
Software Requirements Specification

for

Examsphere

Version 1.0

Prepared by Project Team 29

Name	PRN
Sandesh Pawar	240340520075
Krunal Hatwar	240340320052
Bhavesh Amre	240340320027
Rutvik Tale	240340320084
Amit Mandakhailkar.	240340320010

Centre for Development of Advanced Computing

Raintree Marg, Near Bharati Vidyapeeth, Opp. Kharghar Railway Station, Sector 7, CBD
Belapur, Navi Mumbai - 400 614 - Maharashtra (India)

Phone: +91-22-27565303

Fax: +91-22-2756-0004

Table of Contents

Table of Contents	i
Revision History	ii
1. Introduction	3
1.1 Purpose.....	3
1.2 Intended Audience and Reading Suggestions	3
1.3 Project Scope	3
1.4 References.....	4
2. Overall Description	4
2.1 Product Perspective.....	4
2.2 Product Features.....	4
2.3 User Classes and Characteristics	5
2.4 Operating Environment.....	5
2.5 Design and Implementation Constraints.....	6
2.6 User Documentation.....	6
3. System Features.....	7
3.1 UserFeatures.....	7
3.2 Admin Features	
3.3 Registration module	8
3.4 Login module.....	8
3.5 Category module.....	9
3.6 Quiz module	9
3.7 Exam module	10
3.8 Result module	
4. External Interface Requirements.....	10
4.1 User Interfaces	10
4.2 Hardware Interfaces	10
4.3 Software Interfaces.....	10
4.4 Communications Interfaces.....	11
5. Other Nonfunctional Requirements	11
5.1 Usability.....	11
5.2 Integrity.....	11
5.3 Reliability	
5.4 Performance	
5.5 Security Requirements	11

Appendix A: Glossary	12
Appendix B: Analysis Models	13

Revision History

Name	Date	Reason For Changes	Version

1. Introduction

1.1 Purpose

The purpose of this document is to provide a comprehensive description of online exam portal, an exam Submission & Assessment System. This document outlines the functional and non-functional requirements for the system, provides detailed description of the system's functionalities, such as the submission and review process, result generation. The SRS document also outlines the performance and security requirements for the system.

1.2 Intended Audience and Reading Suggestions

The document is intended for a broad audience, students, and educational institutes.

For developers, the SRS document serves as a comprehensive guide for the design and development of the system. It outlines the functional and non-functional requirements, performance expectations, and security considerations, providing a clear understanding of the scope of the project

For students, the SRS document provides an overview of the system's capabilities helping them to understand how to use the system effectively. They should read the document to gain a general understanding of the system's functionalities and how they can benefit from it.

For educational institutes, the SRS document outlines the goals and objectives of the project and provides a clear understanding of the system's capabilities, helping them to determine whether the system is suitable for their needs. The document should be read by relevant decision-makers to ensure that the system meets their requirements and expectations.

1.3 Project Scope

The scope of the Examsphere project defines the boundaries of the project, outlining what the system will and will not do. The scope of the project includes the following:

1. Platform Development:

	<ul style="list-style-type: none"> • Develop a web-based platform for conducting online examinations. • Provide functionality for students to access and take exams remotely.
2. User Authentication and Authorization:	
	<ul style="list-style-type: none"> • Implement secure user authentication and authorization mechanisms to ensure authorized access to exams.
3. Exam Creation and Management:	
	<ul style="list-style-type: none"> • Allow instructors to create, schedule, and manage online exams. • Provide options for various question types (multiple choice)
4. Real-time Exam Taking:	
	<ul style="list-style-type: none"> • Enable students to take exams online in real-time. • Implement mechanisms to prevent cheating and unauthorized assistance.
5. Automated Result:	
	<ul style="list-style-type: none"> • Develop an automated Result for objective questions to provide instant results.
6. User-Friendly Interface:	
	<ul style="list-style-type: none"> • Design an intuitive and user-friendly interface for both students and admin to navigate the system.
7. Result Generation:	
	<ul style="list-style-type: none"> • Generate immediate exam results • Provide students access to their exam scores.

