CS19442-SOFTWARE ENGINEERING CONCEPTS

ONLINE TEST SYSTEM

S.NO	NAME	REGISTER.NO
1	A. ABHILASH	220701004
2	P.S. ABISHEAK	220701010
3	R.AGISHRAJ	220701016
4	A.AKILAN	220701022
5	C. ARTHEY	220701028
6	S.AVINASH	220701034

ONLINE TEST SYSTEM

Table of contents

EX.NO:1 DATE:20/02/2024

SOFTWARE REQUIREMENT SPECIFICATIOIN

1. Introduction

The Online Test System is a web-based application designed to facilitate the administration, creation, and management of online tests and examinations. This system aims to provide a convenient platform for educators and institutions to conduct assessments remotely, ensuring efficiency, security, and fairness in the testing process.

2. Purpose

The purpose of this document is to provide a comprehensive overview of the functional and non-functional requirements of the Online Test System, outlining its features, capabilities, and constraints.

3. Scope

The Online Test System will include features such as user authentication, test creation, test-taking interface, grading, and reporting. It will cater to both educators and students, allowing educators to create and administer tests while enabling students to participate in online examinations.

4. Functional Requirements

4.1 User Authentication

- Users (educators and students) must be able to register for an account.

- Users must be able to log in securely using their credentials.

- Password recovery functionality should be provided.

4.2 Test Creation and Management

- Educators must be able to create tests by specifying details such as title, duration, instructions, and question types (e.g., multiple choice, short answer).

- Educators should be able to add, edit, and delete questions from a test.

- Tests should support multimedia content such as images, audio, and video.

- Test scheduling functionality should be provided, allowing educators to set start and end times for tests

4.3 Test-Taking Interface

- Students should be able to view available tests upon logging in.

- Students must be able to select and start a test within the specified time frame.

- The test interface should display questions one at a time with navigation controls for moving between questions.

- Support for various question types including multiple choice, true/false, short answer, and essay questions.

- Timer functionality should be displayed to students indicating the remaining time for the test.

4.4 Grading and Reporting

- Automated grading should be provided for objective type questions.

- Educators should be able to manually grade subjective questions.

- Students should be able to view their test scores and detailed feedback post-assessment.

- Educators should have access to comprehensive reports, including test performance metrics and analytics.

6. Constraints

- The system must be compatible with modern web browsers such as Google Chrome, Mozilla Firefox, and Safari.

- The application should be developed using scalable and maintainable technologies to accommodate future enhancements and updates.

7.Conclusion

The Online Test System aims to streamline the process of conducting online assessments, offering a comprehensive solution for educators and students alike. By adhering to the

specified requirements and constraints outlined in this document, the system will provide a robust and reliable platform for administering tests securely and efficiently.

EX.NO:2 DATE:01/03/2024

AGILE AND SCRUM METHODOLOGY

Project Vision:

To develop a secure and efficient online test system that ensures fairness and integrity during exams while providing teachers with a convenient platform to review student performance.

Scrum Team:

- Product Owner (PO): S.Avinash

- Scrum Master (SM): A.Akilan

- Development Team: C.Arthey, P.S.Abisheak, A.Abhilash, R.Agishraj

Sprint Duration:

- Two-week sprints

Backlog Refinement:

- Product Owner works closely with stakeholders to prioritize features based on value and urgency.

- Define user stories based on requirements provided.

Sprint Planning:

- Sprint Planning Meeting:

- Product Owner presents the prioritized user stories from the backlog.

- Development Team estimates the effort required for each user story.

- Select user stories for the sprint based on capacity and priority.

Sprint Execution:

- Daily Stand-up Meetings:
- Development Team discusses progress, impediments, and plans for the day.
- Development:
- Developers implement features according to the user queries.
- Testers conduct testing to ensure functionality and security.
- Designers work on UI/UX enhancements.
- Continuous Integration and Deployment:
- Features are integrated regularly and deployed to a testing environment for feedback.

Sprint Review:

- Sprint Review Meeting:
- Demo of completed user stories to stakeholders.
- Feedback gathered for future iterations.

Sprint Retrospective:

- Sprint Retrospective Meeting:
- Scrum Team reflects on the sprint's process and identifies areas for improvement.
- Action items created to address issues and enhance team collaboration.

Key Features to Implement:

1. Auto-submission after time limit:

- Development Team implements a timer feature that automatically submits the exam when the time limit is reached.

2. Tab switching prevention:

- Developers implement a feature to restrict tab switching during the exam session.

- If tab switching occurs, the exam session is locked, and the attempt is invalidated.

3. Teacher's access to student marks:

- Develop a feature for teachers to access and review student marks after the exam submission.

Iterative Development:

- Features are developed iteratively and incrementally, allowing for continuous feedback and improvement.

