

# LearnHub – Your Center for Skill Enhancement

## ***Project Documentation:***

### **1. Introduction**

- Project Title: LearnHub – Your Center for Skill Enhancement
- Stack: MERN (MongoDB, Express.js, React.js, Node.js)
- Tech Team Members:
  - [Your Name] – Full Stack Developer (Frontend, Backend, DB Integration, Deployment)
  - [Optional] – UI/UX Specialist / QA Engineer / DevOps, etc.

### **2. Project Overview**

#### **Purpose**

LearnHub is a full-featured Online Learning Platform (OLP) designed to empower users with self-paced, skill-based education. The platform provides a responsive, scalable, and interactive learning environment for students, instructors, and admins.

#### **Key Functionalities**

- User registration and role-based login (Student, Instructor, Admin)
- Dynamic course catalog with category filtering

- Course enrollment and progress tracking
- Multimedia learning content: videos, PDFs, quizzes
- Real-time discussion forums and webinar integration
- Certificate auto-generation upon course completion
- Payment gateway integration for premium content
- Admin dashboard for platform and content governance

### **3. System Architecture**

#### **Frontend – *React.js***

- Component-driven architecture using functional components and hooks (`useState`, `useEffect`, `useContext`)
- Routing handled with React Router v6
- State managed via Context API
- API communication using Axios
- UI/UX implemented using Material-UI and Bootstrap 5

#### **Backend – *Node.js & Express.js***

- RESTful API architecture with modular routing
- Middleware stack for authentication (JWT), error handling, and logging
- Role-based access control (RBAC) for protected routes
- Secure REST APIs with data validation using express-validator

#### **Database – *MongoDB***

- Non-relational document storage using MongoDB Atlas
- Mongoose ODM for defining schemas and establishing model relationships

- **Collections:**
  - **Users** – stores user credentials, roles, and progress
  - **Courses** – metadata and content references
  - **Enrollments** – user-course mapping
  - **Certificates** – issued certificates with completion metadata
  - **Payments** – stores transaction records

## 4. Setup Instructions

### Prerequisites

- **Node.js (v18 or higher)**
- **MongoDB (Local or Atlas)**
- **Git CLI**
- **Visual Studio Code or any modern IDE**

### Installation Guide

```
# Clone repository
git clone https://github.com/your-username/learnhub.git
cd learnhub
```

```
# Backend setup
cd server
npm install
touch .env
# Add MONGO_URI, JWT_SECRET, PORT
npm start
```

```
# Frontend setup
cd ../client
npm install
npm start
```

## 5. Folder Structure

### Frontend (React)

```
client/
|
└── src/
    ├── components/   # Reusable UI components
    ├── pages/        # Route-based views
    ├── services/     # Axios services
    ├── context/      # Auth and global state management
    └── App.js         # Main routing and layout
    └── package.json
```

### Backend (Express)

```
server/
├── controllers/   # Request handlers
├── models/         # Mongoose schemas
├── routes/         # Express route definitions
├── middleware/    # Auth, validation, error handlers
├── utils/          # Helper functions (e.g., certificate gen)
└── server.js       # Entry point
```

## 6. Running the Application

### Development Mode

```
# Start backend  
  
cd server  
  
npm start  
  
# Start frontend  
  
cd client  
  
npm start
```

## 7. API Documentation

### Authentication

| Method | Endpoint           | Description                    |
|--------|--------------------|--------------------------------|
| POST   | /api/auth/register | Register a new user            |
| POST   | /api/auth/login    | Login and return JWT           |
| GET    | /api/auth/profile  | Retrieve user info (protected) |

### Course Management

| Method | Endpoint                | Description                                |
|--------|-------------------------|--|
| GET    | /api/courses            | Fetch all available courses                |
| GET    | /api/courses/:id        | Get course details                         |
| POST   | /api/courses            | Create course ( <i>Instructor only</i> )   |
| POST   | /api/courses/:id/enroll | Enroll in a course ( <i>Student only</i> ) |

## 8. Authentication & Authorization

- JWT-based token authentication with expiration and refresh logic
- Middleware: `verifyToken`, `checkRole`

- Roles: Admin, Instructor, Learner
- Protected API routes with RBAC
- Passwords hashed using bcryptjs