1.4 References

2. Overall Description

2.1 Product Perspective

The Examsphere project addresses the limitations of traditional exam methods in the digital age. With the rise of online learning and the need for efficient, secure assessments, a modern solution is crucial.

This system aims to provide a centralized platform for conducting exams online, allowing students to take tests remotely. It empowers instructors to create and manage exams while enhancing the assessment experience.

By embracing technology, the project seeks to streamline exams and adapt to the evolving education landscape. Its implementation stands to revolutionize assessment methods by offering a comprehensive, efficient, and modern approach to online examinations.

2.2 Product Features

- User Authentication and Authorization:

Users log in securely, with their roles and permissions determined for proper access.

- Real-time Exam Taking:

- Students take exams remotely within specified time frames, with measures to prevent cheating.

- User-friendly Interface:

- Intuitive interface for both students and instructors, enhancing user experience.

- Result Generation:

- Instant generation of exam results for objective questions, empowering students with feedback..

- Reporting and Analytics:

- Administrators access insights into exam performance and student trends for analysis.

- User history of exams:

User can check all the exams he has given with his score

- Exam Category: admin can create new category type for exams
- Create question paper: admin can create questions, he can update or delete the quiz and add answers for that

2.3 User Classes and Characteristics

The Assignment Submission & Review System has three main user classes:

1. Students:

- Characteristics: They are the primary users of the system and use it to submit quiz for review and track the result. Student can check his history of previous exams given and check his progress. And can download pdf of result generated

2. Admins:

- Characteristics: They are responsible for managing the exams
- Requirements: They should have the ability to create category of exam and quiz and have access to all system data and information .

2.4 Operating Environment

The operating environment for the project consists of the following hardware and software components:

Hardware:

A machine with at least 8GB of RAM and a fast processor, such as Intel Core i5 or higher, to ensure smooth and efficient execution of the project.

For developing .NET applications, you'll need a system with sufficient resources to run an integrated development environment (IDE) like Visual Studio or Visual Studio Code, as well as the necessary .NET SDKs and tools.

Minimum Requirements:

- **Processor:** 1.8 GHz or faster processor (Quad-core or better recommended)
- **RAM:** 4 GB (8 GB or more recommended)
- **Storage:** At least 20 GB of free disk space
- **Graphics:** DirectX 9-capable video card running at 1024 x 768 or higher-resolution display

Recommended Requirements:

- **Processor:** Quad-core processor or better
- **RAM:** 16 GB or more
- **Storage:** SSD with at least 100 GB of free disk space
- **Graphics:** DirectX 11-capable video card running at a resolution of 1920 x 1080 or higher

Additional Considerations:

- **Scalability:** Ensure that the hardware can scale with the application's growth.
- **Redundancy:** For production environments, consider redundancy and failover strategies.
- **Security:** Ensure that the hardware meets security requirements for your application.

Software:

- AngularJS for the frontend development.
- .Net Framework (Entity Framework) for the backend development.
- MySQL for the database management.
- Identity Framework for Security.

Other Applications:

- Spring Suit Tool
- Git version control software
- Command line interface (CLI) or terminal
- A browser for testing the application.
- Postman for testing APIs.
- MySQL Workbench or another database management tool

2.5 Design and Implementation Constraints

- User interface is only in English. No other language option is available.
- Compatibility issues while mapping Frontend to two backends.
- Parallel Operations may load the system and not function smoothly
- Limited to HTTPS.

2.6 User Documentation

User documentation mainly comprises of Help menu of application. It will give all the minute details about the project, if any user has any query about any module or functionality, one can refer it and see how to operate the application. This report is the complete documentation of our project. It gives complete details about the project, its functionality, users, software used, hardware requirement, environment and so on.

3. System Features

The features of all 2 User Modules have been described below.

ADMIN

1. Register themselves.
2. Login themselves.
3. Add/Delete/Update Categories.
4. Add/Delete/Update Quizzes.
5. Add/Delete/Update Questions.
6. Able to view results.