- Regular releases ensure that stakeholders can provide input throughout the development process.

Conclusion:

This Agile Scrum methodology provides a framework for the development of the online test system, focusing on delivering value to stakeholders while ensuring the integrity and security of the exam process. Adjustments can be made as needed based on feedback and changing requirements throughout the project lifecycle.

EX.NO:3 DATE:12/03/2024

USER STORIES

1. **Title**:

As a teacher, I want to create multiple-choice questions for my online test so that I can assess my students' understanding of the subject.

Description:

Teachers should have the ability to create multiple-choice questions with various options and designate the correct answer.

Acceptance Criteria:

- Teachers can access the question creation functionality from their dashboard.

- Multiple-choice questions are displayed in a structured format, teachers to input question text and options.

- Teachers can mark the correct answer for each question.

Priority:

High

2. **Title**:

As a student, I want to join an online test session with a unique code so that I can participate in the assessment.

Description:

Students should be able to enter a unique code provided by their teacher to access the online test session.

Acceptance Criteria:

- Students can navigate to the "Join Test" section and enter the unique code.

- Upon entering the correct code, students gain access to the test questions.

- Incorrect codes should prompt an error message indicating invalid access.

Priority:

High

3. **Title**:

As a teacher, I want to set a time limit for the online test so that student's complete assessment within a specified duration.

Description:

Teachers should be able to specify the duration of the online test, after which students' submissions are automatically collected.

Acceptance Criteria:

- Teachers can set the test duration while creating the test session.

- A countdown timer is displayed to students during the test session, indicating the remaining time.

- Upon reaching the time limit, the test session ends automatically, and student submissions are collected.

Priority:

Medium

4. **Title**:

As a student, I want to review my answers before submitting the online test so that I can ensure accuracy.

Description:

Students should have the ability to review and modify their answers before final submission.

Acceptance Criteria:

- Students can navigate to a "Review Answers" section after completing all questions.

- During the review, students can navigate through the questions and modify their answers if needed.

- Changes made during the review are reflected in the final submission.

Priority:

Medium

5. ## Title:

As a teacher, I want to view and analyze students' test results in a detailed report format so that I can assess their performance.

Description:

Teachers should have access to comprehensive reports detailing students' test results, including scores and question-wise performance.

Acceptance Criteria:

- Teachers can access the "Test Results" section from their dashboard.

- The report provides an overview of the test, including average score, highest and lowest scores, and completion rate.

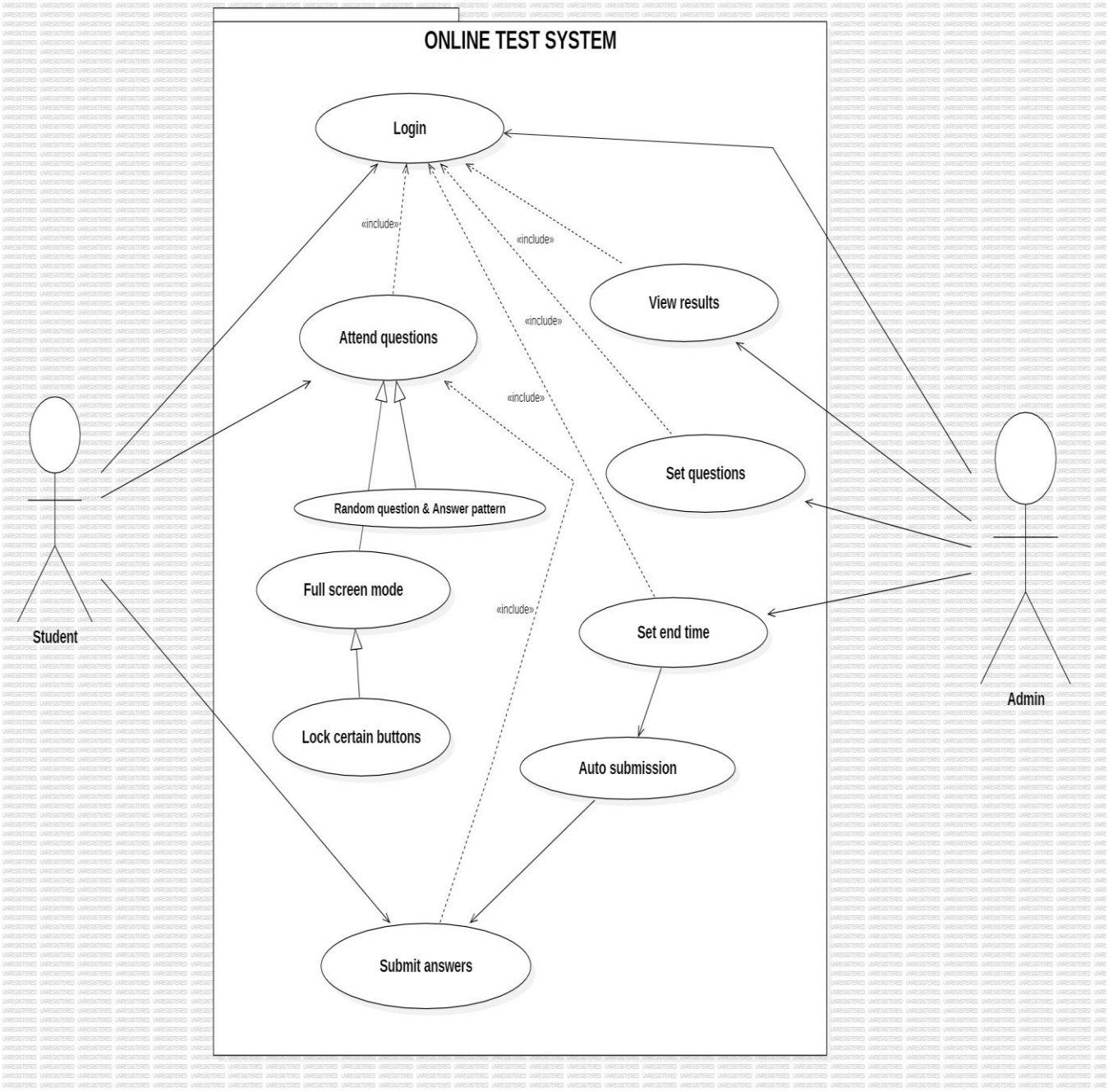
- Detailed insights into individual student performance, including scores and time taken for each question, are available.

