**Title:** LogAnything

**One liner:** An all-in-one logging platform

**Description:** LogAnything is a MERN stack web app that lets users log and track anything—journals, tasks, fitness, food, sleep, movies, and more—in one customizable dashboard.

**How to run the application:**

**Frontend**

1. cd LogAnything/frontend
2. npm install
3. npm start

**Backend**

1. cd LogAnything/backend
2. npm install
3. node server.js

**Database**

1. Install MongoDB and use MongoDB Compass
2. Connect to localhost:27017

**Chatbot**

1. Navigate to Google's AI Studio and create a free Gemini API key
2. Add the API key to the backend's env file

**Project Architecture Overview:**

LogAnything is a full-stack journaling/logging application with AI integration, built using modern web technologies with a clear separation between frontend and backend.

**Frontend Technologies & Stack:**

1. **Core Framework & Libraries:**
    - **React 19.1.1** - Latest version of React for the UI framework
    - **React Router DOM 7.8.2** - Client-side routing and navigation
    - **React Icons 5.5.0** - Icon library for UI elements
    - **Axios 1.11.0** - HTTP client for API communication
    - **React Toastify 11.0.5** - Toast notifications for user feedback
2. **Development Tools:**
    - **Create React App (CRA)** - Build tooling and development server
    - **React Scripts 5.0.1** - Build and test scripts
    - **Testing Library** - Unit and integration testing framework
3. **Styling & UI:**
    - **CSS3** with custom styling
    - **Poppins font family** - Modern typography
    - **Responsive design** with CSS Grid and Flexbox
    - **Dark/Light theme support** via React Context

**Backend Technologies & Stack:**

1. **Core Framework:**
    - **Node.js** - JavaScript runtime
    - **Express.js 5.1.0** - Web application framework
    - **CommonJS modules** - Module system
2. **Database:**
    - **MongoDB** - NoSQL document database
    - **Mongoose 8.18.1** - MongoDB object modeling for Node.js
3. **Authentication & Security:**

- **JWT (JSON Web Tokens) 9.0.2** - Stateless authentication
- **bcryptjs 3.0.2** - Password hashing and salting
- **Custom auth middleware** - Token verification

4. **File Handling:**
   - **Multer 1.4.5** - File upload middleware (supports video uploads up to 100MB)
   - **Static file serving** - For uploaded video files
5. **AI Integration:**
   - **Google Generative AI SDK 0.19.0** - Official Google AI SDK
   - **Axios 1.4.0** - Fallback HTTP client for AI API calls
   - **Gemini AI model** - AI chat functionality
6. **Utilities:**
   - **CORS 2.8.5** - Cross-origin resource sharing
   - **dotenv 17.2.2** - Environment variable management

**Key Features & Functionalities:**

**User Management:**

- User registration and login
- JWT-based authentication
- Profile management (firstName, lastName updates)
- Password hashing with bcrypt

**Logging System:**

- Multi-category logging: Journal, Tasks, Fitness, Food, Sleep, Movies, Video, Custom
- Mood tracking: Happy, Sad, Normal, Angry, or No Mood
- Video logging: Upload videos up to 100MB (MP4, WebM, MOV, MKV)
- Text-based logs: Rich text content for various log types
- Date tracking: Automatic and manual date assignment

**Log Management:**

- Create new logs with modal interface
- View logs in organized layout

- Edit existing logs
- Delete logs
- Real-time updates

**AI Integration:**

- Chat Widget: AI assistant powered by Google Gemini
- Context-aware responses: AI can access user's recent logs for personalized responses
- Fallback system: Local help tips when AI backend is unavailable
- Real-time chat interface

**User Interface:**

- Responsive design - Works on desktop and mobile
- Dark/Light theme toggle - User preference persistence
- Modern UI components - Clean, intuitive interface
- Toast notifications - User feedback for actions
- Modal-based interactions - Add/Edit/View logs

**Pages & Navigation:**

- Dashboard - Landing page with features overview
- HomePage - Main logging interface
- Login/Signup - Authentication pages
- Account - Profile management
- About - Information about the app
- Analytics - Data insights
- Featured - Highlighted content

**Connections & API Structure:**

1. **Frontend-Backend Communication:**

   - Proxy configuration: Frontend proxies API calls to http://localhost:5000

   - RESTful API endpoints:

   - /api/auth/* - Authentication (login, signup)

   - /api/logs/* - Log CRUD operations

   - /api/user/* - User profile management

- /api/ai/* - AI chat functionality

2. **Database Schema:**
    - User Model: firstName, lastName, username, email, password, timestamps
    - Log Model: user (reference), title, type, mood, content (mixed), date, timestamps

3. **File Storage:**
    - Local file system - Video uploads stored in /backend/uploads/
    - Static serving - Files served via Express static middleware

**Authentication System:**

- JWT-based authentication with 1-hour token expiration
- Protected routes - Frontend route guards for authenticated pages
- Middleware protection - Backend route protection via auth middleware
- Token storage - LocalStorage for client-side token persistence
- Password security - bcrypt hashing with salt rounds

**Tech Stack:**

| Layer | Technology |
|---|---|
| **Frontend** | React 19, React Router, Axios, React Icons, CSS3 |
| **Backend** | Node.js, Express.js, MongoDB, Mongoose |
| **Authentication** | JWT, bcryptjs |
| **AI Integration** | Google Generative AI (Gemini) |

| | |
|---|---|
| **File Handling** | Multer, Express Static |
| **Development** | Create React App, Nodemon |
| **Database** | MongoDB (Cloud/Atlas) |

**Notable Features:**

1. AI-Powered Assistant - Context-aware chat widget that can reference user's logs
2. Video Logging - Upload and store video logs with notes
3. Multi-Modal Logging - Support for text, video, and various content types
4. Theme System - Dark/light mode with React Context
5. Responsive Design - Mobile-first approach with modern UI
6. Real-time Updates - Immediate UI updates after log operations
7. Secure Architecture - JWT authentication with protected routes