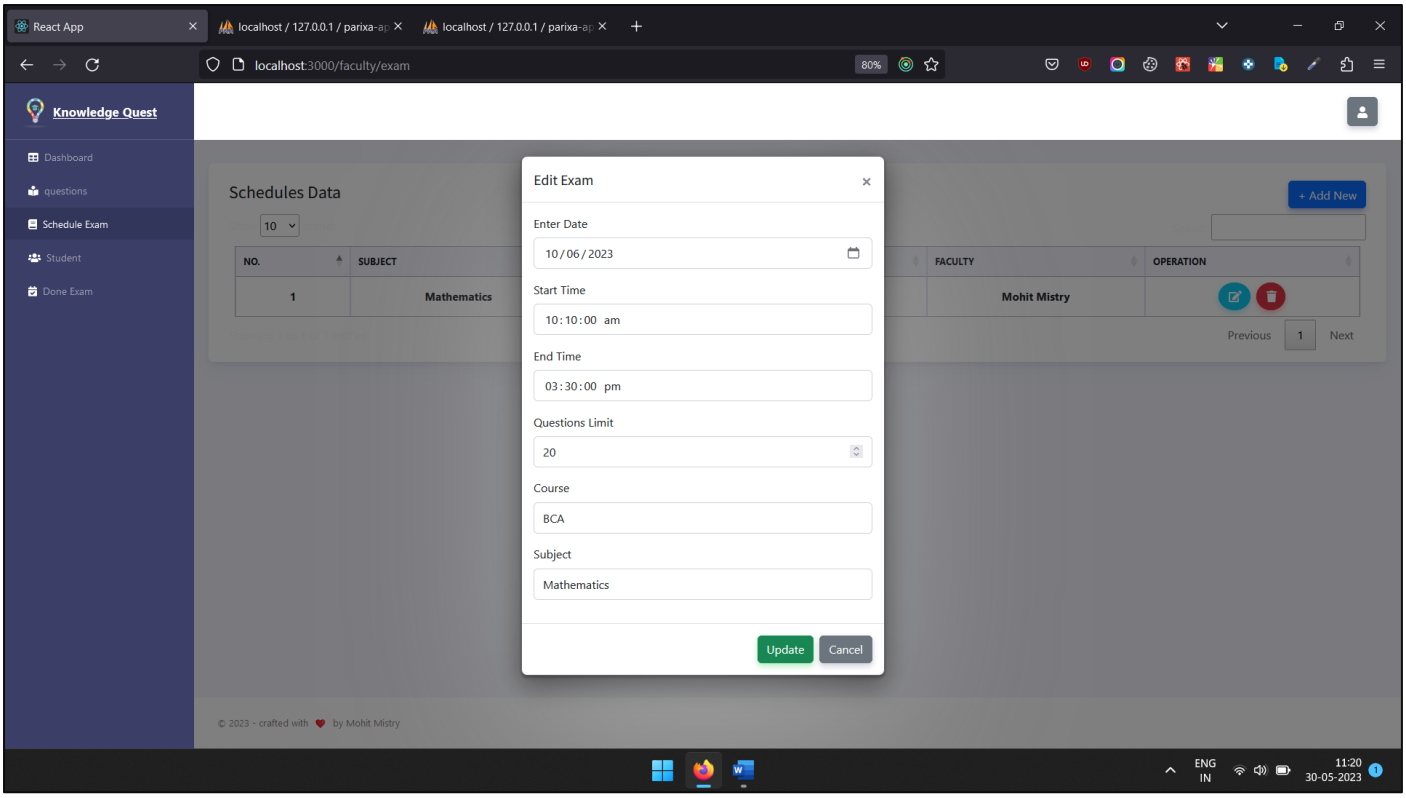
Schedule Future Exam list



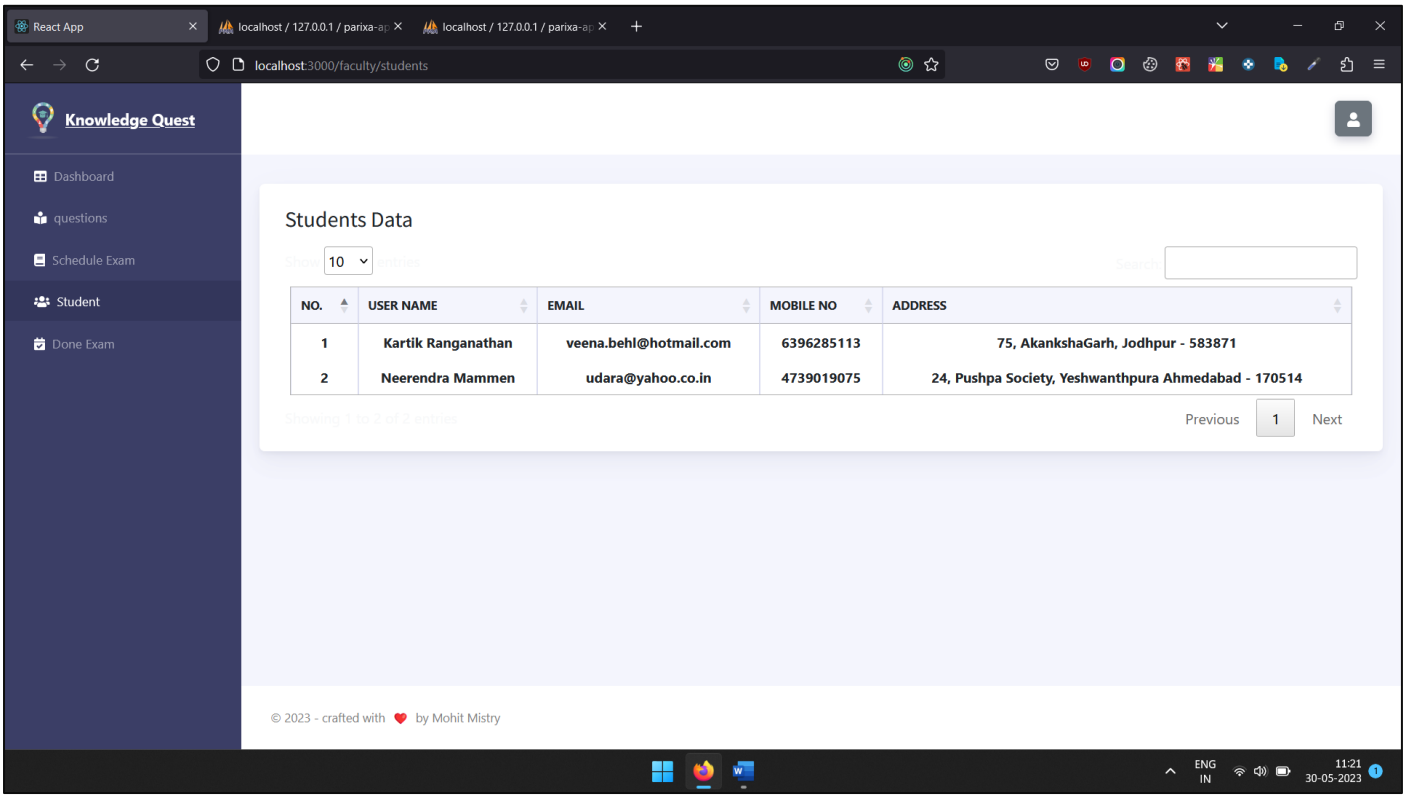
