| **Section** | **Title** |
| --- | --- |
| **1. Introduction** | 1.1 Purpose |
| 1.2 Scope |
| 1.3 Definitions, Acronyms, and Abbreviations |
| 1.4 References |
| **2. Overall Description** | 2.1 Product Perspective |
| 2.2 Product Functions |
| 2.3 User Characteristics |
| 2.4 Constraints |
| 2.5 Assumptions and Dependencies |
| **3. Specific Requirements** | 3.1 Functional Requirements |
| 3.1.1 User Registration and Authentication |
| 3.1.2 Course Management |
| 3.1.3 Content Delivery |
| 3.1.4 Assessments and Grading |
| 3.1.5 Communication Tools |
| 3.1.6 Progress Tracking |
| **3.2 Non-Functional Requirements** | Performance |
| Security |
| Usability |
| Availability |
| **4. Appendices** | API Documentation |
| UI Wireframes |
| DFD Diagram |

Software Requirement Specification (SRS) Document

**1. Introduction**

**1.1 Purpose**

The purpose of this document is to define the functional and non-functional requirements for the E-learning platform. This platform aims to provide students, instructors, and administrators with a seamless digital learning experience, including course management, assessments, and progress tracking.

**1.2 Scope**

The E-learning platform will provide a comprehensive digital environment that enables students, instructors, and administrators to interact seamlessly. The platform’s core functionalities include course creation, online assessments, student progress tracking, live classes, interactive forums, and a payment system for monetization.

**Course Creation and Management**

1. **Instructor Dashboard**: Instructors can create, update, and manage courses from a dedicated dashboard.
2. **Course Structure**: Courses will be structured into modules and lessons, with support for video lectures, PDFs, interactive quizzes, and assignments.
3. **Multimedia Support**: Courses can include videos, audio clips, text documents, and presentations for a richer learning experience.
4. **Drip Content**: Instructors can schedule lessons to be released gradually to maintain engagement.
5. **Course Approval**: Admins can review and approve courses before they are published to ensure quality standards.

**Online Assessments and Grading**

1. **Quizzes and Exams**: Instructors can create multiple-choice questions (MCQs), fill-in-the-blanks, short answers, and coding assessments.
2. **Automated Grading**: The system can auto-grade MCQs and other objective-type questions.
3. **Manual Grading**: Instructors can review and grade subjective answers and assignments.
4. **Timed Assessments**: Support for timed quizzes and exams to simulate real test environments.
5. **Retake and Feedback**: Students can retake quizzes (if allowed) and receive detailed feedback on their answers.

**Student Progress Tracking**

1. **Course Dashboard**: Students can track their enrolled courses, completed modules, and pending lessons.
2. **Performance Metrics**: Displays quiz scores, assignment results, and engagement analytics.
3. **Completion Certificates**: Upon successful course completion, students receive digitally verifiable certificates.
4. **Instructor Analytics**: Instructors can analyze student engagement, course completion rates, and quiz performance to refine teaching strategies.

**Live Classes and Interactive Forums**

1. **Live Sessions**: Instructors can host live classes via Zoom, Google Meet, or an inbuilt video streaming service.
2. **Real-Time Q&A**: Students can ask questions during live sessions through chat and voice interactions.
3. **Discussion Forums**: Each course includes a dedicated forum where students can post queries and discuss topics with instructors and peers.
4. **Community Engagement**: Students can join interest-based groups, share knowledge, and participate in study discussions.

**Payment Integration for Paid Courses**

1. **Subscription Model**: Users can purchase monthly or annual subscriptions to access premium content.
2. **One-Time Payments**: Students can pay for individual courses.
3. **Revenue Sharing**: The system will handle instructor payments based on revenue-sharing models.
4. **Discounts and Coupons**: Admins and instructors can offer discounts and promotional codes.
5. **Secure Transactions**: Integrated with payment gateways like Stripe, Razorpay, or PayPal for secure transactions.

