main

Learnify / README.md

md-shahid-ansari documentation edited 91a17b5 · now

259 lines (197 loc) · 11.2 KB

Preview Code Blame Raw

Learnify is an online learning platform where tutors can create courses, students can enroll in courses, and both can manage their learning experience. Admins oversee the platform, manage users, and ensure the smooth operation of the system. Upon course completion, students receive certificates.Learnify is an online learning platform where tutors can create courses, students can enroll in courses, and both can manage their learning experience. Admins oversee the platform, manage users, and ensure the smooth operation of the system. Upon course completion, students receive certificates.Learnify is an online learning platform where tutors can create courses, students can enroll in courses, and both can manage their learning experience. Admins oversee the platform, manage users, and ensure the smooth operation of the system. Upon course completion, students receive certificates.Learnify is an online learning platform where tutors can create courses, students can enroll in courses, and both can manage their learning experience. Admins oversee the platform, manage users, and ensure the smooth operation of the system. Upon course completion, students receive certificates.Learnify is an online learning platform where tutors can create courses, students can enroll in courses, and both can manage their learning experience. Admins oversee the platform, manage users, and ensure the smooth operation of the system. Upon course completion, students receive certificates.Learnify is an online learning platform where tutors can create courses, students can enroll in courses, and both can manage their learning experience. Admins oversee the platform, manage users, and ensure the smooth operation of the system. Upon course completion, students receive certificates.Learnify is an online learning platform where tutors can create courses, students can enroll in courses, and both can manage their learning experience. Admins oversee the platform, manage users, and ensure the smooth operation of the system. Upon course completion, students receive certificates.Learnify is an online learning platform where tutors can create courses, students can enroll in courses, and both can manage their learning experience. Admins oversee the platform, manage users, and ensure the smooth operation of the system. Upon course completion, students receive certificates.Learnify is an online learning platform where tutors can create courses, students can enroll in courses, and both can manage their learning experience. Admins oversee the platform, manage users, and ensure the smooth operation of the system. Upon course completion, students receive certificates.Learnify is an online learning platform where tutors can create courses, students can enroll in courses, and both can manage their learning experience. Admins oversee the platform, manage users, and ensure the smooth operation of the system. Upon course completion, students receive certificates.

# Learnify 🚀

Learnify is an online learning platform where tutors can create courses, students can enroll in courses, and both can manage their learning experience. Admins oversee the platform, manage users, and ensure the smooth operation of the system. Upon course completion, students receive certificates.

## Features

- **Student Functionality**:
  - Register, login, and manage personal profiles.
  - Enroll in courses and access lessons and quizzes.
  - Earn certificates upon completing courses.
- **Tutor Functionality**:
  - Register and create courses with modules, lessons, and quizzes.
  - Manage course content and interact with enrolled students.
- **Admin Functionality**:
  - Manage tutors, students, and courses.
  - Access platform-wide data and oversee operations.
- **Certificate Issuance**:
  - Students earn certificates automatically upon course completion.
  - Students can download it as PDF.

## Technology Stack

- **Backend**: Node.js, Express.js
- **Frontend**: React.js, Tailwind CSS
- **Database**: MongoDB (with GridFS for file storage)
- **Authentication**: JWT for secure role-based access control

## Requirement Analysis

The requirement analysis focuses on understanding the needs of all stakeholders—students, tutors, and administrators—to design a platform that fulfills its objectives effectively.

### Stakeholder Needs:

1. **Students**:
   - Easy registration and access to courses.
   - A user-friendly interface for managing courses and tracking progress.
   - Automated certificate generation for completed courses.
2. **Tutors**:
   - Tools for creating and managing detailed courses with modules, lessons, and quizzes.
   - A platform to interact with students and track their progress.
3. **Admins**:
   - Oversight of platform operations, including user and content management.
   - Tools for resolving issues and maintaining platform integrity.
4. **Platform Requirements**:
   - Secure authentication and role-based access control.
   - Scalable backend to handle growing user data.
   - Efficient file storage for course materials and certificates.

### Functional Requirements:

- User registration and role-based authentication.
- CRUD operations for courses, lessons, and quizzes.
- Enrollment management and progress tracking.
- Certificate generation and storage.
- Administrative tools for monitoring and maintenance.

