EduQuiz An Online Examination System

A Major Project Report Submitted to
Rajiv Gandhi Proudyogiki Vishwavidyalaya

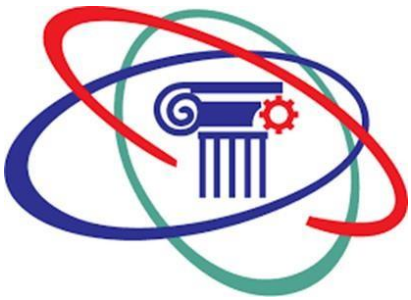
Towards Partial Fulfillment for the Award of
Bachelor of Technology in Computer Science
& Engineering

Submitted by:

Aadesh Garg (0827CS201001)
Aashish Pagare (0827CS201005)
Aman Verma (0827CS201028)
Bhumika Patidar (0827CS201060)

Guided by:

Prof. Juhi Shrivastava
Assistant Professor
Computer Science & Engg.



Acropolis Institute of Technology & Research, Indore

Jul - Dec 2023

EXAMINER APPROVAL

The Major Project titled ***"EduQuiz : An Online Examination System"*** submitted by **Aadesh Garg (0827CS201001), Aashish Pagare (0827CS201005), Aman Verma (0827CS201028), Bhumika Patidar (0827CS201060)** has been examined and is hereby approved towards partial fulfillment for the award of ***Bachelor of Technology degree in Computer Science Engineering*** discipline, for which it has been submitted. It is understood that by this approval the undersigned do not necessarily endorse or approve any statement made, opinion expressed, or conclusion drawn therein, but approve the project only for the purpose for which it has been submitted.

(Internal Examiner) **(External Examiner)**

Date: **Date:**

RECOMMENDATION

This is to certify that the work embodied in this major project entitled **"*EduQuiz : An Online Examination System*"** submitted by **Aadesh Garg (0827CS201001), Aashish Pagare (0827CS201005), Aman Verma (0827CS201028), Bhumika Patidar (0827CS201060)** is a satisfactory account of the bonafide work done under the supervision of **Prof. Juhi Shrivastava**, is recommended towards partial fulfillment for the award of the Bachelor of Technology (Computer Science Engineering) degree by Rajiv Gandhi Proudyogiki Vishwavidhyalaya, Bhopal.

(Project Guide)

(Project Coordinator)

(Dean Academics)

STUDENTS' UNDERTAKING

This is to certify that the major project titled ***"EduQuiz : An Onilne Examination System "*** has developed by us under the supervision of **Prof. Juhi Shrivastava**. The whole responsibility of the work done in this project is ours. The sole intension of this work is only for practical learning and research.

We further declare that to the best of our knowledge; this report does not contain any part of any work which has been submitted for the award of any degree either in this University or in any other University / Deemed University without proper citation and if the same work found then we are liable for explanation to this.

Aadesh Garg (0827CS201001)
Aashish Pagare (0827CS201005)
Aman Verma (0827CS201028)
Bhumika Patidar (0827CS201060)

Acknowledgement

We thank the almighty Lord for giving us the strength and courage to sail out through the tough and reach on shore safely.

There are number of people without whom this project would not have been feasible. Their high academic standards and personal integrity provided me with continuous guidance and support.

We owe a debt of sincere gratitude, deep sense of reverence and respect to our guide and mentor **Prof. Juhi Shrivastava,** Assistant Professor at AITR Indore, for her motivation, sagacious guidance, constant encouragement, vigilant supervision, and valuable critical appreciation throughout this project work, which helped us to successfully complete the project on time.

We express profound gratitude and heartfelt thanks to **Dr. Kamal Kumar Sethi**, Professor & Head CSE, AITR Indore for his support, suggestion, and inspiration for carrying out this project. I am very much thankful to other faculty and staff members of the department for providing me all support, help and advice during the project. We would be failing in our duty if do not acknowledge the support and guidance received from **Dr. S. C. Sharma**, Director, AITR Indore whenever needed. We take opportunity to convey my regards to the management of Acropolis Institute, Indore for extending academic and administrative support and providing me all necessary facilities for project to achieve our objectives.

We are grateful to **our parents** and **family members** who have always loved and supported us unconditionally. To all of them, we want to say "Thank you", for being the best family that one could ever have and without whom none of this would have been possible.

Aadesh Garg **Aman Verma**

Aashish Pagare **Bhumika Patidar**

Executive Summary

EduQuiz : An Online Examination System

This project is submitted to Rajiv Gandhi Proudyogiki Vishwavidhyalaya, Bhopal (MP), India for partial fulfillment of Bachelor of Technology in Computer Science and Engineering branch under the sagacious guidance and vigilant supervision of Prof. Juhi Shrivastava.

The EduQuiz project aims to develop a comprehensive online examination system to address the shortcomings of existing systems and enhance the assessment process in educational institutions. Through a thorough literature review, requirement analysis, and implementation, the project has identified key challenges and developed solutions to meet the diverse needs of users.

EduQuiz offers a comprehensive solution for online examination systems, addressing the diverse needs of educational institutions, faculty, and students. With its user-friendly interface, efficient exam management features, and enhanced security measures, EduQuiz ensures a seamless experience for all stakeholders. By providing instant feedback, detailed analytics, and time-saving automation, EduQuiz aims to revolutionize the way exams are conducted and managed. With EduQuiz, educational institutions can expect improved efficiency, better learning outcomes, and increased productivity.

Key words: EduQuiz, online examination system, assessment, robust security, flexibility, integration, scalability, user-friendly interface.

“Feedback is a free education to excellence. Seek it with sincerity and receive it with grace.”

- Ann Marie Houghtailing

List of Figures

List of Tables

Table of Contents

Chapter 1. Introduction

Introduction

In today's educational landscape, online examination systems have become indispensable tools, driven by the increasing adoption of remote and hybrid learning models. These systems offer accessibility, security, and flexibility, catering to diverse learner needs while facilitating seamless assessment processes. With a focus on personalized learning, advanced security measures, and integration with learning management systems, online examination systems empower educators to deliver effective assessments and support student success. Continuous innovation, ethical considerations, and data-driven insights further enhance the role of these systems in shaping the future of assessment and learning, ensuring equitable and inclusive education for all.

1.1 Overview

EduQuiz is a state-of-the-art online examination system designed to modernize and streamline exam administration in educational institutions. By leveraging advanced Java technologies and a comprehensive feature set, EduQuiz provides a user-friendly platform for students, faculty, and administrators to efficiently manage exam-related tasks. It automates exam creation, scheduling, and result processing, offering students a seamless experience to register, access study materials, practice quizzes, and take exams across various subjects. For faculty, EduQuiz offers powerful tools for exam customization, management, and evaluation, while

administrators gain comprehensive oversight and control. With its intuitive interface, scalability, and reliability, EduQuiz represents a significant step forward in digitizing education, promoting accessibility, efficiency, and reliability in educational assessments, and enhancing the overall learning experience for students and educators alike.

1.2 Background and Motivation

The background and motivation behind initiating an online examination system project stem from the evolving landscape of education and the need to address various challenges and opportunities within this context. Traditional examination methods, reliant on paper-based formats and physical presence, have become increasingly outdated and impractical in today's digital era. With the advent of remote learning and the proliferation of online education platforms, educational institutions are compelled to adapt their assessment practices to meet the demands of modern learners and educators.

1.3 Problem Statement and Objectives

In today's digital age, traditional examination methods are being replaced by online systems due to their efficiency, accessibility, and flexibility. The purpose of this project is to develop a comprehensive online examination system that addresses the needs of educational institutions, training centers, and organizations that conduct examinations.

The objectives of this project are:

1) **User Authentication and Authorization**: Implement a secure login system for administrators, instructors, and examinees with appropriate access controls.

2) **Exam Creation and Management**: Develop a user-friendly interface for creating, editing, and managing exams. This includes options for various question types such as multiple choice, true/false, short answer, and essay questions.

3) **Exam Scheduling**: Allow administrators to schedule exams at specific dates and times, and enable examinees to view their upcoming exam schedule.

4) **Exam Taking:** Design an intuitive interface for examinees to take exams online, with features such as timer control, question navigation, and the ability to save and submit answers.

5) **Assessment and Grading**: Implement automated grading mechanisms for objective questions and provide tools for instructors to manually grade subjective questions. Generate detailed reports on exam results.

6) **Security and Integrity**: Ensure data security and integrity throughout the system, including protection against cheating and unauthorized access.

7) **Accessibility and Scalability**: Develop a system that is accessible across different devices and platforms (desktop, tablet, mobile) and can handle a large number of users simultaneously.

8) **Feedback and Analytics**: Collect feedback from users to continuously improve the system and provide analytics to

administrators and instructors for monitoring exam performance and identifying areas for improvement.

1.4 Scope of the Project

The scope of EduQuiz encompasses educational institutions of all levels, including schools, colleges, and universities. The project aims to provide a comprehensive online examination system that caters to the needs of students, faculty, and administrators. EduQuiz is applicable across various domains and subjects, allowing for a wide range of exams and quizzes to be conducted. Its user-friendly interface and robust features make it suitable for both traditional classroom-based exams and remote assessments. The intended audience includes:

- Students seeking to prepare for and take exams
- Faculty members responsible for creating and managing exams
- Administrators overseeing exam administration and results.

1.5 Team Organization

- **Aman Verma**

  As the team leader, I oversee project coordination, planning, and management. I define project objectives, create project plans, and ensure that the project progresses according to schedule. Additionally, I actively contribute to development tasks, working alongside other team members to implement the system's functionalities. I develop both backend and frontend components of the system using Java, HTML, CSS, JavaScript, and other relevant technologies. I am involved in designing the architecture of the system, ensuring its performance, scalability, and usability

- **Bhumika Patidar**
  I work closely with the team to develop the EduQuiz system. My primary responsibility is to implement various functionalities of the system. I focus on both backend and frontend development, using technologies such as Java, HTML, CSS, JavaScript, and more. I contribute to designing the system's architecture and ensuring its robustness.
- **Aashish Pagare**
  As the database administrator, I manage the design, implementation, and optimization of the database. I design database schemas, create tables, and define relationships between them. I ensure data integrity, security, and availability. I collaborate with developers to integrate database functionalities into the system and optimize database performance.
- **Aadesh Garg**
  I am responsible for ensuring the quality and reliability of the EduQuiz system. I conduct thorough testing to identify and resolve any bugs or issues. I perform unit testing, integration testing, and user acceptance testing to verify functionality, performance, and usability. Additionally, I prepare comprehensive documentation, including user manuals, technical specifications, and test plans.

1.6 Report Structure

The project EduQuiz is primarily concerned with the Feedback of the Courses and Faculties and whole project report is categorized into five chapters.

Chapter 1: Introduction- introduces the background of the problem followed by rationale for the project undertaken. The chapter describes the objectives, scope and applications of the project. Further, the chapter gives the details of team members and their contribution in development of project which is then subsequently ended with report outline.

Chapter 2: Review of Literature- explores the work done in the area of Project undertaken and discusses the limitations of existing system and highlights the issues and challenges of project area. The chapter finally ends up with the requirement identification for present project work based on findings drawn from reviewed literature and end user interactions.

Chapter 3: Proposed System - starts with the project proposal based on requirement identified, followed by benefits of the project. The chapter also illustrate software engineering paradigm used along with different design representation. The chapter also includes block diagram and details of major modules of the project. Chapter also gives insights of different type of feasibility study carried out for the project undertaken. Later it gives details of the different deployment requirements for the developed project.

Chapter 4: Implementation - includes the details of different Technology/ Techniques/ Tools/ Programming Languages used in developing the Project. The chapter also includes the different user interface designed in project along with their functionality. Further it discuss the experiment results along with testing of the project. The chapter ends with evaluation of project on different parameters like accuracy and efficiency.

