# Full Stack Development with MERN Learn Hub - Your Center for Skill Enhancement Online Learning Platform

#### 1. Introduction

- Project Title: LearnHub Your Center for Skill Enhancement
- Team Members:
  - o Ramireddygari Laxmiharikaa
  - o Namila Soni

## 2. Project Overview

# Purpose:

The purpose of LearnHub is to offer an interactive and accessible online learning platform that enables students to learn new skills and teachers to deliver content effectively. The platform supports registration, course management, and certification.

## Key Features:

- o User roles: Admin, Teacher, Student
- Course creation, categorization, and enrollment
- o Interactive elements: forums, webinars
- Certificate generation
- Payment integration for paid courses
- Responsive UI with filtering and search

## 3. Architecture

#### • Frontend:

Built with React.js using Vite for development, integrated with:

- Bootstrap
- Material UI

- MDB React UI Kit
- Axios for API calls
- React Router for navigation

#### Backend:

Developed using Express.js and Node.js.

- o RESTful API design
- o Authentication with JWT
- Middleware (authMiddleware.js) for protected routes

# • Database (MongoDB):

Two main collections:

```
    Users: { _id, name, email, password, type }
    Courses: { _id, userID, C_educator, C_categories, C_title, C_description, sections, C_price, enrolled }
```

## 4. Setup Instructions

## • Prerequisites:

- o Node.js and npm
- o MongoDB
- o Vite
- o Git

## Installation Steps:

```
git clone [your-repo-link]
cd your-project-folder

# Frontend setup
cd frontend
npm install

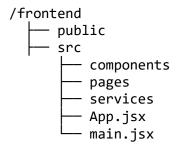
# Backend setup
cd ../backend
npm install
```

• Environment Variables: Create a . env file in the backend folder:

```
PORT=5000
MONGO_URI=your_mongo_connection_string
JWT_SECRET=your_jwt_secret
```

# 5. Folder Structure

## • Client (Frontend):



## • Server (Backend):



## 6. Running the Application

• Frontend:

cd frontend npm run dev

• Backend:

cd backend npm start

Access App:

http://localhost:5172

## 7. API Documentation

- POST /api/auth/register Register user
- POST /api/auth/login Login user
- GET /api/courses List courses
- POST /api/courses Add new course
- PUT /api/courses/:id Update course
- DELETE /api/courses/:id Delete course

• POST /api/enroll - Enroll in a course

## 8. Authentication

- JWT-based authentication
- Tokens issued upon login
- Middleware checks token validity before accessing protected routes

#### 9. User Interface Screens include:

- Landing Page
- Register/Login Page
- Admin Dashboard
- Teacher Dashboard
- Student Dashboard
- Course Detail and Enrollment

# 10. Testing

- Manual testing done across different roles
- Bug fixing during implementation
- UI verification after final build

### 11. Screenshots or Demo

Access all materials here: LearnHubDrive

## 12. Known Issues

- Payment integration may require real-world gateway
- No file upload for assignments or student submissions yet
- Limited error handling on some backend routes

#### 13. Future Enhancements

- Chat feature between student and instructor
- Al-based course recommendation
- Assignment submission & grading
- Mobile app version
- Multi-language support