Project Name: **Pariksha Portal**

Project Member:

Virkhare Dwarkesh Sanjay **230943120027**

Shubham Chandrashekhar Pokale **230943120080**

Sujata Rajaram Natave **230943120089**

Syed Mohammad Mehdi **230943120092**

Abstract:

The Pariksha Portal is an innovative examination management system designed to streamline the process of conducting and managing exams efficiently. This project encompasses a comprehensive High-Level Design (HLD) incorporating React for the frontend, Spring Boot for the backend, and MySQL for the database management system. The system caters to both administrators and students, offering a seamless experience tailored to their respective needs.

Admin Panel:
The Admin Panel provides functionalities for administrators to manage exams, upload question papers, add student details, and monitor live exam participation. Administrators can create exams, view created exams, and access detailed analytics of student performance.

Student Panel:
The Student Panel offers students access to exams, allowing them to log in, attempt exams. It includes features such as warning alerts for tab switching, noise detection, and web camera use to maintain exam integrity.

Backend (Spring Boot):
The backend system handles authentication and authorization, exam management, result management, and real-time monitoring. It enables secure login for both admins and students, facilitates exam creation and management, stores student results, and monitors exam activities in real-time.

Database (MySQL):
The MySQL database stores crucial data such as admin credentials, exam details, questions, student information, exam participation records, and results. It ensures data integrity, reliability, and scalability for the entire system.

Workflow:
The workflow involves distinct processes for administrators and students. Administrators can create exams, manage student details, and monitor live exams. On the other hand, students can log in, access exams, attempt questions, and receive real-time warnings during exams.

Integration Points:
The integration between frontend and backend is achieved through REST APIs, enabling seamless communication between React frontend and Spring Boot backend. Additionally, the system integrates with external libraries/tools for webcam and microphone access, tab-switching detection, and noise detection to ensure exam integrity.

Technologies:
The system utilizes ReactJS for the frontend, Spring Boot for the backend, and MySQL for database management. Real-time monitoring using Face-Api.js. Security measures include HTTPS and JWT for secure authentication and exam proctoring.

Conclusion:
The Pariksha Portal offers a robust, user-friendly platform for conducting exams efficiently while ensuring exam integrity and security. Its comprehensive features cater to the needs of both administrators and students, making it a valuable tool for educational institutions seeking a modernized examination management system.

Implementation Technologies:

ReactJS Description:

ReactJS is a popular JavaScript library for building user interfaces. It allows developers to create reusable UI components and manage the state of the application efficiently.

Key Features:

Component-Based Architecture: ReactJS follows a component-based architecture, where UIs are broken down into reusable components. This makes it easier to maintain and scale applications.

Virtual DOM:

React uses a virtual DOM to optimize the rendering process. It updates only the components that have changed, resulting in better performance.

One-Way Data Binding: React implements one-way data binding, meaning data flows in one direction, from parent components to child components. This simplifies data management and reduces the risk of bugs.

JSX:

JSX is a syntax extension for JavaScript that allows developers to write HTML-like code within JavaScript. This makes it easier to create and maintain UI components.

Use Cases: ReactJS is commonly used for building single-page applications (SPAs), progressive web apps (PWAs), and large-scale web applications with complex user interfaces.

Spring Boot Description:

Spring Boot is a Java-based framework for building web applications and microservices. It provides a streamlined way to develop production-ready applications with minimal configuration.

Key Features:

Auto-Configuration: Spring Boot uses auto-configuration to automatically configure the application based on the dependencies present in the classpath. This reduces boilerplate code and simplifies development.

Embedded Servers: Spring Boot comes with embedded servlet containers like Tomcat, Jetty, and Undertow, allowing developers to run applications as standalone JAR files.

Spring Ecosystem: Spring Boot integrates seamlessly with the broader Spring ecosystem, including Spring Framework, Spring Data, Spring Security, etc. This provides a wide range of features and tools for building enterprise-grade applications.

Actuator: Spring Boot Actuator provides built-in support for monitoring and managing applications in production environments. It includes features like health checks, metrics, and application endpoints.

MySQL Description: MySQL is an open-source relational database management system (RDBMS) that is widely used for storing and managing structured data.