Priority:

High

EX.NO:4 DATE:19/03/2024

USECASE DIAGRAM



Use Case Diagram Explanation

Actors

Teacher

The user who can set up and manage tests.

Student

The user who takes the tests.

Use Cases for Teacher

Login

Allows the teacher to securely access the system.

Set Questions

Enables the teacher to create and set questions for the test.

Set End Time

Allows the teacher to specify the end time for the test.

View Results

Permits the teacher to view the results after the test is completed by the students.

Use Cases for Student

Login

Allows the student to securely access the system.

Attend Questions

Enables the student to attend the test questions.

Submit Answers

Allows the student to submit their answers upon completion of the test.

System Features

Auto Submission

Automatically submits the student's answers when the set end time is reached.

Full Screen Mode

Forces the test to be taken in full screen mode to prevent distractions and malpractice.

Lock Certain Buttons

Locks specific buttons such as window switch key or minimize to maintain test integrity.

Random Question & Answer Pattern

Presents questions and answer options in a random order to reduce the likelihood of cheating.

Relationships

Includes Relationship

Attend Questions includes Login for Students.

Set Questions, Set End Time, and View Results include Login for Teachers.

Submit Answers includes Attend Questions for Students.

Generalization Relationship

Full Screen Mode is a general feature that includes Lock Certain Buttons.

Attend Questions includes Random Question & Answer Pattern.

Summary

The Teacher can log in to the system, set questions, set the end time, and view results after students complete the test.

The student can log in, attend questions, and submit answers.

The System includes features such as auto submission of tests at the end time, full screen mode with certain locked buttons to prevent malpractice, and randomization of questions and answers to ensure test integrity.

EX.NO:5 DATE:29/03/2024

NON-FUNCTIONAL REQUIREMENT

1. Performance

- The system should maintain responsiveness even under peak loads.

- Performance tests should be conducted to ensure that the system handles expected user traffic without slowdowns or crashes.

2. Security

- User authentication and authorization mechanisms should be robust.

- Data transmission should be encrypted to prevent interception and unauthorized access.

- Regular security audits and vulnerability assessments should be conducted.

3. Scalability

- The system architecture should support horizontal scaling.

- Scalability tests should be performed to verify that the system can scale seamlessly as user traffic grows.

4. Availability

- The system should have built-in redundancy and failover mechanisms.

- Scheduled maintenance activities should be planned during off-peak hours.