**1.3 Definitions, Acronyms, and Abbreviations**

1. **LMS (Learning Management System)**: A software application that enables educational institutions and instructors to create, manage, and deliver online courses.
2. **CMS (Content Management System)**: A platform that allows users to create, manage, and modify digital content efficiently.
3. **UI/UX (User Interface/User Experience)**: UI refers to the design and layout of the platform, while UX focuses on the user’s overall experience and interaction with the system.
4. **API (Application Programming Interface)**: A set of protocols that allows different software systems to communicate and share data

**1.4 References**

1. Software Engineering principles
2. Standard SRS template
3. Best practices for E-learning platforms

**2. Overall Description**

**2.1 Product Perspective**

The E-learning platform will be a web-based and mobile-friendly application designed to facilitate online education. It will integrate with third-party APIs for payment, video hosting, and live classes.

**2.2 Product Functions**

1. User Registration & Authentication
2. Course Management
3. Content Delivery
4. Assessment & Evaluation
5. Progress Tracking
6. Communication Tools

**2.3 User Characteristics**

1. **Students**: Engage with course materials, take quizzes, and track progress.
2. **Instructors**: Create and manage courses, assign assessments, and interact with students.
3. **Administrators**: Manage platform settings, users, and content approvals.

**2.4 Constraints**

1. The platform must be accessible on both desktop and mobile devices.
2. Adherence to security and data privacy regulations.
3. Scalable to support multiple concurrent users.

**2.5 Assumptions and Dependencies**

1. Requires a stable internet connection for accessing content.
2. Relies on third-party integrations for video streaming and online payments.

**3. Specific Requirements**

**3.1 Functional Requirements**

**3.1.1 User Registration and Authentication**

1. Users can register using email and password or social media logins.
2. Secure login with role-based access.

**3.1.2 Course Management**

1. Instructors can create, update, and delete courses.
2. Courses can include videos, PDFs, quizzes, and assignments.

**3.1.3 Content Delivery**

1. Support for multimedia content (videos, documents, interactive quizzes).
2. Scheduled content release (drip courses).

**3.1.4 Assessments and Grading**

* Multiple types of assessments (MCQs, descriptive answers, file uploads).
* Auto-grading for objective-type questions.

**3.1.5 Communication Tools**

* Discussion forums for courses.
* Direct messaging between students and instructors.

**3.1.6 Progress Tracking**

* Dashboard displaying course completion status.
* Certificates upon course completion.

**3.2 Non-Functional Requirements**

* **Performance**: The system should be designed to handle 1000+ concurrent users efficiently. It should ensure smooth operation under high traffic loads, optimizing database queries, caching, and load balancing to maintain responsiveness.
* **Security**: Data encryption (AES, SSL/TLS) should be implemented to protect user data. Secure authentication methods, such as OAuth and multi-factor authentication, will enhance security.
* **Usability**: The UI should be intuitive and user-friendly, ensuring seamless navigation. Accessibility features such as screen reader compatibility, high contrast mode, and keyboard navigation should be included.
* **Availability**: The platform should ensure 99.9% uptime by utilizing cloud-based infrastructure, redundancy strategies, and automatic failover mechanisms to prevent downtime.

**4.1 API Documentation**

The API documentation provides a detailed overview of the various endpoints that enable communication between the front-end and back-end of the e-learning platform. This includes:

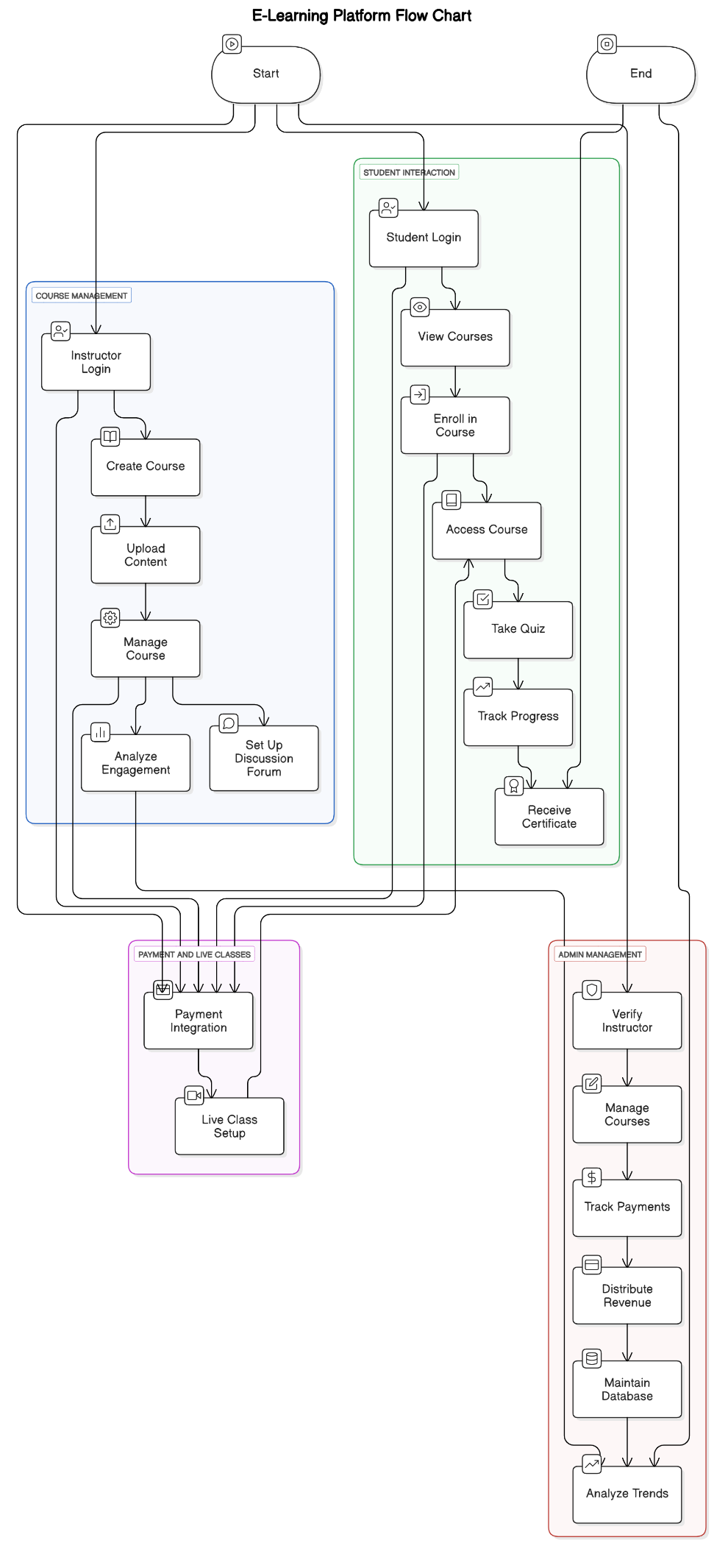
* **Authentication API:** Handles user registration, login, and role-based access.
* **Course Management API**: Allows instructors to create, update, and delete courses.
* **User Management API**: Enables administrators to manage students and instructors.
* **Assessment API**: Supports quiz creation, submission, and grading.
* **Payment API**: Handles transactions for paid courses and instructor revenue distribution.
* **Progress Tracking API**: Fetches student progress, course completion status, and certificate generation.