Key Features:

MySQL is a database management system.

A database is a structured collection of data. It may be anything from a simple shopping list to a picture gallery or the vast amounts of information in a corporate network. To add, access, and process data stored in a computer database, you need a database management system such as MySQL Server. Since computers are very good at handling large amounts of data, database management systems play a central role in computing, as standalone utilities, or as parts of other applications.

MySQL databases are relational.

MySQL is a relational database that organizes data into tables with rows and columns. It supports SQL (Structured Query Language) for querying and manipulating data. A relational database stores data in separate tables rather than putting all the data in one big storeroom. The database structures are organized into physical files optimized for speed. The logical model, with objects such as databases, tables, views, rows, and columns, offers a flexible programming environment.

MySQL software is Open Source.

Open Source means that it is possible for anyone to use and modify the software. Anybody can download the MySQL software from the Internet and use it without paying anything.

The MySQL Database Server is very fast, reliable, scalable, and easy to use.

MySQL is highly scalable and can handle large volumes of data and high concurrent loads. It supports replication and clustering for horizontal scalability. MySQL Server was originally developed to handle large databases much faster than existing solutions and has been successfully used in highly demanding production environments for several years. Although under constant development, MySQL Server today offers a rich and useful set of functions. Its connectivity, speed, and security make MySQL Server highly suited for accessing databases on the Internet.

Security: MySQL provides robust security features, including encryption, access control, and user authentication. It also supports SSL/TLS for secure communication between clients and servers.

MySQL Server works in client/server or embedded systems.

The MySQL Database Software is a client/server system that consists of a multithreaded SQL server that supports different back ends, several different client programs and libraries, administrative tools, and a wide range of application programming interfaces (APIs).

1. Hardware and Software Requirements (Minimum):

Hardware:

1. Intel i3 processor 3rd generation or later / AMD Ryzen 200 2nd generation or later
2. 2 GB ddr3 ram.
3. Windows 7 Home edition or later.
4. 200 GB Sata HDD Space
5. Data Connection 200 kbps
6. WebCam and Microphone

Software:

1. Spring tool suite / IntelliJ IDEA /VS Code
2. MySQL 5.7 with Workbench 8.0
3. Google Chrome version 79.0
4. Maven Dependencies
5. Postman Api platform

