LearnDesk : A Secure and Dynamic Online learning Platform

A PROJECT REPORT

Major Project II (01CE0807)

Submitted by

Ajay Joshi
92201703192

Het Dalsaniya

92201703202

Twisha Gol
92200103018

BACHELOR OF TECHNOLOGY

in

Computer Engineering

Faculty of Engineering & Technology

Marwadi University, Rajkot

February, 2026

Major Project II (01CE0807)

Department of Computer Engineering

Faculty of Engineering & Technology

Marwadi University

A.Y. 2025-26

CERTIFICATE

This is to certify that the project report submitted along with the project entitled **LearnDesk : A Secure and Dynamic Online learning Platform** has been carried out by **Ajay Joshi** (92201703192), **Het Dalsaniya** (92201703202)**, Twisha Gol** (92200103018) under my guidance in partial fulfilment for the degree of Bachelor of Technology in Computer Engineering, 8th Semester of Marwadi University, Rajkot during the academic year 2025-26.

Prof. Paras Shingadiya	Dr. Krunal Vaghela
Professor	Professor & Head
Department of Computer Engineering	Department of Computer Engineering

Major Project II (01CE0807)

Department of Computer Engineering

Faculty of Engineering & Technology

Marwadi University

A.Y. 2025-26

CERTIFICATE

This is to certify that the project report submitted along with the project entitled **LearnDesk : A Secure and Dynamic Online learning Platform** has been carried out by **Ajay Joshi** (92201703192) under my guidance in partial fulfilment for the degree of Bachelor of Technology in Computer Engineering, 8th Semester of Marwadi University, Rajkot during the academic year 2025-26.

Prof. Paras Shingadiya	Dr. Krunal Vaghela
Professor	Professor & Head
Department of Computer Engineering	Department of Computer Engineering

Major Project II (01CE0807)

Department of Computer Engineering

Faculty of Engineering & Technology

Marwadi University

A.Y. 2025-26

CERTIFICATE

This is to certify that the project report submitted along with the project entitled **LearnDesk : A Secure and Dynamic Online learning Platform** has been carried out by **Het Dalsaniya** (92201703202) under my guidance in partial fulfilment for the degree of Bachelor of Technology in Computer Engineering, 8th Semester of Marwadi University, Rajkot during the academic year 2025-26.

Prof. Paras Shingadiya	Dr. Krunal Vaghela
Professor	Professor & Head
Department of Computer Engineering	Department of Computer Engineering

Major Project II (01CE0807)

Department of Computer Engineering

Faculty of Engineering & Technology

Marwadi University

A.Y. 2025-26

CERTIFICATE

This is to certify that the project report submitted along with the project entitled **LearnDesk : A Secure and Dynamic Online learning Platform** has been carried out by **Twisha Gol** (92200103018) under my guidance in partial fulfilment for the degree of Bachelor of Technology in Computer Engineering, 8th Semester of Marwadi University, Rajkot during the academic year 2025-26.

Prof. Paras Shingadiya	Dr. Krunal Vaghela
Professor	Professor & Head
Department of Computer Engineering	Department of Computer Engineering

Major Project II (01CE0807)

Department of Computer Engineering

Faculty of Engineering & Technology

Marwadi University

A.Y. 2025-26

DECLARATION

We hereby declare that the **Major Project II (01CE0807)** report submitted along with the Project entitled **LearnDesk : A Secure and Dynamic Online learning Platform** submitted in partial fulfilment for the degree of Bachelor of Technology in Computer Engineering to Marwadi University, Rajkot, is a bonafide record of original project work carried out by me / us at Marwadi University under the supervision of Prof. Paras Shingadiya and that no part of this report has been directly copied from any students' reports or taken from any other source, without providing due reference.

S.No	Student Name	Sign
1	Ajay Joshi ajay.joshi119418@marwadiuniversity.ac.in	
2	Het Dalsaniya het.dalsaniya119544@marwadiuniversity.ac.in	
3	Twisha Gol twisha.gol116004@marwadiuniversity.ac.in	

Acknowledgement

I would like to express my sincere gratitude to everyone who supported me during the development of the LearnDesk website. First and foremost, I am thankful to Prof. Paras Shingadiya, whose guidance, encouragement, and valuable suggestions helped me shape this project successfully.

I would also like to thank my friends and colleagues who provided constant motivation, feedback, and assistance during the design and development process. Their inputs were invaluable in improving the functionality and user experience of the website.

I am grateful to the Department of Computer Engineering Marwadi University for providing the necessary facilities, resources, and a supportive environment to carry out this work.

Finally, I am grateful to my family for their unwavering support and encouragement throughout this journey. This project would not have been possible without their understanding and patience.

Abstract

LearnDesk is a smart learning platform built using the MERN stack(MongoDB, Express.js, React.js, Node.js) that helps teachers and students connect and learn in an interactive way. Teachers can create classes, add students, and make their own customized quizzes. Students can join these classes to attempt quizzes or learn freely by exploring different topics and practice tests available on the platform. Default aptitude quizzes—covering areas like verbal ability, logical reasoning, arithmetic, and quantitative aptitude. Users can also learn new topics by reading our articles..

All quizzes are time-based and get automatically submitted when the timer ends. After submission, the result is shown instantly. All quizzes are highly secure: they run in full-screen mode, restrict exiting or copying, and include watermarks to prevent malpractice or unauthorized sharing.

LearnDesk also includes features like email verification, mail notifications, secure login with JWT tokens, and data encryption and hashing to keep user information safe. Additional features include motivational "Quote of the Day," announcements, file uploads, and notification-based updates to keep students engaged. By combining learning resources, secure assessments, and performance tracking in one platform, LearnDesk offers a comprehensive, efficient, and user-friendly solution for placement preparation.

Table of Contents

Acknowledgement ..... **i**
Abstract ..... **ii**
Table of Contents ..... **iii**
List of Figures ..... **v**
List of Tables ..... **vi**
Abbreviations ..... **vii**
CHAPTER 1 ..... **1**
1.1 Introduction 1
1.2 Background 1
1.3 Problem Statement 2
1.4 Objectives 3
1.5 Scope of the Project 3
1.5.1 Scope for Students 3
1.5.2 Scope for Teachers 4
1.5.3 Technical Scope 4
CHAPTER 2 ..... **5**
2.1 Overview of Current Solutions 5
2.2 Limitations of Existing Systems 6
2.3 Proposed Solution 7
Chapter 3: **9**
3.1 Requirement Analysis 9
3.1.1 Functional Requirements 9
3.1.2 Non-Functional Requirements 10
3.2 Feasibility Study 10
3.2.1 Technical Feasibility 10
3.2.2 Operational Feasibility 11
3.2.3 Economic Feasibility 11
Chapter 4: **12**
4.1 High-Level Architecture Diagram 12
4.2 Use Case Diagram 13
4.3 ER Diagram / Database Design 13
4.4 Flowchart 14
4.5 Class Diagram 16
4.6 Database Schema 16
Chapter 5 ..... **18**
5.1 Technologies Used 18
5.2 Key Features Implemented 19
5.3 Interface Samples 19

Chapter 6:..... 22
6.1 Testing Strategies..... 22
6.1.1 Unit Testing..... 22
6.1.2 Integration Testing..... 22
6.2 Test Cases & Results..... 23
6.3 Testing Summary..... 25
Chapter 7..... 26
7.1 Results & Discussion..... 26
7.2 Performance Analysis..... 26
7.3 Achievements vs Objectives..... 27
7.4 Conclusion & Future Scope..... 28
7.4.1 Conclusion..... 28
7.4.2 Future Scope..... 28
References..... 29

List of Figures

Figure 4.1:High level architecture.....12
Figure 4.2: Use Case Diagram for learndesk.....13
Figure 4.3: ER Diagram.....14
Figure 4.4: Flowchart.....15
Figure 4.5: Class Diagram.....16
Figure 4.6: Database schema.....17
Figure 5.1: Login Page.....20
Figure 5.2: Admin dashboard.....20
Figure 5.3: Admin Class management.....21
Figure 5.4: Student dashboard.....21

List of Tables

Table 1.1: Technologies Used..... 2

Table 1.2: Common Issues in Existing Online Learning Systems..... 2

Table 2.1: Comparison of Existing Platforms with LearnDesk.....6

Table 5.1 Technologies Used in LearnDesk.....18

Table 6.1 Student dashboard Test Case Table.....23

Table 6.2 Teacher dashboard Test Case Table.....24

Abbreviations

API	-	Application Programming Interface
UI	-	User Interface
URL	-	Uniform Resource Locator
HTTP	-	HyperText Transfer Protocol
JSON	-	JavaScript Object Notation
DB	-	Database
ER	-	Entity Relationship Diagram
PK	-	Primary Key
ID	-	Identifier
MERN	-	MongoDB, Express.js, React.js, Node.js
NPM	-	Node Package Manager
JS	-	JavaScript
OTP	-	One-Time Password
JWT	-	JSON Web Token
MCQ	-	Multiple Choice Question
CSS	-	Cascading Style Sheets
DOM	-	Document Object Model

CHAPTER 1

INTRODUCTION

1.1 Introduction

The increasing use of online learning platforms has highlighted the need for systems that not only deliver study content but also offer secure, effective assessment tools. While many platforms provide learning material, very few combine learning, practice, class management, and secure testing in a single system.