5. Usability

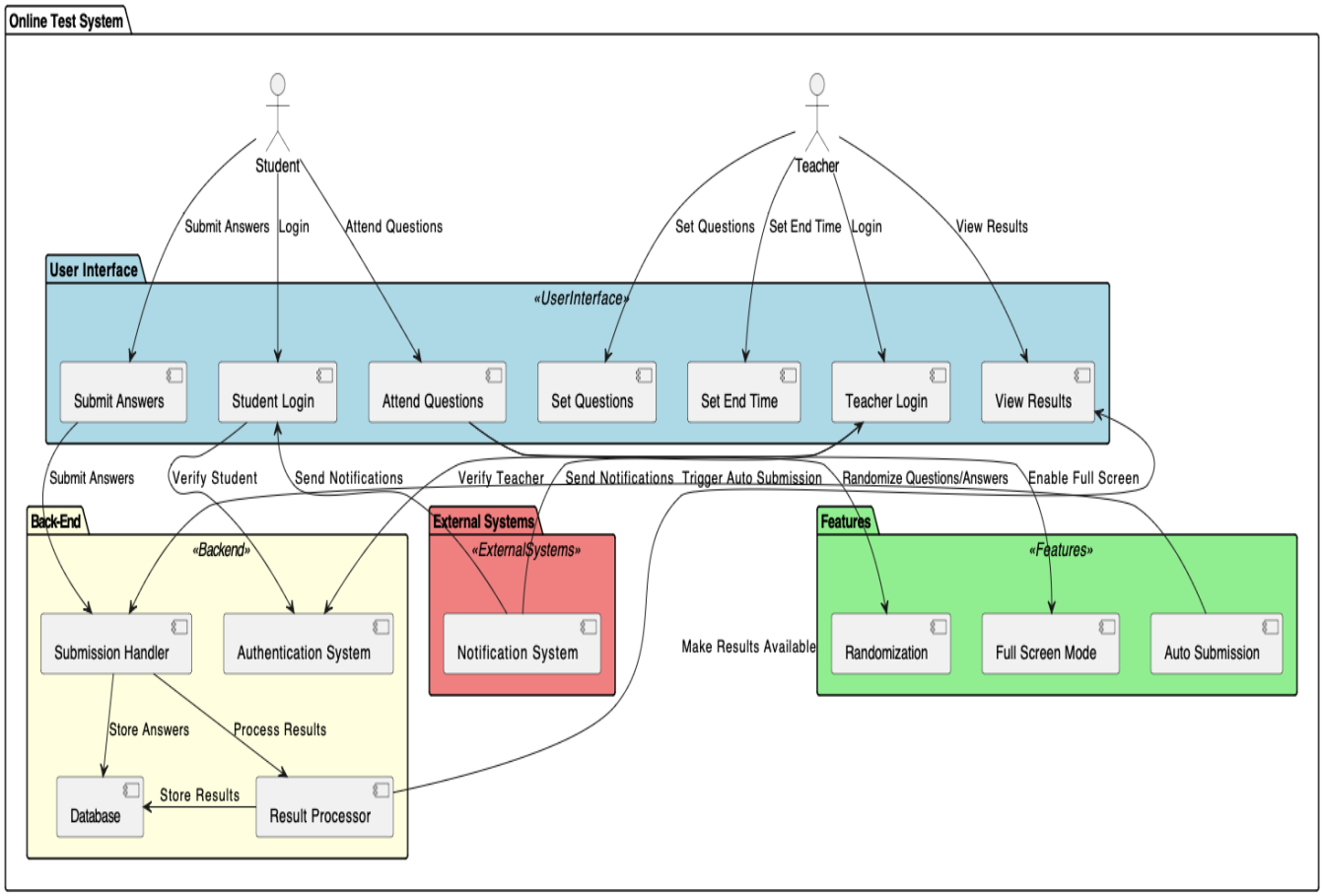
- The system should be intuitive and user-friendly.

- User interface design should follow best practices for usability.

- User feedback mechanisms should be in place to gather input on usability improvements

EX.NO:6 DATE:09/04/2024

OVERALL PROJECT ARCHITECTURE



User Interface components are hosted on the "Web Server".

Features are managed by the "Application Server".