### Non-Functional Requirements:

- Scalability to accommodate increasing users.
- High availability and performance.
- Secure data handling with encryption for sensitive information.

- Intuitive UI/UX for all user roles.

## Feasibility Study

The feasibility study evaluates the practicality of implementing Learnify within the defined constraints of time, technology, and resources.

### Technical Feasibility:

- The technology stack (Node.js, React.js, MongoDB) is capable of meeting the platform's requirements.
- Using **GridFS** for file storage ensures efficient handling of course materials and certificates.
- JWT authentication provides a secure mechanism for managing user roles and access control.

### Operational Feasibility:

- The platform addresses the core needs of all stakeholders, promoting ease of use and accessibility.
- The learning curve for the system is manageable for tutors, students, and admins.

### Economic Feasibility:

- Open-source tools and technologies minimize development costs.
- Future scalability ensures long-term cost-effectiveness.

### Schedule Feasibility:

- Development timelines are realistic with modular milestones.
- The team is equipped to deliver the platform within the stipulated time frame.

### Legal and Compliance Feasibility:

- The platform ensures data privacy and adheres to standards such as GDPR.
- Transparent user policies align with legal requirements for online platforms.

## API Documentation

### Authentication Endpoints

### Student Authentication

- `POST /student-register`: Register a new student.
- `POST /student-verify`: Verify student email.
- `POST /student-login`: Login for students.
- `POST /student-forgot`: Request password reset.
- `POST /student-reset/:token`: Reset password.
- `POST /student-logout`: Logout from the platform.
- `GET /student-auth`: Authenticate student session.

### Tutor Authentication

- `POST /tutor-register`: Register a new tutor.
- `POST /tutor-verify`: Verify tutor email.
- `POST /tutor-login`: Login for tutors.
- `POST /tutor-forgot`: Request password reset.
- `POST /tutor-reset/:token`: Reset password.
- `POST /tutor-logout`: Logout from the platform.
- `GET /tutor-auth`: Authenticate tutor session.

### Admin Authentication

- `POST /admin-register`: Register a new admin.
- `POST /admin-verify`: Verify admin email.
- `POST /admin-login`: Login for admins.
- `POST /admin-forgot`: Request password reset.
- `POST /admin-reset/:token`: Reset password.
- `POST /admin-logout`: Logout from the platform.
- `GET /admin-auth`: Authenticate admin session.

---

## Course Management Endpoints

### For Tutors

- `POST /create-course`: Create a new course.
- `POST /update-course`: Update an existing course.
- `POST /delete-course`: Delete a course.
- `POST /fetch-courses`: Fetch all courses created by the tutor.
- `POST /upload-image`: Upload course images.
- `GET /files/:fileId`: Retrieve uploaded files.

#### For Students

- `POST /enroll`: Enroll in a course.
- `POST /course`: Get details of a specific course.
- `POST /enrollments`: Get all enrolled courses.
- `POST /get-enrollment`: Get details of a specific enrollment.
- `POST /add-lesson`: Add a lesson to a course module.
- `POST /add-quiz`: Add a quiz to a course module.
- `POST /certificates`: Fetch all certificates for the student.

#### Admin Management

- `POST /get-all-courses`: Fetch all courses without modules.
- `POST /get-all-enrollments`: Retrieve all enrollments.

---

### User Management Endpoints

#### Profile Updates

- `POST /tutor-update`: Update tutor details.
- `POST /student-update`: Update student details.
- `POST /admin-update`: Update admin details.

#### Admin Actions

- `POST /get-all-admins`: Fetch all admin accounts.
- `POST /get-all-tutors`: Fetch all tutor accounts.
- `POST /get-all-students`: Fetch all student accounts.
- `POST /get-all-certificates`: Fetch all certificates issued.
- `POST /delete-batch`: Delete multiple entities in a batch.

