

Full Stack Development with MERN

Project Documentation format

1. Introduction

- **Project Title:** Learn Hub
- **Team Members:** Padarth Akanksha – code written, documentation
Depuri Keerthana -demo video,
Dogiparthi Sai Tejasri-demo video,
Gummadi Thiru Naga Komali-documentation

2. Project Overview

- **Purpose:** Learn hub is an online platform designed to provide a comprehensive learning experience for users. To platform aims to offer a wide range of courses, tutorials, and educational resources, creating to diverse learning needs.
- **Features:** Learning Management and Administrative

3. Architecture

- **Frontend:** It provides client-side framework – reusable UI components, UI Library-Material-UI for consistent styling and layout, state management-Redux for managing global state and side effects, Routing, API Integration
- **Backend:** It is used to Node.js with Express.js for building a RESTful API and error handling errors and exceptions
- **Database:** Implemented indexing and caching mechanisms for improved query performance.

4. Setup Instructions

- **Prerequisites:** Visual Studio Code
- **Installation:** 1. Install VS code
2.Install language extensions (JavaScript)
3.Run code in VS code

5. Folder Structure

Backend Folder Structure:

backend/

└─ app.js

└─ config/

| └─ database.js

| └─ authentication.js

| └─ ...

└─ controllers/

| └─ courseController.js

| └─ userController.js

| └─ ...

└─ models/

| └─ courseModel.js

| └─ userModel.js

| └─ ...

└─ routes/

| └─ courseRoutes.js

| └─ userRoutes.js

| └─ ...

└─ services/

| └─ courseService.js

| └─ userService.js

| └─ ...

└─ utils/

| └─ authentication.js

| └─ authorization.js

| └─ ...

└─ package. Json

Frontend Folder Structure:

frontend/

└─ public/

| └─ index.html

| └─ ...

└─ src/

| └─ components/

| | └─ CourseComponent.js

| | └─ UserComponent.js

| | └─ ...

| └─ containers/

| | └─ CourseContainer.js

| | └─ UserContainer.js

| | └─ ...

| └─ actions/

| | └─ courseActions.js

| | └─ userActions.js

| | └─ ...

| └─ reducers/

| | └─ courseReducer.js

| | └─ userReducer.js

| | └─ ...

| └─ utils/

| | └─ api.js

| | └─ constants.js

```

| | └─ ...
| └─ App.js
| └─ index.js
| └─ ...
└─ package.json

```

6. Running the Application

o Frontend: 1.cd frontend

2. npm install

o Backend: 1.cd backend

2. npm install.

Database: MongoDB net start mongod.

7. API Documentation

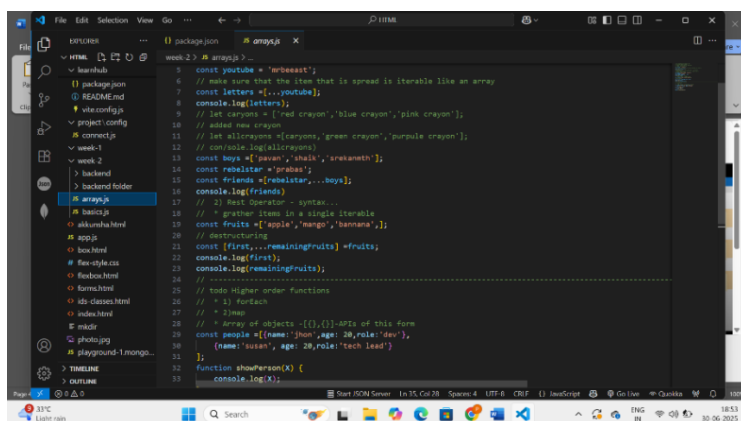
This API documentation provides a comprehensive overview of the available endpoints, request and response formats, error handling, authentication, and authorization. It also includes example requests to demonstrate how to use the API.

8. Authentication:

- Authentication Method: JSON Web Tokens (JWT)

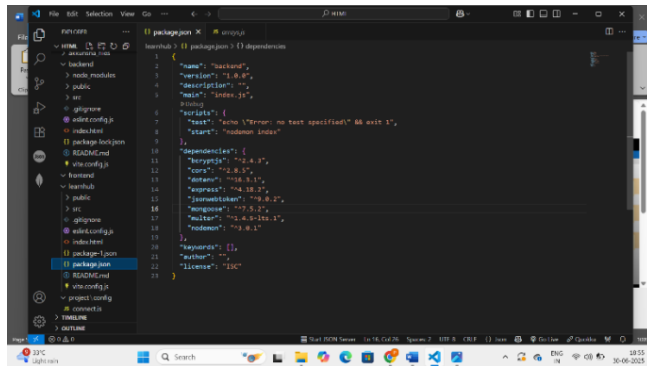
- Authentication Endpoint: /login

9. User Interface



10. Testing: Testing Json server

11. Screenshots or Demo



12. Known Issues: known issues like Browser compatibility and API security.

13. Future Enhancements: Artificial intelligence and Machine Learning ,Gamified and Engagement.