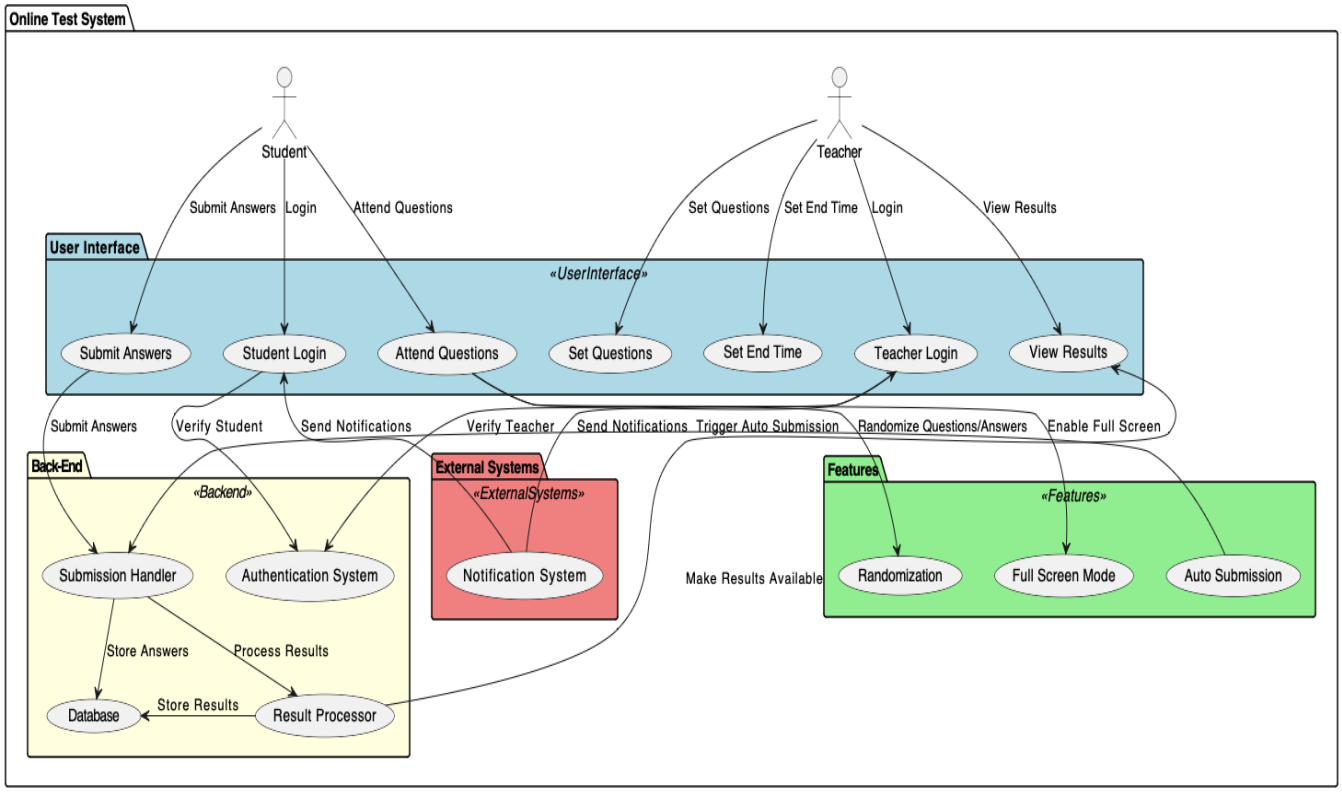
Back-End components reside on the "Database Server".

External Systems include the "Notification System".

The diagram shows the interactions between these components and the actors (Teacher and Student), providing an overview of the system's architecture.

EX.NO:7 DATE:19/04/2024

BUSINESS ARCHITECTURE



User Roles and Differentiation: The system distinguishes between teachers and students, offering tailored experiences based on their roles and responsibilities.

- **Package Organization:** The diagram utilizes packages to categorize system components, emphasizing clarity and structure in outlining roles and interactions.
- **Intuitive User Interfaces**: User interfaces are designed to be user-friendly, facilitating seamless navigation and interaction for both teachers and students.
- **Backend System Core:** A robust backend system forms the foundation, managing data flow and operations to ensure efficiency and reliability.
- **Integration with External Systems**: The diagram highlights interactions with external systems, showcasing the system's potential for integration with additional tools or databases.
- **Comprehensive Overview**: The diagram provides a comprehensive blueprint of the system's functionalities and interactions, aiding in understanding its intricate network.

- **Enhanced User Experience**: The system aims to deliver an enriched experience by facilitating various actions such as test creation, management, participation, and result assessment efficiently.

Current Process:

Login: Users (teachers and students) log in to the online test system using their respective credentials.

Question Setting (Teachers): Teachers set questions for the test, including specifying the end time for the test.

Test Participation (Students): Students log in and attend the test, answering questions within the specified time limit.

Auto Submission: The system automatically submits the test once the end time is reached.

Result Viewing (Teachers): Teachers can view the results of the test after students finish and submit their answers.

Security Features: Certain security features, such as full-screen mode and locked buttons, are implemented to prevent malpractice.

Explanation for Different Personas:

Teachers:

Question Setting: Teachers log in to the system and set questions for the test, specifying the end time.

Result Viewing: After students finish the test, teachers can view the results to assess student performance.

Students:

Test Participation: Students log in to the system and attend the test, answering questions within the specified time limit.

Auto Submission: The system automatically submits the test once the end time is reached, ensuring fairness and integrity.

Business Process:

Authentication and Authorization: Users (teachers and students) authenticate themselves through separate logins.

Authorization levels are set to differentiate between teacher and student privileges.

Test Creation and Configuration: Teachers create tests, set questions, and specify the end time for each test.

Test Participation: Students log in, access the test, and answer questions within the specified time limit.

Auto Submission and Result Processing: The system automatically submits tests once the end time is reached.