Add or conduct future exam



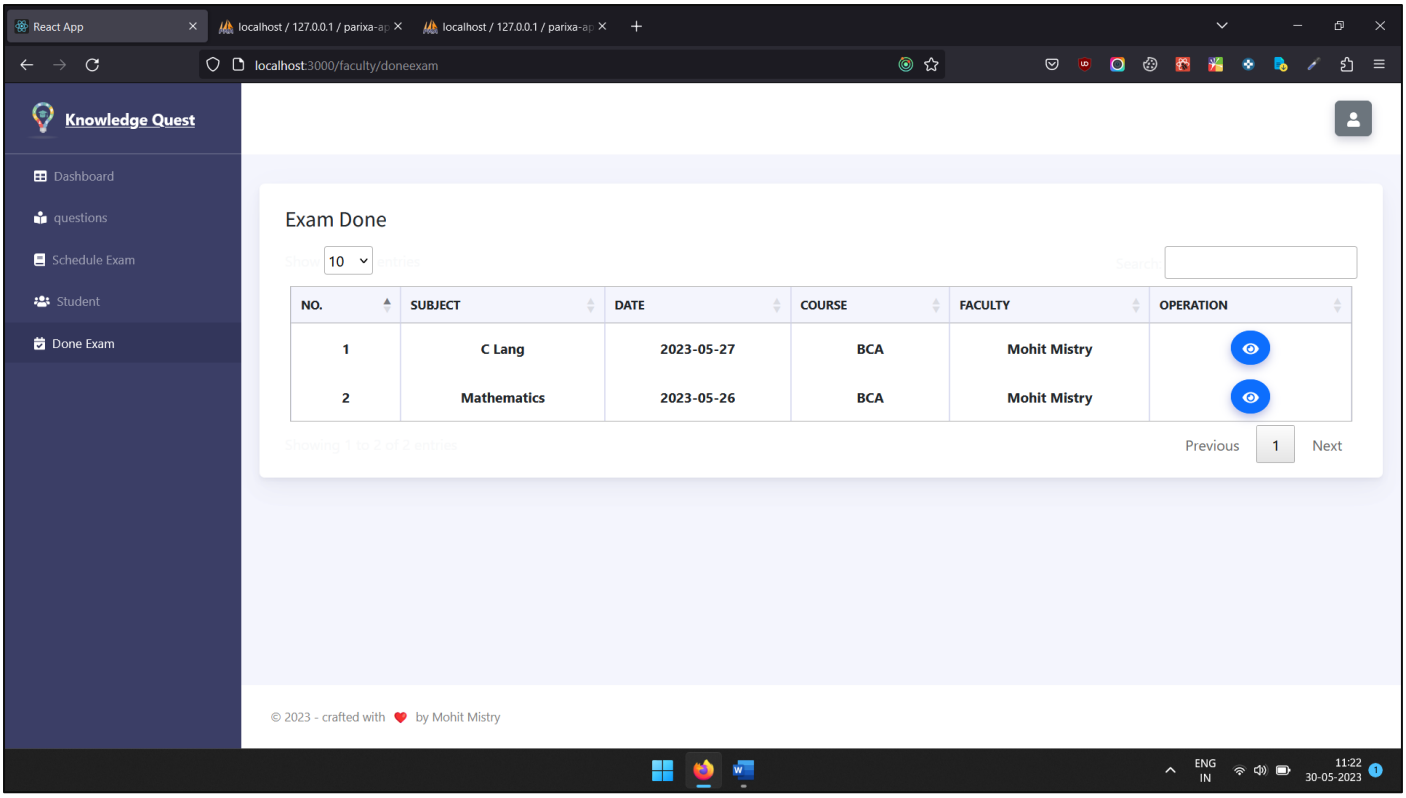
Update Scheduled or Added Exam



