System Analysis & Design

1. Problem Statement & Objectives

Problem Statement:

Traditional learning methods often lack accessibility, flexibility, and engagement. Students and educators need a digital learning platform that enables seamless course management, interactive learning materials, and efficient assessment tools.

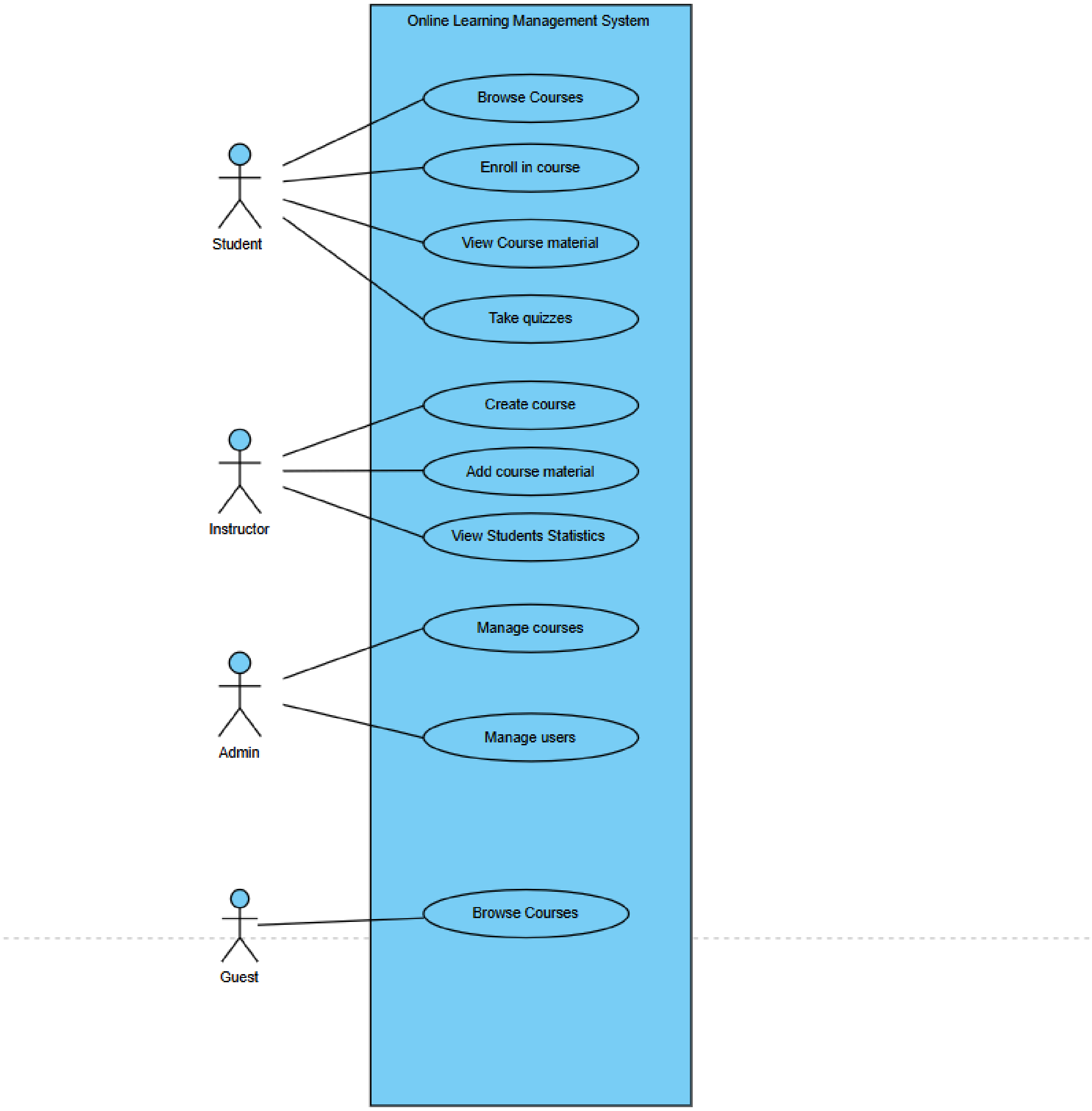
Objectives:

- Develop a user-friendly e-learning platform accessible anytime, anywhere.
- Provide a structured course management system for instructors.
- Enable students to enroll in courses and track their progress.
- Integrate interactive learning tools (quizzes, discussion forums, videos, PDFs).
- Support multiple roles: Admin, Instructor, Student.
- Ensure security, scalability, and smooth user experience.