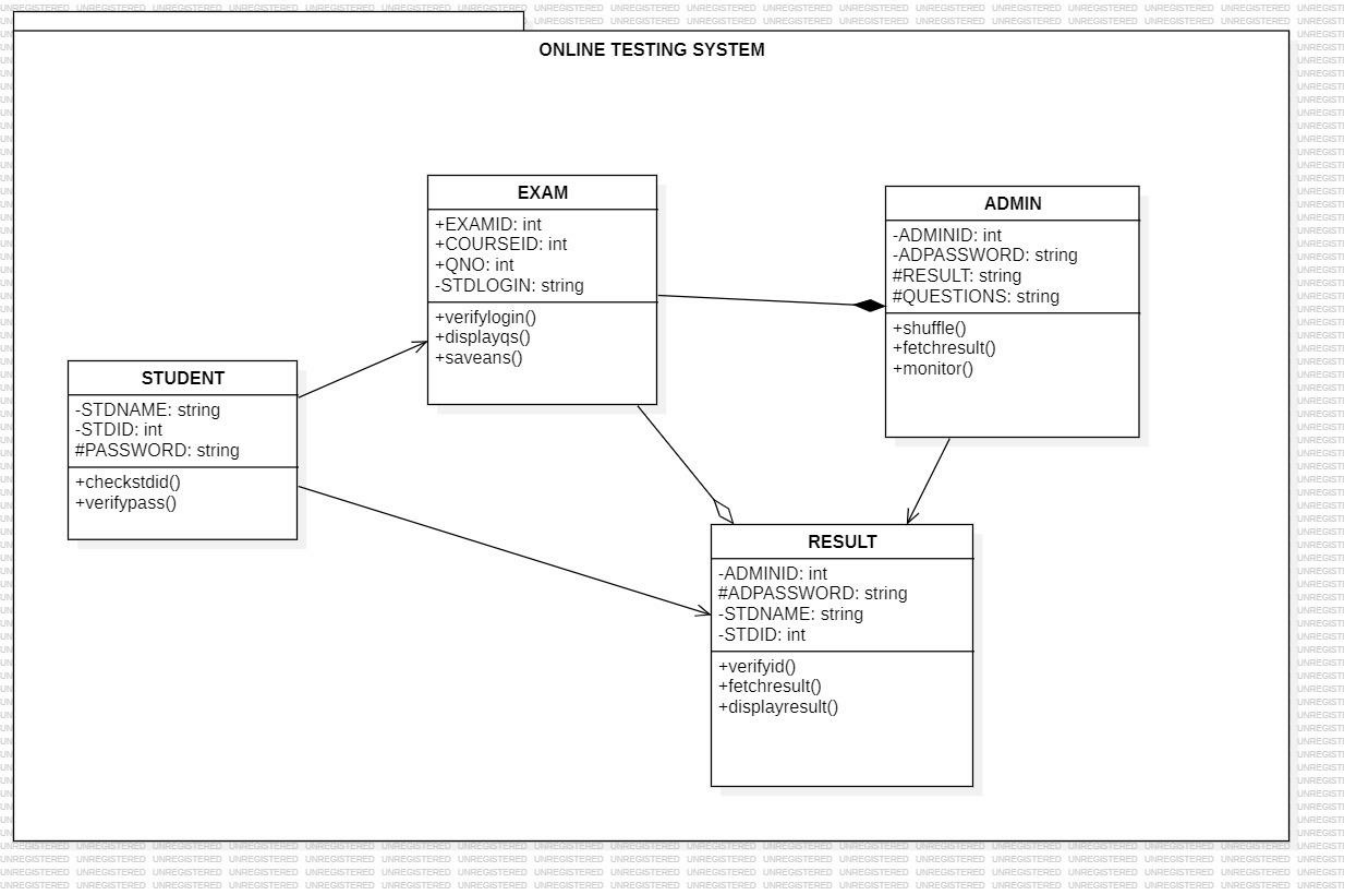
Teachers can then view and assess the results to gauge student performance.

Security Measures: Security features such as full-screen mode and locked buttons are implemented to prevent malpractice and ensure test integrity.

This breakdown outlines the current process, explains how it works for different personas, and provides insights into the overall business process of the online test system

EX.NO:8 DATE:30/04/2024

CLASS DIAGRAM



In this diagram:

User is the base class for Teacher and Student.

The teacher can set questions, set the end time, and view results.

Students can attend questions and submit answers.

Question class represents a question with multiple answers.

Answer class represents an answer to a question.

Test class represents a test with multiple questions and settings for end time and auto submission.

Result class represents the results of a student's test.

Authentication Service, Question Service, Test Service, and Result Service handle the core logic and interactions.