LearnDesk is a MERN-stack–based platform created with the aim of simplifying the learning and testing workflow for both teachers and students. It brings learning resources, aptitude quizzes, performance tracking, and a secure exam environment under one application. Teachers can create classes, add students, prepare quizzes, and share material, while students can attempt quizzes, practise aptitude topics, or learn independently through the articles provided on the platform.

1.2 Background

As students prepare for placements, they often need to switch between multiple websites—one for aptitude practice, another for articles, another for mock tests, and yet another for tracking progress. Teachers face similar issues when trying to conduct online quizzes securely, especially with increasing chances of malpractice in remote assessments.

To solve these challenges, LearnDesk integrates all the required functionalities into one coherent system. It uses modern technologies such as MongoDB, Express.js, React, and Node.js, combined with third-party APIs and secure authentication mechanisms.

A quick summary of the core technologies is shown below:

Table 1.1: Technologies Used

Technology	Purpose
React.js	Frontend interface, component-based UI
Node.js & Express.js	Backend API, routing, server logic
MongoDB	Database for users, quizzes, results, articles
JWT Tokens	Secure login and authentication
Docker	Containerization for deployment
Render Hosting	Deployment of backend & frontend

1.3 Problem Statement

Traditional online learning platforms often suffer from one or more of the following issues:

Table 1.2: Common Issues in Existing Online Learning Systems

Issue	Description
Lack of Security	Many platforms allow copying, tab switching, or screenshots during quizzes.
Fragmented Resources	Students use different platforms for learning, practising, and testing.
No Real-Time Performance Feedback	Tests do not always show instant results or analytics.
Complex Class Management	Teachers do not get simple tools to create classes or add students.
Limited Motivation & Engagement	Platforms lack features that keep students active and consistent.

LearnDesk aims to address all these issues by providing a secure, interactive, and unified solution.

1.4 Objectives

The main objectives of this project are:

1. Create a unified learning platform that allows students to learn new topics, practise aptitude, and attempt quizzes in one place.
2. Provide a simple and efficient interface for teachers to create classes, manage students, and conduct quizzes seamlessly.
3. Ensure secure and fair quiz attempts through full-screen restrictions, copy-prevention features, and dynamic watermarking.
4. Deliver instant, detailed results with proper evaluation and response breakdown for every quiz.
5. Offer a smooth, user-friendly experience using the MERN stack with secure authentication, email verification, and encrypted data handling.
6. Enable scalable deployment using Docker containers for easy hosting and maintenance.

1.5 Scope of the Project

The scope of LearnDesk covers both students and teachers:

1.5.1 Scope for Students

For students, LearnDesk provides a complete environment to learn, practise, and assess themselves in one place. Students can read articles to understand new topics and strengthen their fundamentals. They also get access to a variety of default aptitude quizzes including verbal ability, logical reasoning, quantitative aptitude, and arithmetic. Apart from self-learning, students can join classes created by their teachers and attempt assigned quizzes in a structured manner. After every quiz, the results are shown instantly, helping students identify their strengths and weak areas. The platform also keeps learners motivated by showing a daily “Quote of the Day,” and students receive notifications and announcements to stay updated about upcoming quizzes, new materials, or important class updates.

1.5.2 Scope for Teachers

LearnDesk offers teachers a simple and organized way to manage their classes and assessments. Teachers can create multiple classes, add or manage student lists, and share important learning materials through file uploads or announcements. The platform also allows teachers to create their own customized quizzes with specific time limits and automatic submission when the timer ends. This ensures smooth and fair evaluations even in an online environment. Additionally, teachers can monitor how each student is performing and track progress across different quizzes, making the teaching process more efficient and data-driven.

1.5.3 Technical Scope

From a technical perspective, LearnDesk covers the full implementation of the MERN stack, combining MongoDB, Express.js, React.js, and Node.js to deliver a fast and responsive experience. It includes secure user authentication through JWT to protect login sessions and user data. The platform integrates third-party APIs to fetch dynamic content such as motivational quotes. Both the frontend and backend are fully containerized using Docker, which makes the application easy to deploy, scale, and maintain. Finally, the entire project is deployed on Render, ensuring public accessibility and smooth performance in a real-world environment.

CHAPTER 2
LITERATURE REVIEW / EXISTING SYSTEMS

2.1 Overview of Current Solutions

Several online platforms currently support aptitude learning, coding practice, and general skill development. These platforms provide high-quality study material and practice questions, but each one focuses on a specific category, which often forces students to depend on multiple websites for complete preparation.

One of the most widely used learning portals is GeeksforGeeks (GFG). It offers thousands of articles, tutorials, and placement-related questions. Although it is an excellent source for theoretical learning and practice, it lacks an integrated quiz management system, teacher–student interaction, and real-time assessments. It mainly serves as a self-study resource rather than a structured classroom tool.

Another popular site is IndiaBix, which provides a large collection of multiple-choice aptitude questions across verbal, logical, and quantitative topics. While it is great for quick practice, it does not support secure user authentication, automatic evaluation, class-wise quiz assignments, or detailed performance tracking. As a result, it is useful mainly for individual practice and not for organized learning environments.

In educational institutions, platforms like Canvas LMS are commonly used to share study material and manage courses. However, Canvas focuses more on general classroom activities, assignments, and content delivery. It does not offer placement-oriented aptitude tests, automated MCQ evaluations, or secure, controlled quiz environments required to prevent malpractice during competitive assessments.

These systems each have strengths, but none provide a complete package that covers learning materials, aptitude quizzes, secure examinations, teacher control, class-level management, and instant performance analysis—all in one platform.

Table 2.1: Comparison of Existing Platforms with LearnDesk

Feature / Criteria	GeeksforGeeks (GFG)	IndiaBix	Canvas LMS	LearnDesk (Proposed System)
Aptitude Practice	Provides articles and aptitude questions	Provides MCQs for aptitude practice	Not focused on aptitude tests	Offers topic-wise and random aptitude quizzes
Secure Quiz Environment	Does not provide secure quiz features	No secure quiz features	Basic quiz support only	Provides full-screen mode, watermarking, and copy restrictions
Class-Based Learning	Not available	Not available	Supports class management	Allows teachers to create classes and add students
Instant Results	Not applicable (no quizzes)	Provides results for practice MCQs	Depends on instructor setup	Shows instant automated results with analysis
Progress Tracking	Limited or none	No detailed progress tracking	Basic gradebook functionality	Detailed performance tracking and history
User Authentication	Available	Not required	Available	Secure JWT authentication with email verification
Learning Material	Large collection of articles & tutorials	Provides explanations for MCQs	Supports course content uploads	Articles + teacher-uploaded files and materials
Placement Orientation	Partially supports placement topics	Focused on aptitude practice	Not specific to placement preparation	Fully designed for placement and aptitude-based learning

2.2 Limitations of Existing Systems

Although existing platforms are helpful, they still fall short in meeting the specific needs of placement preparation combined with secure assessments.

Some of the key limitations are:

1. **Lack of secure quizzes**: Platforms like IndiaBix and GFG do not provide full-screen quizzes, copy restrictions, watermarks, or controlled assessment environments. This makes them unsuitable for high-integrity online exams.
2. **No class-based organization**: Students cannot join classes, nor can teachers assign quizzes, track attendance, or monitor performance in a structured manner.
3. **No real-time evaluation**: Many platforms do not offer instant results, detailed analytics, or response breakdowns immediately after submission.
4. **Fragmented resources**: Students must rely on different websites for articles, aptitude practice, quizzes, and performance tracking, which results in an unorganized learning experience.
5. **Limited customization for teachers**: Systems like Canvas allow course creation but lack features for quick MCQ quiz generation specifically designed for placement or aptitude tests.
6. **Absence of progress tracking**: Most aptitude platforms do not store user history, past quiz data, or improvement trends.
7. **No built-in motivation and notification features**: Popular platforms rarely offer features like Quotes of the Day, automated email notifications, or real-time updates.