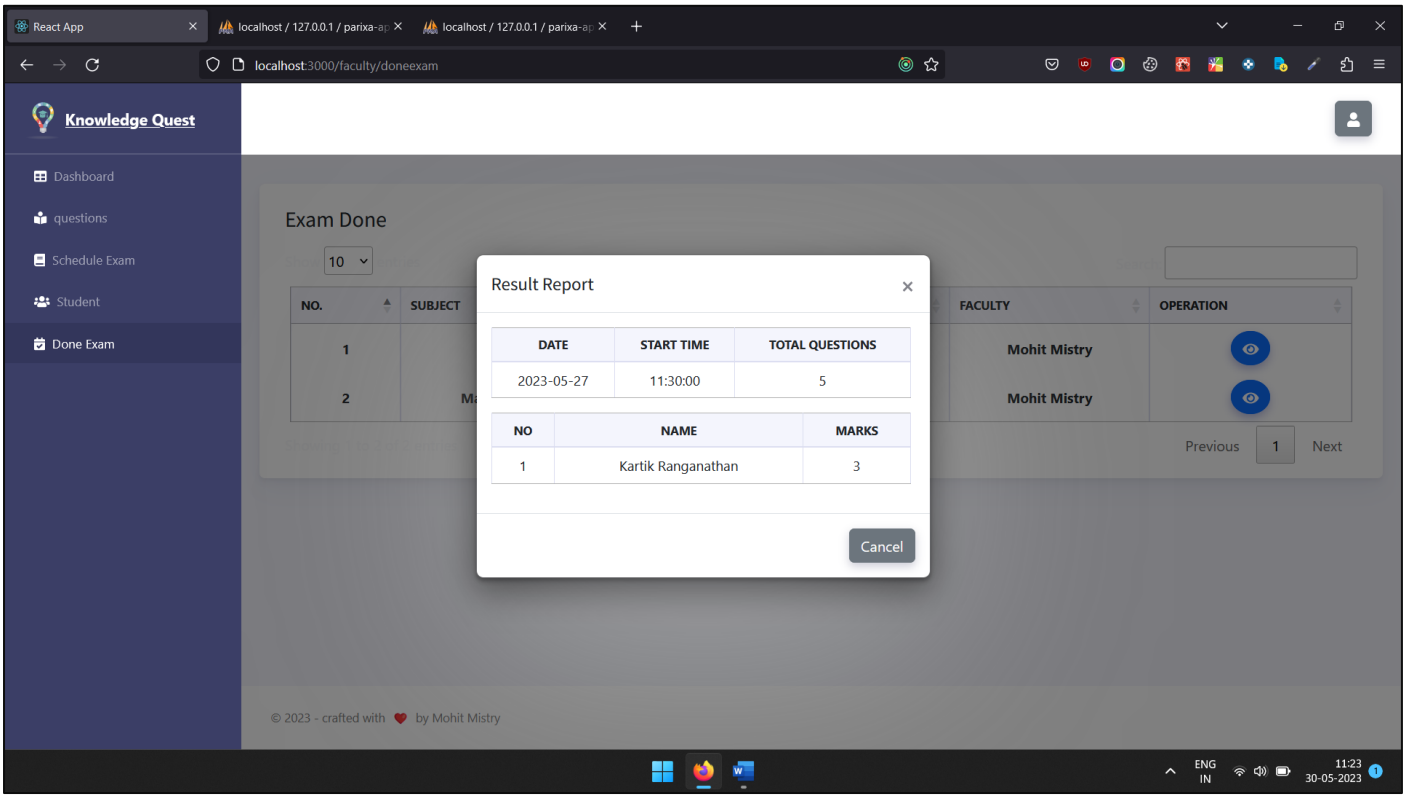
Faculty can see their student list



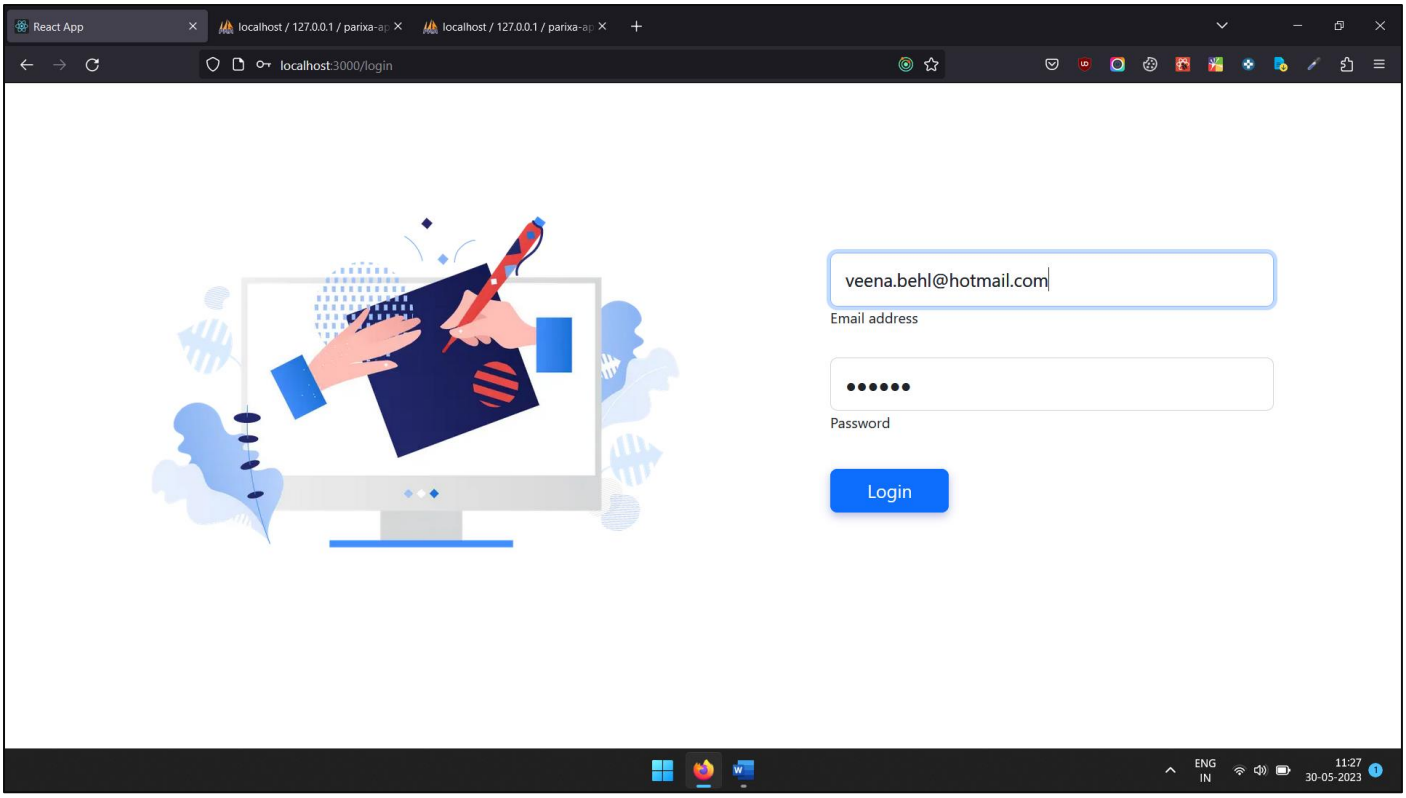