2. ER Diagram:

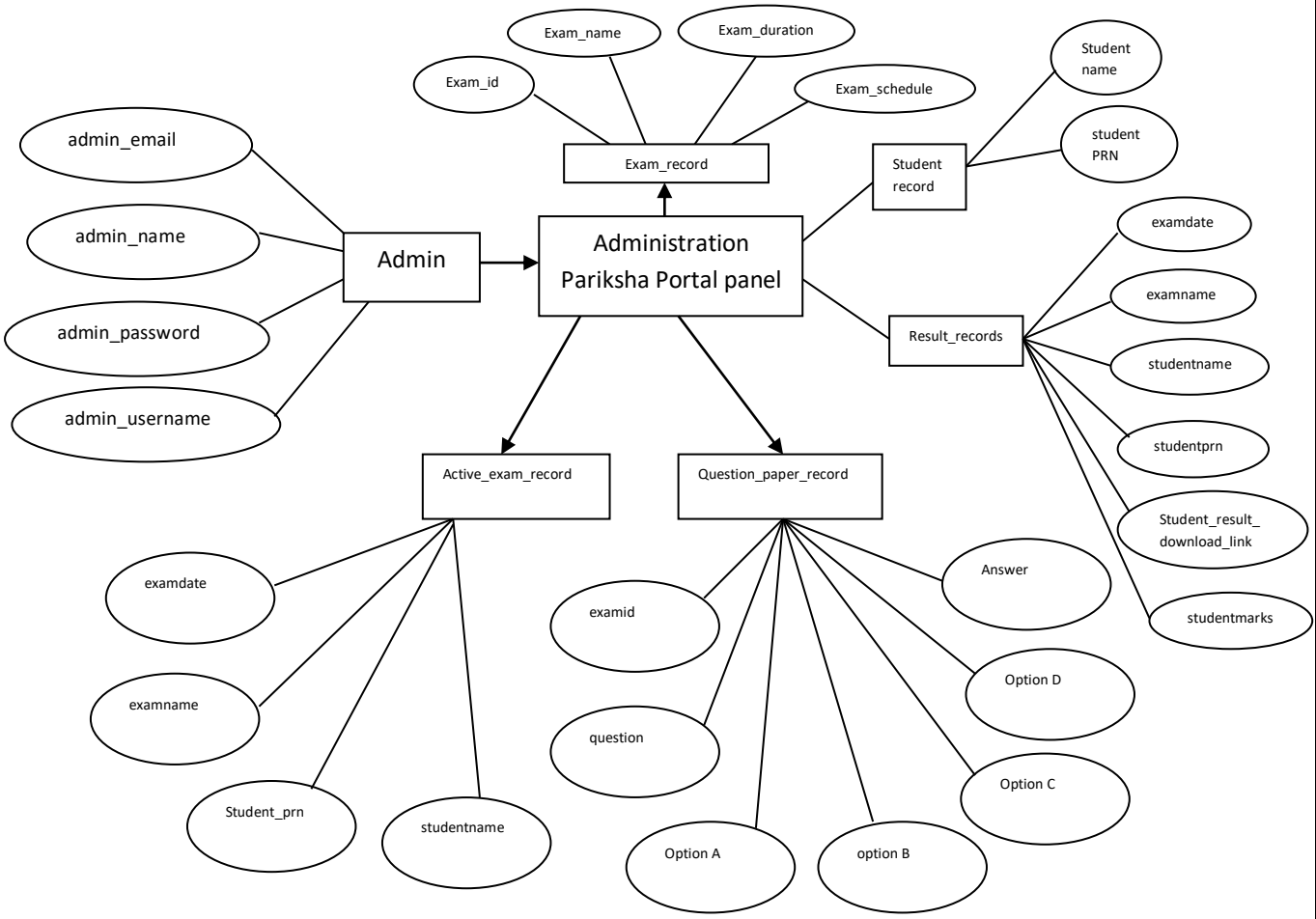

Figure 1: ER Diagram

3. Table Structures:

1. **Table name: active_exam_records**

Field	Type	Null	Key	Default	Extra
id	bigint	NO	PRI		auto_increment
exam_date	varchar(255)	YES			
exam_name	varchar(255)	YES			
student_name	varchar(255)	YES			
student_prn	varchar(255)	YES			

2. **Table name: adminsrecord**

Field	Type	Null	Key	Default	Extra
id	bigint	NO	PRI		auto_increment
exam_date	varchar(255)	YES			
exam_name	varchar(255)	YES			
student_name	varchar(255)	YES			
student_prn	varchar(255)	YES			

3. **Table name: examsrecord**

Field	Type	Null	Key	Default	Extra
id	bigint	NO	PRI		auto_increment
exam_date	varchar(255)	YES			
exam_duration	int	NO			
exam_name	varchar(255)	YES			
exam_id	varchar(255)	YES			

4. **Table name: question_paper_records**

Field	Type	Null	Key	Default	Extra
id	bigint	NO	PRI		auto_increment
question_paper_id	varchar(255)	YES			
question_paper_name	varchar(255)	YES			
answer	varchar(255)	YES			
exam_id	varchar(255)	YES			
optiona	varchar(255)	YES			
optionb	varchar(255)	YES			
optionc	varchar(255)	YES			
optiond	varchar(255)	YES			
question	varchar(255)	YES			

5. **Table name: result_records**

Field	Type	Null	Key	Default	Extra
id	bigint	NO	PRI		auto_increment
exam_date	varchar(255)	YES			
exam_name	varchar(255)	YES			
student_name	varchar(255)	YES			
student_prn	varchar(255)	YES			
student_marks	int	NO			
student_result_download_link	varchar(255)	YES			

6. **Table name: studentsrecord**

Field	Type	Null	Key	Default	Extra
id	bigint	NO	PRI		auto_increment
student_name	varchar(255)	YES			
student_prn	varchar(255)	YES			