Changes made:

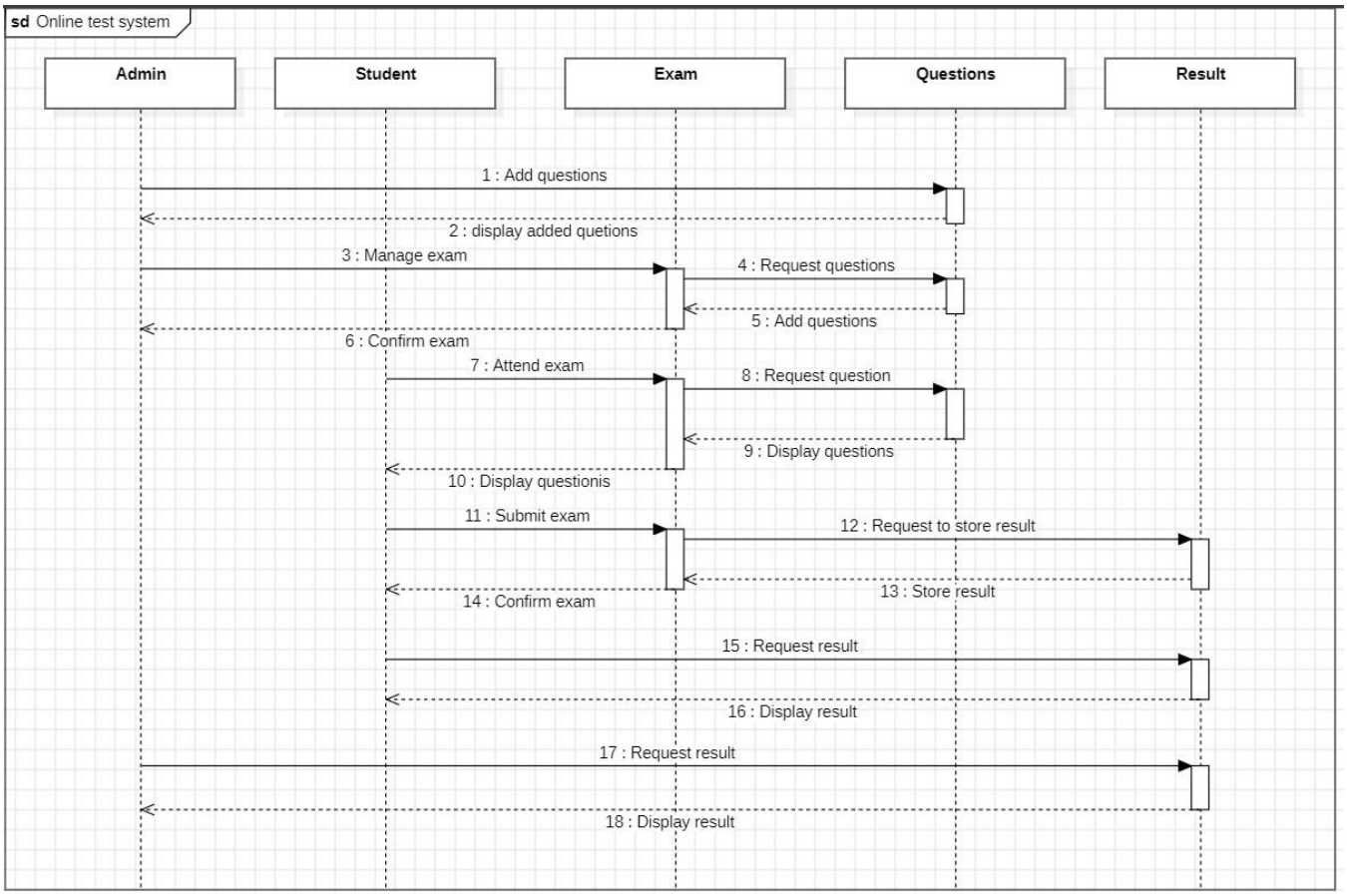
Added class backgrounds for visual differentiation.

Increased spacing between classes for clarity.

Organized classes to ensure clear and readable relationships and dependencies.

EX.NO:9 DATE:10/05/2024

SEQUENCE DIAGRAM



The sequence diagram for the online test system shows the following interactions:

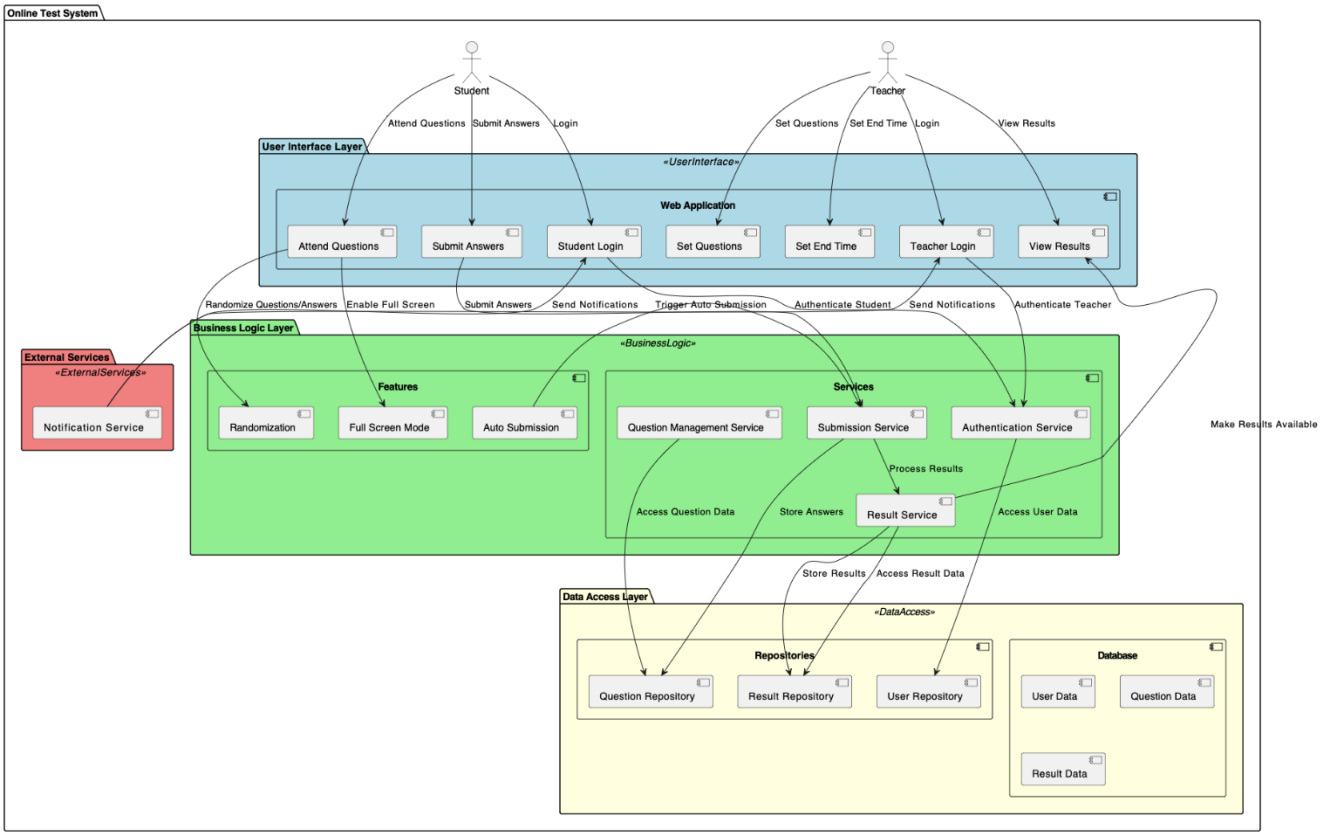
- **Admin:** Adds questions, manages and confirms exams, requests and views results.
- **Student:** Attends and submits exams, requests and views results.
- **Exam Module:** Requests and displays questions, handles exam submissions and result storage.
- **Questions Module:** Supplies questions to the Exam module.
- **Result Module:** Stores and displays exam results.

Key Features:

- **Auto Submission:** Automatically submits exams at the end time.
- **Full-Screen Mode:** Prevents cheating by locking certain keys and buttons.
- **Randomization:** Randomizes question and answer order to prevent malpractice

EX.NO:10 DATE:17/05/2024

ARCHITECTURAL PATTERN



Model:

- **Question:** Holds test questions, options, and correct answers.
- **Student:** Stores student information like name and email.
- **Teacher:** Manages teacher data such as name and email.
- **Test:** Contains questions and end time for tests.
- **Result:** Stores test results, including scores.

View:

- **TeacherDashboardView:** Allows teachers to create tests and view results.
- **StudentTestView:** Provides the test-taking interface for students.
- **LoginView:** Manages login interfaces for both teachers and students.

Controller:

- **TeacherController:** Handles test creation and result viewing for teachers.
- **StudentController:** Manages test-taking and submission for students.
- **AuthController:** Manages authentication for login and logout.

Key Features Addressed:

1. **Separate Logins:** AuthController handles authentication.
2. **Teacher Functions:** TeacherDashboardView and TeacherController facilitate test creation and result viewing.
3. **Student Functions:** StudentTestView and StudentController manage test-taking and submission.
4. **Auto Submission:** StudentController automatically submits tests when the end time is reached.
5. **UI/UX Controls:** StudentTestView ensures full-screen mode and button locking to prevent cheating.
6. **Randomization:** StudentController shuffles questions and options to prevent malpractice.