These limitations highlight the need for a more integrated, secure, and interactive platform designed specifically for placement-oriented learning and evaluation.

2.3 Proposed Solution

To overcome the limitations of existing systems, this project proposes LearnDesk, a unified MERN-based platform that combines learning, practice, assessments, and performance tracking in a single application. LearnDesk provides article-based learning, topic-wise quizzes, random aptitude tests, and teacher-created class quizzes—all accessible through a secure login.

What makes LearnDesk different is its highly secure quiz environment, which includes full-screen mode, exit restrictions, watermarking, and automatic submission when the timer ends. Teachers can create classes, enroll students, assign quizzes, upload files, and send announcements, making the platform suitable for actual academic use. Students

receive instant results with detailed breakdowns, enabling them to track their progress and improve continuously.

With features like JWT authentication, email verification, API-based motivational quotes, notifications, Dockerized deployment, and real-time accessibility through Render, LearnDesk acts as a complete digital solution for placement preparation. It brings together the strengths of existing systems while addressing their gaps, providing a reliable, secure, and user-friendly learning environment.

Chapter 3:
System Analysis

3.1 Requirement Analysis

Requirement analysis helps in understanding what the system should achieve and how it should behave in different situations. It forms the base for system design and development. LearnDesk, being a learning and assessment platform, requires clear definition of both functional and non-functional requirements to ensure smooth operation, high security, and user satisfaction.

3.1.1 Functional Requirements

Functional requirements describe what the system must do. For LearnDesk, the key functional requirements include:

1. **User Registration and Login :** The system should allow users (students and teachers) to register, verify email, and log in using secure authentication.
2. **Role-Based Access :** Teachers and students should have different access privileges. Teachers can create classes and quizzes, while students can join classes and attempt quizzes.
3. **Class Management :** Teachers should be able to create classes, add students, upload materials, and post announcements.
4. **Quiz Creation :** Teachers can create quizzes with customizable time limits, multiple-choice questions, and secure settings.
5. **Quiz Attempt and Auto-Submission :** Students must be able to attempt quizzes, and quizzes should auto-submit when the timer ends or the user exits full-screen mode.
6. **Quiz Security Features :** The system should enforce full-screen mode, block copy/paste, add watermarking, and prevent tab switching.
7. **Article and Topic Learning :** Students should be able to read articles and learn new concepts directly on the platform.
8. **Performance Tracking and Results :** After quiz submission, results should be displayed instantly with detailed question-wise breakdown.

9. **Notifications and Email Alerts :** The system should send notifications for assignments, class updates, quiz reminders, and account-related information.
10. **File Uploads and Material Sharing :** Teachers should be able to upload files and study material for their classes.

3.1.2 Non-Functional Requirements

Non-functional requirements define the quality attributes of the system. For LearnDesk, these are critical for user experience and system reliability.

1. **Performance :** The system should load pages quickly, handle multiple users simultaneously, and ensure smooth navigation.
2. **Security :** Must use JWT authentication, password hashing, data encryption, secure APIs, and strict quiz protection mechanisms.
3. **Usability :** The platform should be easy to use, visually clear, and accessible for both students and teachers.
4. **Scalability :** The system should be able to handle increasing numbers of quizzes, users, and classes without performance issues.
5. **Reliability :** The system should have minimal downtime and consistently save user progress and data.
6. **Maintainability :** Code and architecture should be structured in a way that allows easy updates and new feature additions.
7. **Portability :** Dockerized deployment enables the system to run across different environments with minimal configuration.

3.2 Feasibility Study

A feasibility study evaluates whether the project is practical, achievable, and beneficial. LearnDesk was assessed across three major feasibility areas.

3.2.1 Technical Feasibility

This project is technically feasible because the required technologies—MongoDB, Express.js, React.js, Node.js, and Docker—are open-source, well-documented, and widely used in modern web development. The MERN stack offers flexibility, fast performance, and smooth integration between frontend and backend. Additionally, deployment on Render makes the platform accessible and easy to maintain. All required

features such as authentication, quiz security, and data storage can be implemented using available tools and libraries.

3.2.2 Operational Feasibility

Operationally, LearnDesk is easy for students and teachers to understand and use. Teachers can manage classes, upload materials, and conduct quizzes without any technical difficulty. Students simply log in, join their class, attempt quizzes, and view results instantly. The interface is simple, user-friendly, and designed to reduce learning effort. Features like announcements, notifications, and Quotes of the Day help maintain user engagement. Therefore, the system fits well into real-world academic workflows.

3.2.3 Economic Feasibility

LearnDesk is economically feasible because it uses cost-effective tools. All technologies used (MERN stack, JWT, third-party APIs, and Docker) are open-source, eliminating licensing expenses. Deployment on Render's free/low-cost tiers makes hosting affordable for development and testing purposes. Since the system does not require expensive hardware or paid software frameworks, the overall project cost remains minimal compared to commercial LMS or assessment platforms.

Chapter 4:
System Design

4.1 High-Level Architecture Diagram

The system follows a MERN-based client–server architecture supported by containerized deployment using Docker.
The architecture consists of:

- **Frontend (React.js) :** Handles the user interface, routing, quiz interactions, content display, and communication with the backend through REST APIs.
- **Backend (Node.js + Express.js) :** Handles authentication, API processing, quiz management, class management, security checks, and evaluation logic.
- **Database (MongoDB Atlas / MongoDB) :** Stores users, classes, exams, questions, results, and activity logs using a document-oriented structure.

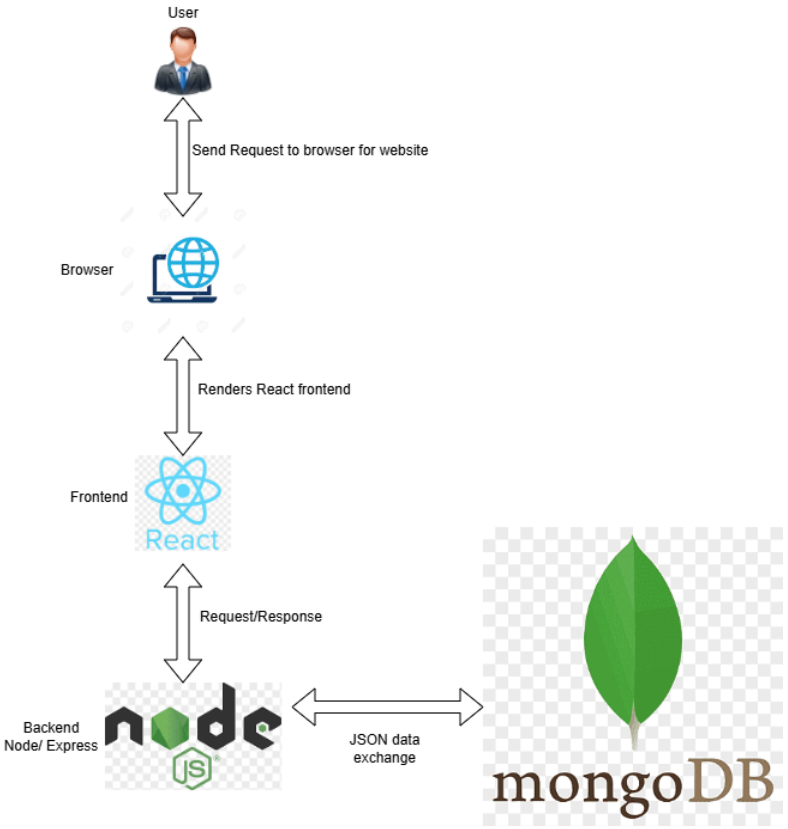

Figure 4.1:High level architecture

4.2 Use Case Diagram

The Use Case Diagram for the LearnDesk Platform illustrates the interaction between different users (students and teachers) and the system's key functionalities. It provides a clear visual representation of how each actor engages with features such as quiz creation, class management, topic learning, secured quiz attempts, and result viewing. The diagram helps identify the system boundaries, user roles, and the primary actions performed by each actor, ensuring a well-defined functional overview of the application.