4. UML Diagrams:

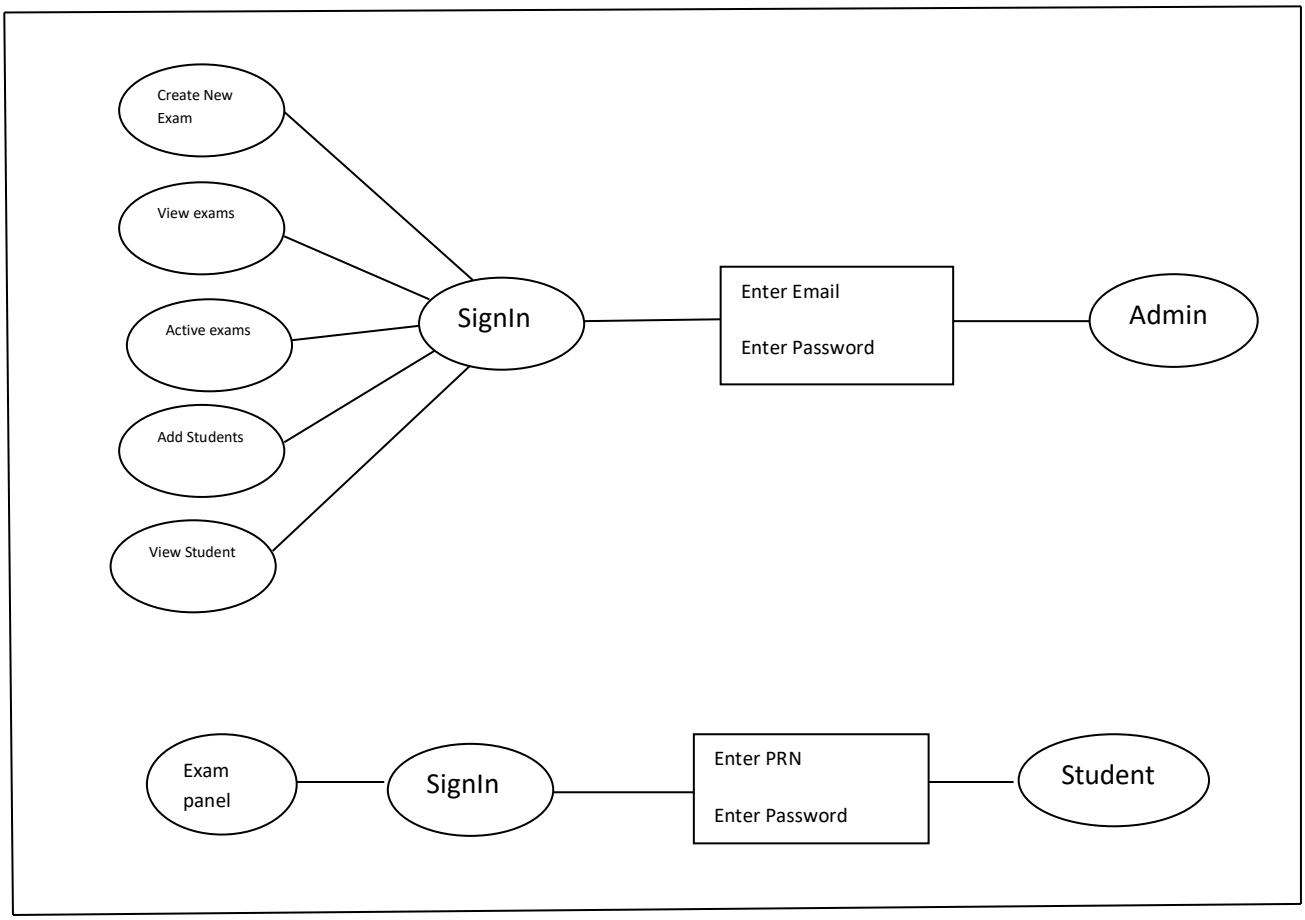

Figure 1: Use Case Diagram

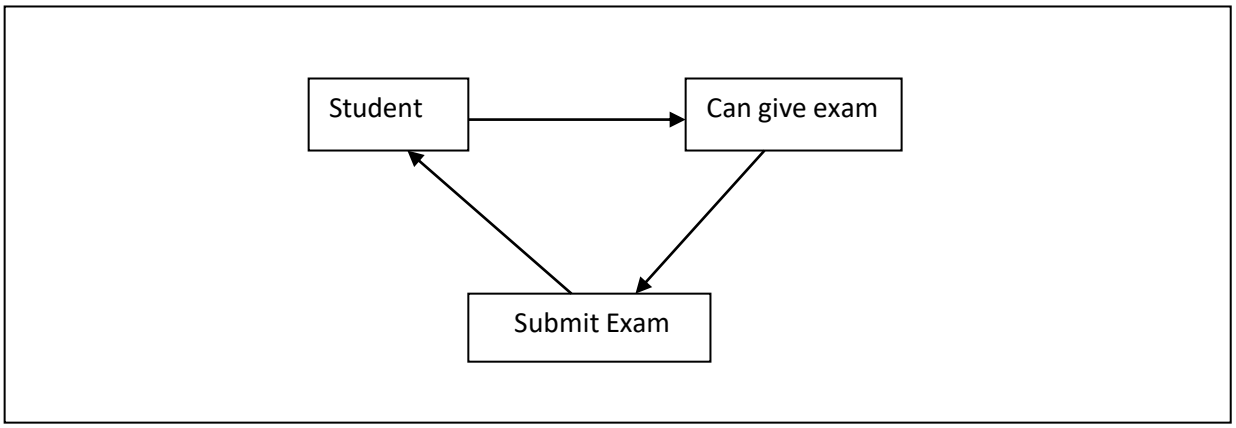

