**PROF FINDER**

**COURSE REVIEW PLATFORM USING AI**

**Links**

[client repo link](https://github.com/Dharshini750/course_review_client)

[server repo link](https://github.com/Dharshini750/course_review_server)

[figma link](https://www.figma.com/design/JUrne2Z4lzbRpBnDVEjymf/Untitled?node-id=0-1&t=43bdI3pPVNf7fKa5-1)

**env file(frontend)**

REACT\_APP\_API\_URL=https://course-review-server-1.onrender.com

VITE\_API\_URL=https://course-review-server-1.onrender.com

**.env file (backend)**

MONGO\_URI=mongodb+srv://dharshiniramalingam2004:Dharshu\*04@cr.zu8khks.mongodb.net/?retryWrites=true&w=majority&appName=cr

PORT=5000

JWT\_SECRET=mysecret123

GEMINI\_API\_KEY=AIzaSyCNVq0Dc\_1v4xl92p8oHxtGAG006EFkxG0

Readme File

ONLINE COURSE REVIEW PLATFORM WITH AI

A platform to browse courses, add reviews, view analytics, and manage course-related tasks. This platform supports various user roles such as **\*\*Students\*\***, **\*\*Instructors\*\***, and **\*\*Admins\*\***, each with unique permissions and analytics.

**---**

**##  Features**

- **\*\*User Authentication\*\*** (Student, Instructor, Admin)

  - Sign Up, Login, and Logout functionality

  - Admin account: `admin@gmail.com`, password: `admin123`

- **\*\*Course Management:\*\***

  - **\*\*Browse Courses\*\***: View courses with detailed info like description, prerequisites, rating, difficulty, and instructor.

  - **\*\*Filter Courses\*\***: Filter by Department, Course, Instructor, Difficulty.

  - **\*\*Add Courses\*\*** (For Instructors): Instructors can add new courses.

- **\*\*Review System:\*\***

  - **\*\*Add Reviews\*\***: Students can add reviews to courses.

  - **\*\*View Reviews\*\***: Students can see reviews; Admins and Instructors can see course reviews for their respective courses.

- **\*\*Analytics:\*\***

  - **\*\*Student Analytics\*\***: View course analytics by department and instructor.

  - **\*\*Instructor Analytics\*\***: Instructors can see sentiment analytics for their courses.

  - **\*\*Admin Analytics\*\***: Admin can see overall analytics for all courses, departments, and instructors.

- **\*\*User Profiles:\*\***

  - Save and manage user data for **\*\*Students\*\*** and **\*\*Instructors\*\***.

  - Instructors have access to their course management and analytics on their profiles.

- **\*\*Responsive UI\*\***: Fully responsive, built with Tailwind CSS.

**---**

**##  Tech Stack**

- **\*\*Frontend\*\***:

  - React.js

  - React Router for navigation

  - Tailwind CSS for styling

  - Context API for global state management (AuthContext, CourseContext, ReviewContext)

- **\*\*Backend\*\***:

  - Express.js (Server)

  - MongoDB (Database for storing users, courses, and reviews)

  - JWT (For authentication)

- **\*\*AI Tools\*\***:

  -  **\*\*Gemini\*\*** (for course recommendations based on user skills and preferences)

- **\*\*Data Handling\*\***:

  - Filtering courses by department, course name, instructor, and difficulty

- **\*\*Data Management\*\***: MongoDb atlas for data storage

**---**

**##  Setup Instructions**

1. **\*\*Clone the Repository:\*\***

Install Dependencies: For both frontend and backend:

bash

Copy

Edit

npm install

Environment Variables: Create a .env file at the root of the project and add the necessary environment variables for the server (like MongoDB connection string, JWT secret, etc.).

Run the Development Server:

For Frontend:

bash

Copy

Edit

npm run dev

Open your browser and navigate to http://localhost:5173 to view the frontend.

For Backend:

bash

Copy

Edit

node server.js

 AI Tools Used

Gemini (OpenAI):

- bug fixing.

GitHub Copilot:

Provided suggestions for component structure .

**\*\*Deployment Steps\*\***

You can deploy the project easily on Vercel  for the frontend and Render for the backend.

Frontend Deployment (Vercel ):

-Go to Vercel

-Click New Project and import the repository.

-Choose React (Vite) as the framework.

-Deploy the project.

Backend Deployment (Render):

-Go to Render.

-Click New App and connect to GitHub.

-Set up the environment variables for MongoDB and JWT secret in the app settings.

-Deploy the app.

**\*\*Build for Production\*\***

bash

Copy

Edit

npm run build

\*\* Default User Credentials\*\*

Admin: admin@gmail.com | Password: admin123

Instructor: Use the signup page to create an account and add courses.

Student: Use the signup page to register and browse courses.

\*\* Key Features to Remember:\*\*

Admin Dashboard for global analytics

Instructor Dashboard to manage courses and view sentiment analytics

Student Experience with course filtering, reviews, and personalized analytics

Run the frontend (npm run dev) and backend (node server.js).

**\*\*Full Dependency List\*\***

Frontend Dependencies:

@emotion/react: ^11.14.0

@emotion/styled: ^11.14.0

@google/generative-ai: ^0.24.0

@reduxjs/toolkit: ^2.7.0

@tailwindcss/vite: ^4.1.4

axios: ^1.9.0

chart.js: ^4.4.9

echarts: ^5.6.0

echarts-for-react: ^3.0.2

fa: ^1.0.1

framer-motion: ^12.9.2

lucide-react: ^0.503.0

react: ^19.1.0

react-chartjs-2: ^5.3.0

react-dom: ^19.0.0

react-icons: ^5.5.0

react-redux: ^9.2.0

react-router-dom: ^7.5.2

react-tables: ^1.0.2

recharts: ^2.15.3

redux: ^5.0.1

tailwindcss: ^4.1.4

Backend Dependencies:

@google/generative-ai: ^0.24.0

@reduxjs/toolkit: ^2.7.0

axios: ^1.9.0

bcryptjs: ^3.0.2

body-parser: ^2.2.0

chart.js: ^4.4.9

cors: ^2.8.5

dotenv: ^16.5.0

express: ^5.1.0

jsonwebtoken: ^9.0.2

mongoose: ^8.13.3

react-redux: ^9.2.0

react-tables: ^1.0.2

recharts: ^2.15.3

sentiment: ^5.0.2

Dev Dependencies:

nodemon: ^3.1.10