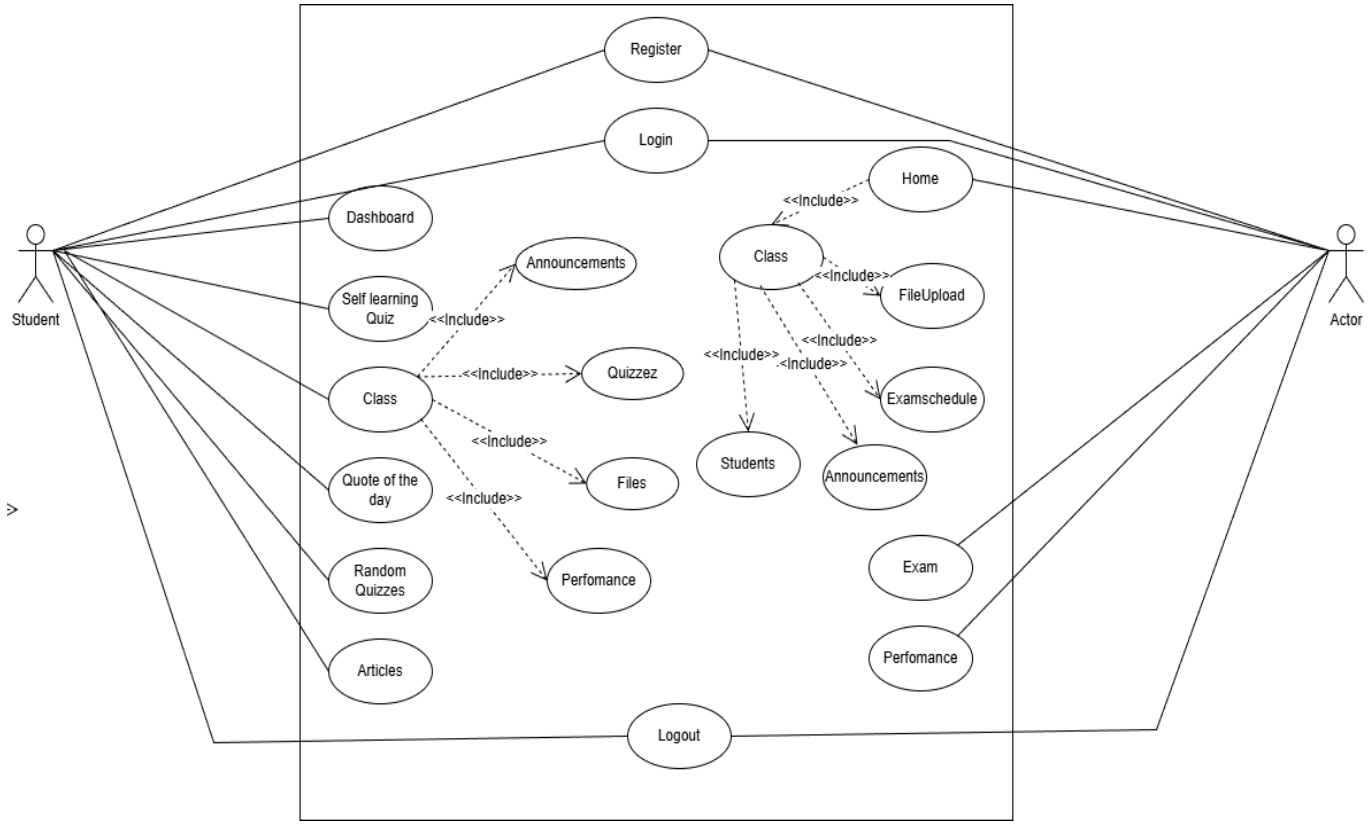

Figure 4.2: Use Case Diagram for learndesk

4.3 ER Diagram / Database Design

The ER Diagram or Database Design section showcases the structural layout of the system's data and how different entities relate to one another. Although the project uses MongoDB, a document-oriented database, the ER diagram is used to conceptually represent collections such as Users, Classes, Exams, Questions, and Results. It visually explains relationships such as one-to-many (e.g., teachers to classes, classes to exams) and many-to-many (e.g., students associated with multiple classes). This conceptual

design ensures clarity in data organization and aids in implementing efficient database operations.

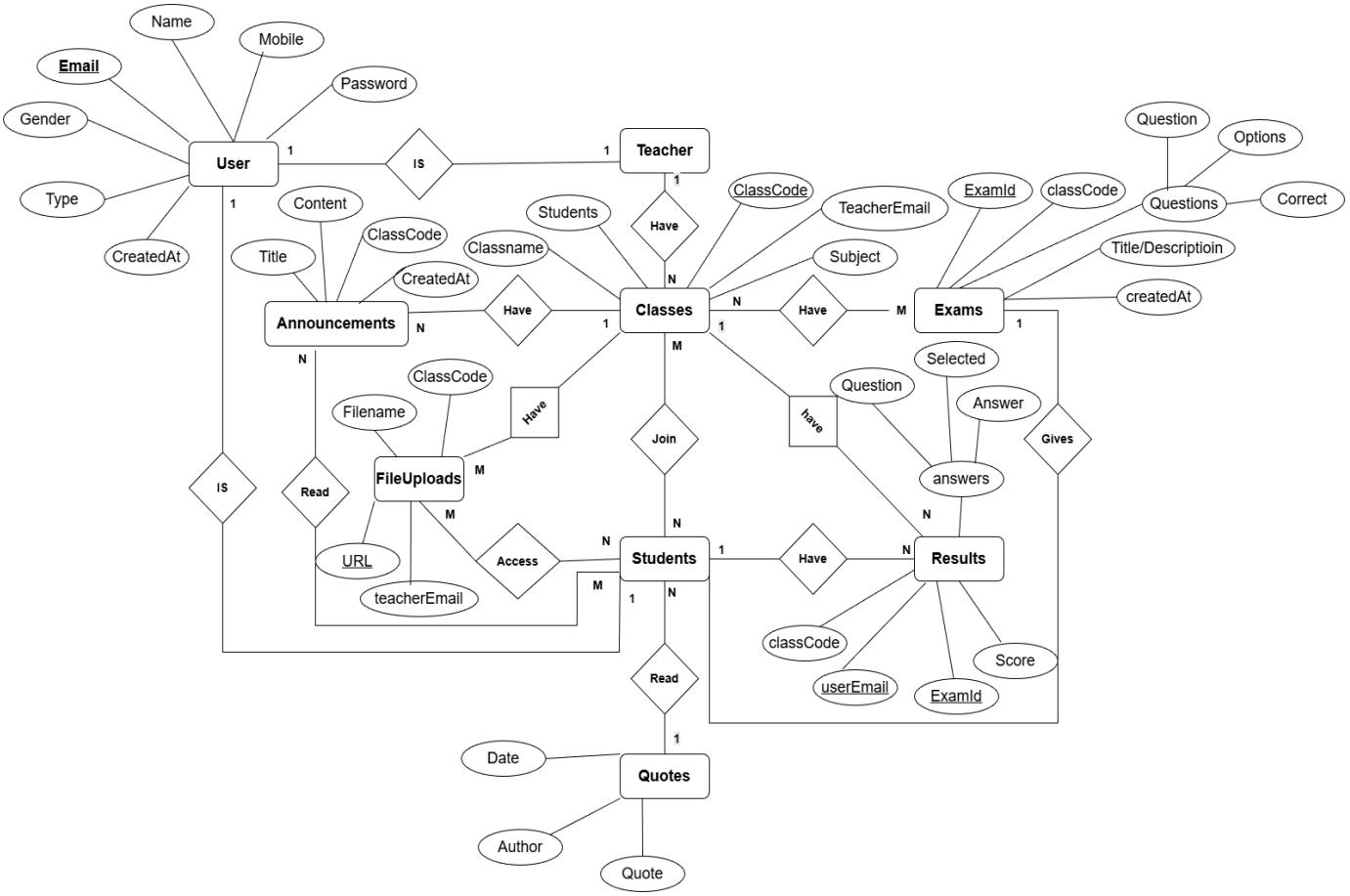

Figure 4.3: ER Diagram

4.4 Flowchart

The Flowchart outlines the logical flow of operations within the LearnDesk system, starting from user authentication to quiz completion and result generation. It demonstrates step-by-step processes for both students and teachers, including class creation, quiz assignment, quiz security checks, question attempting, and evaluation. The diagram simplifies the understanding of system functionality by visually breaking down complex procedures into sequential actions. It ensures that every stage of system operation is transparent, predictable, and easy to analyze.