USER

1. Register themselves.
2. Login themselves.
3. See the available Categories.
4. See and attempt the available Quizzes.
5. See and attempt to solve the Questions.
6. Able to view results.

Modules, architecture and various elements that are combined to form the whole system's framework:

- **Registration Module:** It will allow new user to get register on the website. User needs to provide his/her full name, email id, contact number, & password while registering.
- **Login Module:** In this module user can login to the website by entering login ID and password. After Login the system links to Profile page.
- **Category Module:** It allows the admins to add, update, or delete category. Normal user (students) can see the quizzes based on their categories.

- **Quiz Module:** It allows the admins to add, update, or delete quiz. Each quiz belongs to one category. This allows to filter the quiz by category. Normal user (students) can see the quizzes and if they desire, they can attempt the quiz.
- **Exam Module:** This is the main module which handles the logic used when a student starts the quiz. Timer will run for each quiz, & once it finishes, quiz will auto-submit. The quiz can also be submitted by the student beforehand, by clicking on the Submit button. The exam module will pass the data entered by student to result module.
- **Result Module:** This module produces the result for each attempted quiz. The attempted question by students is matched against the correct answer saved in database. And based on that the result is generated.

4. External Interface Requirements

4.1 User Interfaces

The new system shall provide a very intuitive and simple interface to the user and the administrator, so that the user can easily navigate through pages, start discussion threads, upload quizzes, generate reports and the administrator can easily manage user details.

4.2 Hardware Interfaces

Server side

- The web application will be hosted on a web server which is listening on the web standard port, port 8000.

Client side

- Monitor screen – the software shall display information to the user via monitor screen
- Mouse – the software shall interact with the movement of the mouse and the mouse buttons. The mouse shall activate areas for data input, command buttons and select options from menus.
- Keyboard – the software shall interact with the keystrokes of the keyboard. The keyboard will input data into the active area of the database.

4.3 Software Interfaces

Server side

An Apache web server will accept all requests from the client and forward it accordingly. A database will be hosted centrally using MySQL

Client side

An OS which is capable of running a modern web browser which supports JavaScript and HTML5

4.4 Communications Interfaces

Communication interfaces refer to the various methods and protocols used for exchanging data between systems, devices, software components, or users. These interfaces can be classified based on different criteria, such as the type of communication (e.g., wired, wireless), the protocol used (e.g., HTTP, Bluetooth), and the layer of the network model they operate in (e.g., physical layer, application layer).

In this Project the HTTP or HTTPS protocol(s) will be used to facilitate communication between the client and server.

5. Other Nonfunctional Requirements

5.1 Usability

Usability is a quality attribute used to access how easy the interface is to use. Usability is ease of use. It tells how user friendly the interface is. It includes memorability, learnability, and satisfaction. Our software interface has all the above quality. Any kind of user can easily understand the interface.

5.2 Reliability

Reliability is how much the system is consistent in different platforms. The ability of an apparatus, system to consistently perform its required function, on demand and without degradation or failure.

5.3 Integrity

Integrity means doing the right thing in a reliable way. Data integrity is a fundamental component of security. In its broadcast use, "Data Integrity" refers to the accuracy and consistency of data stored in a database, data mart or another construct. Data integrity is the overall completeness, accuracy, and consistency of data.

5.4 PERFORMANCE

Performance is also a major non-functional requirement. Performance Requirements about resources required, response time, transaction rate or anything else having to do with performance.