Figure 2: Collaboration Diagram

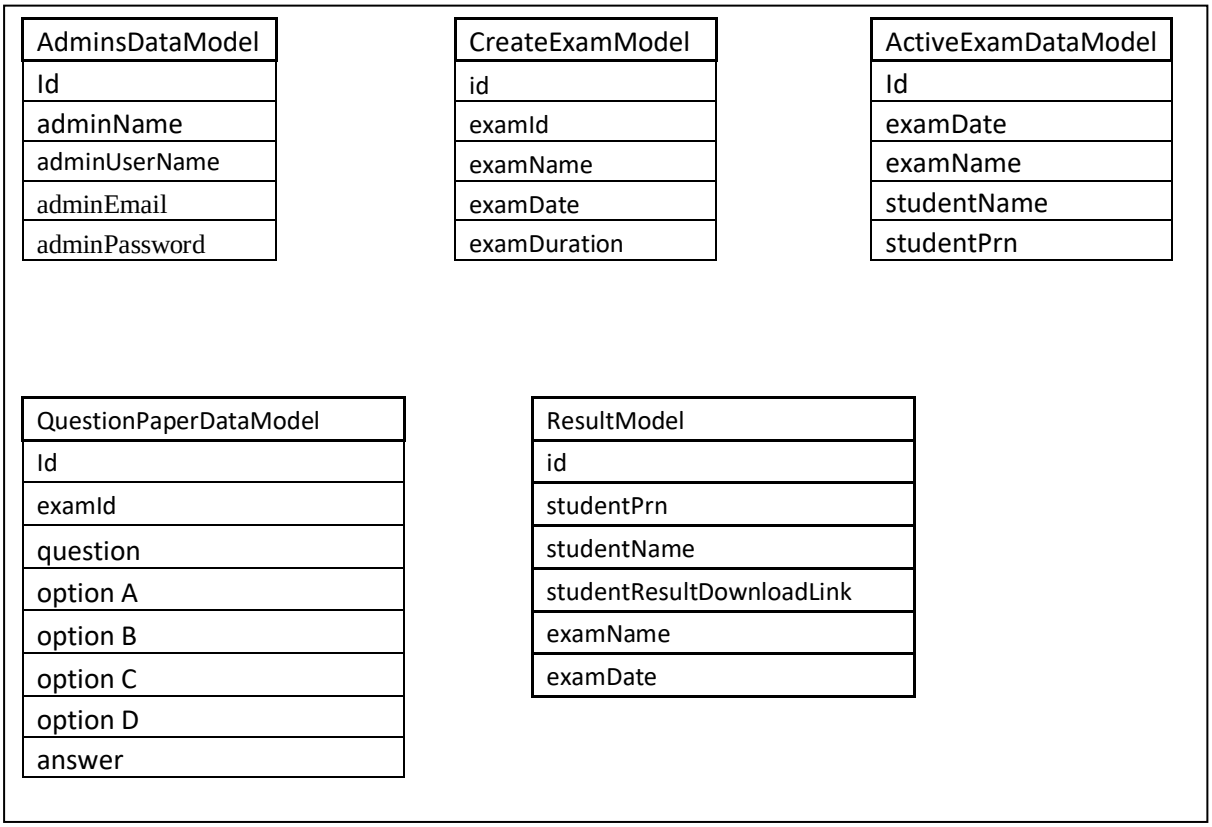

Figure 3: Entity Diagram[model]

