

# Full Stack Development with MERN

## Learnhub-Your Center for Skill Enhancement

### 1. Introduction

- **Project Title:** LearnHub – Online Learning Platform
- **Team Members:**
  - *Jhansi* – Full Stack Developer
  - *Navya Sree* – UI/UX Designer
  - *Rajasri* – Backend Developer
  - *Karunasri*– QA Engineer

### 2. Project Overview

- **Purpose:**

LearnHub is an online learning platform that allows students to enroll in courses, interact with instructors, and track their learning progress. It also enables teachers to create and manage their courses.
- **Features:**
  - User roles: Student, Teacher, Admin
  - Course creation and enrollment
  - Authentication and authorization
  - Video and document support for lessons
  - Admin dashboard for user and course management
  - Review and rating system

### 3. Architecture

- **Frontend:**
  - Built using **React.js** with functional components and hooks
  - State management with **Context API**
  - Routing handled with **React Router DOM**
  - API calls with **Axios**
- **Backend:**
  - **Node.js** runtime with **Express.js** framework
  - RESTful APIs structured around resources (users, courses, enrollments)
  - Middleware for error handling, authentication, and validation
- **Database:**
  - **MongoDB** (NoSQL database) with **Mongoose** for schema modeling
  - Collections: Users, Courses, Enrollments, Reviews
  - Relationships:
    - One-to-many (Teacher → Courses)
    - Many-to-many (Student ↔ Courses through Enrollment)

### 4. Setup Instructions

- **Prerequisites:**
  - Node.js v18+
  - MongoDB
  - Git
  - npm or yarn
- **Installation:**

```
# Clone the repository
git clone https://github.com/yourusername/learnhub.git
cd learnhub

# Setup Backend
cd backend
npm install
# Create a .env file with:
# MONGO_URI=<your-mongodb-uri>
# JWT_SECRET=<your-secret-key>
npm run dev

# Setup Frontend
cd ../frontend
npm install
npm start
```

## 5. Folder Structure

- Client (React):

```
frontend/
├─ public/
├─ src/
│   ├─ components/
│   ├─ pages/
│   ├─ context/
│   ├─ services/
│   ├─ App.js
│   └─ index.js
```

- Server (Node.js):

```
backend/  
├── controllers/  
├── models/  
├── routes/  
├── middleware/  
├── config/  
├── index.js  
└── .env
```

## 6. Running the Application

- Frontend:

```
cd frontend  
npm start
```

- Backend:

```
cd backend  
npm run dev
```

## 7. API Documentation

Endpoint	Method	Description	Auth Required
/api/auth/register	POST	Register a new user	No
/api/auth/login	POST	Login user and return token	No
/api/courses	GET	Get all courses	Yes
/api/courses	POST	Create a new course (teacher)	Yes (Teacher)
/api/enroll/:id	POST	Enroll in a course	Yes (Student)