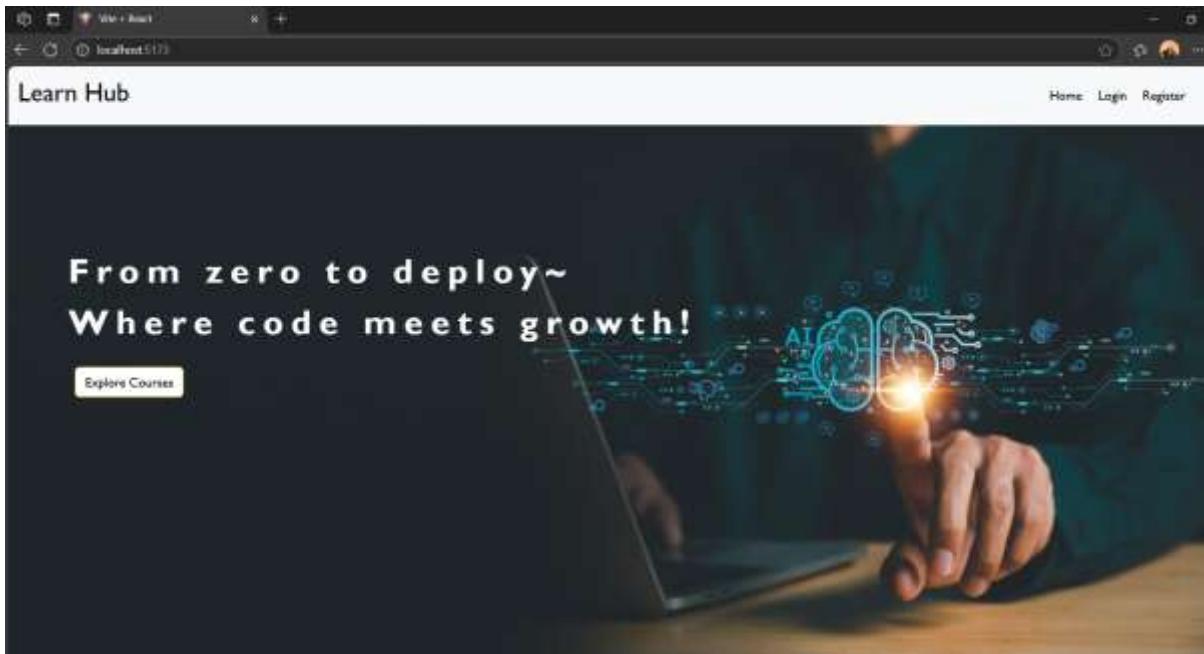
## 9. User Interface Highlights

- Responsive design (mobile-first) using Bootstrap Grid
- Dynamic rendering of UI elements based on user roles
- Dashboard view varies by user type
- Course cards, progress bars, status indicators
- Accessible modals, alerts, and tooltips

## 10. Testing Strategy

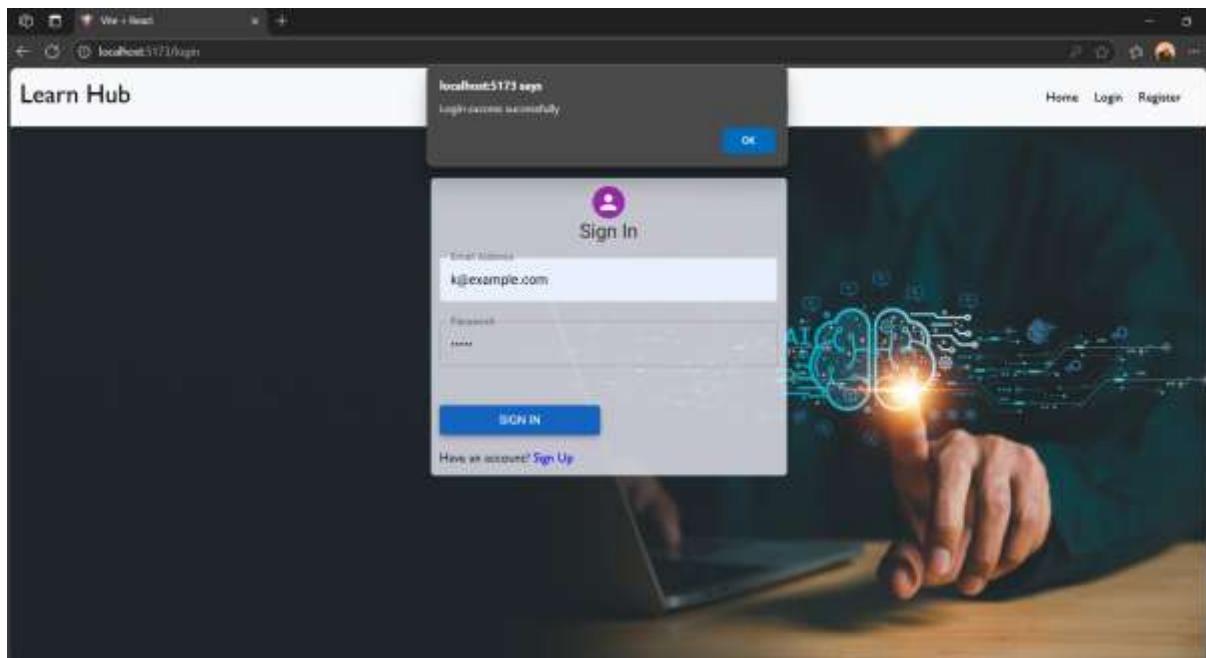
- API tested using Postman
- Manual UI Testing across browsers (Chrome, Firefox, Edge)
- Schema validation using Mongoose
- Basic unit testing with Jest (*optional*)