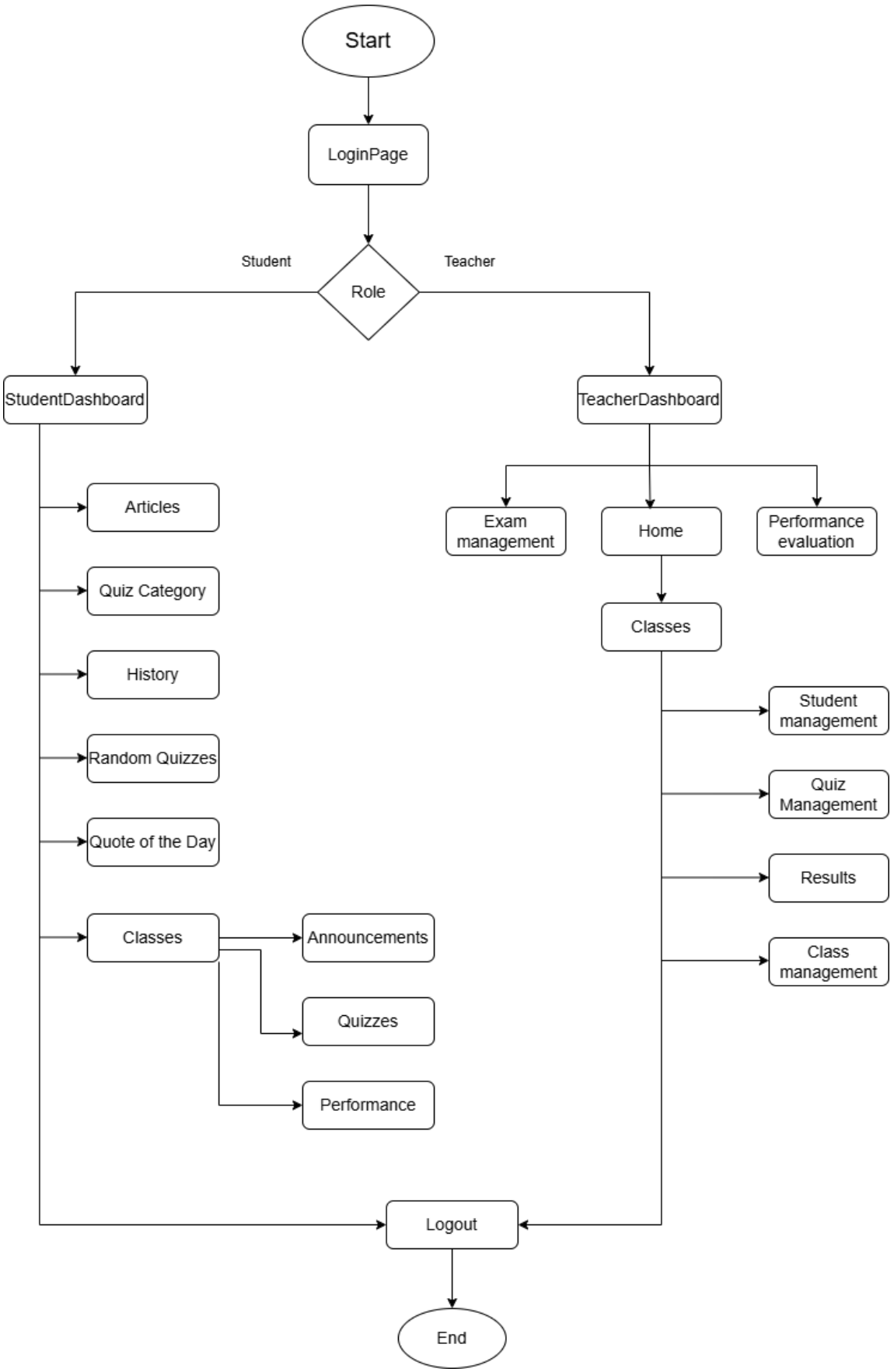

Figure 4.4: Flowchart

4.5 Class Diagram

The Class Diagram represents the structural design of the key components and modules of the system. Although the MERN stack is not purely object-oriented, the class diagram conceptually outlines major logical units such as controllers, services, models, and frontend modules. It shows how these units interact with each other, what data they hold, and what functionalities they provide. This diagram helps in understanding the internal architecture of the application and ensures proper separation of concerns throughout the system.

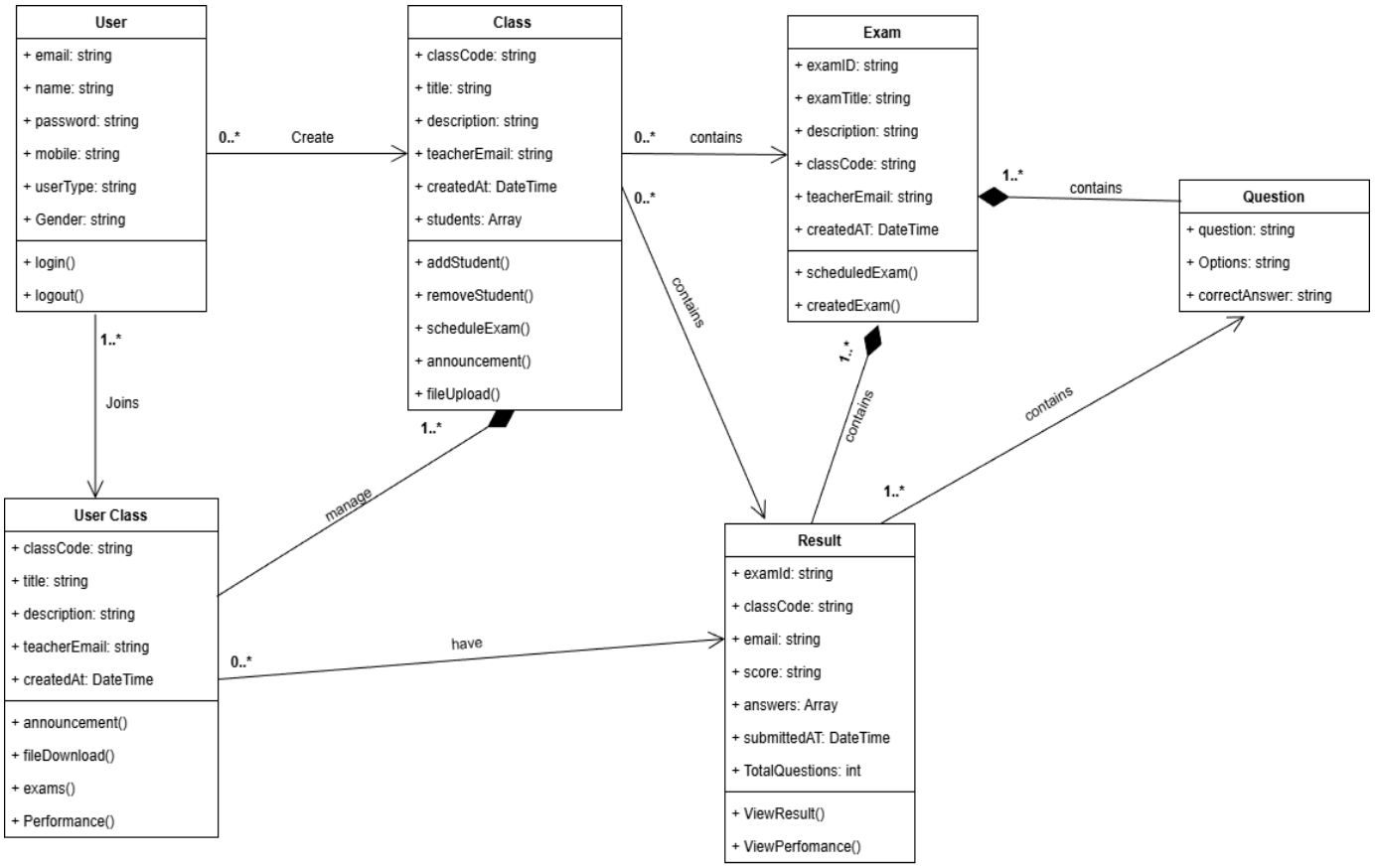

Figure 4.5: Class Diagram

4.6 Database Schema

The Database Schema section provides a detailed description of each collection used in the system along with their fields and relationships. It outlines how data is stored, retrieved, and connected across Users, Classes, Exams, Questions, and Results. The schema ensures structured data management by defining key attributes, references, embedded objects, and indexes required for efficient performance. This section serves as

a blueprint for the backend implementation and guarantees a systematic and organized database design.