Done Exam List



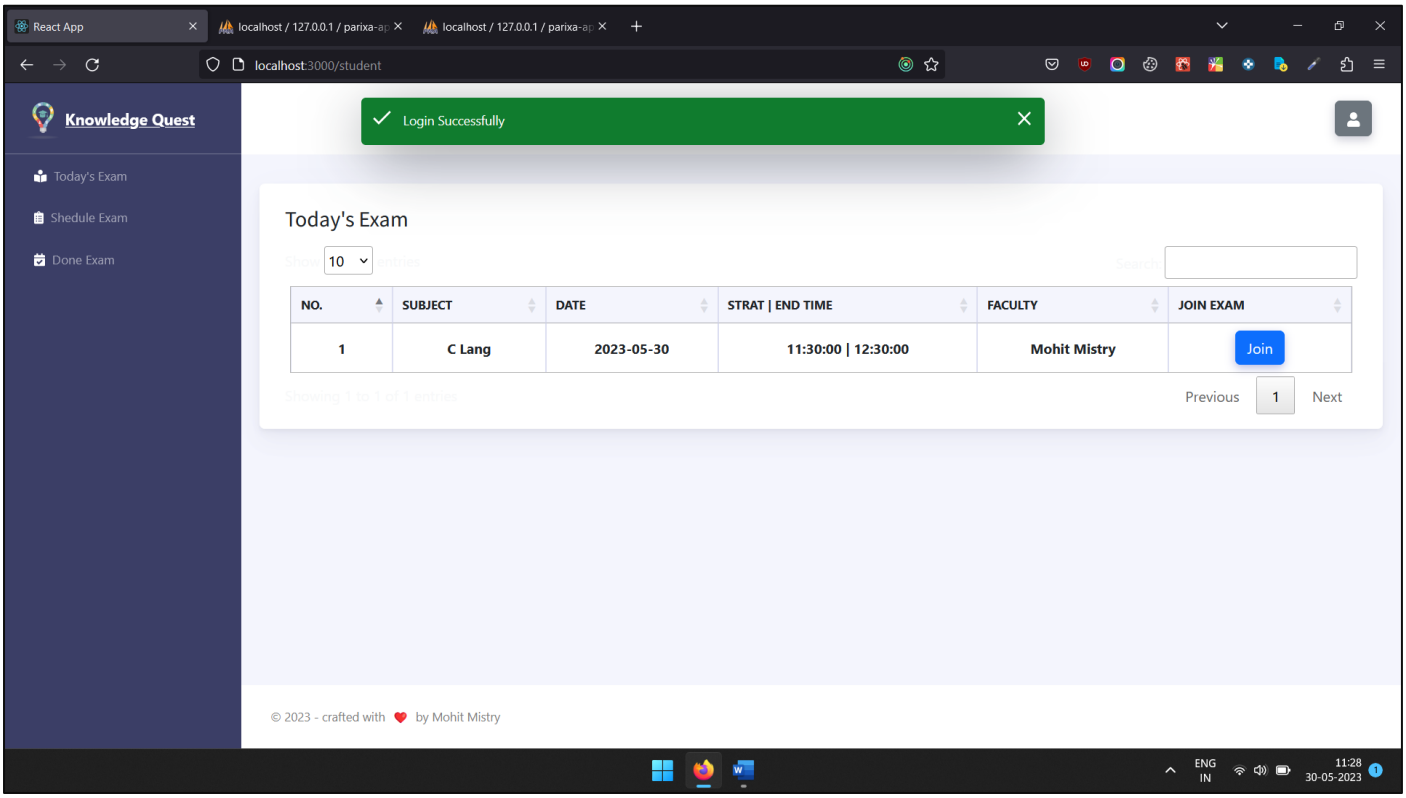
Can see how many student give exam with marks