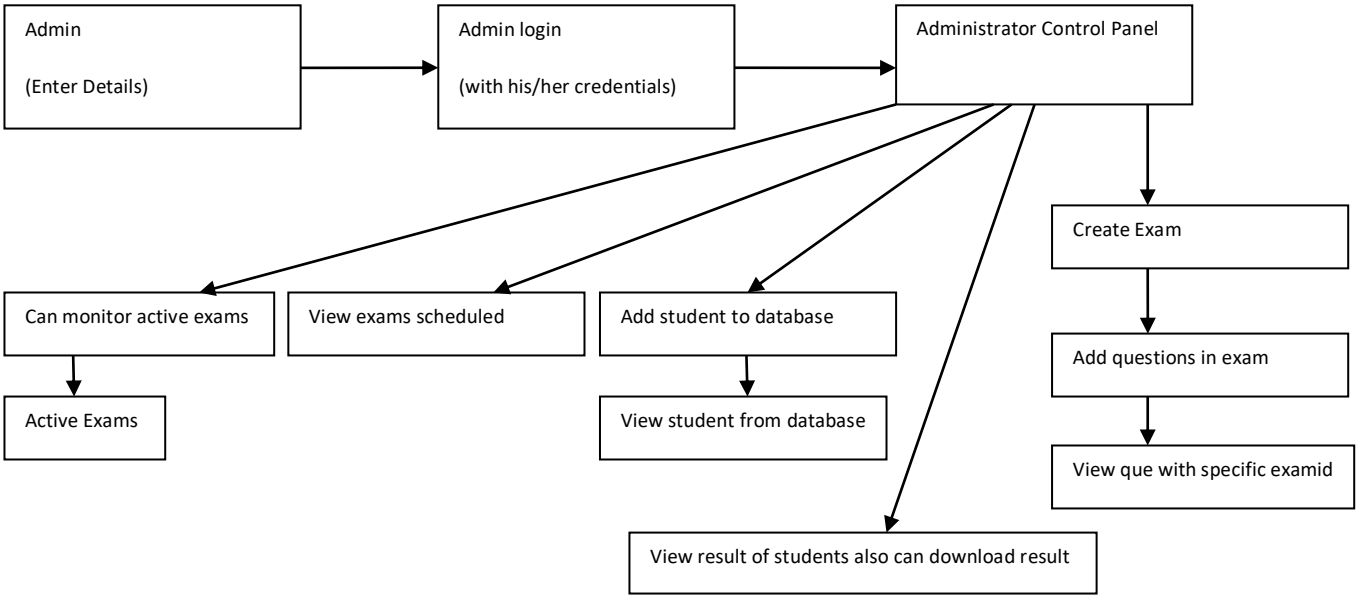

Figure 3: Admin Sequence Diagram

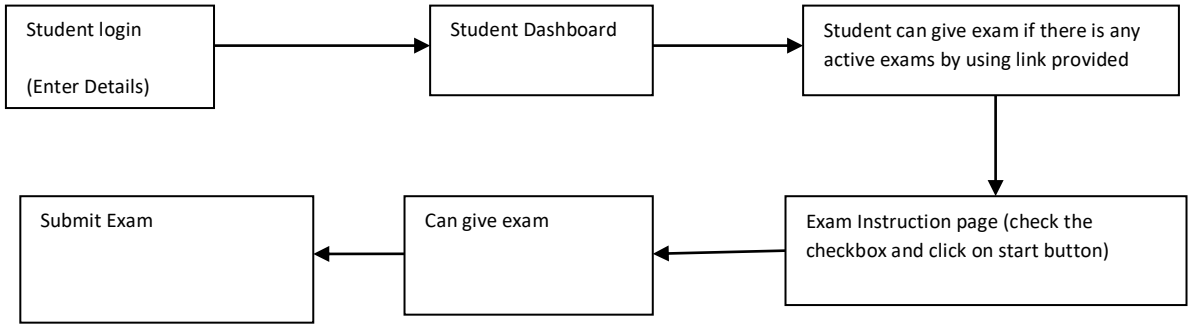

