**KakshaDev**

Design Document

**Overview**

KakshaDev is a modern e-learning marketplace built on the MERN stack, designed to bridge traditional education with digital innovation. The platform facilitates knowledge sharing through high-quality video content and supplementary materials, connecting instructors with learners in an engaging online environment.

**Target Audience**

* Self-motivated learners seeking professional and personal development
* Subject matter experts and instructors looking to share their knowledge
* Working professionals aiming to upskill
* Students seeking supplementary learning resources

## **1. Features**

### Core Features:

 Course Management:

* Instructor dashboard for course creation and management
* Video content upload and organization
* Course pricing configuration

 User Management:

* Separate interfaces for learners and instructors
* Course enrolment
* Payment processing

 Content Delivery:

* Video streaming with playback controls

### Additional Features:

* Secure user authentication and account management system for seamless login and registration.
* Curate a personalized pages for users and admins.
* Secure payment gateway

## **2. Architecture**

KakshaDev follows a client-server architecture, with the frontend and backend components interacting via APIs.

### Frontend:

* Framework: React.j
* Routing: React Router
* Styling: CSS
* HTTP Requests: Axios

### Backend:

* Framework: Express.js
* Database: MongoDB (via Mongoose)
* API Documentation: Postman

## **3. User Interface Design**

* **Landing Page for** introducing user to KakshaDev
* **Course Page for** Featured courses carousel
* **Lecture Page for** Video player and Course content navigation
* **User Dashboard for** Enrolled courses
* **Instructor Dashboard for** Course analytics, Content management and Revenue tracking

## **4. Security Considerations**

* Implement proper input validation and sanitization to prevent injection attacks.
* Use secure authentication mechanisms (if implemented) such as JWT tokens or OAuth.
* Ensure data privacy and protection by encrypting sensitive information and passwords stored in the database.

## **5. Testing Strategy**

* Conduct integration testing to ensure seamless interaction between frontend and backend.
* Test user flows and edge cases to identify and resolve any bugs or issues.

**6. Database Schema:**

* **Users:**
* name
* email
* password
* role
* mainrole
* subscription
* timestamps
* **Courses:**
* title
* description
* image
* price
* duration
* category
* timestamps
* **Payment:**
* razorpay\_order\_id
* razorpay\_payment\_id
* razorpay\_signature
* createdAt
* **Progress**
* course
* completedLectures
* user
* timestamps
* **Lecutre**
* title
* description
* video
* course
* createdAt

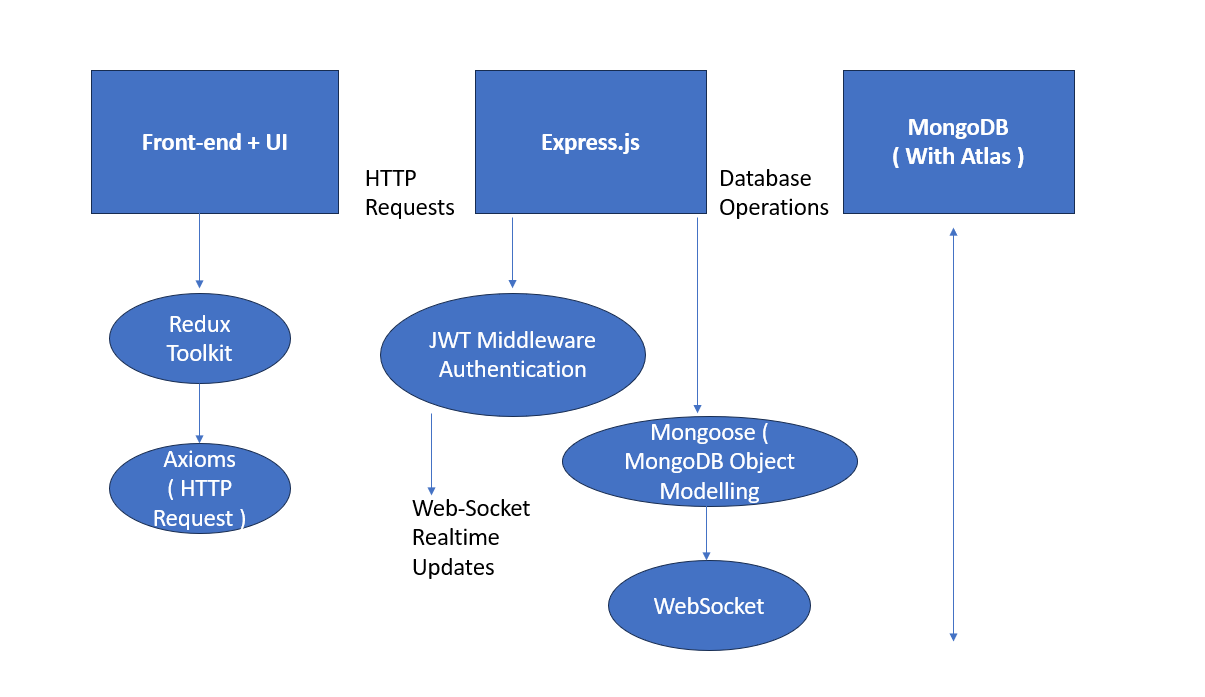
## 

## **Diagram depicting an overview of the Database**

## **7. API Endpoints:**

## Endpoint definitions for User Authentication, User Profiles, Community Management, Content Sharing, Messaging, and Notifications.

## **8. Technology Stack:**



## **9. Conclusion**

KakshaDev aims to redefine the eLearning experience by providing users with a platform for authentic connections, engaging lectures, and vibrant communities. With its intuitive interface, comprehensive features, and robust architecture, KakshaDev promises to deliver a seamless and enjoyable eLearning experience for users across the globe.