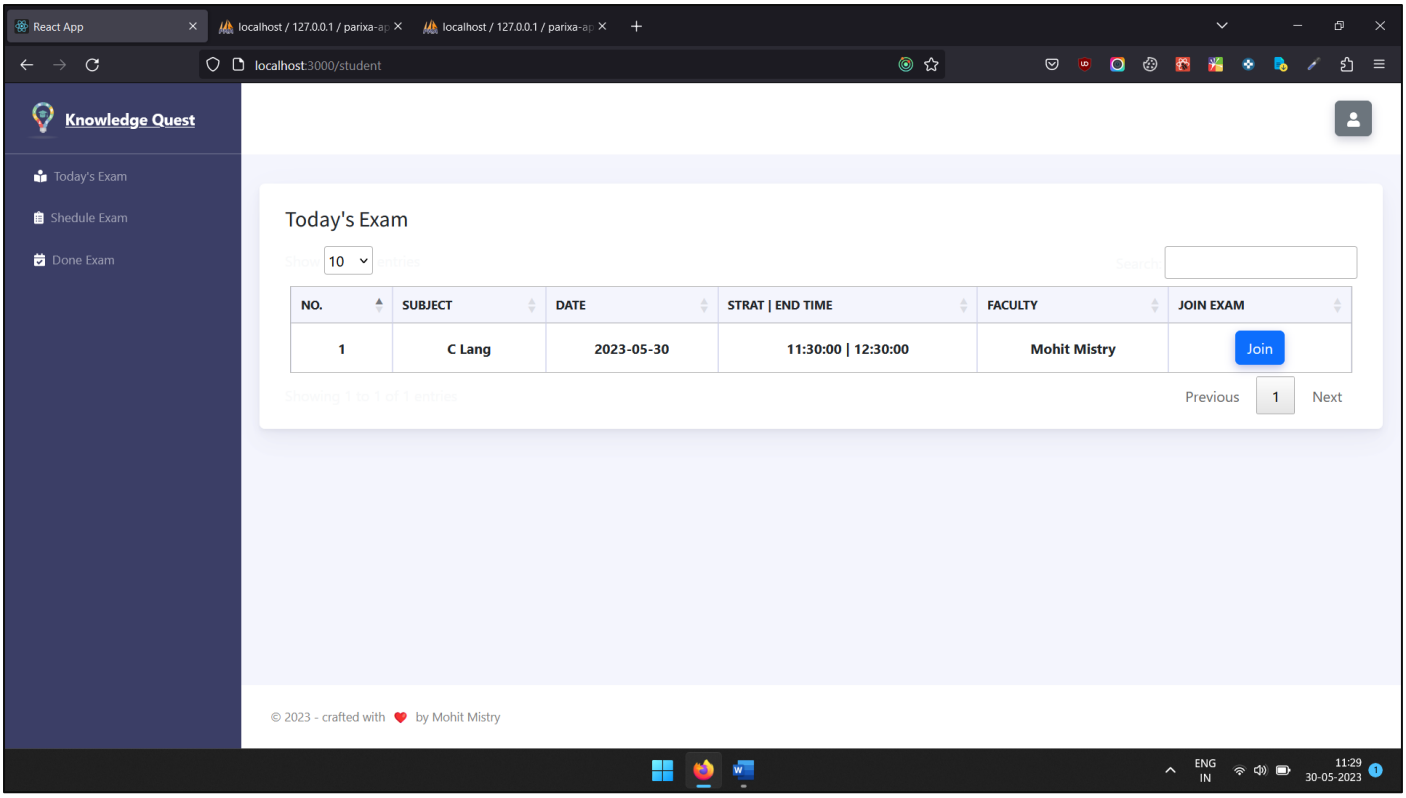
Student Login



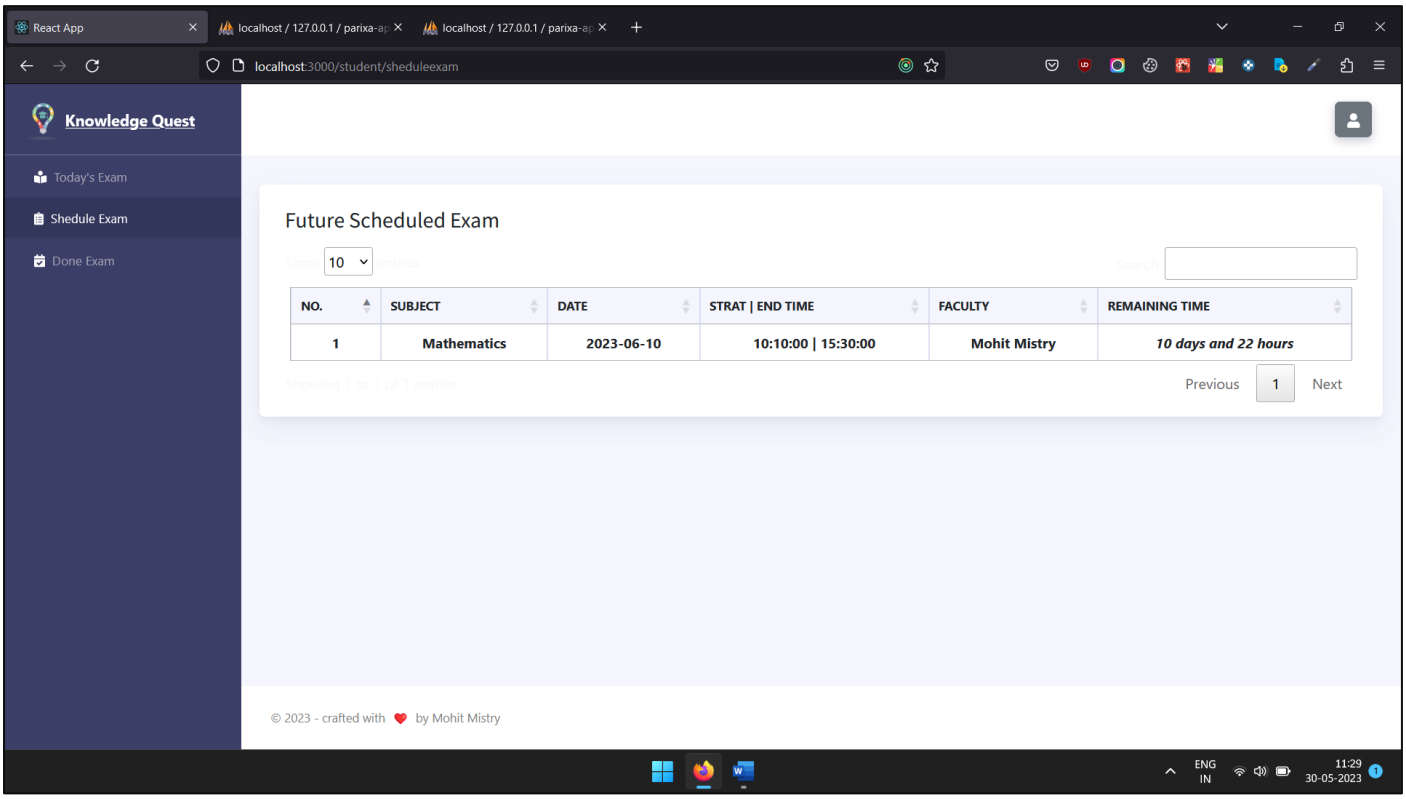
After Login Student Dashboard



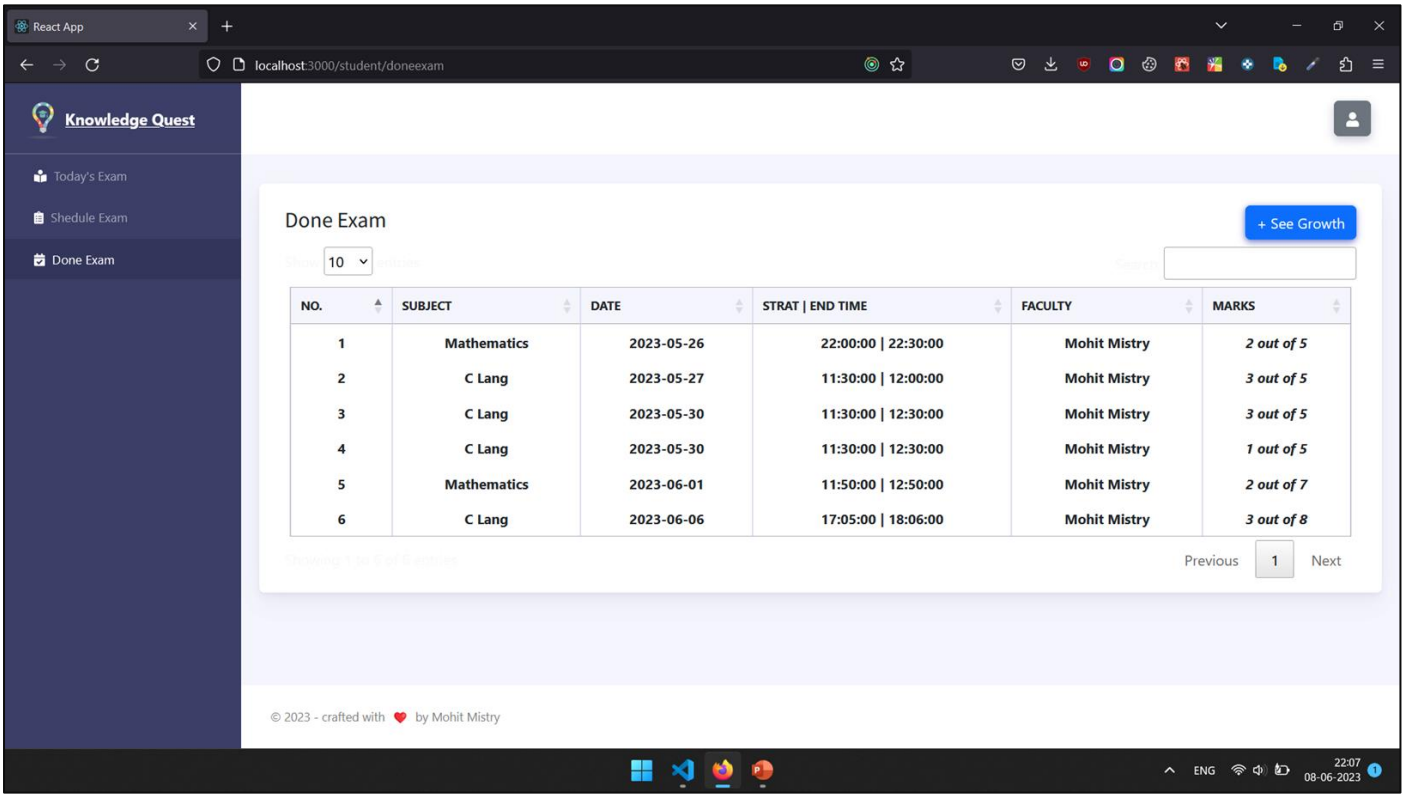
Today's Exam List



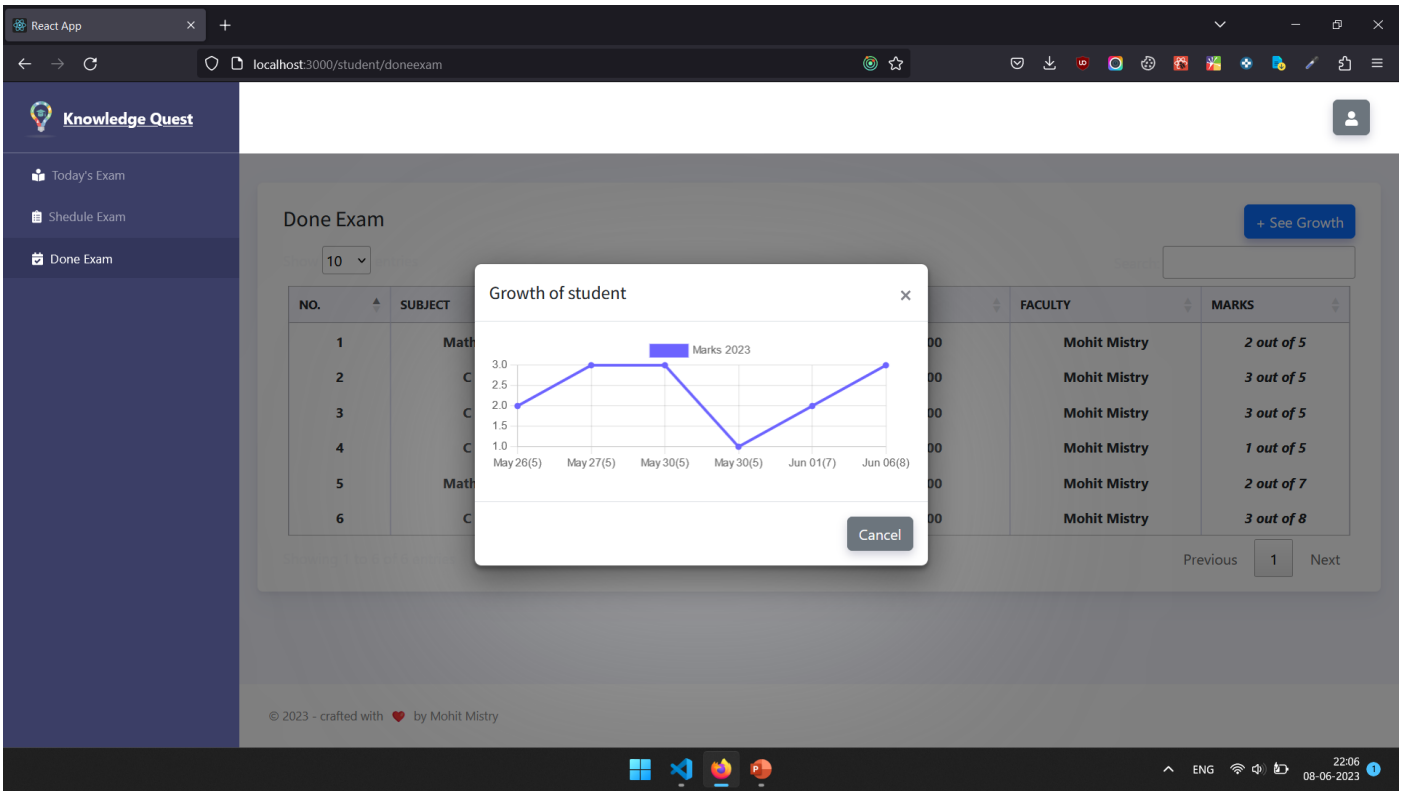
Future Scheduled Exam list



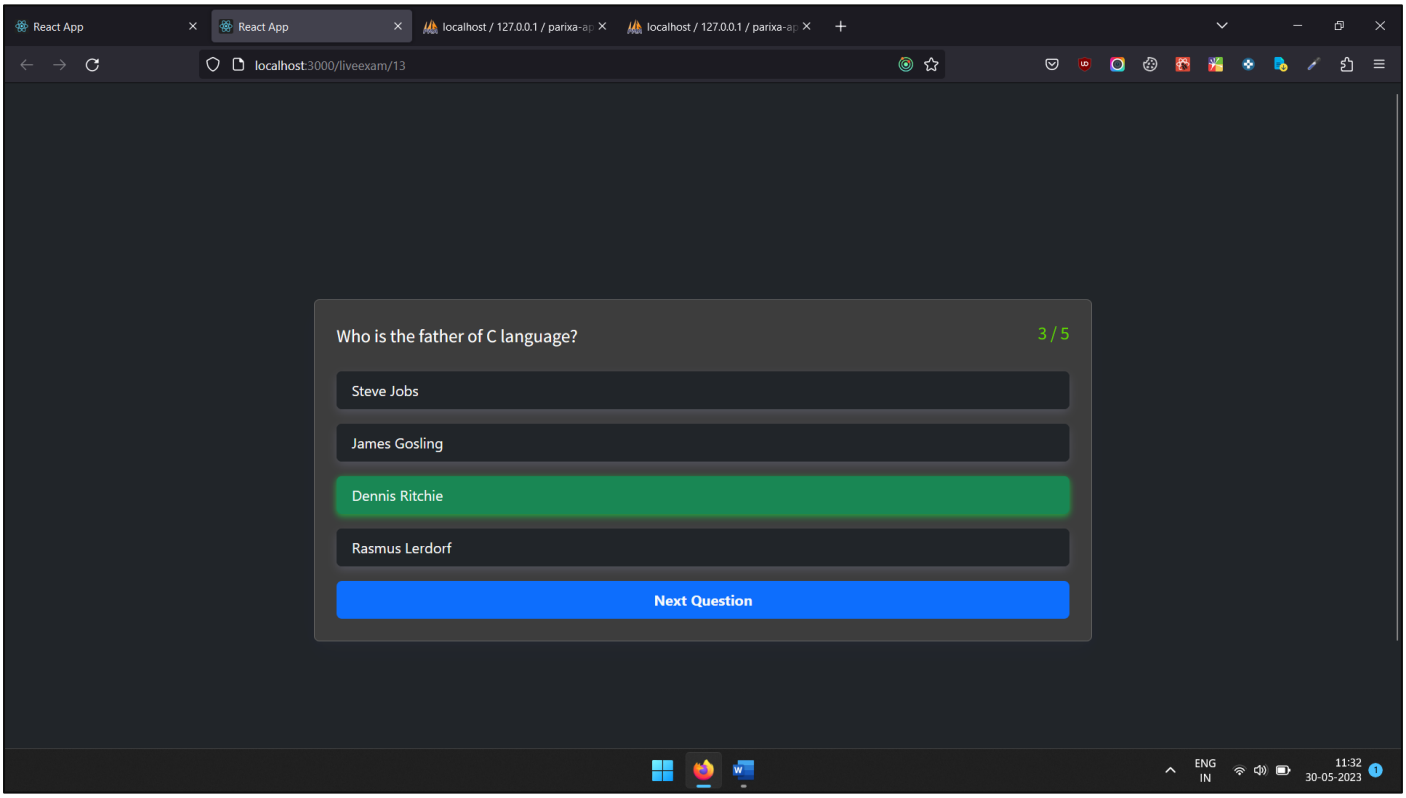
Done Attend Exam List With Marks



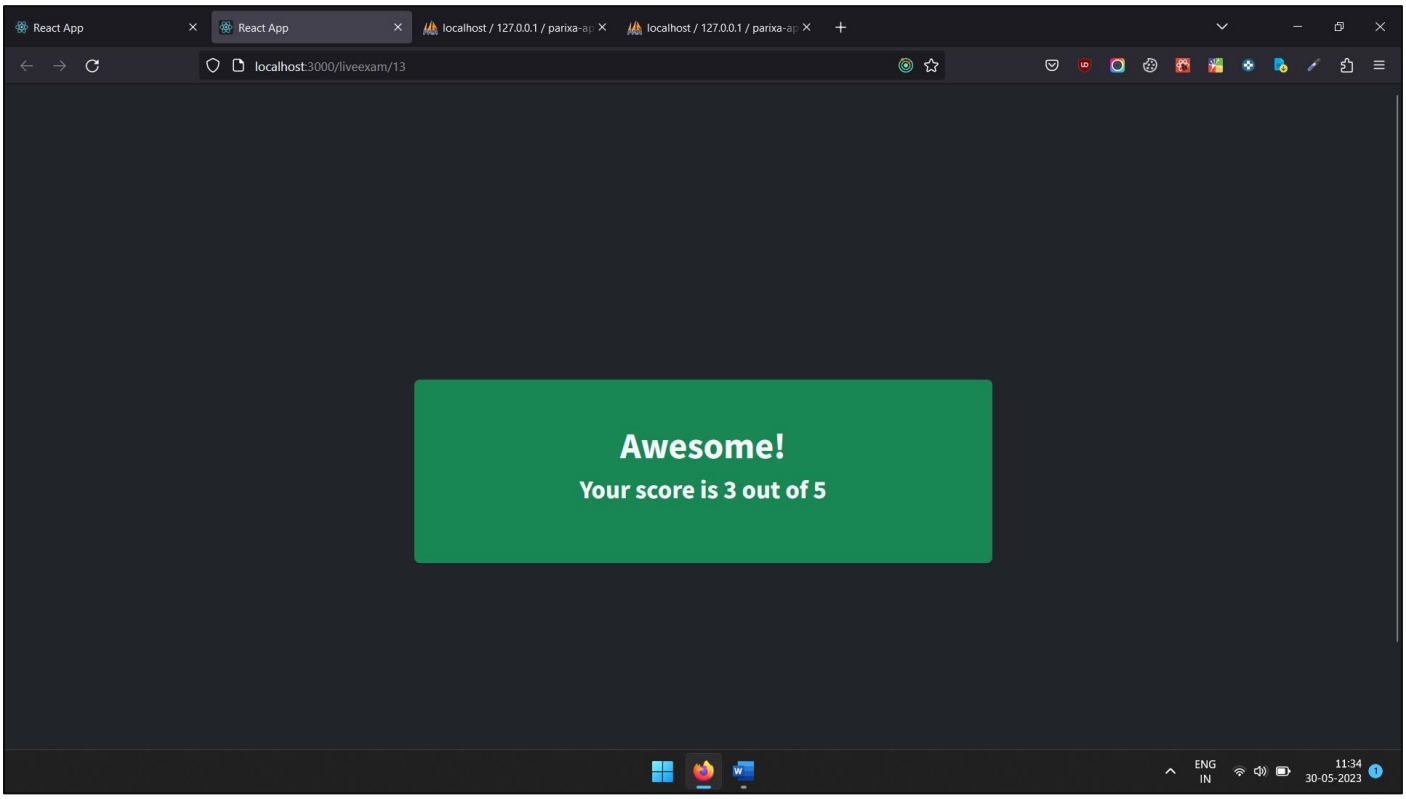
Student can see their growth graph



After Join Today's Exam Student give exam



After Exam it's show result



5. Proposed Enhancements

- No project can be perfect. With change in technology and security measures, updates are required every now and then. All the features in the system