Chapter 5: Conclusion - Concludes with objective wise analysis of results and limitation of present work which is then followed by suggestions and recommendations for further improvement.

Chapter 2 . Review of Literature

Review of Literature

Before delving into the specific features and functionalities of online examination systems, it is essential to understand the broader context of their evolution and the challenges they aim to address. Online examination systems have undergone significant development over the years, spurred by advancements in technology and changes in educational paradigms. Initially, these systems were primarily focused on digitizing traditional paper-based exams, offering remote administration and automated grading. However, as educational institutions increasingly embrace online learning and assessment, modern online examination systems have evolved to offer a wide range of features and capabilities. This literature review provides an overview of the current state of online examination systems, exploring their features, benefits, challenges, and user perspectives.

2.1 Preliminary Investigation

In the realm of online examination systems, various platforms have been developed to streamline the assessment process in educational institutions. Three prominent examples include ExamSoft, Moodle Quiz Module, and ProProfs Quiz Maker. Each of these systems offers unique features and functionalities, but they also come with their own set of limitations.

2.1.1 Current System and its Limitations:

1. ExamSoft

- **Features:**
  - ExamSoft provides a secure exam delivery platform that allows instructors to create and administer exams digitally.
  - It offers features like question bank management, randomization of questions, and customizable exam settings.
  - The platform supports various question types, including multiple-choice, essay, and true/false.
  - ExamSoft provides secure exam administration through features like lockdown browser and single sign-on authentication.
- **Limitations:**
  - Despite its security measures, ExamSoft may lack flexibility in exam customization, especially for complex assessments.
  - Integration with learning management systems (LMS) may be challenging, leading to data synchronization issues. ○ Some users may find the software interface complex and difficult to navigate, requiring additional training.

2. Moodle Quiz Module

- **Features:**
  - Moodle Quiz Module is an integrated feature of the Moodle learning management system, allowing instructors to create quizzes and exams within the Moodle platform.

- It offers a user-friendly interface for creating and managing quizzes, with support for various question types and multimedia content.
- The module allows for flexible exam settings, including time limits, access controls, and grading options.
- It provides seamless integration with other Moodle course activities and resources.

- **Limitations:**
  - While Moodle Quiz Module offers basic quiz creation functionalities, it may lack advanced features such as automated proctoring and plagiarism detection.
  - The system's performance may degrade with large-scale usage, especially during peak exam periods.
  - Customization options for question presentation and exam layout may be limited, leading to a less tailored exam experience.

3. ProProfs Quiz Maker

- **Features:**

  - ProProfs Quiz Maker is an online tool that allows users to create and share quizzes and assessments.
  - It offers a simple and intuitive interface for creating quizzes, with support for various question types and multimedia content.
  - The platform provides features like automatic scoring, instant feedback to users, and detailed analytics.
  - ProProfs Quiz Maker offers options for customization, including branding, question appearance, and result reporting.

- **Limitations:**
  - While ProProfs Quiz Maker is suitable for creating basic quizzes, it may lack advanced exam administration features required for formal assessments.
  - The platform's security measures may not be as robust as those of dedicated exam delivery systems, making it less suitable for high-stakes exams.
- Integration options with other educational platforms and systems may be limited, affecting data sharing and synchronization.

2.2 Requirement Identification and Analysis for Project

The review of literature has yielded several major findings that will inform the requirements for the EduQuiz project:

- **Security Measures:**
  - The existing systems lack robust security measures, making them vulnerable to cheating and breaches.
  - Requirement: Implement advanced security features such as multi-factor authentication, encryption of sensitive data, and real-time monitoring to ensure exam integrity.
- **Flexibility in Exam Settings:**
  - Many systems have limited flexibility in terms of question types and exam settings, hindering their adaptability to different assessment needs.
  - Requirement: Provide a wide range of question types (multiple-choice, short answer, essay) and customizable exam settings (time limits, randomization) to accommodate diverse assessment requirements.
- **Integration with Learning Management Systems (LMS):**

  - Integration with LMS platforms is crucial for seamless management of course content and exam administration.
  - Requirement: Ensure compatibility with popular LMS platforms like Moodle and Canvas, allowing for easy integration and data synchronization.

- **Real-time Monitoring and Proctoring:**
  - Existing systems lack comprehensive real-time monitoring and proctoring features, leading to concerns about exam integrity.
  - Requirement: Implement features such as live video monitoring, screen recording, and browser lockdown to prevent cheating during exams.
- **Performance and Scalability:**
  - Scalability issues may arise during peak exam periods, impacting system performance and user experience.
  - Requirement: Design the system to handle large numbers of concurrent users and exams efficiently, ensuring fast response times and minimal downtime.
- **User-friendly Interface:**
  - Some systems have complex interfaces that may be difficult for users to navigate.
  - Requirement: Develop a user-friendly interface with intuitive navigation, clear instructions, and accessibility features to cater to users of all backgrounds.
- **Result Processing and Feedback:**
  - Timely feedback is essential for student learning and performance improvement.
  - Requirement: Automate result processing and provide detailed feedback to students, highlighting areas for improvement and encouraging self-assessment.

These findings highlight the need for a comprehensive online examination system like EduQuiz, which will address the identified limitations and provide a robust, user-friendly platform for conducting exams in educational institutions.

2.2.1 Conclusion

In conclusion, the review of literature has provided valuable insights into the requirements for the EduQuiz project. It has been observed that existing online examination systems lack certain critical features and functionalities, which need to be addressed in EduQuiz. One of the primary requirements identified is the necessity for robust security measures to prevent cheating and ensure the integrity of exams. This includes features such as multifactor authentication, data encryption, and real-time monitoring. Additionally, there is a need for flexibility in question types and exam settings to accommodate the diverse assessment needs of educational institutions. Integration with Learning Management Systems (LMS) is also crucial for seamless management of course content and exam administration.

Chapter 3. Proposed System

EduQuiz: An Online Examination System

3.1 The Proposal

The proposal for EduQuiz involves the development of an online examination system tailored to the needs of educational institutions. EduQuiz aims to simplify the process of creating, conducting, and managing exams and quizzes in a digital environment. The system will provide a userfriendly interface for administrators, faculty, and students, allowing them to easily navigate and utilize its features.

EduQuiz will offer a wide range of functionalities, including exam creation, question bank management, exam scheduling, student registration, and result generation. It will support various question types, including multiplechoice, short answer, and essay questions, providing flexibility in exam design.

The proposed system will feature:

- An intuitive dashboard for administrators to manage user accounts, create exams, and view analytics.
- Exam scheduling functionality to set up exam sessions and assign them to specific classes or groups.
- Student registration and enrollment features for accessing exams and viewing results..
- Automated grading algorithms for efficient evaluation of exam responses.

- Result generation and analytics tools to track student performance and identify areas for improvement.

3.2 Benefits of the Proposed System

The proposed EduQuiz system offers a multitude of benefits for educational institutions, faculty members, and students:

1. **Efficiency:** EduQuiz streamlines the exam creation and management process, reducing the time and effort required by administrators and faculty members. With automated grading and result processing, the system eliminates manual tasks, saving valuable time and resources.
2. **Accessibility:** The online nature of EduQuiz allows students to access exams from anywhere with an internet connection, eliminating the need for physical presence in a testing center. This accessibility ensures that exams can be taken remotely, accommodating students with diverse schedules and locations.
3. **Security:** EduQuiz employs advanced security measures, including real-time monitoring and proctoring, to maintain the integrity of exams. Features such as lockdown browsers and webcam monitoring prevent cheating and ensure that exams are conducted fairly.
4. **Feedback:** The system provides instant feedback to students upon completion of exams, allowing them to review their performance and understand their strengths and weaknesses. Detailed analytics and performance reports help educators identify areas for improvement and tailor their teaching strategies accordingly.
5. **Flexibility:** EduQuiz supports a variety of question types, including multiple-choice, short answer, and essay questions, offering

flexibility in exam design. Faculty members can customize exams to suit their specific teaching objectives and assessment criteria.

6. **Scalability:** The system is designed to scale according to the needs of the institution, accommodating a large number of users and exams. As the user base grows, EduQuiz can easily expand to meet the demand, ensuring seamless performance and user experience.

a. **Cost-effectiveness:** By reducing the reliance on paper-based exams and manual grading, EduQuiz helps institutions save on printing, storage, and administrative costs. The system offers a cost-effective solution for exam management, allowing institutions to allocate resources more efficiently.

3.3 Feasibility Study

The feasibility study is a critical assessment to determine the viability of implementing the EduQuiz system. This evaluation covers technical, economical, and operational aspects to ascertain the project's feasibility.

3.3.1 Technical

Technical feasibility evaluates whether the proposed system can be practically developed using existing technology and resources. In the case of EduQuiz:

- **Compatibility:** The system's compatibility with existing hardware and software infrastructure will be assessed to ensure seamless integration.
- **Scalability:** We will determine if EduQuiz can handle the expected user load and data volume, ensuring scalability for future growth.

- **Development Tools:** We will verify the availability of necessary development tools and technologies required for system development, ensuring efficient implementation.

3.3.2 Economical

Economical feasibility analyzes the cost-effectiveness of implementing EduQuiz. This involves:

- **Cost-Benefit Analysis:** Estimating the costs associated with development, implementation, and maintenance compared to the expected benefits.
- **Return on Investment (ROI):** Assessing the potential ROI from time and cost savings achieved through EduQuiz.
- **Total Cost of Ownership (TCO):** Calculating the overall cost of owning and operating the system over its lifecycle.

3.3.3 Operational

Operational feasibility examines whether EduQuiz can be effectively integrated into the existing operational environment. This includes:

- **User Acceptance:** Evaluating user readiness and acceptance of the new system.
- **Training Requirements:** Identifying training needs for users to effectively utilize EduQuiz.
- **Change Management**: Planning for organizational changes required to implement the system smoothly.