**Sample Request:**

POST /api/auth/register

```
{} package.json backend JS authRoutes.js JS authController.js .env {} packo

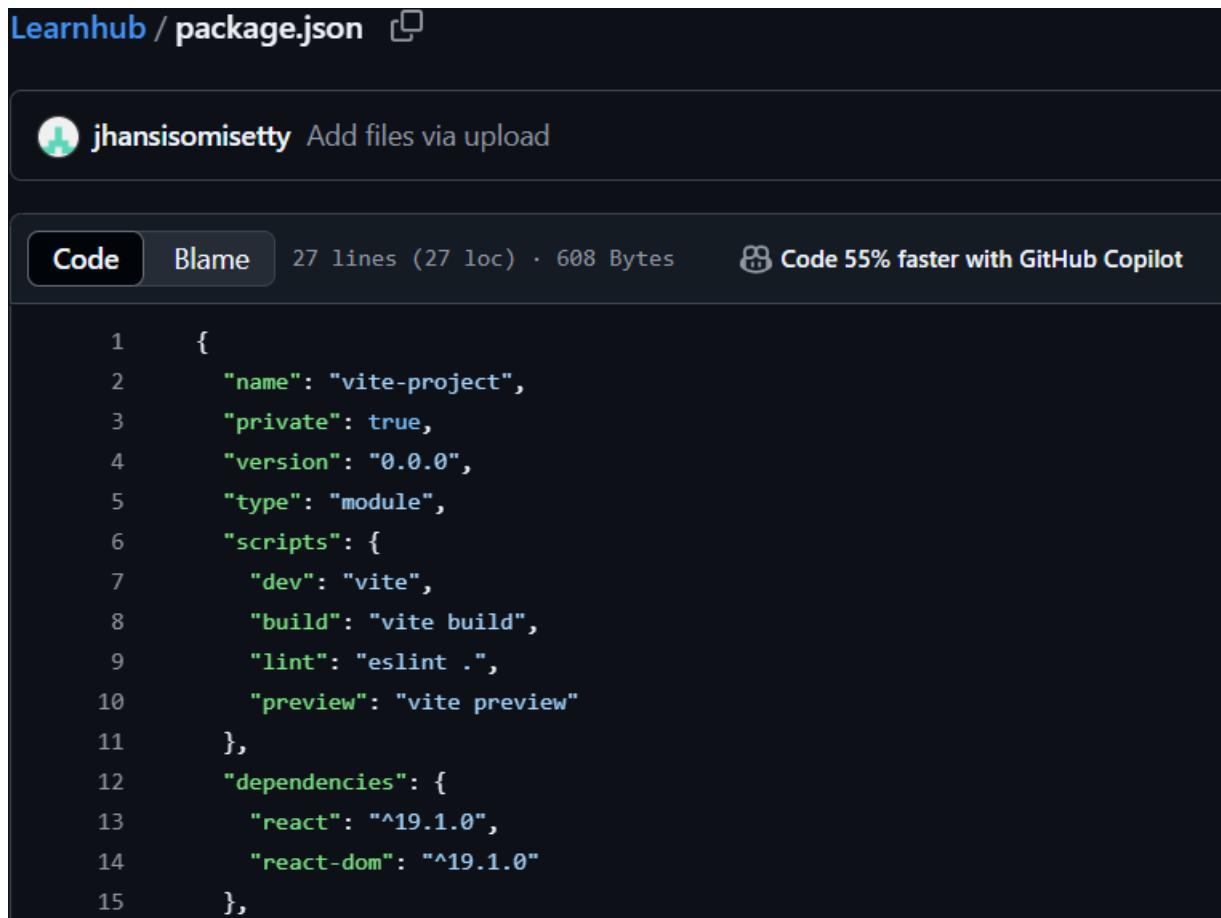
backend > middleware > JS authMiddleware.js > ...
 1  const jwt = require("jsonwebtoken");
 2  function auth(req, res, next) {
 3      const token = req.headers.authorization;
 4      if (!token) return res.status(401).json({ message: "Unauthorized" });
 5      try {
 6          const decoded = jwt.verify(token, process.env.JWT_SECRET);
 7          req.user = decoded;
 8          next();
 9      } catch (err) {
10          res.status(403).json({ message: "Forbidden" });
11      }
12  }
13  module.exports = auth;
```

**Sample Response:**

```
{
  "message": "User registered successfully",
  "token": "JWT_TOKEN_HERE"
}
```

## 8. Authentication

- **Method Used:** JWT (JSON Web Tokens)



The screenshot shows a GitHub repository page for 'package.json' in the 'Learnhub' repository. The repository is owned by 'jhansisomisetty'. The file is 27 lines (27 loc) and 608 Bytes. It includes a GitHub Copilot badge indicating 'Code 55% faster with GitHub Copilot'. The code is as follows:

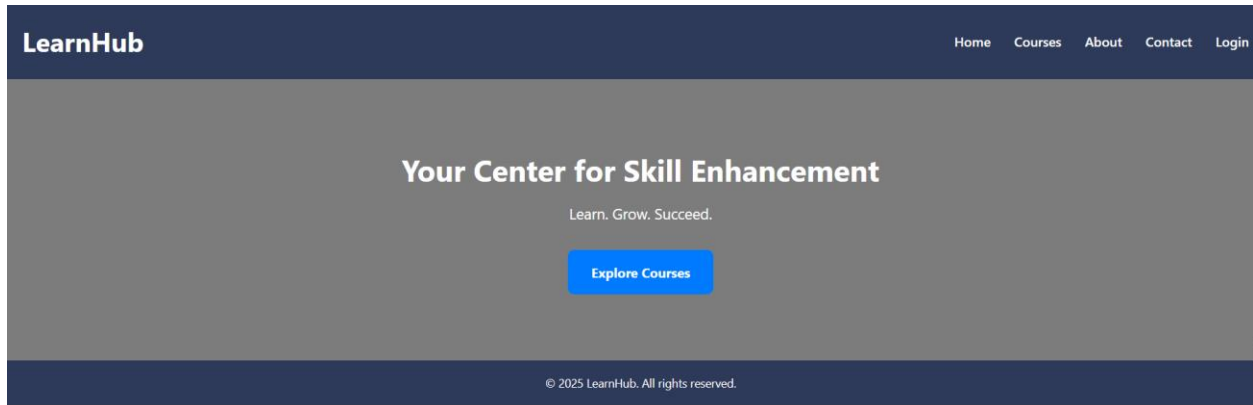
```
1  {
2    "name": "vite-project",
3    "private": true,
4    "version": "0.0.0",
5    "type": "module",
6    "scripts": {
7      "dev": "vite",
8      "build": "vite build",
9      "lint": "eslint .",
10     "preview": "vite preview"
11   },
12   "dependencies": {
13     "react": "^19.1.0",
14     "react-dom": "^19.1.0"
15   },
```