---

## How It Works

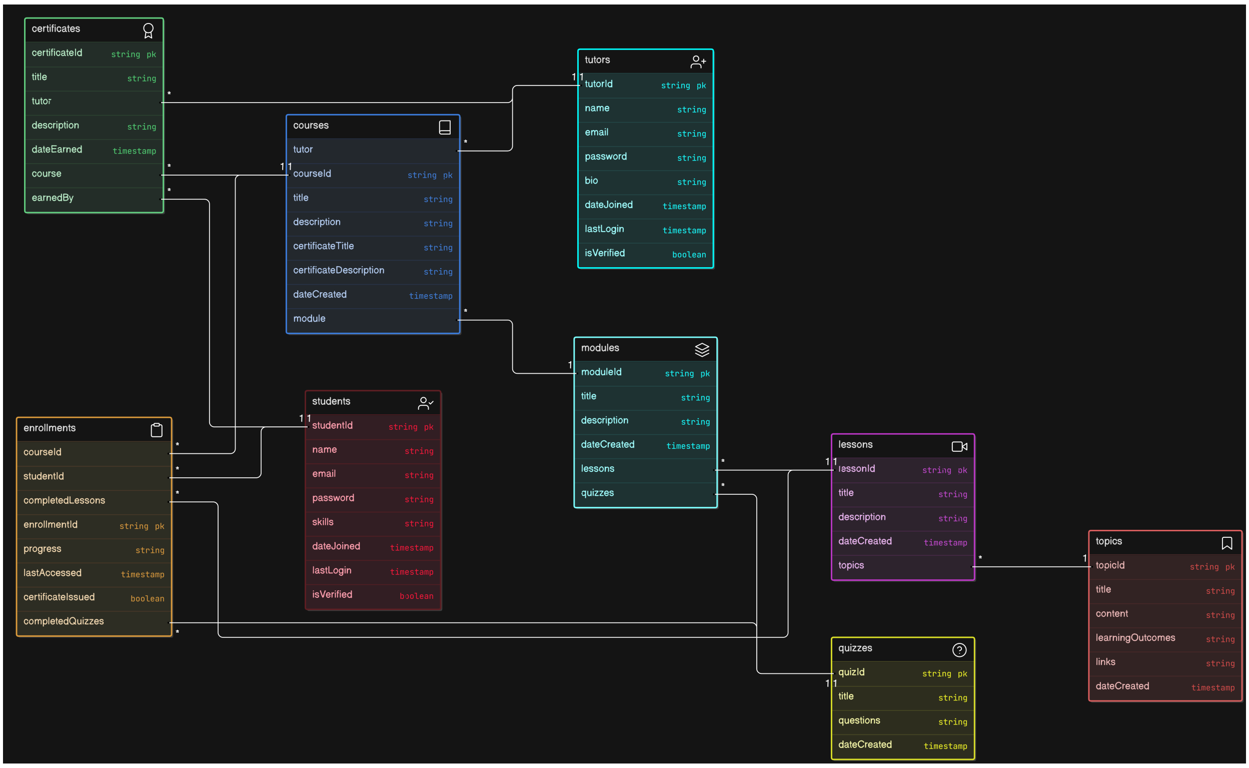
1. **Students**:
   - Register, log in, and enroll in courses.
   - Access course materials (lessons, quizzes) and complete them at their own pace.
   - Receive certificates upon course completion.
2. **Tutors**:

- Register, log in, and create courses with detailed modules, lessons, and quizzes.
- Manage course content, including updates and deletions.

3. **Admins**:

- Oversee platform operations, manage users, and access platform-wide data.
- Perform administrative tasks like deleting data in batches.

# Entity-Relationship Diagram



| certificates | |
|---|---|
| certificateId | string pk |
| title | string |
| tutor | |
| description | string |
| dateEarned | timestamp |
| course | |
| earnedBy | |

| courses | |
|---|---|
| tutor | |
| courseId | string pk |
| title | string |
| description | string |
| certificateTitle | string |
| certificateDescription | string |
| dateCreated | timestamp |
| module | |

| tutors | |
|---|---|
| tutorId | string pk |
| name | string |
| email | string |
| password | string |
| bio | string |
| dateJoined | timestamp |
| lastLogin | timestamp |
| isVerified | boolean |

| modules | |
|---|---|
| moduleId | string pk |
| title | string |
| description | string |
| dateCreated | timestamp |
| lessons | |
| quizzes | |

| students | |
|---|---|
| studentId | string pk |
| name | string |
| email | string |
| password | string |
| skills | string |
| dateJoined | timestamp |
| lastLogin | timestamp |
| isVerified | boolean |

| enrollments | |
|---|---|
| courseId | |
| studentId | |
| completedLessons | |
| enrollmentId | string pk |
| progress | string |
| lastAccessed | timestamp |
| certificateIssued | boolean |
| completedQuizzes | |