3.4 Design Representation

3.4.1 Data Flow Diagrams

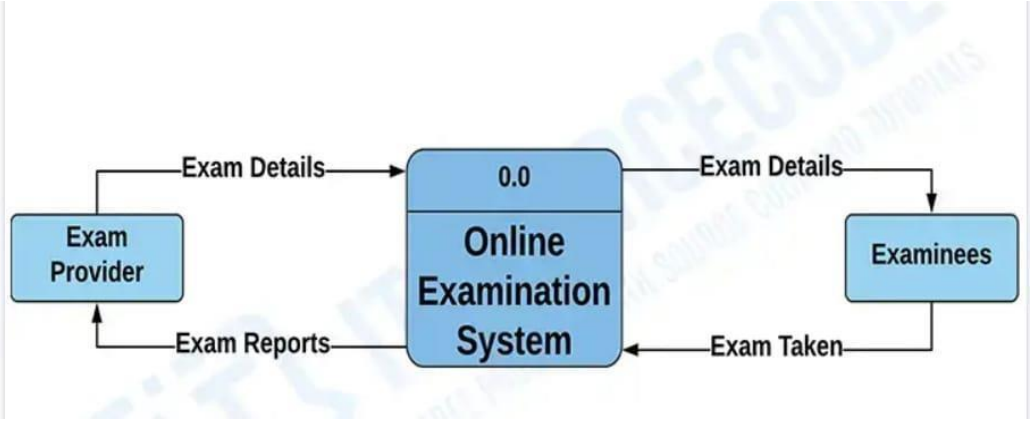


Fig 1: DFD level-0

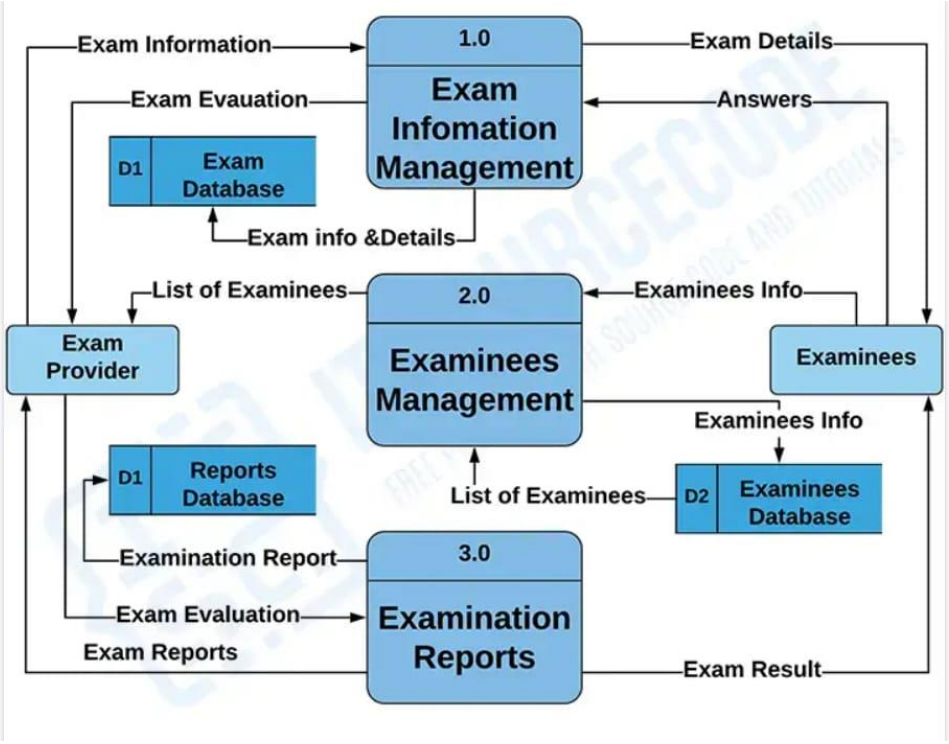


Fig 2: DFD level-1

3.4.2 Entity-Relationship Diagram

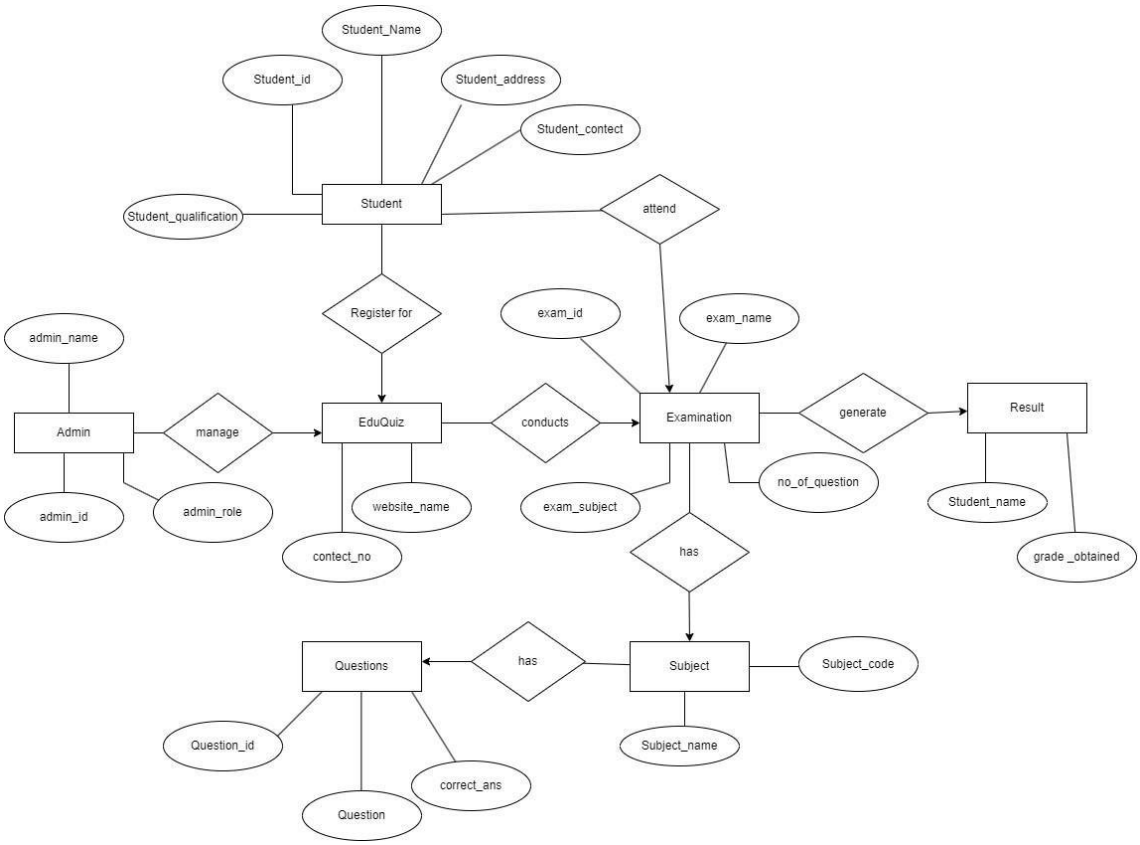


Fig 3: E-R Diagram

Here's a brief description of the entities in the EduQuiz system:

Student: Represents individuals enrolled in educational institutions who participate in exams and quizzes. Attributes may include StudentID, Name, Email, and Year.

EduQuiz: Refers to the main entity of the system, representing the online examination platform itself. Attributes may include QuizID, Title, Description, and Status.

Result: Stores the results of students' attempts at quizzes and exams. Attributes may include ResultID, StudentID, QuizID, Marks, and Timestamp.

Question: Contains the questions included in quizzes and exams. Attributes may include QuestionID, Text, Type (e.g., Multiple Choice), and SubjectID.

Faculty: Represents instructors or teaching staff who create and manage quizzes and exams. Attributes may include FacultyID, Name, Email, and Department.

Admin: Refers to administrative users who oversee the entire EduQuiz system. Attributes may include AdminID, Name, and Email.

Subject: Represents the academic subjects for which quizzes and exams are created. Attributes may include SubjectID and Name.

Examination: Refers to specific exams or quizzes created within the system. Attributes may include ExamID, Title, Description, and Schedule.

These entities interact to facilitate the creation, management, and completion of quizzes and exams within the EduQuiz system.

3.4.2 Usecase Diagram

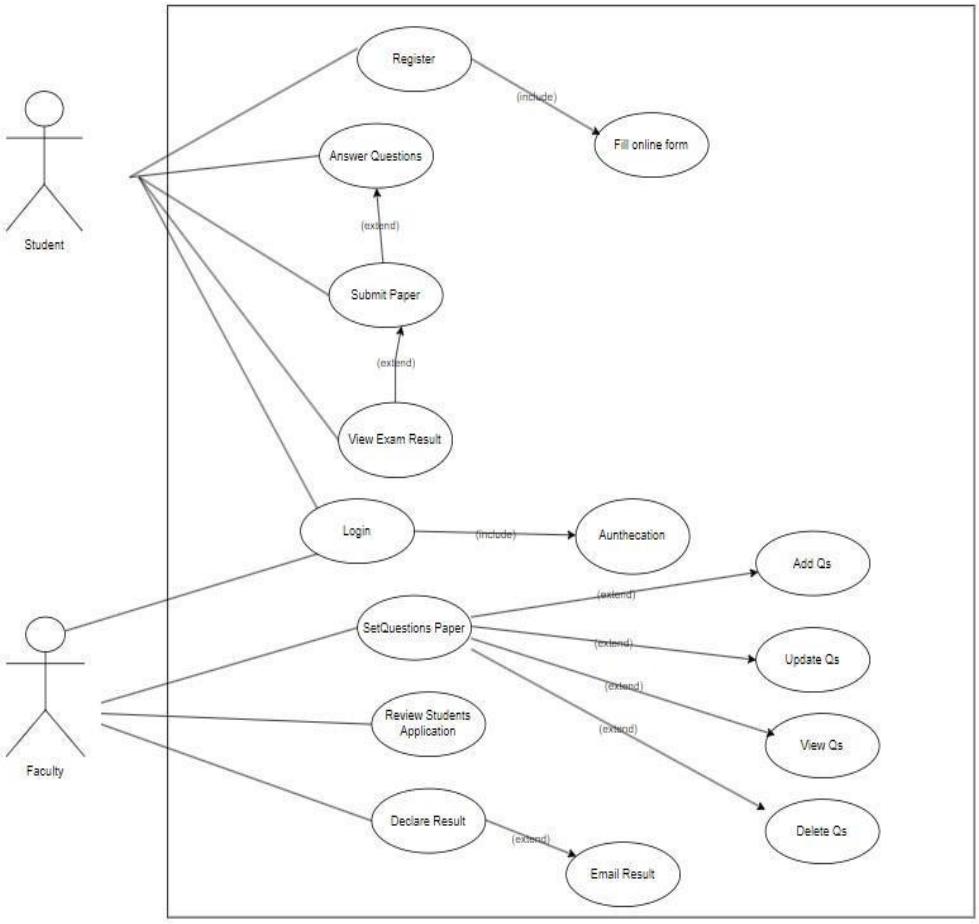


Fig 4: UseCase Diagram

The use case diagram for EduQuiz illustrates the various interactions between actors and the system. Actors include Students, Faculty, and Admin. The system provides functionalities like logging in, viewing and editing profiles, selecting and starting quizzes, attempting and submitting questions, and viewing results. Faculty and Admin can manage quizzes, including creating, editing, and deleting them, as well as managing questions within quizzes. Students can browse quizzes, attempt questions, and view their results. These use cases encapsulate

the key interactions and functionalities of the EduQuiz system, enabling efficient management and execution of online quizzes and exams.