5.5 Security Requirements

The project would include measures to protect the confidentiality, integrity, and availability of sensitive information such as students' assignments, grades, and personal information. Some of the security requirements are:

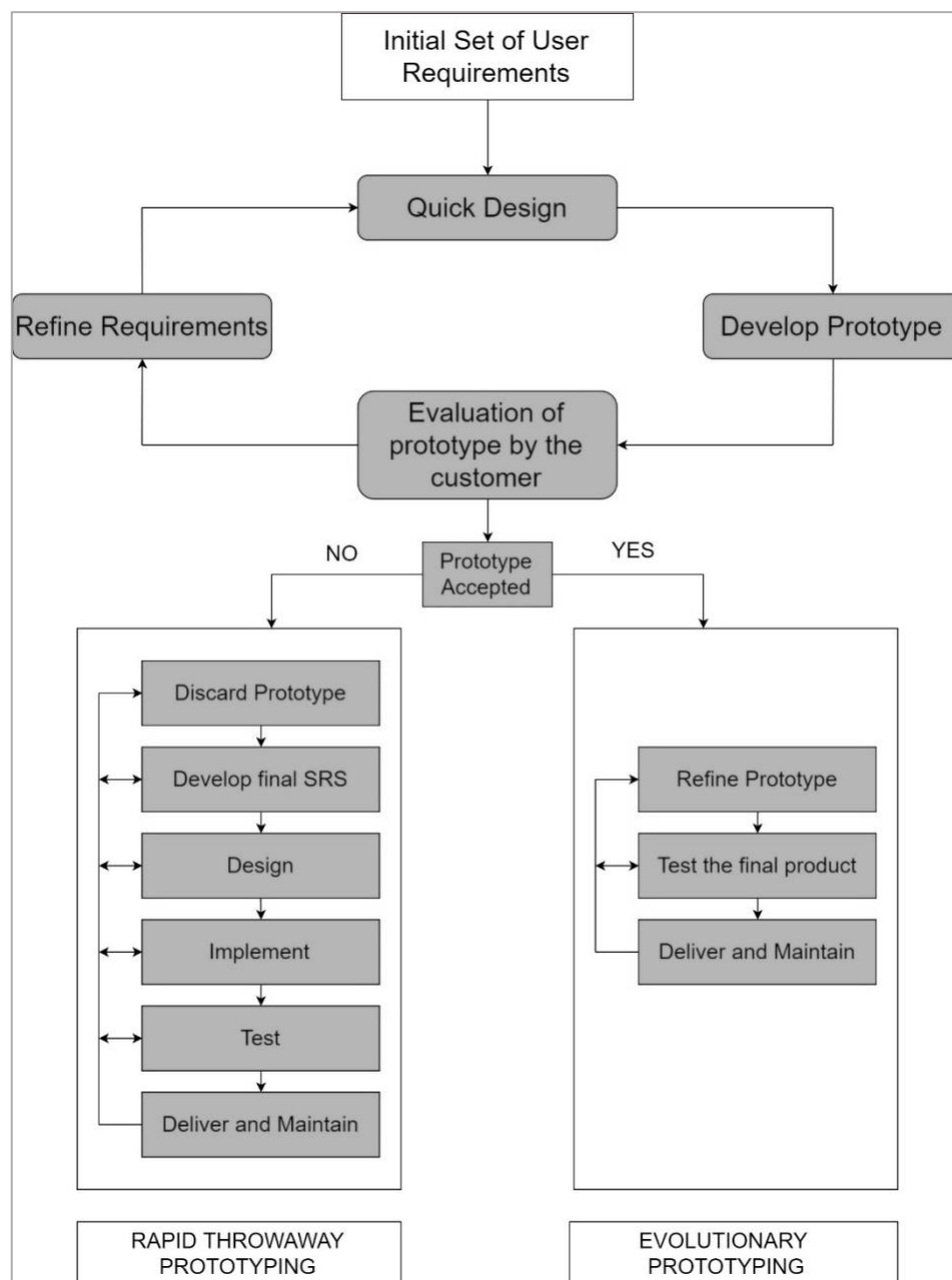
- **Authentication:** A secure authentication system that ensures only authorized users can access the system.
- **Authorization:** An authorization mechanism to determine what actions a user can perform within the system based on their role and permissions.
- **Encryption:** Data in transit and at rest should be encrypted to protect sensitive information from being intercepted or accessed by unauthorized parties.
- **Access Control:** A mechanism to control access to sensitive information within the system, including the ability to set permissions for different users and roles.