- **Workflow:**
  - On login, a JWT is issued and stored on the client (localStorage).
  - Protected routes are validated via middleware.
  - Token is verified using jsonwebtoken package on each request.

## 9. User Interface

- **Tech Stack:** React, TailwindCSS
- **Key UI Features:**
  - Responsive Dashboard for Teachers and Students
  - Admin panel with user management

- Course cards, modals for adding/editing
- Mobile-first design



## 10. Testing

- **Tools Used:** Jest, Postman, React Testing Library
- **Strategy:**
  - Unit tests for backend logic
  - Integration tests for routes and middleware
  - Manual testing for frontend components
  - API testing with Postman

## 11. Screenshots or Demo

- **Demo:** Live Site
- **Screenshots:**

Learnhub / Quiz.js

jhansisomisetty Add files via upload

Code Blame 11 lines (9 loc) · 307 Bytes Code 55% faster with GitHub Copilot

```
1 const mongoose = require("mongoose");
2
3 const quizSchema = new mongoose.Schema({
4   courseId: { type: mongoose.Schema.Types.ObjectId, ref: "Course" },
5   sectionIndex: Number,
6   question: String,
7   options: [String],
8   correctAnswer: String
9 });
10
11 module.exports = mongoose.model("Quiz", quizSchema);
```

Learnhub/courseRoutes.js at main · jhansisomisetty/Learnhub · GitHub

55f83c6 · 32 minutes ago History

Files

- main
- .gitignore
- .package-lock.json
- Course.js
- Quiz.js
- README.md
- User.js
- authController.js
- authMiddleware.js
- authRoutes.js
- courseController.js
- courseRoutes.js
- eslint.config.js
- index.html
- main.jsx
- package-lock.json
- package.json
- paymentRoutes.js

Learnhub / courseRoutes.js

jhansisomisetty Add files via upload

Code Blame 42 lines (35 loc) · 1.11 KB Code 55% faster with GitHub Copilot

```
1 import express from "express";
2 import auth from "../middleware/authMiddleware.js";
3 import {
4   getAllCourses,
5   getCourseById,
6   createCourse,
7   enrollInCourse,
8   getEnrolledCourses,
9   checkCourseCompletion,
10  markCourseCompleted,
11  getCertificate,
12  getCourseQuiz,
13  getQuizScore,
14  submitQuiz,
15  addComment,
16 } from "../controllers/courseController.js";
17
18 const router = express.Router();
19
20 // Public Routes
21 router.get("/", getAllCourses);
22 router.get("/:id", getCourseById);
23
24 // Student Routes (Requires Auth)
25 router.post("/enroll/:courseId", auth, enrollInCourse);
26 router.get("/enrolled", auth, getEnrolledCourses);
```



Learnhub / userRoutes.js

jhansomisetty Add files via upload

Code Blame 15 lines (13 loc) · 636 Bytes Code 55% faster with GitHub Copilot

```
1 router.post("/login", async (req, res) => {
2   const { email, password } = req.body;
3   try {
4     const user = await User.findOne({ email });
5     if (!user) return res.status(401).json({ error: "Invalid credentials" });
6
7     const isMatch = await bcrypt.compare(password, user.password);
8     if (!isMatch) return res.status(401).json({ error: "Invalid credentials" });
9
10    const token = jwt.sign({ id: user._id }, process.env.JWT_SECRET, { expiresIn: "1d" });
11    res.json({ token, userType: user.type, userId: user._id }); // ✅ Return userId
12  } catch (err) {
13    res.status(500).json({ error: "Login failed" });
14  }
15  });
```

## 12. Known Issues

- Video uploads not yet optimized for large files
- No email verification system
- Limited testing coverage on the frontend

## 13. Future Enhancements

- Add real-time chat between students and teachers
- Support for quizzes and certificates
- Implement email and SMS notifications
- Add Stripe integration for paid courses
- Progressive Web App (PWA) support