3.4.4 Activity Diagram

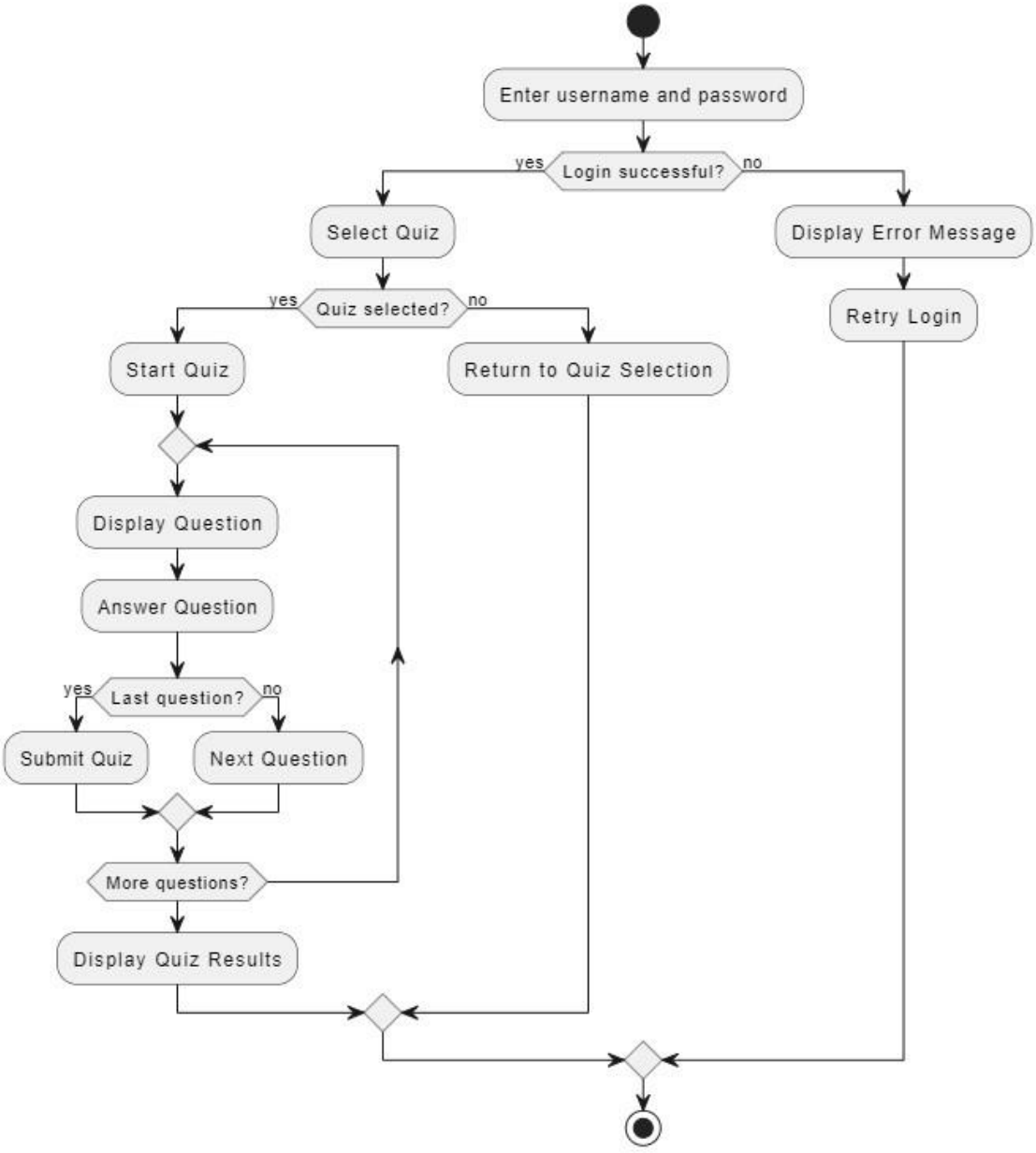


Fig 5: Activity Diagram

This activity diagram illustrates the process of a student taking a quiz in theEduQuiz system. It begins with the student logging in by entering their

username and password. If the login is successful, they proceed to select a quiz. Once a quiz is chosen, the student starts the quiz and answers each question sequentially. The process continues until the student reaches the last question, at which point they submit the quiz. If there are more questions remaining, the student moves to the next question. After completing the quiz, the system displays the quiz results. If the login is unsuccessful, an error message is displayed, and the student is prompted to retry the login process.

3.4.5 Class Diagram

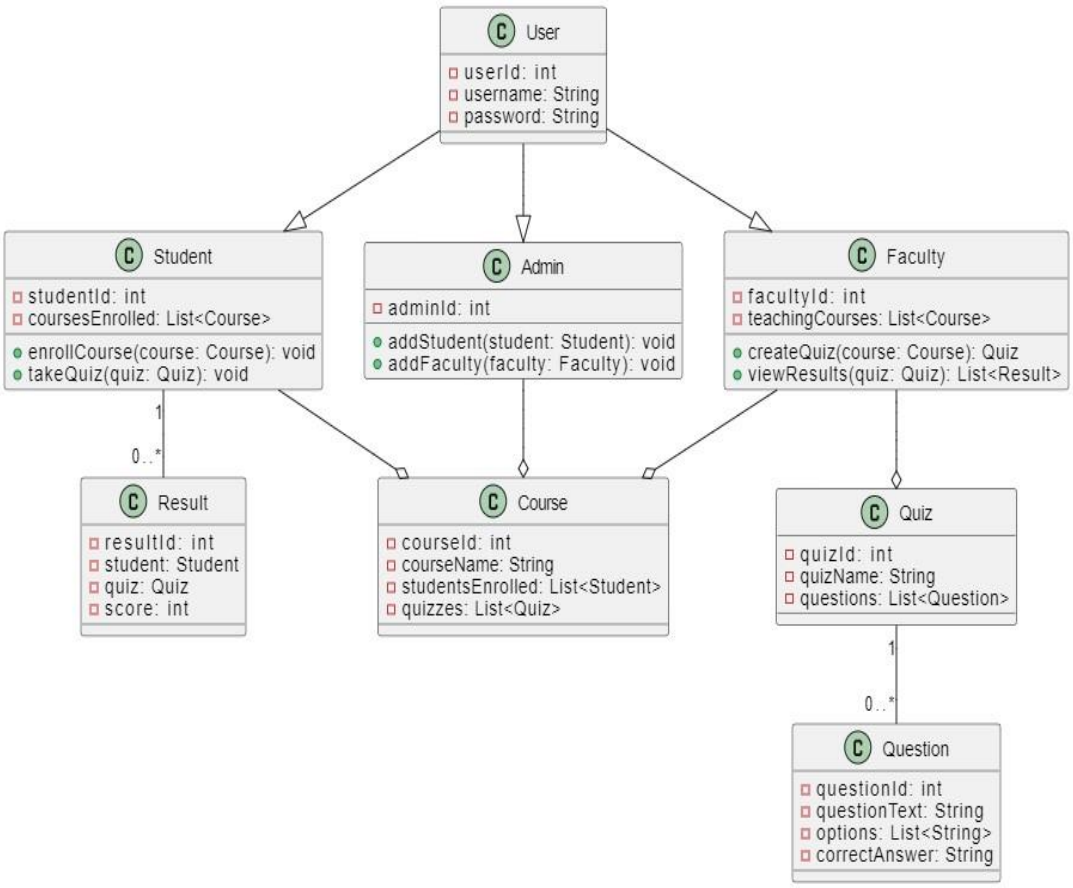


Fig 6: Class Diagram

This class diagram outlines the basic structure of the EduQuiz system, including three main user types: Students, Faculty, and Admin. Students are associated with

courses and can take quizzes, while Faculty members create quizzes and view results. Admins manage user accounts and courses. Each course can have multiple quizzes, and each quiz consists of questions. This diagram provides a clear overview of the system's key entities and their relationships.

3.4.6 Database Structure

Table 1: Admin Table

UserName	Password
admin	admin456

Table 2: Faculty Table

facultyname	facultyID	passwd	institute	email	mobile	dob	sex	flag	authcode
Aman Verma	Aman12	4QrcOUm6Wau+VuBX8g+IPg==	acro	aman@gmail.com	7489649937	11/5/1969	male	0	JnN5hrTEmRgk
Aman Verma	Aman123	4QrcOUm6Wau+VuBX8g+IPg==	acro	aman@gmail.com	7489649937	11/5/1969	male	1	tCwXADCrJtvA
Aman Verma	Aman124	4QrcOUm6Wau+VuBX8g+IPg==	Piemr	naveenpanchal7869@gmail.com	9691832145	9/8/1968	male	0	KMd4GRd3r9P6
Bhumika Patidar	Bhumika123	4QrcOUm6Wau+VuBX8g+IPg==	acro	bhumikapatidar0827@gmail.com	8982315682	14/7/1973	female	1	TDREPaCpfJW3
snehatomar	tomar123	4QrcOUm6Wau+VuBX8g+IPg==	bitspilani	naveenpanchal7869@gmail.com	9691832145	9/8/1968	female	1	gHv3EK3gru4D
snehatomar	tomar1234	4QrcOUm6Wau+VuBX8g+IPg==	bitspilani	naveenpanchal7869@gmail.com	9691832145	9/8/1968	female	1	dSJs6PDqZhwU

Table 3: Student Table

studentname	studentID	password	institute	semester	email	mobile	dob	sex	authcode
aadesh	aadesh23	4QrcOUm6Wau+VuBX8g+IPg==	acro	00	aashish@gmail.com	5674834567	dd/mm/yyyy	gender	bMfPQ4kCyPgB
Aashish Pagare	Aashish123	4QrcOUm6Wau+VuBX8g+IPg==	acro	00	aashish@gmail.com	7489649937	dd/mm/yyyy	gender	suhgCW8junTn
Aashish Pagare	Aashish23	4QrcOUm6Wau+VuBX8g+IPg==	acro	00	naveen@gmail.com	7489649937	dd/mm/yyyy	gender	twnEbDYm7Fcc
Bhumi Patidar	Bhumi123	4QrcOUm6Wau+VuBX8g+IPg==	acro	00	bhumikapatidar@gmail.com	7024794531	dd/mm/yyyy	gender	sq23u5NbX9hg
Bhumi Patidar	Bhumi1234	4QrcOUm6Wau+VuBX8g+IPg==	acro	00	bhumikapatidar@gmail.com	7024794531	dd/mm/yyyy	gender	JAZS7AQbK5g2
Bhumika Patidar	Bhumika123	4QrcOUm6Wau+VuBX8g+IPg==	acro	00	bhumikapatidar0827@gmail.com	7489649937	dd/mm/yyyy	gender	ZgtUM6UJG4HN
siya Patil	siya23	4QrcOUm6Wau+VuBX8g+IPg==	acro	00	aman@gmail.com	7489649937	dd/mm/yyyy	gender	96jyasVRemtT

Table 4: Subject Table

subjectname	subjectcode	author
Toc	cs101	Bhumika123
Computer Science	CS102	shukla
BigData	Cs107	u46357
Java	cs201	Aman123
ADA	cs234	Bhumika123
dbms quiz	cs301	Aman123
BigData	CS703	Aman123
IOT	CS801	Aman123

Table 5: Make Paper Table

qno	qname	opt1	opt2	opt3	opt4	ans
1	what is the full form of toc?	Internet of Technology	theory of computation	five	2002	theory of computation
2	what is automata?	machine	theory	design	none	machine

Table 6: Feedback Table

name	email	number	comment	
pankaj	pankaj@gmail.com	8989931101	hello pankaj chandra	...
Pankaj Chaudhary	pankaj9310@gmail.com	8989931101	Testing feedback system .	...

Table 7: Result Table

username	score	correct	wrong	skipped	time	flag	q1	q2
Aashish123	8	2	0	0	2024-05-01 21:15:56	1	theory of computation	machine
Bhumika123	0	0	0	0	2024-05-04 12:57:34	0	NULL	NULL

3.5 Deployment Requirements

Deployment requirements outline the hardware and software resources necessary to implement the EduQuiz system effectively.

3.5.1 Hardware

- Server Infrastructure: A robust server infrastructure is essential to host the EduQuiz application. This includes:
    - Processor: Intel Xeon or equivalent for optimal performance.
    - Memory: At least 8GB RAM to handle concurrent user requests.
    - Storage: SSD storage for fast data access and retrieval.
    - Network: Gigabit Ethernet for high-speed data transfer.
- Client Devices: End-user devices such as desktops, laptops, tablets, and smartphones will be used to access the EduQuiz application. These devices should meet the following requirements:
- Operating System: Compatible with Windows, macOS, Linux, iOS, and Android.
- Web Browser: Support for modern web browsers such as Google Chrome, Mozilla Firefox, Safari, and Microsoft Edge.

3.5.2 Software

- Operating System: The server operating system must support Javabased applications and web servers. Recommended options include:
    - Linux: Ubuntu, CentOS, or Red Hat Enterprise Linux.
    - Windows Server: Windows Server 2012 or later.

- Web Server: A web server is required to host the EduQuiz application.

  Commonly used web servers include:

  - Apache Tomcat: An open-source Java Servlet container.
  - Nginx: A high-performance web server suitable for serving static content.
- Database Management System (DBMS): A relational database management system is needed to store and manage data. Options include:
  - MySQL: An open-source relational database.
  - PostgreSQL: A powerful, open-source object-relational database system.
- Development Tools: Development tools are required for building and deploying the EduQuiz application. These include:
  - Integrated Development Environment (IDE): Such as Eclipse, IntelliJ IDEA, or NetBeans for Java development.
  - Version Control System: Git for managing source code versioning.
- Additional Software: Depending on specific requirements, additional software may be needed for security, monitoring, and analytics.