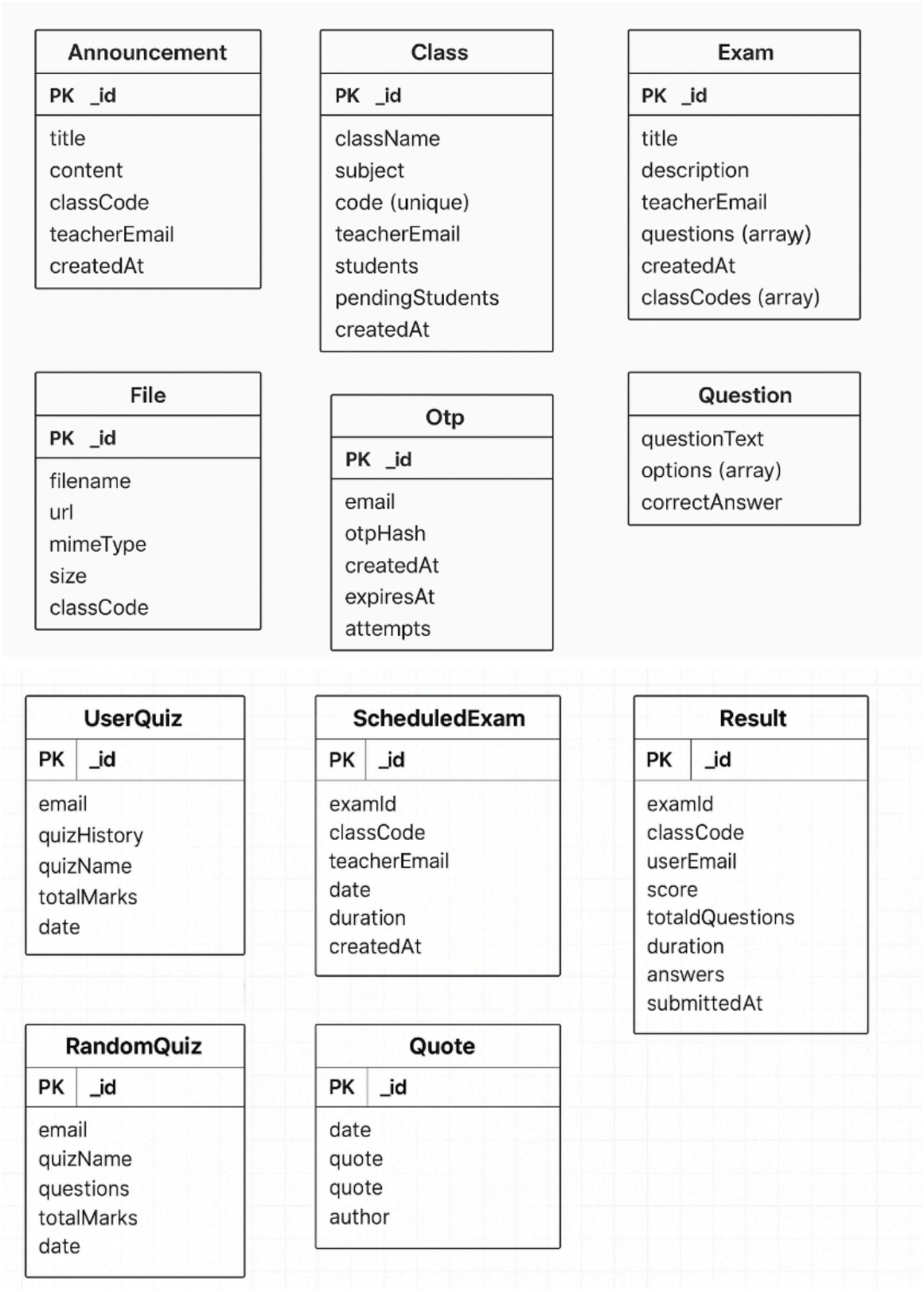

Figure 4.6: Database schema

Chapter 5
System Implementation

5.1 Technologies Used

The implementation of LearnDesk is carried out using the MERN stack, which provides a full-fledged environment for efficient web application development. React.js is used to build a responsive and interactive user interface that enhances the learning and quiz experience. The backend is developed using Node.js and Express.js, enabling smooth API handling, secure authentication, and efficient server-side processing. MongoDB serves as the primary database, storing user details, classes, quizzes, questions, and results in a flexible document-oriented format.

Table 5.1 Technologies Used in LearnDesk

Technology	Category	Purpose in Project
React.js	Frontend Framework	Builds the user interface, dashboards, quiz screens, and interactive components.
Node.js	Backend Runtime	Executes server-side logic, handles API requests, manages authentication and quiz processing.
Express.js	Backend Framework	Simplifies route handling, middleware functions, and REST API development.
MongoDB	Database	Stores users, quizzes, classes, results, and article data in a flexible NoSQL schema.
JWT (JSON Web Tokens)	Security / Authentication	Ensures secure login sessions, protects routes, and manages token-based access.
Bcrypt	Security	Hashes user passwords before storing them in the database.
Third-party APIs	External Services	Provides “Quote of the Day” and random Quizzes.
Docker	Containerization	Encapsulates frontend and backend into separate containers for easier deployment.

Render	Deployment Platform	Hosts the application online and manages automated builds and deployment.
Nodemailer	Email Service	Sends verification emails, notifications, file upload messages and class announcements.
Mongoose	ODM (Object Data Modeling)	Simplifies communication between Node.js and MongoDB, and defines database schemas.

5.2 Key Features Implemented

The LearnDesk system incorporates a wide range of features that support both learning and assessment. Students can access articles, attempt quizzes, join classes, and view instant results with detailed evaluation. Teachers can create classes, upload study materials, manage students, and prepare secure, time-based quizzes with automatic submission.

Security is a major focus in implementation. The system enforces full-screen mode during quizzes, restricts copying or switching tabs, and includes watermarking to prevent unauthorized sharing of content. Email verification and email-based notifications keep users informed about important updates.

The platform also features automatic result computation, announcement delivery, notifications, and a clean user-friendly interface to make learning intuitive. Together, these features provide a complete solution for placement preparation and classroom-based assessments.

5.3 Interface Samples

This section presents sample interfaces from the LearnDesk application to demonstrate the usability and layout of the system. Screens such as the login page, dashboard, class management panel, quiz creation interface, quiz attempt screen, and result summary are showcased to illustrate how different functionalities are visually organized. Each interface is designed with clarity and simplicity in mind to ensure that both students and teachers can interact with the system effortlessly.

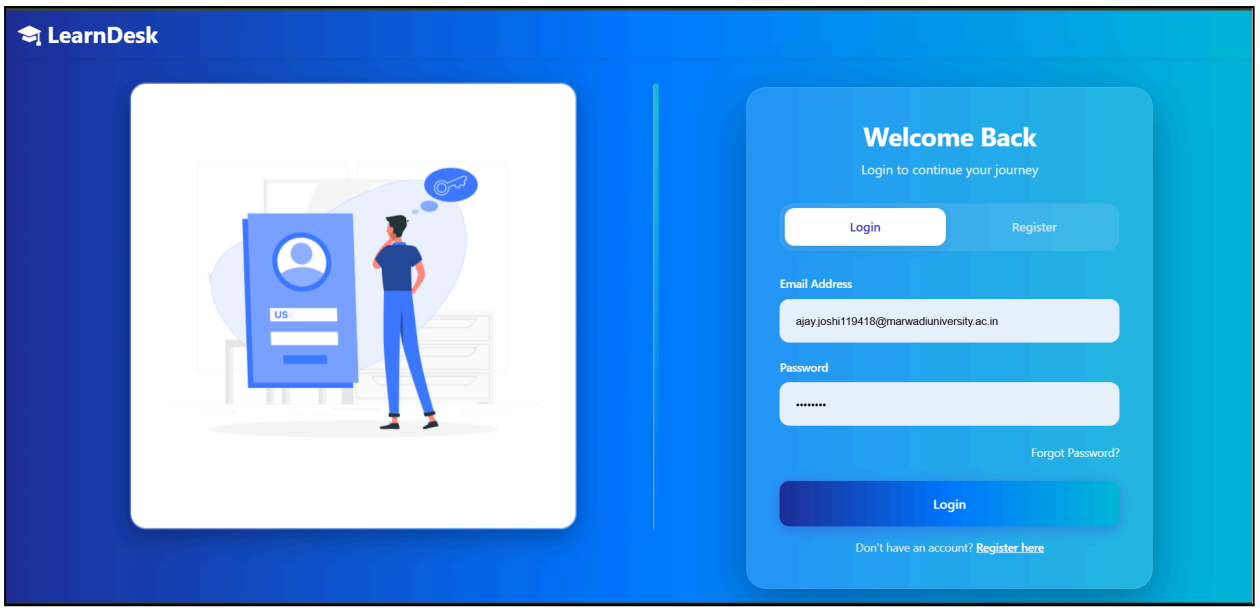

Figure 5.1: Login Page

The Teacher Dashboard provides teachers with tools to manage classes, add students, create quizzes, and share study materials. It offers a structured overview of all teaching activities, including announcements, files, and performance tracking for each student. The interface focuses on simplicity and efficiency, enabling teachers to manage learning tasks effortlessly.