## 11. Screenshots:

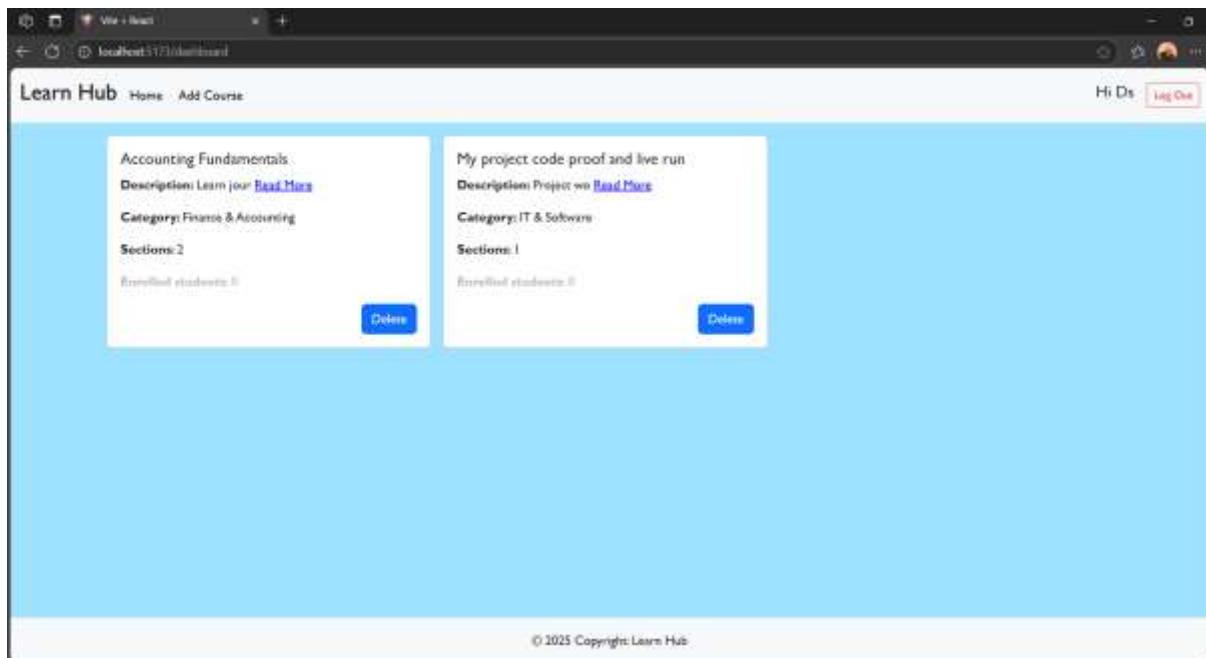


The screenshot shows the "Trending Courses" page. At the top, there is a search bar labeled "Search By: Title" and a dropdown menu set to "All Courses". Below the search bar, three course modules are listed in boxes:

| Modules  | Modules   | Modules   |
|--|---|---|
| Title:<br>Description:<br>many more to watch.. | Title: demo<br>Description: working space<br>Title: demo<br>Description: working window<br>many more to watch.. | Title: Live video<br>Description: The video contains som...<br>many more to watch.. |



A screenshot of a web browser window titled "Learn Hub" with the URL "localhost:5173/dashboard". The top navigation bar includes "Learn Hub", "Home", "Enrolled Courses", "Hi K", and "Log Out". The main content area displays a table with columns: "Course ID", "Course Name", "Course Educator", "Course Category", and "Action". A single row in the table contains the text "yet to be enrolled courses". At the bottom of the page, a copyright notice reads "© 2025 Copyright: Learn Hub".



## 12. Known Issues

- Certificate PDF generation delay on large files
- No push notifications implemented yet
- Some routes lack comprehensive unit tests

## 13. Future Enhancements

- Integration with third-party webinar tools (e.g., Zoom API)
- Gamification (badges, leaderboards, XP)
- AI-powered course recommendations
- Dedicated mobile app (React Native or Flutter)
- Admin analytics dashboard (e.g., charting enrollments, revenue)