Appendix A: Glossary

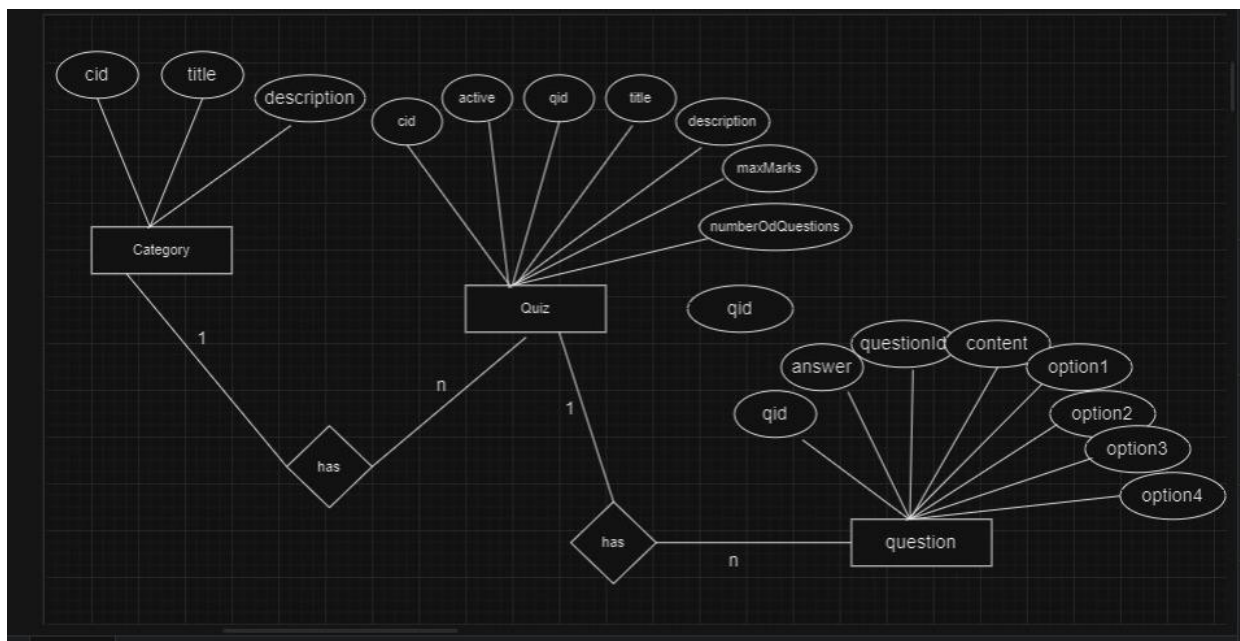
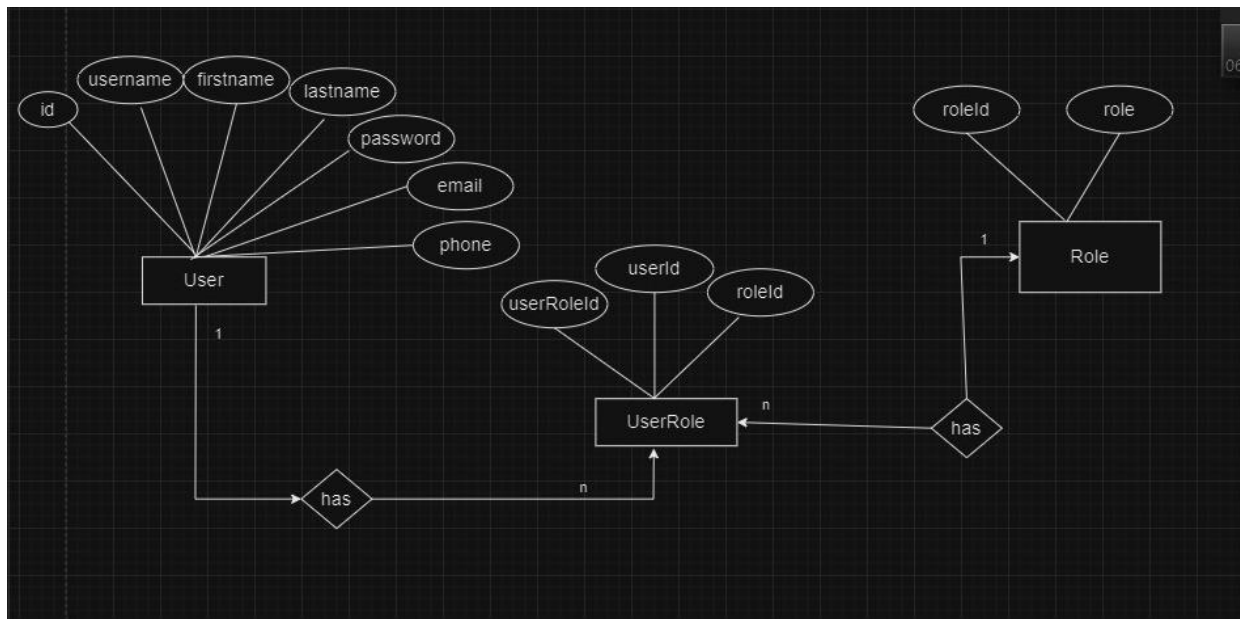
Sr. No.	Abbreviation	Full Form
1.	API	Application Programming Interface
2.	AWS	Amazon Web Service
3.	CLI	Command line Argument
4.	GB	Gigabyte
5.	HTML	Hypertext Markup Language
6.	HTTP / HTTPS	Hypertext Transfer Protocol / Hypertext Transfer Protocol Secure
7.	ID	Identification
8.	JS	JavaScript
9.	JWT	Java Web Token
10.	OS	Operating System
11.	RAM	Random Access Memory
12.	SQL	Structured Query Language
13.	SRS	Software Requirement Specification
14.	URL	Uniform Resource Locator
15.	ER	Entity Relationship