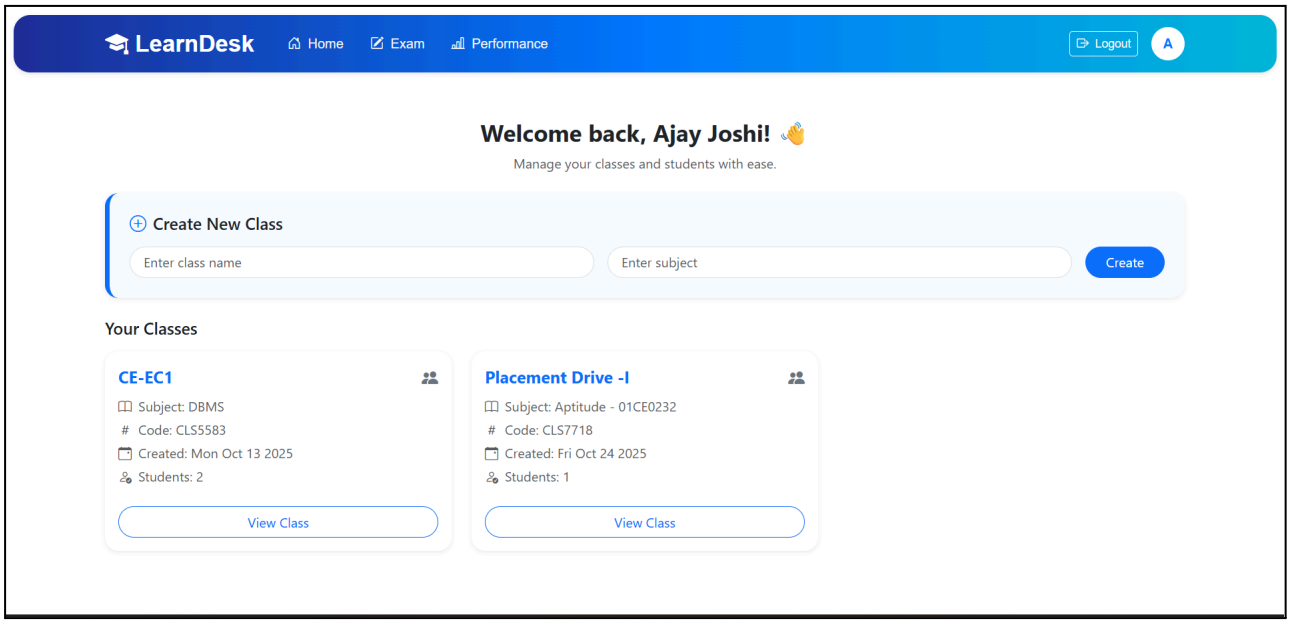

Figure 5.2: Admin dashboard

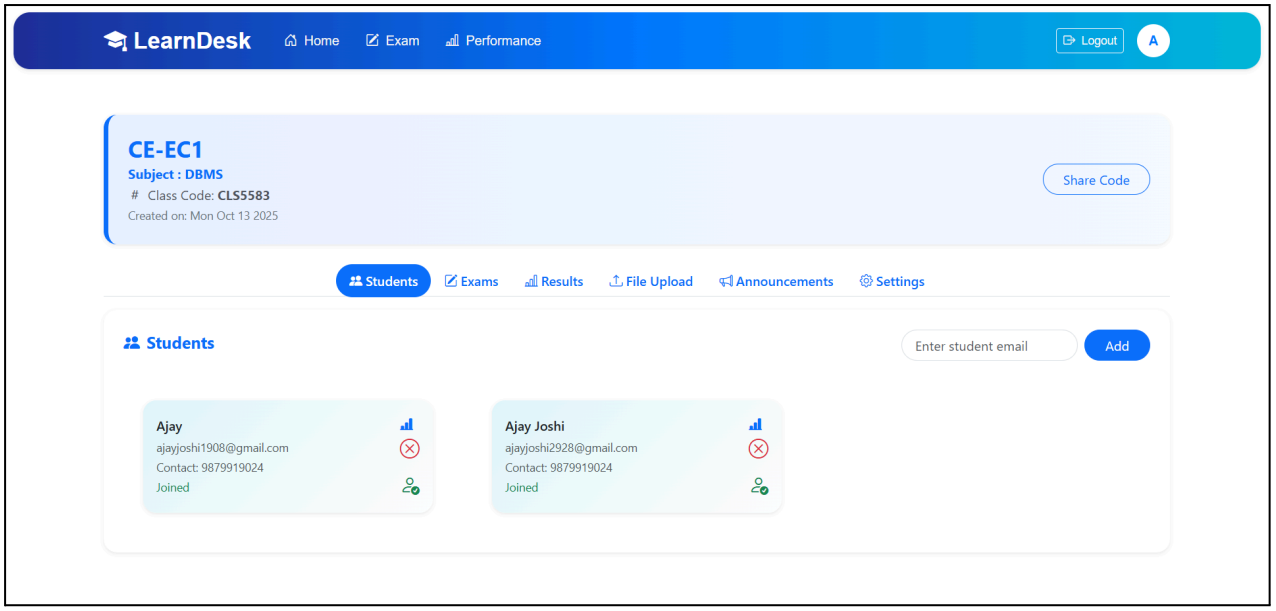

Figure 5.3: Admin Class management

The Student Dashboard displays all essential learning and assessment options in one place. Students can access articles, explore aptitude quizzes, join teacher-created classes, and attempt assigned tests. Quick links, notifications, and motivational quotes help students stay engaged. The dashboard layout is designed for smooth navigation and provides a clear view of ongoing and completed activities.

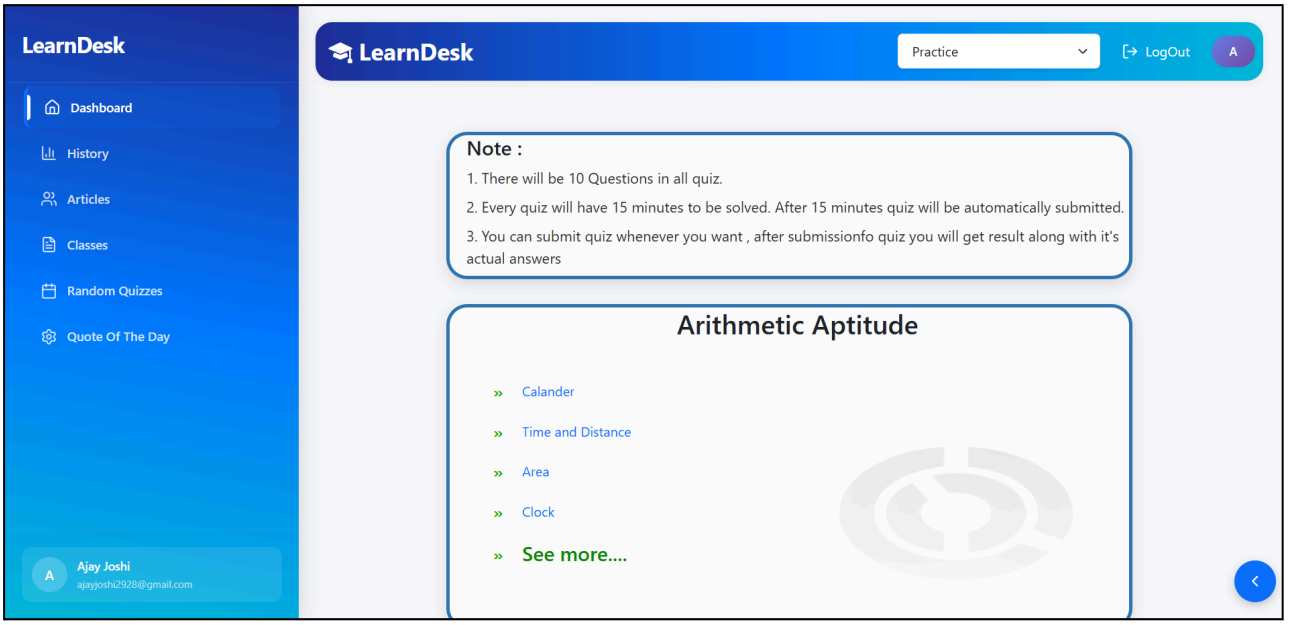

Figure 5.4: Student dashboard

Chapter 6:
Testing

6.1 Testing Strategies

To ensure the LearnDesk platform worked accurately and reliably, multiple testing approaches were adopted throughout development. The two major techniques applied were **Unit Testing** and **Integration Testing**, along with manual functional verification.