Figure 4: Student Sequence Diagram

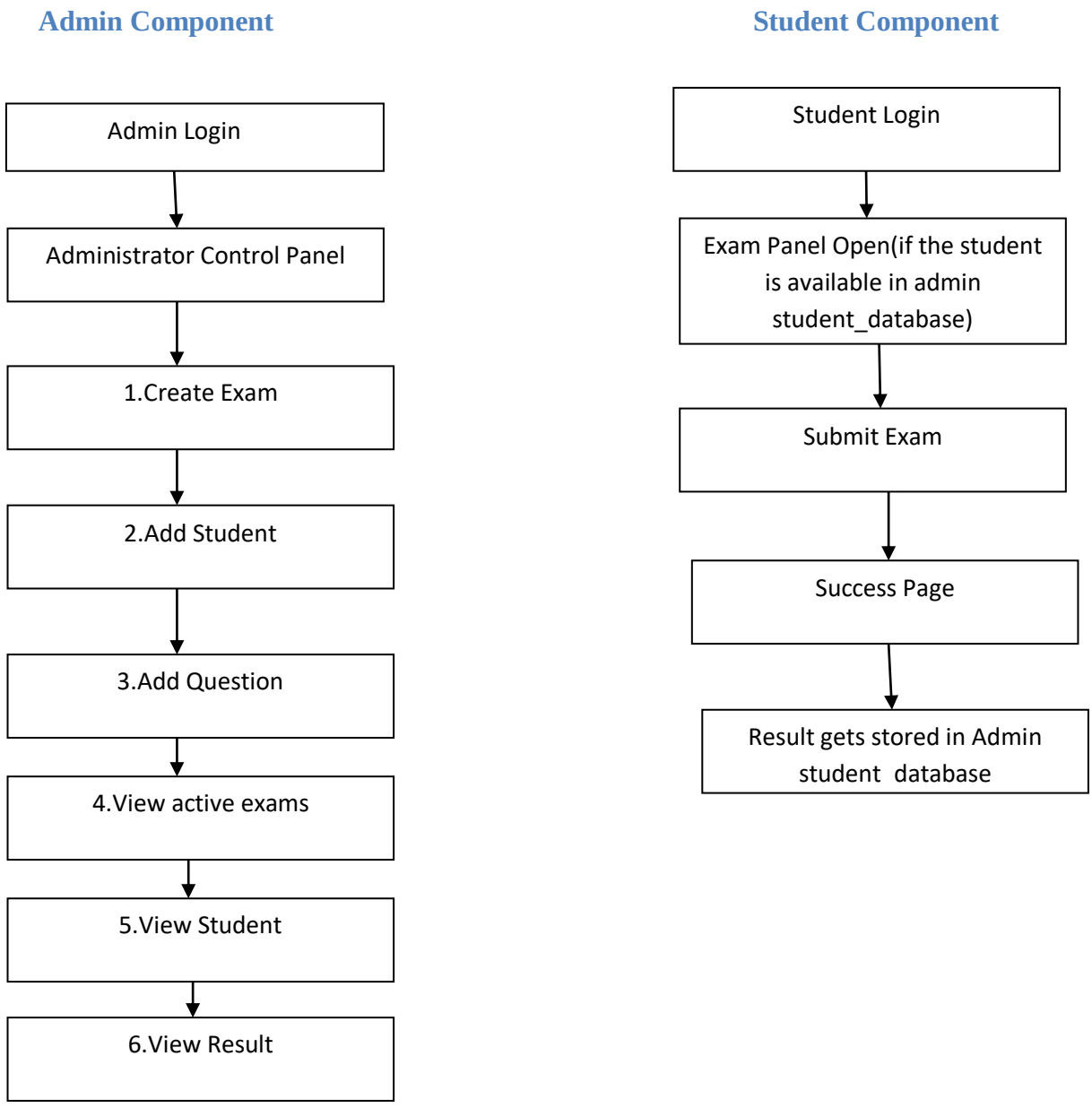

Figure 5: Component Diagram

5. End to End Flow of Application:

Admin:

i. On Website Homepage **Admin** will **login** to the Pariksha portal or will have to signup if he/she is not a registered user by clicking on the help button.

ii. After successfully login Admin will login to the Admin dashboard page. **On admin dashboard** page all the functionalities are provided which Admin can do for instance **create new exam**, **view exams**, **active exam** , **add students**, **view students**, **view results**, etc.

iii. When Admin clicks on **create new exam** it will redirect to next component where he will be able to create exam, admin can choose exam question paper file at the time of creating the exam after clicking on create exam, exam will be generated .

iv. When admin clicks on **view exams** it will redirect to next component where he will able to view exam scheduled.

v. When admin clicks on **Active exams**, the admin will be redirected to the next component where he will able to view active exams.

vi. When admin clicks on **Add students** he/she will be redirect to next component where he will able to add specific students manually or choose a excel file.

vii. By clicking on **Add student** button student will be created and by clicking on Add Students List File all the students in the list will be added.

viii. When Admin clicks on **View Students, he** will be redirect to next component where he will able to see all the students present on the pariksha portal . Admin can search students by entering student PRN and can delete students manually and can delete all students at once.

ix. When Admin clicks on **View Results** he will be redirect to next component where Admin will able to see all the students Results present . Admin can search students result by entering student PRN and can delete students result manually and can delete all students result at once.

Student:

i. Students will get the exam link and will enter the link on compatible browser .Link will redirect towards **Student Dashboard** After entering **PRN** and **Password** ,click on login button and successfully login of student .[Note-Entered PRN Student should be present in the Students List added by the admin]

ii. After Login if exam is started then student will be able to redirect to the exam otherwise student will have to wait till the exam starts.
iii. After exam starts, students have to click on enter full screen, it will redirect to instruction page. After reading all the instructions students have to check on checkbox and then after clicking on start button exam will start after time up exam will submit or student can submit exam manually by clicking on submit button.
iv. The exam will be monitored by web camera , noise detection monitoring by accessing microphone, tab switch monitoring etc, all these security measures to avoid cheating in exam. If the student is caught performing any malpractice the exam will be automatically submitted due to excessive warnings.

6. Future Scope of Project

Enhanced User Experience:

Continuously improving the user interface and experience to make the platform more intuitive and user-friendly.

Implementing personalized dashboards for students, teachers, and administrators to cater to their specific needs and preferences.

Mobile Accessibility:

Developing mobile applications for Android and iOS platforms to provide access to the exam portal on smartphones and tablets, ensuring accessibility anytime, anywhere.

Data Analytics and Insights:

Employing advanced data analytics techniques to extract valuable insights from exam data, such as identifying trends, strengths, and weaknesses in student performance.

Generating comprehensive reports and analytics dashboards for administrators and students to track progress and make data-driven decisions.

Security and Integrity:

Implementing robust security measures to ensure the integrity of the exam process, including features such as secure encryption protocols, and anti-cheating mechanisms.

Continuous Updates and Maintenance:

Regularly updating the exam portal with new features, security patches, and performance enhancements to keep pace with technological advancements and user requirements.

Providing responsive customer support and soliciting feedback from users to identify areas for improvement and address any issues promptly.

Scalability and Flexibility:

Designing the exam portal with scalability in mind to accommodate growing user bases and increasing demand for online assessments.

Overall, the future scope of an exam portal application revolves around leveraging emerging technologies, enhancing user experience, ensuring security and integrity, and fostering engagement and inclusivity in the assessment process. By continuously innovating and adapting to evolving educational needs and technological trends, exam portals can play a pivotal role in transforming the way assessments are conducted and learning outcomes are measured.

Thank You!