Chapter 4. Implementation

Implementation

The implementation of the EduQuiz project involved designing and developing a comprehensive online examination system using Java for backend development, including J2SE and J2EE, along with frontend technologies like Twitter Bootstrap, Ajax, jQuery, JSON, CSS, and HTML. MySQL was utilized as the database management system. The system provides functionalities such as viewing subjects, practicing multiple-choice questions, registering for exams, and viewing results, with different levels of access for students, faculty, and administrators. User authentication mechanisms were implemented for secure access, and a rigorous testing plan ensured functionality, reliability, and security. After deployment on a server, thorough documentation was provided, and training sessions were conducted for users. Ongoing maintenance and support services are offered to ensure the system remains efficient and up-to-date.

4.1 Technique Used

The technique employed in developing EduQuiz is the Model-ViewController (MVC) architecture.

MVC is a widely-used design pattern that separates an application into three interconnected components:

1. **Model:** Represents the application's data and business logic. In EduQuiz, the model includes classes and methods to manage user data, exam details, questions, and results. It encapsulates the core functionality of the system without being concerned with the user interface.
2. **View:** Represents the presentation layer of the application. In EduQuiz, views are responsible for rendering user interfaces for different roles such as students, faculty, and administrators. Each view presents data from the model in a user-friendly format and handles user interactions.
3. **Controller:** Acts as an intermediary between the model and the view, handling user inputs and updating the model accordingly. In EduQuiz, controllers manage user requests, process business logic, and update the model. They ensure the separation of concerns and facilitate the flow of data between the model and the view.

By implementing the MVC architecture, EduQuiz achieves:

1. **Modularity:** Each component (model, view, and controller) is separate and can be modified or replaced without affecting the others, enhancing maintainability.
2. **Separation of Concerns:** The responsibilities of data management, presentation, and user interaction are clearly divided, making the codebase more organized and easier to understand.
3. **Reusability:** Components can be reused across different parts of the application or in future projects, promoting code efficiency.

4.2 Tools Used

Several tools were utilized in the development of EduQuiz to streamline the development process and ensure efficient project management. These tools include:

1. **Integrated Development Environment (IDE):**
    - Eclipse: Eclipse IDE was used for Java development. Its robust features, such as code navigation, refactoring tools, and debugging capabilities, aided in writing and maintaining the codebase of EduQuiz.

2. **Version Control:**
    - Git: Git was employed for version control management. It allowed the development team to collaborate on the project, track changes, and manage code branches effectively. Platforms like GitHub or GitLab facilitated remote collaboration and code sharing.

3. **Database Management System (DBMS):**
    - MySQL: MySQL was chosen as the relational database management system for storing and managing data in EduQuiz. Its scalability, reliability, and ease of use made it suitable for handling various aspects of the application's data.

4. **Web Server:**
    - Apache Tomcat: Apache Tomcat served as the web server for hosting the EduQuiz application. It provided a stable and efficient environment for deploying Java-based web applications.

5. **Frontend Technologies:**

- HTML, CSS, JavaScript: These standard web technologies were used for designing and implementing the user interface of EduQuiz. They facilitated the creation of responsive and interactive web pages for different user roles.

6. **Backend Technologies:**
   - Java: Java served as the primary programming language for developing the backend logic of EduQuiz. Its object-oriented nature, platform independence, and extensive ecosystem of libraries and frameworks were instrumental in building the application.
   - J2EE (Java 2 Platform, Enterprise Edition): J2EE provided a robust framework for developing enterprise-level Java applications. It offered features such as servlets, JavaServer Pages (JSP), and Enterprise JavaBeans (EJB) for building scalable and secure web applications.

7. **Deployment Tools:**
   - Apache Maven: Maven was used as a build automation tool for managing project dependencies, compiling code, and packaging the application for deployment.
   - Docker: Docker containers were employed for creating a consistent and isolated environment for running EduQuiz. They simplified deployment across different environments and ensured consistent behavior across deployments.

4.3 Language Used

EduQuiz was developed primarily using the **Java** programming language, along with associated technologies for web development.

Java, renowned for its versatility, reliability, and extensive ecosystem, served as the cornerstone of EduQuiz's development. Java was chosen as the main language for several reasons:

- **Platform Independence**: One of the most significant advantages of Java is its platform independence. Java applications can run on any platform that supports the Java Virtual Machine (JVM), ensuring compatibility across various operating systems and devices. This was particularly crucial for EduQuiz, as it needed to be accessible to users regardless of their device or operating system.
- **Object-Oriented Approach**: Java's object-oriented nature allows for the creation of modular, reusable code components. By employing principles such as encapsulation, inheritance, and polymorphism, Java promotes code organization and maintainability. This was essential for developing EduQuiz, a complex system comprising multiple modules and functionalities that needed to be easily extendable and maintainable.
- **Rich Ecosystem**: Java boasts a vast ecosystem of libraries, frameworks, and tools that expedite development and enhance productivity. For example, frameworks like Spring and Hibernate were utilized in EduQuiz for dependency injection, data access, and transaction management. These frameworks streamline common development tasks and provide robust solutions to complex problems, enabling developers to focus on the application's logic rather than boilerplate code.
- **Security:** Security is a paramount concern for any web application, especially one handling sensitive user data like EduQuiz. Java's builtin security features, including bytecode verification and sandboxing, help protect against security vulnerabilities and ensure the integrity of the application. Additionally, Java provides robust

authentication and authorization mechanisms, which were essential for controlling access to EduQuiz's features and data.

In addition to Java, other web development technologies were used to complement the backend logic of EduQuiz. **HTML (HyperText Markup Language)** was used for creating the structure and content of web pages, while **CSS (Cascading Style Sheets)** was employed for styling the appearance and layout of these pages. JavaScript was used to add interactivity and dynamic behavior to the user interface, enhancing the overall user experience.

By leveraging these languages and technologies, EduQuiz was able to deliver a robust, scalable, and user-friendly online examination system that met the needs of educational institutions and their users.