6.1.1 Unit Testing

Unit testing was carried out on individual backend modules such as authentication, quiz creation, result calculation, JWT validation, and API route handling. Each function was tested in isolation to ensure correct input handling, error responses, and expected output behavior. These tests helped identify early bugs and improved the stability of core logic.

6.1.2 Integration Testing

Integration testing was performed to confirm that various modules communicate properly with each other. This included:

- Frontend ↔ Backend API communication
- Backend ↔ MongoDB interactions
- Quiz timer ↔ Auto-submission logic
- Class management ↔ Student list updates
- Authentication ↔ Role handling (Student / Teacher)

Integration testing ensured that combined features behaved smoothly when used together, replicating real-world scenarios such as a teacher creating a quiz and students attempting it. Additionally, functional testing, UI testing, and security checks were performed to ensure consistency and smooth user experience across different modules.

6.2 Test Cases & Results

This section contains the major test cases executed during system evaluation along with their results. Test cases were designed based on core functionalities such as authentication, quiz operations, class management, and data validation.

Table 6.1 Student dashboard Test Case Table

Test Case ID	Test Case Description	Expected Result	Actual Result	Status
TC-01	Login with valid credentials	User should be logged in successfully	Logged in and redirected to dashboard	Pass
TC-02	Login with invalid password	Display error message	Error shown correctly	Pass
TC-03	Student attempts a quiz	Quiz should start with timer	Quiz opens and timer starts	Pass
TC-04	Quiz timer ends	Quiz should auto-submit	Auto-submission triggered	Pass
TC-05	Teacher creates a new class	Class should be stored in database	Class created successfully	Pass
TC-06	Student joins a class	Student should be added to class list	Successfully added to class	Pass
TC-07	Full-screen mode enforcement	Restrict quiz page exit	Warning displayed and attempt logged	Pass
TC-08	Announcement creation	Students should receive announcement	Notification displayed	Pass

Table 6.2 Teacher dashboard Test Case Table

Test Case ID	Test Description	Expected Result	Actual Result	Status
TC-T01	Create a new class with valid details	Class should be stored and appear in teacher dashboard	Works as expected	Pass
TC-T02	Create class with missing fields	System should show validation error	Validation error displayed	Pass
TC-T03	Add student via email to a class	Student should be added to class list	Student added successfully	Pass
TC-T04	Add student who already exists in class	System should show appropriate error	Duplicate error displayed	Pass
TC-T05	Create quiz with all valid inputs	Quiz saved and visible to class students	Works as expected	Pass
TC-T06	Create quiz without questions	System should prevent quiz creation	Error shown correctly	Pass
TC-T07	Upload PDF or file to class	File should be stored and visible for students	File uploaded successfully	Pass
TC-T08	Post an announcement to class	Students should receive notification	Notification delivered	Pass
TC-T09	View student results after quiz	Results should load accurately from DB	Loaded correctly	Pass
TC-T10	Modify an existing quiz	Updates should reflect in DB	Quiz updated successfully	Pass

6.3 Testing Summary

The testing process confirmed that the LearnDesk platform performs reliably across all major modules. All primary functionalities such as authentication, quiz management, class operations, and secure assessment worked as intended under multiple test scenarios. The system demonstrated stable performance, quick response times, and consistent behavior across different devices and browsers.

Most importantly, security measures like full-screen detection, copy restriction, and auto-submission were verified thoroughly and behaved accurately. Minor UI adjustments were made during testing to improve clarity and navigation. Overall, the system met the expected functional and non-functional requirements, making it ready for deployment and real-world usage.

Chapter 7
Results & Conclusion

7.1 Results & Discussion

The LearnDesk platform successfully delivers a secure, user-friendly, and efficient environment for learning and assessment. The system integrates multiple features such as class management, quiz creation, topic exploration, and instant result generation into a single platform. The results obtained during testing and real-time usage show that students can attempt quizzes without facing disruptions, and teachers can manage classes and assessments with ease.

Quiz security measures like full-screen enforcement, watermarking, and copy restrictions functioned exactly as intended, minimizing malpractice during assessments. The use of Docker and Render for deployment ensured smooth hosting with stable performance. Overall, the system demonstrates reliability, accuracy in evaluation, and intuitive navigation for both students and teachers.

7.2 Performance Analysis

The performance of the LearnDesk system was evaluated based on speed, responsiveness, resource usage, and overall stability. The MERN stack architecture ensured quick API responses and fast page rendering. MongoDB’s flexible schema allowed efficient storage and retrieval of quiz data, user information, and results.

During testing, quiz pages loaded quickly, timers ran without interruption, and auto-submission occurred consistently at the exact timer endpoint. The system handled multiple user operations such as simultaneous quiz attempts, class creation, and file uploads without performance drops. Docker-based deployment further improved consistency by isolating frontend and backend services, resulting in predictable behavior across different environments.

Overall, the platform performs efficiently for real-world educational use, supporting both learning and assessment workflows.

7.3 Achievements vs Objectives

The primary objectives of the project were to build a unified learning and assessment platform, ensure quiz security, provide teacher–student interaction tools, and deploy the system in a real hosting environment. All key objectives were successfully met.

Objective	Achievement
Build a unified learning platform	Implemented student dashboard, articles, aptitude quizzes, class joining, etc.
Provide tools for teachers	Delivered class management, quiz creation, announcements, file uploads.
Ensure secure quiz attempts	Integrated full-screen mode, copy restriction, watermarking, and auto-submission.
Instant evaluation & results	Results generated instantly with detailed responses.
Smooth user experience	Clean UI with structured navigation for both roles.
Secure authentication	Implemented JWT, email verification, hashing, and encryption.
Containerized & deployed	Successfully dockerized and deployed the system on Render.

The results clearly show that the implemented system meets and, in some areas, exceeds the original goals.

7.4 Conclusion & Future Scope

7.4.1 Conclusion

LearnDesk provides a complete and reliable solution for placement preparation and classroom-based assessments. By combining learning materials, quizzes, class management, real-time evaluation, and strong security measures, it creates an efficient environment for both students and teachers. The use of modern web technologies, secure authentication, and cloud deployment further enhances the platform's stability and accessibility. The project demonstrates how interactive learning and secure online assessments can be brought together in one cohesive application.

7.4.2 Future Scope

Although the system is feature-rich, several enhancements can be introduced in future versions:

- Mobile application for Android and iOS users.
- AI-based performance analysis and personalized learning recommendations.
- Live classes, video lectures, and interactive discussion forums.
- Support for subjective questions and automated evaluation using NLP.
- Real-time proctoring features such as webcam monitoring, voice detection, and behavior tracking.

These improvements can make LearnDesk even more scalable, intelligent, and suitable for large-scale educational use.

References

[1] MongoDB Inc. “MongoDB Manual.” Available at: https://www.mongodb.com/docs/

[2] Meta Platforms Inc. “React.js Official Documentation.” Available at: https://react.dev/

[3] Node.js Foundation. “Node.js v18.x Documentation.” Available at: https://nodejs.org/en/docs/

[4] Express.js Team. “Express Official Documentation.” Available at: https://expressjs.com/

[5] bcrypt npm package. “Bcrypt Library Documentation.” Available at: https://www.npmjs.com/package/bcrypt

[6] Docker Inc. “Docker Overview & Guides.” Available at: https://docs.docker.com/

[7] Mongoose ODM. “Mongoose Documentation.” Available at: https://mongoosejs.com/docs/

[8] Render.com. “Deploying Web Services.” Available at: https://render.com/docs/

[9] GeeksforGeeks. “Aptitude Questions and Answers.” Available at: https://www.geeksforgeeks.org/aptitude/aptitude-questions-and-answers/

[10] IndiaBix. “Aptitude and Reasoning Practice.” Available at: https://www.indiabix.com/

[11] Instructure Inc. “Canvas Learning Management System.” Available at: https://www.instructure.com/canvas

[12] Mozilla Foundation. “JavaScript Reference & Tutorials.” Available at: https://developer.mozilla.org/

[13] Nodemailer. “Nodemailer – Sending Emails with Node.js.” Available at: https://nodemailer.com/about/