# Project Report

**Final Project Report**

**Online Learning Platform (OLP)**

**1. Introduction**

An Online Learning Platform (OLP) is a digital solution designed to facilitate education through the internet. It provides interactive tools, flexible schedules, and accessible content to learners worldwide. The OLP project focuses on delivering a real-world experience of an e-learning ecosystem with core features such as user authentication, course enrollment, progress tracking, teacher-student interaction, and course certification.

**2. Objective**

To develop a scalable and user-friendly full-stack application that allows learners to register, browse, and enroll in courses while enabling educators to create, manage, and track their courses. Admins oversee user activities and content integrity.

**3. Technical Architecture**

* **Frontend:** React.js + Vite + Axios
* **Styling:** Bootstrap, Material UI, Ant Design, MDB UI Kit
* **Backend:** Node.js + Express.js
* **Database:** MongoDB
* **APIs:** RESTful API using Express
* **Middleware:** JWT Authentication, CORS, Multer
* **Tools:** Nodemon, dotenv

**4. Project Features**

**User Interface**

* User-friendly and responsive design
* Accessible on all devices (mobile, tablet, desktop)
* Real-time updates using Axios

**Authentication and Roles**

* JWT-based login for Students, Teachers, and Admins
* Secure password storage with bcryptjs

**Course Features**

* Teachers can create, update, and delete courses
* Students can enroll, view content, and earn certificates
* Admins can manage all users and content

**5. Scenario-Based Use Case**

**User: Sarah (Student)**

* Registers and browses the course catalog
* Enrolls in “Web Development Fundamentals”
* Learns at her own pace and earns a certificate
* Pays and enrolls in an advanced course

**User: John (Teacher)**

* Creates and manages web development courses
* Adds sections and monitors enrollment

**Admin**

* Monitors overall activity
* Manages users and course listings

**6. ER Diagram**

**Collections**

* **Users:**
  + \_id, name, email, password, type
* **Courses:**
  + \_id, userID, C\_title, C\_description, C\_categories, C\_educator, sections, C\_price, enrolled

**7. Project Structure**

* **Frontend:** React structure with folders for components, pages, services, and assets
* **Backend:** Express server with folders for routes, controllers, models, middlewares, config

**8. Milestones**

**Milestone 1: Setup**

* Initialize backend and frontend folders
* Install dependencies (npm install)

**Milestone 2: Backend Development**

* Setup Express server and middleware
* Implement authentication
* Create MongoDB schemas

**Milestone 3: Database**

* Connect MongoDB with Mongoose
* Define user and course models

**Milestone 4: Frontend Development**

* Build UI using React and MUI
* Integrate Axios for API calls
* Role-based dashboards (Admin, Teacher, Student)

**Milestone 5: Final Testing**

* Run full-stack app locally using npm start
* Fix bugs and finalize functionality

**9. Pre-requisites & Tools**

* Node.js & npm
* MongoDB
* Vite
* React.js
* Express.js
* Axios, Mongoose
* Bootstrap, MUI, Antd
* JWT, bcryptjs, Multer

**10. Running the App**

**# Backend**

**cd backend**

**npm install**

**npm start**

**# Frontend**

**cd frontend**

**npm install**

**npm run dev**

**11. Conclusion**

**The Online Learning Platform successfully simulates a real-world education system with clear role separation and robust architecture. The project highlights the use of modern web technologies and best practices in full-stack development. It is scalable and can be extended with features like video streaming, quizzes, and AI-based recommendations.**