**4.2 UI Wireframes**

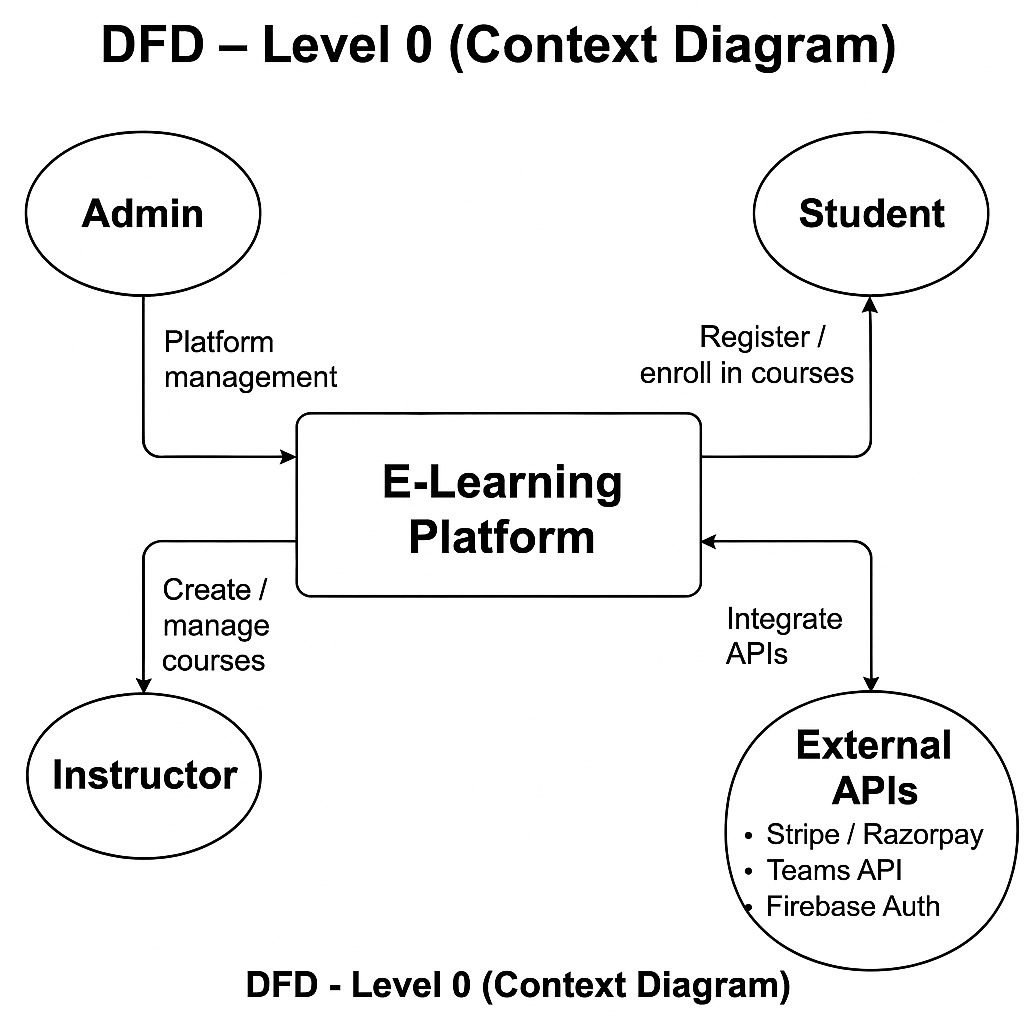
The UI wireframes illustrate the user interface design and navigation flow, including:

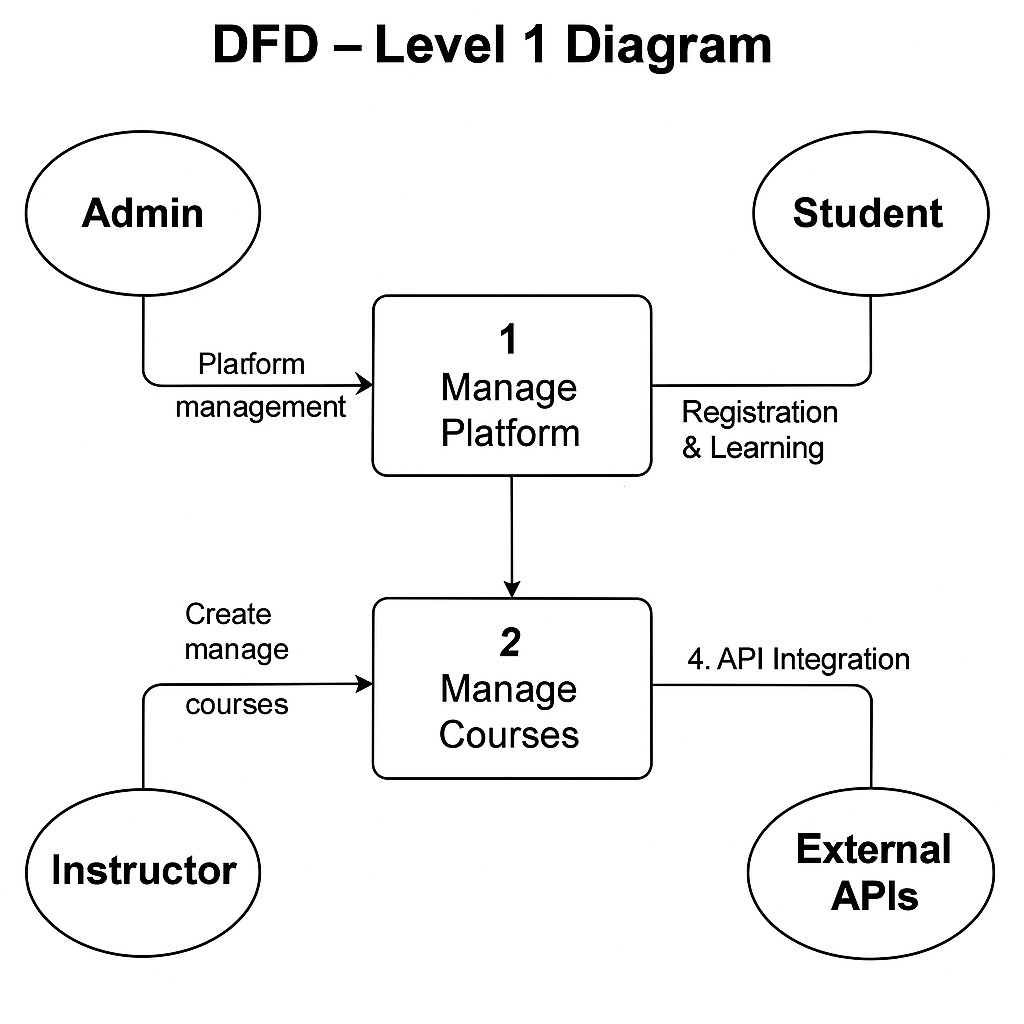
* Home Page: Displays featured courses and categories.
* Student Dashboard: Shows enrolled courses, progress tracking, and quiz results.
* Instructor Dashboard: Provides course management tools, analytics, and earnings reports.
* Admin Panel: Manages user roles, payments, and platform settings.
* Course Detail Page: Contains course content, instructor information, and discussion forums.
* Quiz Interface: Allows students to attempt quizzes and view results.