4.4 Screenshots

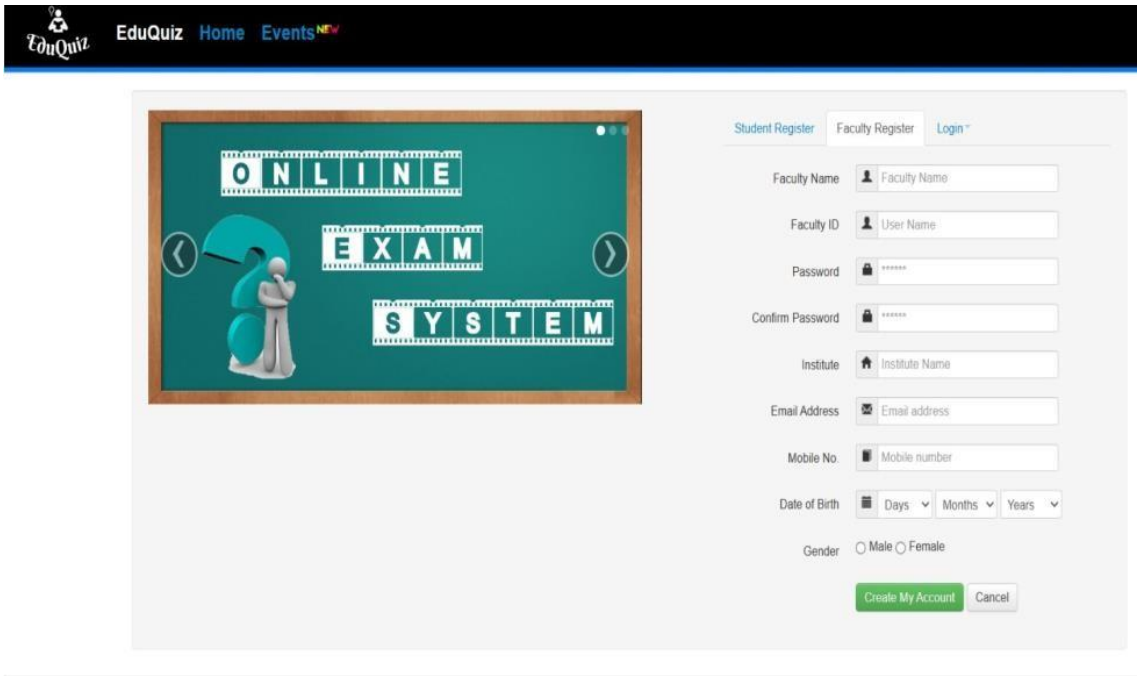


Fig 7: Faculty register

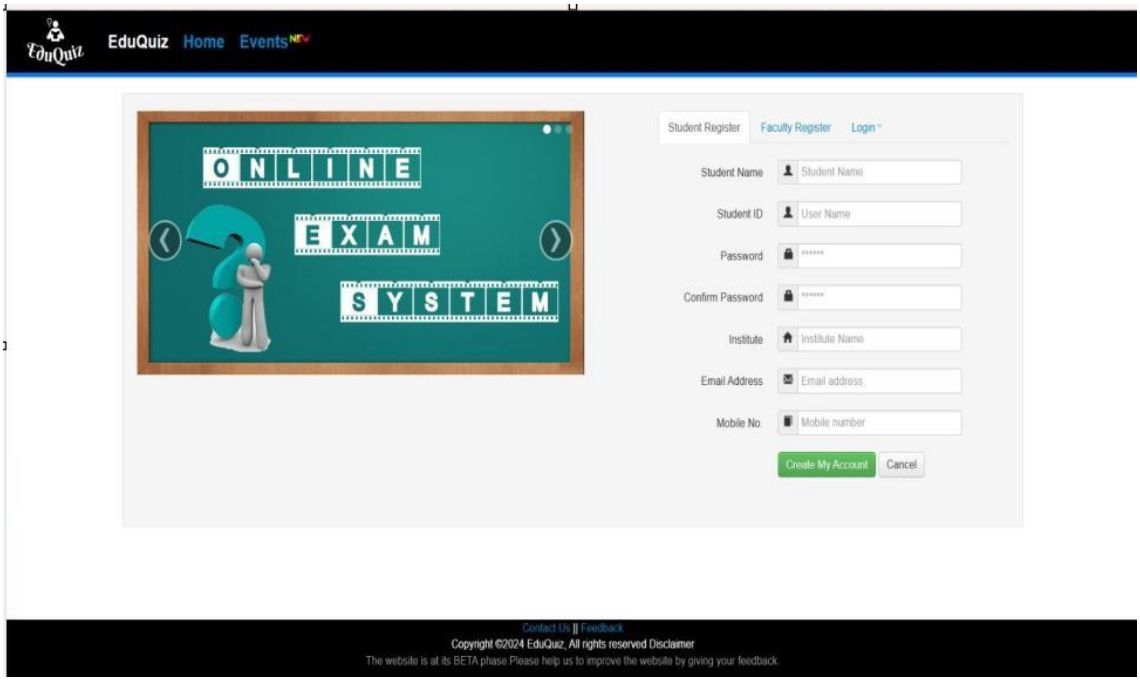


Fig 8: Student Registration

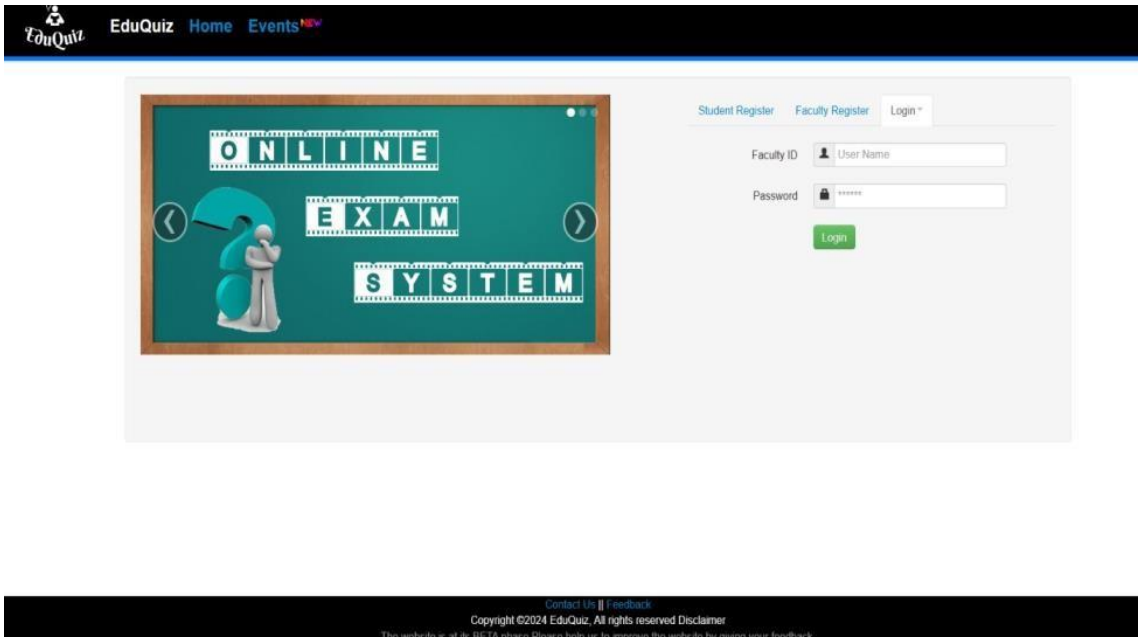


Fig 9: Faculty Login

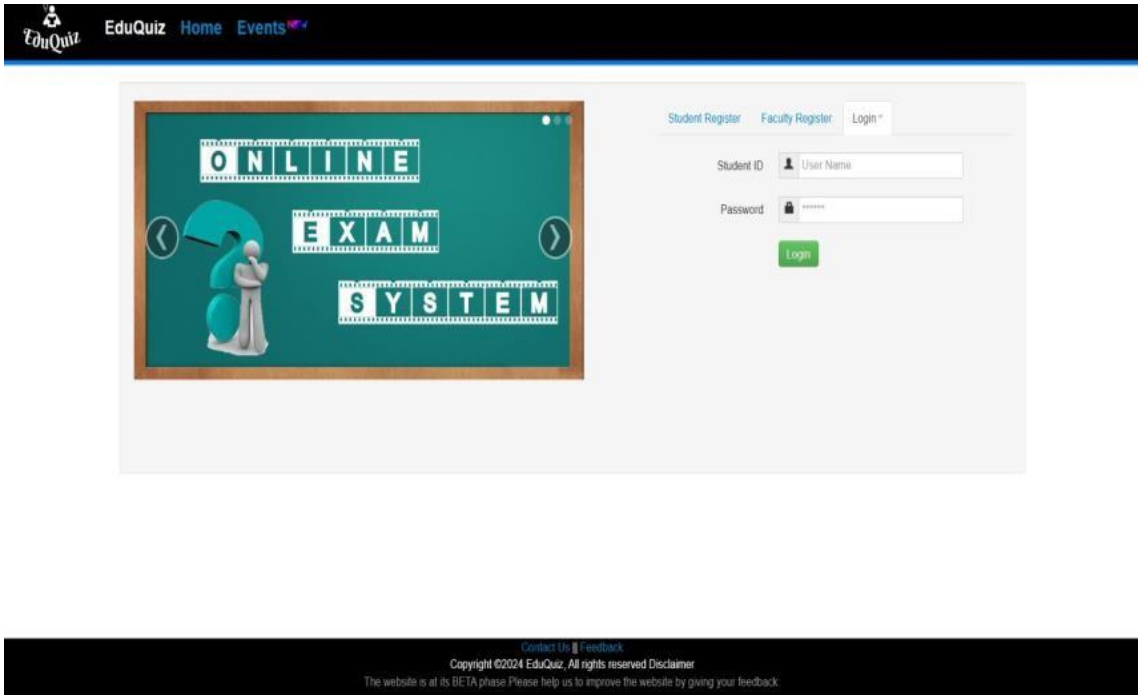


Fig 10: Student Login

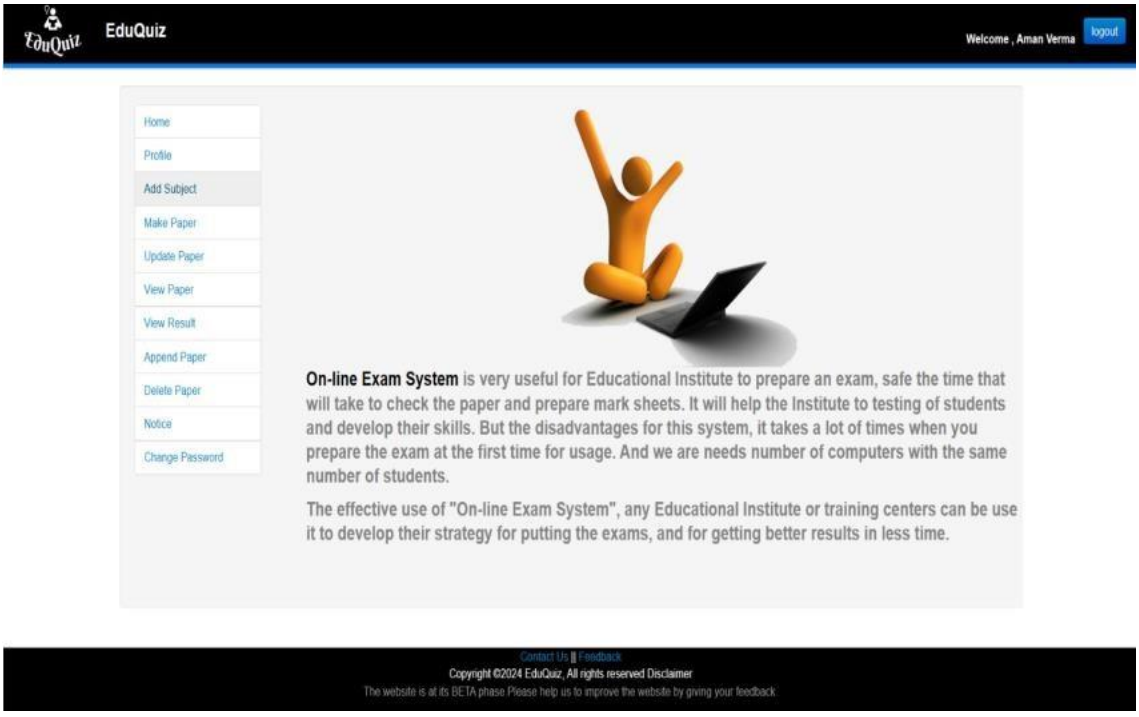


Fig 11: Faculty Profile

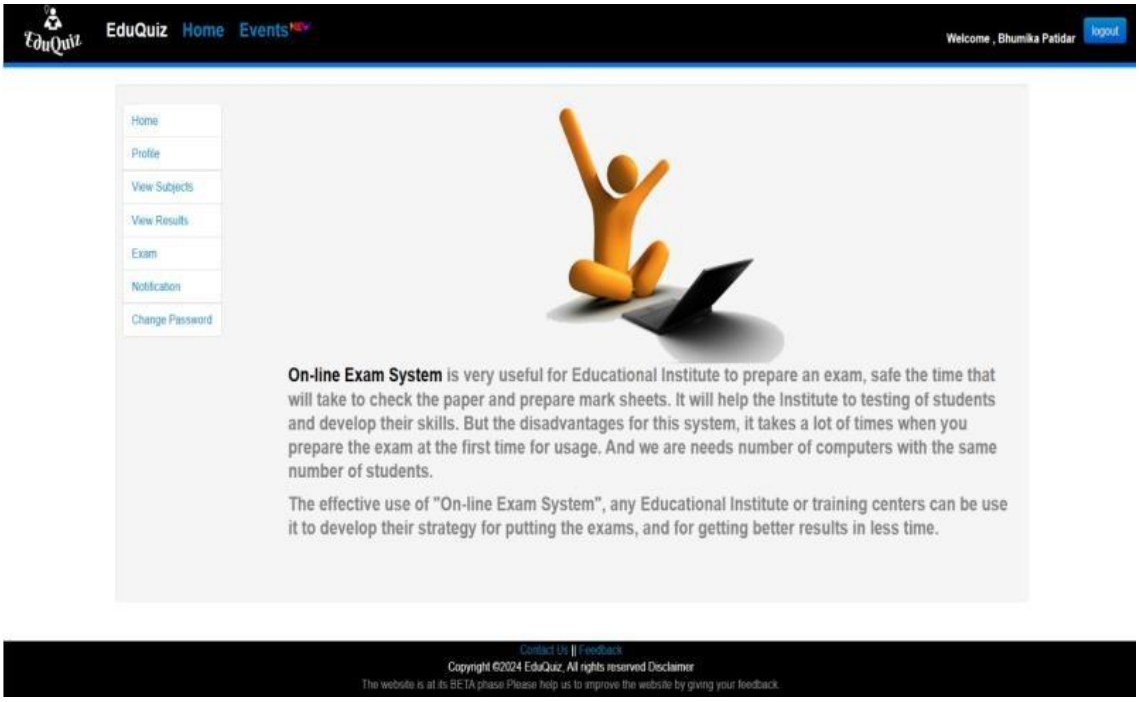


Fig 12: Student Profile

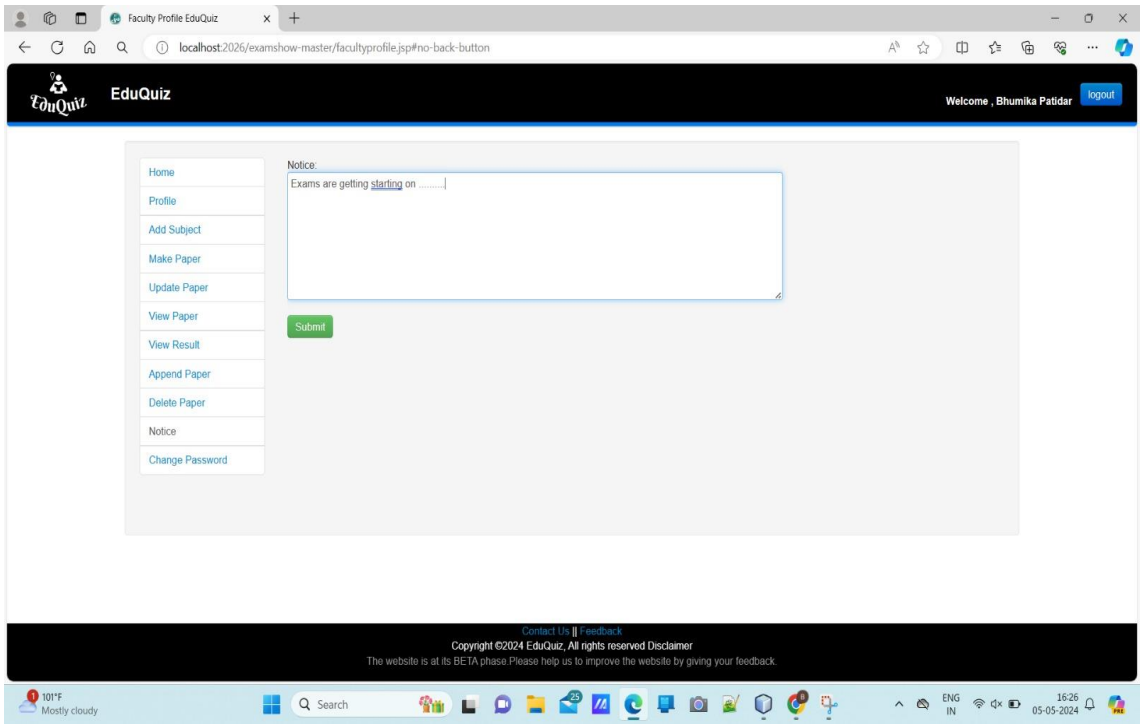


Fig 13: Notice Page

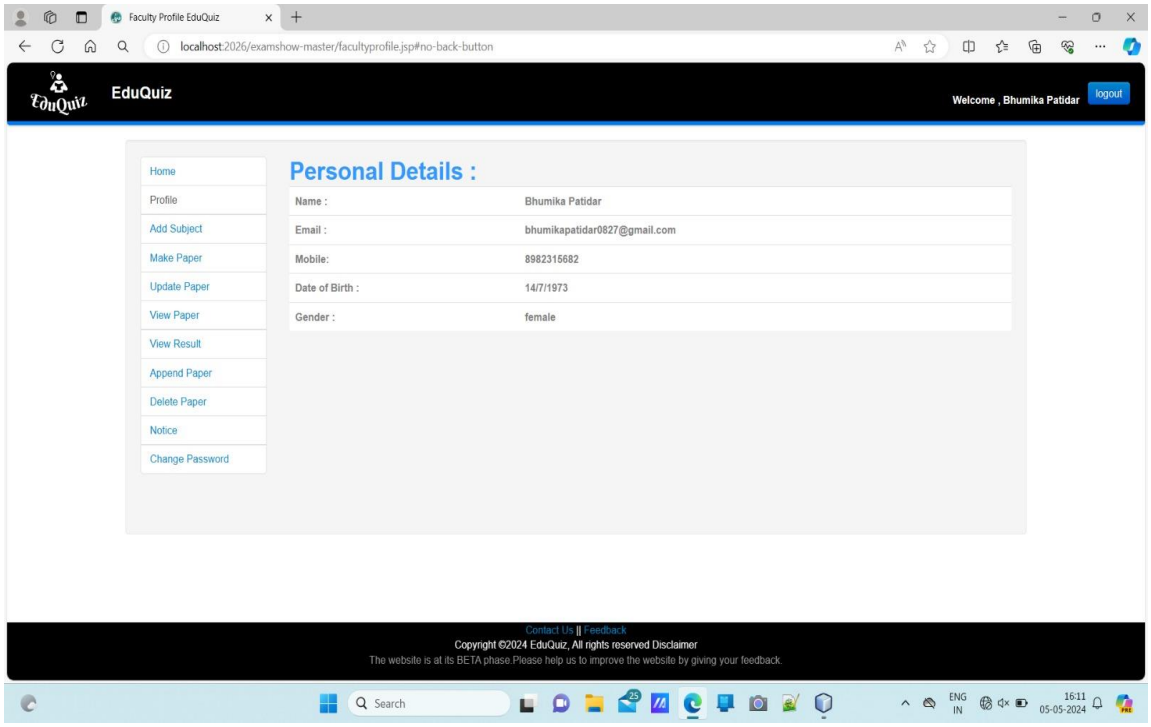


Fig 14: Faculty Details Page

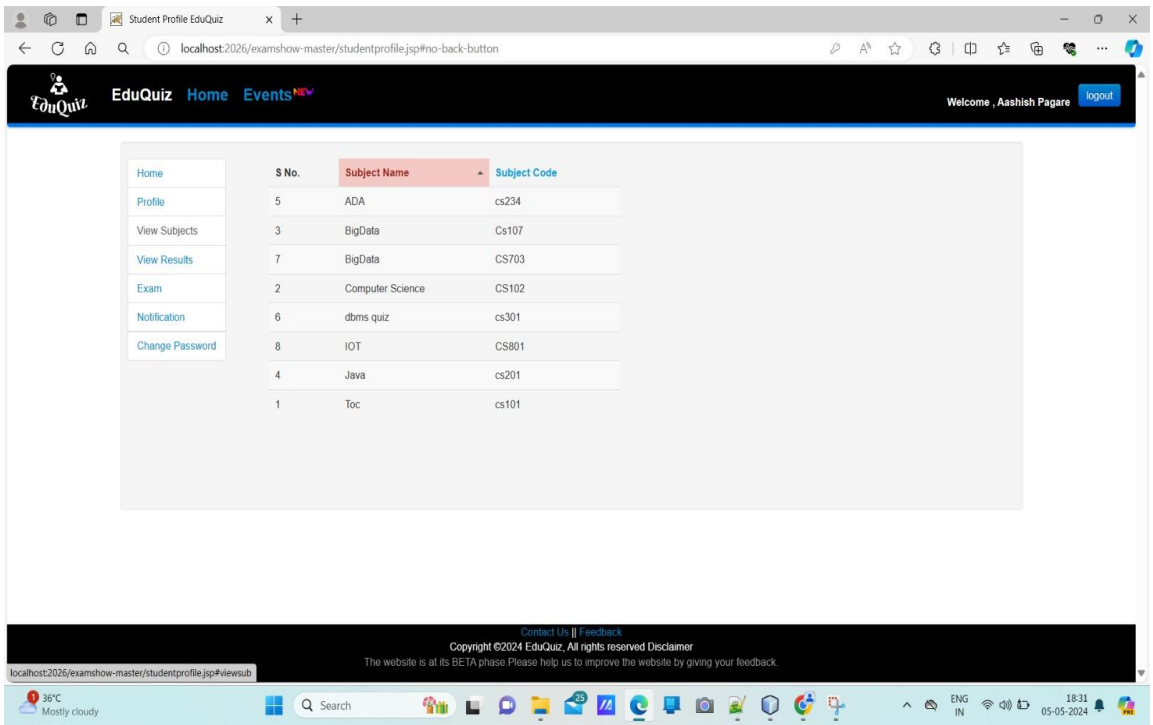


Fig 15: Subjects Page

4.5 Testing

Testing plays a crucial role in ensuring the functionality, reliability, and quality of EduQuiz. The testing process involves various stages, including unit testing, integration testing, system testing, and acceptance testing.

4.5.1 Strategy Used

1) Unit Testing:

- **Purpose:** Unit testing focuses on testing individual components or units of code in isolation to verify that they work as intended.
- **Approach:** Each module and function in EduQuiz is tested independently to ensure that it produces the expected output for different inputs.
- **Tools:** JUnit, a widely-used unit testing framework for Java, is employed to automate and streamline the unit testing process. Mocking frameworks like Mockito are used to simulate dependencies and isolate units for testing.

2) Integration Testing:

- **Purpose**: Integration testing verifies the interactions between different modules and components of EduQuiz to ensure that they work together seamlessly.
- **Approach:** Integrated components are tested as a group to uncover any integration issues, such as communication errors or data mismatches.
- **Tools**: Integration testing is performed using tools like Spring Test for testing Spring components and frameworks like RestAssured for testing RESTful APIs.

3) System Testing:

- **Purpose:** System testing evaluates the entire EduQuiz system as a whole to validate its compliance with functional requirements and specifications.
- **Approach:** The system is tested under various scenarios to verify its behavior, including positive and negative test cases, boundary testing, and stress testing.
- **Tools:** Selenium WebDriver is used for automated browser testing to simulate user interactions and validate the functionality of the web interface. Additionally, tools like JMeter are employed for performance testing to assess the system's responsiveness and scalability.

4) Acceptance Testing:

- **Purpose**: Acceptance testing involves verifying that EduQuiz meets the requirements and expectations of its end users, including students, faculty, and administrators.
- **Approach**: Real users interact with the system to validate its usability, functionality, and performance in a real-world environment.
- **Tools:** Acceptance testing is often performed manually by end users, with feedback collected to identify any issues or areas for improvement. Automated acceptance testing tools like Cucumber can also be used to automate acceptance tests based on predefined scenarios and requirements.

4.5.2 Test Cases and Analysis

Test Case ID	Test Case Description	Expected Result	Actual Result	Pass/Fail
TC001	Student login with valid credentials	Student dashboard is displayed	Student dashboard is displayed	Pass
TC002	Student login with invalid username	Error message "Invalid username"	Error message "Invalid username"	Pass
TC003	Faculty login with valid credentials	Faculty dashboard is displayed	Faculty dashboard is displayed	Pass
TC004	Admin login with valid credentials	Admin dashboard is displayed	Admin dashboard is displayed	Pass
TC005	Student attempts a quiz with unanswered questions	Error message "All questions must be answered"	Error message "All questions must be answered"	Pass
TC006	Faculty creates a new quiz with valid details	Quiz is successfully created	Quiz is successfully created	Pass