Actors:

1. **Admin** – Manages users, content, and platform settings.
2. **Instructor** – Creates and manages courses, uploads materials, and View Students Statistics.
3. **Student** – Enrolls in courses, accesses learning materials, and takes quizzes
4. **Guest** – Browses available courses but cannot enroll without registration.

Case Diagram:



Case Descriptions:

- **Login/Signup:** Users (students, instructors, admins) can register and log in.
- **Manage Courses:** Instructors can create, edit, and delete courses.
- **Enroll in Course:** Students can search and enroll in courses.
- **View Course Content:** Students can access videos, PDFs, and quizzes.
- **Take Assessments:** Students can complete quizzes and assignments.
- **Admin Management:** Admin can monitor platform activities and manage users ,add courses, add jobs, and add blogs.

Functional Requirements:

- User authentication and role-based access control.
- Course creation and management.
- Enrollment and progress tracking.
- Content delivery (videos, PDFs, interactive quizzes).
- Assessment and grading system.
- Notifications and reminders.

Non-Functional Requirements:

- **Performance:** Fast response time and optimized load handling.
- **Accessibility:** Support for multiple devices and assistive technologies.
- **Usability:** Intuitive UI/UX.

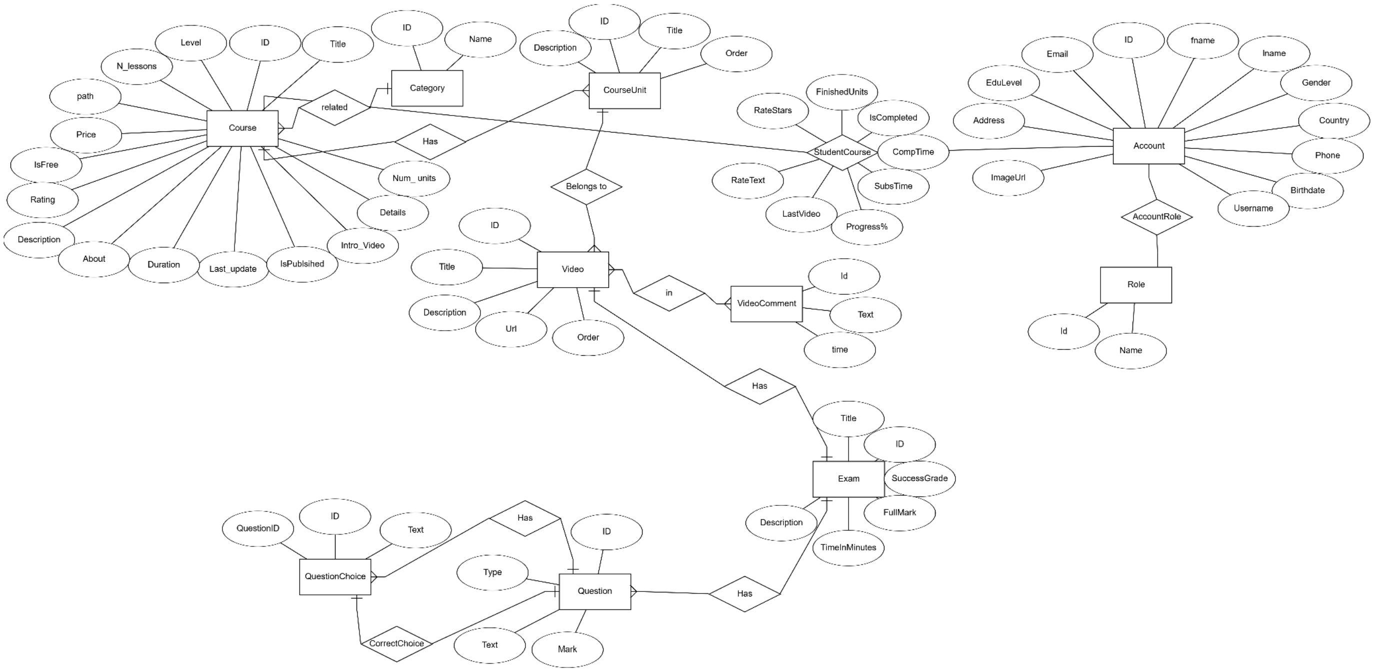
Architecture Style:

- **RESTful API**: A structured API exposing endpoints for handling courses, users, and payments.

System Components:

- **Frontend (React)** – Handles user interaction.
- **Backend (ASP.NET Core)** – Processes requests, business logic.
- **Database (SQL Server)**– Stores user and course data.

2. Database Design & Data Modeling



3- UI/UX Design