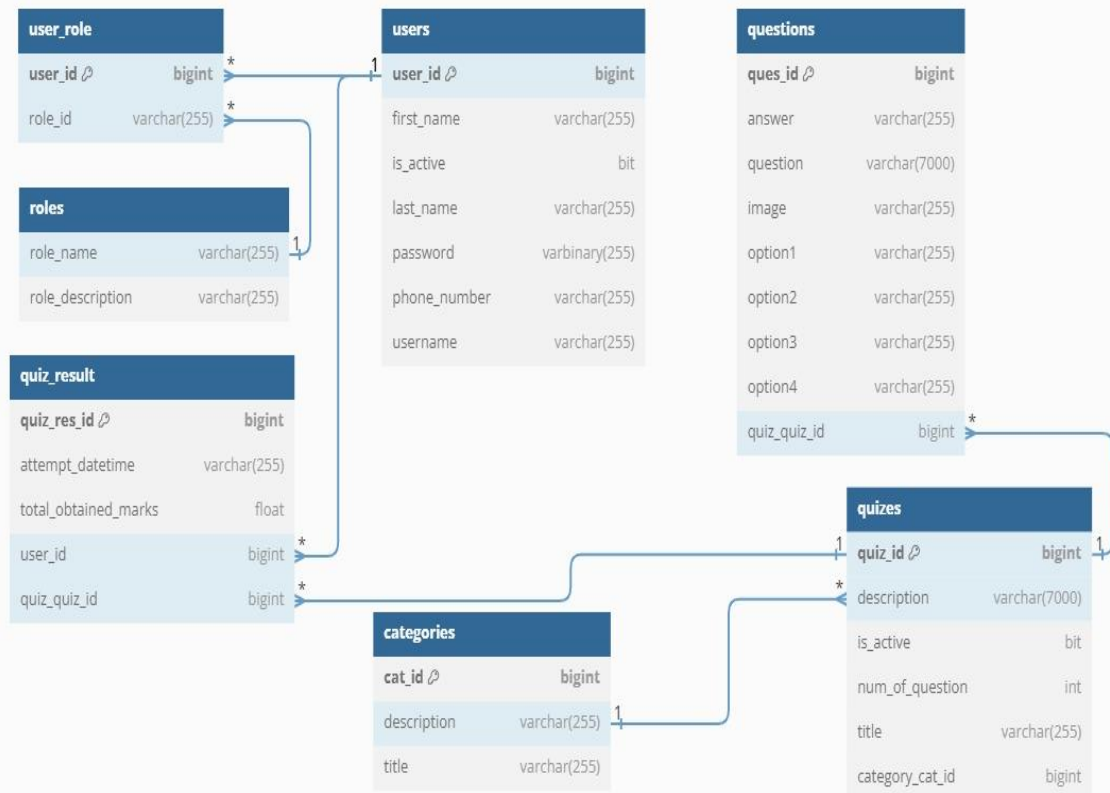
Appendix B: Analysis Models

- **Software Development Approach in Our System**