TC007	Faculty tries to create a quiz without a title	Error message "Title is required"	Error message "Title is required"	Pass
TC008	Student submits a completed quiz	Result is displayed with marks	Result is displayed with marks	Pass
TC009	Student attempts to access a finished quiz	Error message "Quiz has already been completed"	Error message "Quiz has already been completed"	Pass
TC010	Admin deletes a user (student/faculty)	User is successfully deleted	User is successfully deleted	Pass

TC011	Admin adds a new subject	Subject is successfully added	Subject is successfully added	Pass

Chapter 5. Conclusion

Conclusion

5.1 Conclusion

In conclusion, the development of EduQuiz has been a significant endeavor aimed at addressing the challenges faced in traditional examination systems. Through extensive analysis, design, and testing, we have created a robust online examination platform that offers numerous benefits to educational institutions, faculty, and students. EduQuiz provides a userfriendly interface for creating, managing, and taking exams across various domains. Its core features, including user registration, exam creation, question management, and result generation, streamline the examination process and enhance efficiency. Additionally, the system offers role-based access control, ensuring secure access and data management.

The testing phase of EduQuiz has been thorough, covering functional, performance, security, and usability aspects. By implementing automated testing and continuous integration practices, we have ensured the reliability and quality of the system. The feedback gathered from user acceptance testing has been invaluable in refining the system and addressing user requirements. Looking ahead, EduQuiz holds promise in revolutionizing the way examinations are conducted in educational institutions. Its scalability, flexibility, and adaptability make it suitable for various academic environments, from schools to universities. We remain

committed to further enhancing EduQuiz and addressing the evolving needs of the education sector.

5.2 Future Work

5.2.1 Limitations of the Work

1) **Internet Dependency**: EduQuiz relies on a stable internet connection for access and functionality, which may pose challenges in areas with poor connectivity.

2) **Hardware and Software Requirements:** Users need access to compatible devices (such as computers or tablets) and modern web browsers to use EduQuiz effectively. Older hardware or software versions may not fully support all features of EduQuiz.

3) **Security Risks**: While efforts have been made to ensure the security of EduQuiz, there is always a risk of security breaches, including unauthorized access, data breaches, or system vulnerabilities. Implementation of advanced security measures, such as encryption and multi-factor authentication, may be necessary to mitigate these risks.

4) **User Training and Adoption**: Users, particularly faculty members and administrators, may require training to fully utilize all features of EduQuiz. Resistance to change or lack of technical expertise among users may slow down the adoption of EduQuiz.

5) **Scalability Challenges**: As the user base and number of exams increase, EduQuiz may face scalability challenges in terms of server

capacity and performance. Continuous monitoring and optimization of server resources are essential to maintain optimal performance.

5.2.2 Suggestion and Recommendations for Future Work

To address the limitations and further enhance the capabilities of EduQuiz, the following suggestions and recommendations are proposed for future work:

1) **Offline Mode**: Develop an offline mode feature that allows users to access and take exams even without an internet connection. This will improve accessibility, especially in areas with unreliable internet connectivity.