are implanted to fill the gap between the person who wants toinvest on a daily basis and the person who has the knowledge about it.
- the online exam management system is no exception. Though it has proven to be an efficient and effective tool for conducting exams remotely, there is always room for improvement.
- One proposed enhancement for the system could be the implementation of AI-based proctoring. With advancements in artificial intelligence, it is now possible to monitor exam takers more accurately and prevent cheating in a more sophisticated manner. This would not only ensure the integrity of exams but also save time and resources for the administrators.
- Another potential enhancement could be the integration of biometric authentication for exam takers. This would provide an added layer of security and prevent impersonation or identity fraud during the exam. With the use of facial recognition technology, fingerprint scanning, or voice recognition, the system can verify the identity of the test-taker before allowing them to start the exam.
- In addition, the system could benefit from improved accessibility features for differently-abled users. The exams could be designed to accommodate various disabilities including visual impairment, hearing loss, or motor disabilities. This would ensure that everyone has equal opportunities to take the exam and demonstrate their knowledge regardless of their physical limitations.
- Overall, these enhancements to the online exam management system would improve its functionality, efficiency, and security, making it a more reliable tool for exams.

6. Conclusion

- It is an externally wonderful movement while concluding this report. This my first experience to perform such professional work. Objectives of this project were to satisfy user's requirement, successful implementation of the system, design a user friendly and easy to operate system.
- This software provides appropriate information to user.
- This project is developed for material management.