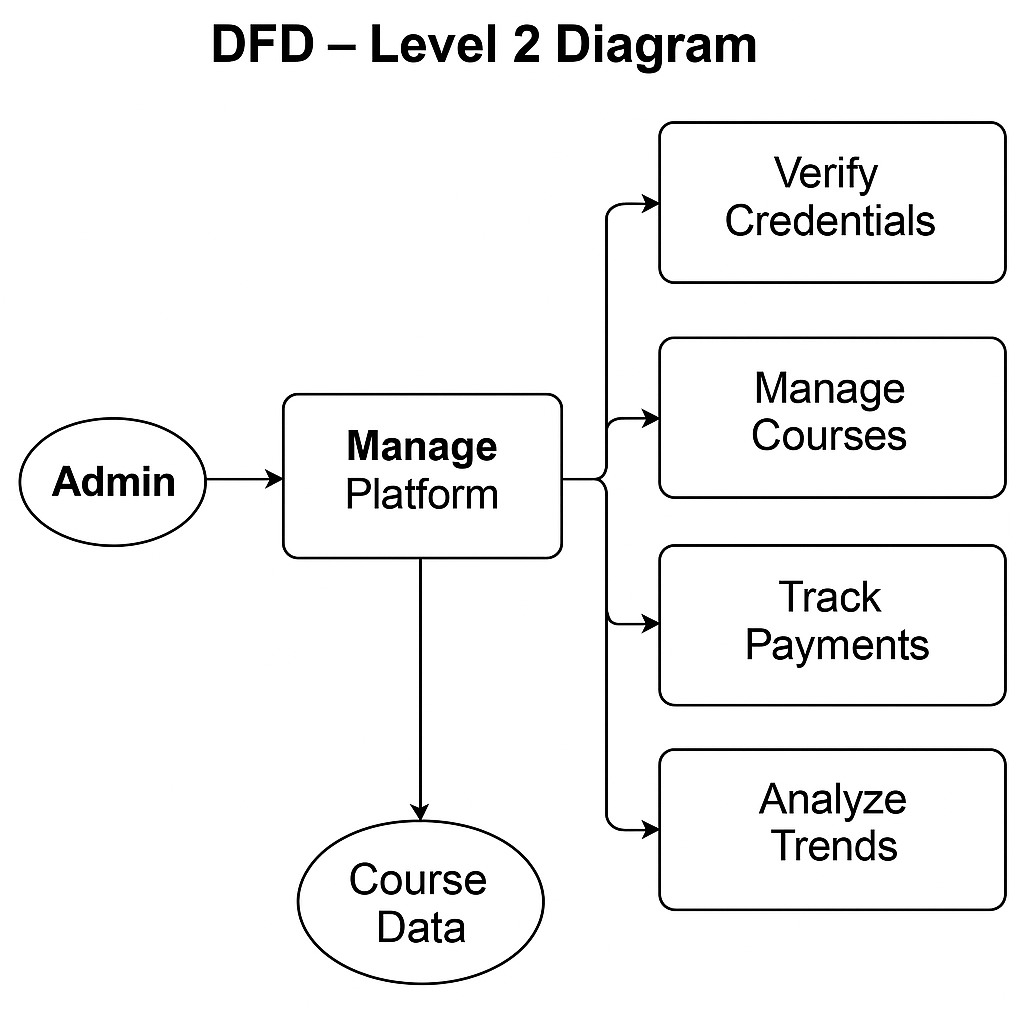
**4.3 FLOW CHART**

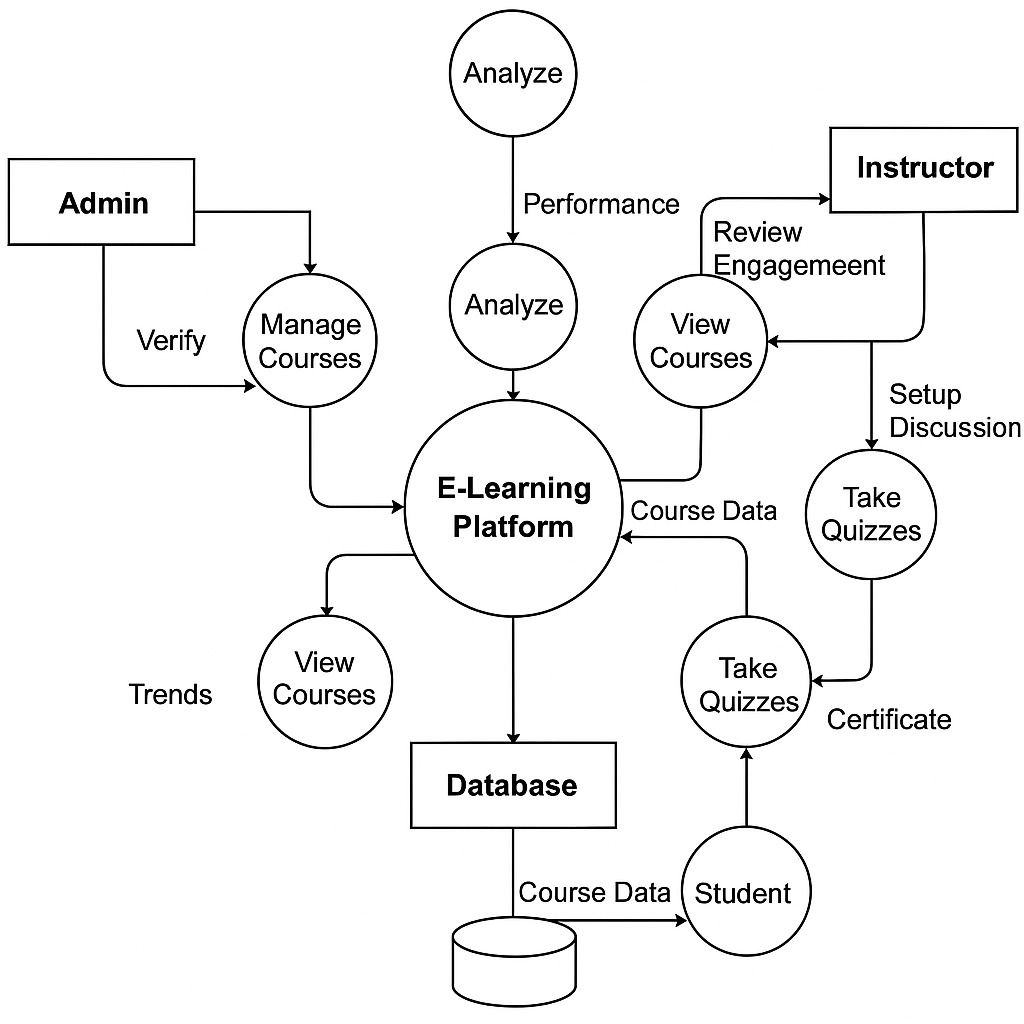
****

**4.3 DATA FLOW DAIGRAM**

****

****

****

****