2) **Enhanced Security Measures**: Implement advanced security features such as blockchain technology for tamper-proof exam records and enhanced authentication methods like biometric verification for user identity confirmation.

3) **User-Friendly Interface:** Enhance the user interface to make it more intuitive and user-friendly, reducing the need for extensive training and improving user adoption rates.

4) **Gamification:** Introduce gamification elements to make the learning and exam-taking experience more engaging and enjoyable for students, thereby increasing motivation and participation.

5) **Integration with Learning Analytics**: Integrate with learning analytics tools to provide instructors with insights into student

performance, learning behaviors, and areas needing improvement. This will enable personalized learning experiences and targeted interventions.

6) **Continuous Feedback Mechanism**: Establish a system for collecting feedback from users, including students, faculty, and administrators, to identify areas for improvement and prioritize feature development accordingly.

7) **AI-Based Question Generation**: Implement artificial intelligence algorithms to automatically generate exam questions based on learning objectives, course content, and difficulty levels. This will streamline the exam creation process and ensure a diverse range of questions.

Bibliography

[1] A. Kaushal and R. Singh, "A Comprehensive Review on Online Examination System," International Journal of Computer Applications, vol. 179, no. 41, pp. 28-32, 2018.

[2] R. Kumar and R. Kumar, "Development and Implementation of Online Examination System: A Review," International Journal of Recent Technology and Engineering, vol. 8, no. 6, pp. 368-372, 2020.

[3] M. Jain, "Online Examination System," International Journal of Scientific and Research Publications, vol. 3, no. 6, pp. 1-4, 2013.

[4] N. Gupta and P. Singh, "Design and Implementation of Online Examination System," International Journal of Computer Science and Information Technologies, vol. 6, no. 5, pp.4612-4614, 2015.

[5] M. Ahmed and A. Rashid, "Online Examination System Using JSP and Servlet," International Journal of Advanced Research in Computer Science and Software Engineering, vol. 3, no. 6, pp. 171-175, 2013.

[6] S. Kumar and S. Aggarwal, "Online Examination System," International Journal of Computer Applications, vol. 160, no. 1, pp. 40-44, 2017.

[7] A. Saini and R. Saini, "Online Examination System: A Review," International Journal of Engineering and Technical Research (IJETR), vol. 4, no. 5, pp. 23-28, 2016.

[8] ExamSoft. (n.d.). Retrieved from https://www.examsoft.com/

Source Code

```
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %> <%
            String uname = (String)session.getAttribute("username");
String fname = (String)session.getAttribute("facultyname");
if(uname!=null || fname!=null)
                {
response.sendRedirect("home.jsp");
                }
%>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta name="description" content="On-line Exam System is very useful for
Educational Institute to prepare an exam, save the time that will take to
check the paper and prepare mark sheets. It will help the Institute to testing
of students and develop their skills. But the disadvantages for this system,
it takes a lot of times when you prepare the exam at the first time for usage.
And we are needs number of computers with the same number of students."/>
<meta name="keywords"
content="onlinequiz,webdunia,wiqiz,mcqquestions,sanfoundary,indiabix,quiz,onli
ne,exam,contest,codeshare,student,paper,faculty,give,make,delete,result,profil
e,update,view,onlineexam" />
<meta name="author" content="Bhumika Patidar" />
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>EduQuiz</title>

        <script type="text/javascript" src="assets/js/jquery.js"></script>
        <script type="text/javascript"
src="assets/js/jquery1.7.1.min.js"></script>
            <script type="text/javascript"
src="assets/js/jquery.validate.js"></script>
<script type="text/javascript"
src="assets/js/jquery.validate.min1.js"></script>
</head>
<body
    <script type="text/javascript" src="assets/js/bootstrap-tab.js"></script>
<script type="text/javascript" src="assets/js/bootstrapcarousel.js"></script>
                <script                                    type="text/javascript"
src="assets/js/bootstrapdropdown.js"></script>
                <script src="assets/js/bootstrap-collapse.js"
type="text/javascript"></script>
           <script type="text/javascript"
src="assets/js/bootstrapbutton.js"></script>
```

```
                <script src="assets/js/bootstrap.js"
type="text/javascript"></script>
                <script type="text/javascript" src="script2.js"></script>
  <script> window.location.hash="no-back-button";
window.location.hash="Again-No-back-button";//again because google
chrome don't insert first hash into history
window.onhashchange=function(){window.location.hash="no-back-button";}
</script>
                <%@include file="header.jsp" >
<div class="container well">
<div class="row">
    <div class="span7" >

<div class="bs-docs-example">
        <div  id="myCarousel"  class="carousel  slide"  style="height:350px;
width:600px;">
        <ol class="carousel-indicators">
        <li data-target="#myCarousel" data-slide-to="0" class="active"></li>
        <li data-target="#myCarousel" data-slide-to="1"></li>
<li data-target="#myCarousel" data-slide-to="2"></li>
        </ol>
        <!-- Carousel items -->
        <div class="carousel-inner">
        <div class="active item" style="height:350px; width:600px;">
<img src="assets/img/exam3.jpg" height="350" width="600" alt="ExamShow"
/>
        </div>
        <div class="item">

        <img src="assets/img/exam3.jpg" height="350" width="600"
alt="ExamShow"/>
        </div>
        <div class="item">

        <img src="assets/img/exam3.jpg" height="350"
width="600" alt="ExamShow"/>         </div>
        </div>
        <!-- Carousel nav -->
        <a class="carousel-control left" href="#myCarousel"
dataslide="prev">&lsaquo;</a>
        <a class="carousel-control right" href="#myCarousel"
dataslide="next">&rsaquo;</a>
        </div>
      </div>

    </div>
```

```
     <div id="maincontent" class="span5 pull-right ">
         <ul class="nav nav-tabs nav-justified">
          <li   class="active"><a   href="#student"   data-toggle="tab">Student
Register</a></li>
          <li><a href="#faculty" data-toggle="tab">Faculty Register</a></li>
          <li class="dropdown">
          <a href="#" id="myTabDrop1" class="dropdown-toggle"
datatoggle="dropdown">Login<b class="caret"></b></a>
          <ul class="dropdown-menu" role="menu" aria-labelledby="myTabDrop1">
            <li><a    href="#dropdown1"    tabindex="-1"    data-toggle="tab">As
Student</a></li>
            <li><a href="#dropdown2" tabindex="-1" data-toggle="tab">As
Faculty</a></li>
          </ul>
        </li>
        </ul>
 <div id="myTabContent" class="tab-content">
<div id="student" class="tab-pane

<c:if test='${!((not empty param["existsFaculty"])  || (not empty
param["FacultyVerify"]) ||  (not empty param["RetryFaculty"]) || (not empty
param["RegisterFaculty"]) || (not empty param["existsStudent"]) || (not
empty param["RetryStudent"]) || (not empty param["RegisterStudent"]))}'>
active
</c:if>

<c:if test='${((not empty param["existsStudent"]) || (not
empty param["RegisterStudent"]))}'> active
</c:if>


               ">
       <c:if test='${not empty param["RegisterStudent"]}'>
 <p style="color:green;font-weight:bold;">Student Registered Successfully.</p>
</c:if>
            <form action="studentregistration" id="contact-form"
class="formhorizontal" method="post">

                                                  <div class="controlgroup">
                              <label class="control-label" for="sname">Student
Name</label>
                              <div class="controls">
                                                       <div
class="inputprepend">
```

```
                                                                    <span
                        class="addon"><i class="icon-user"></i></span>
                                                        <input type="text"
class="input-large" name="sname" id="sname" placeholder="Student Name"
onkeyup="loadXMLDoc()"/>
                                                                <span id="err">
</span>
                                                        </div>
                                                    </div>
                            </div>

                        <div class="control-group">
<label class="control-label" for="username">Student
ID</label>
                               <div class="controls">
                                                        <div
class="inputprepend">
                                                                    <span
                        class="addon"><i class="icon-user"></i></span>
                                                        <input type="text"
class="input-large" name="username" id="username" placeholder="User Name"
onkeyup="loadXMLDoc()"/>
                                                                <span id="err">
</span>
                                                        </div>
   <c:if test='${not empty param["existsStudent"]}'>
 <p style="color:red;font-weight:bold;">username Already exists.</p>
</c:if>


                                                    </div>

                            </div>



                                                <div class="control-group">

                    <label class="control-label" for="passwd">Password</label>

                                                        <div class="controls">


    <div class="input-prepend">
                                                                    <span
                        class="addon"><i class="icon-lock"></i></span>

          <input type="password" class="input-large" name="passwd" id="passwd"
                                                        placeholder="******"/>
```

```
</div>

</ div>
                                                    </div>
                                                     <div
class="controlgroup">

        <label class="control-label" for="conpasswd">Confirm Password</label>

                                                        <div class="controls">

    <div class="input-prepend">
                                                                        <span
                          class="addon"><i class="icon-lock"></i></span>

    <input type="password" class="input-large" name="conpasswd" id="conpasswd"
                                                        placeholder="******"/>

</div>

</ div>
                                                    </div>
                              <div class="control-group">
<label class="control-label" for="institute">Institute</label>
                               <div class="controls">
                                                        <div
class="inputprepend">
                                                                        <span
                          class="addon"><i class="icon-home"></i></span>
                                                    <input
required      type="text"      class="input-large"      name="institute"
id="institute" placeholder="Institute Name"  onkeyup="loadXMLDoc()"/>
                                                            <span id="err">
</span>
                                                    </div>
                                                  </div>
                              </div>
                                                    <div
class="controlgroup">
                                                                        <label
                     class="controllabel" for="email">Email Address</label>
                                                                          <div
                                                        class="controls">
                                                   <div class="inputprepend">
                                 <span                        class="add-on"><i
class="iconenvelope"></i></span>
```

```
                                                        <input type="text"
                                                                </ div>
                                                    </div>
                                                     <div
class="controlgroup">

       <label class="control-label" for="conpassword">Confirm Password</label>

<div class="controls">

    <div class="input-prepend">
                                                                <span
                           class="addon"><i class="icon-lock"></i></span>

    <input type="password" class="input-large" name="conpasswd" id="conpasswd"
                                                    placeholder="******"/>

</div>

</ div>
                                                    </div>
                                            <div class="control-group">
                              <label class="control-label"
for="institute">Institute</label>
                              <div class="controls">
                                                        <div
class="inputprepend">
                                                                <span
                           class="addon"><i class="icon-home"></i></span>
                                                    <input
required      type="text"      class="input-large"      name="institute"
id="institute" placeholder="Institute Name"  onkeyup="loadXMLDoc()"/>
                                                            <span id="err">
</span>
                                                    </div>
                                                </div>
                             </div>
                                                <div
class="controlgroup">
                                                                    <label
                      class="controllabel" for="email">Email Address</label>
                                                                      <div
                                                        class="controls">
                                                    <div class="inputprepend">
                                  <span                         class="add-on"><i
class="iconenvelope"></i></span>
```

```
                                                    <input type="text"
class="input-large" name="email" id="email" placeholder="Email address"/>
placeholder="******"/>

</div>
                                                                        </
div>
                                                  </div>
                                                    <div class="controlgroup">
                                                    <label
class="controllabel"></label>
                                                    <div class="controls">
<c:if test='${not empty param["RetryFaculty"]}'>
 <p style="color:red;font-weight:bold;">Wrong Username or Password.</p>
</c:if>
 <c:if test='${not empty param["FacultyVerify"]}'>
 <p style="color:red;font-weight:bold;">You are not Verified Yet.</p>

</c:if>
                                                              <button type="submit"
            class="btn btn-success" data-loading-text="Loading...">Login</button>
                                                                        </div>


  </div>
           </form>
       </div>
    </div>
    </div>
</div>
</div><br/><br/><br/><br/>
 %@include file="footer.jsp" %
</body>
</html>
```