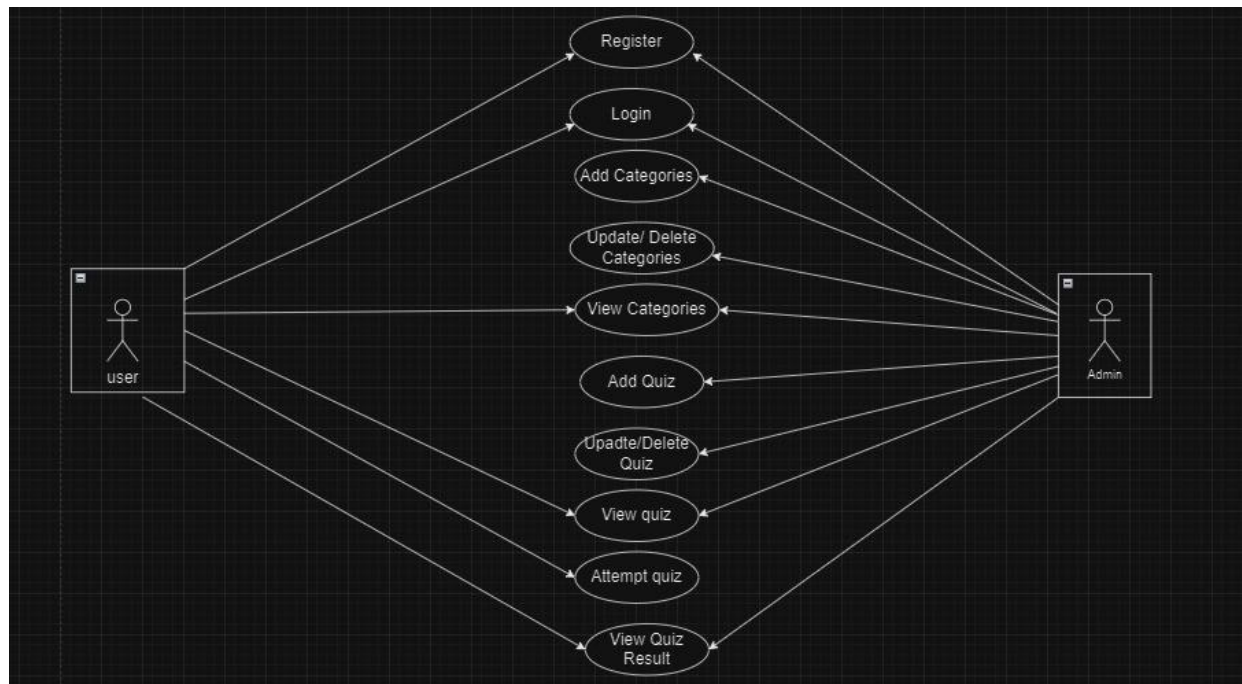


- **ER Diagram**





- **Use-Case Diagram**



- **Flow chart Diagram**