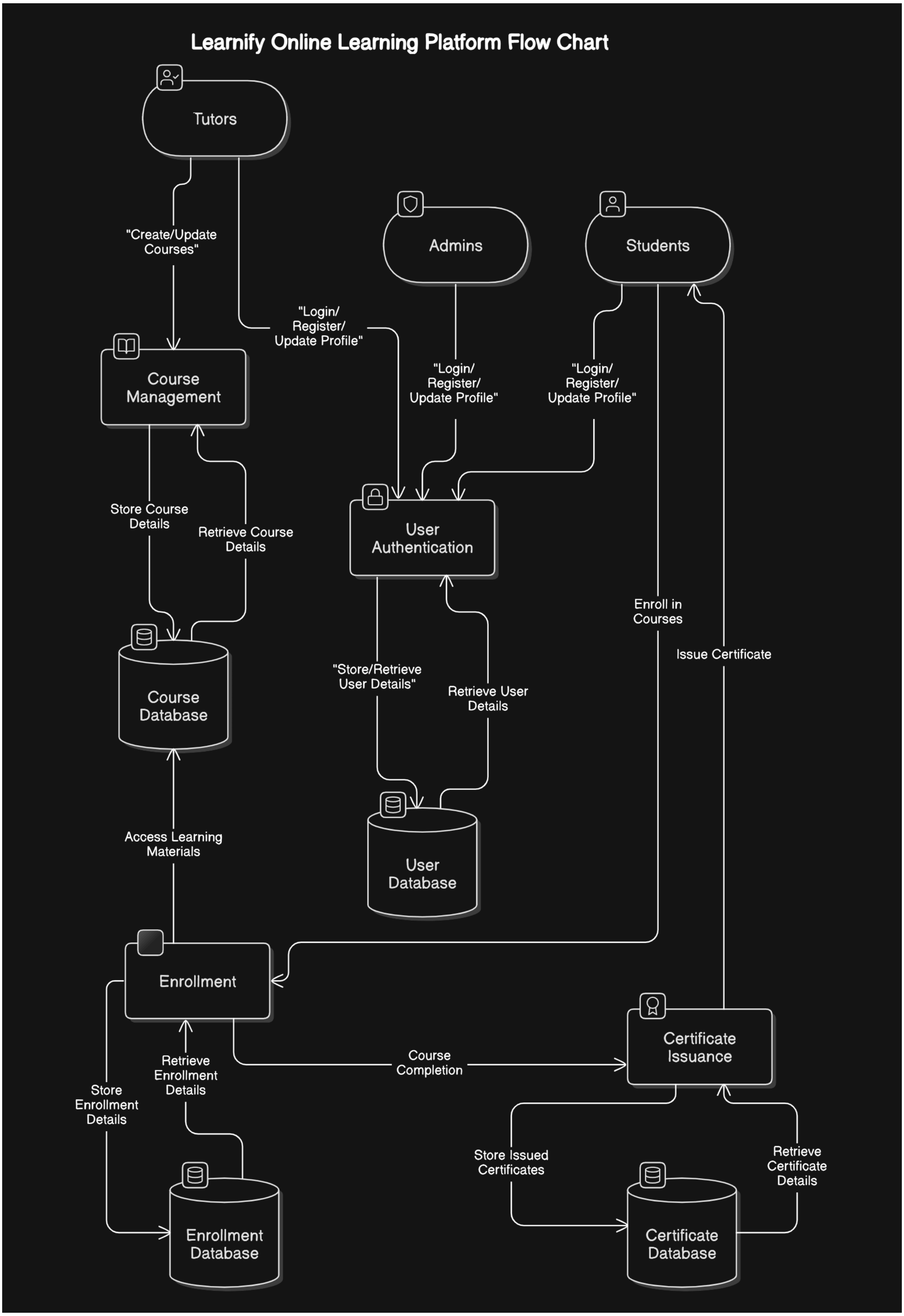
| lessons | |
|---|---|
| lessonId | string pk |
| title | string |
| description | string |
| dateCreated | timestamp |
| topics | |

| quizzes | |
|---|---|
| quizId | string pk |
| title | string |
| questions | string |
| dateCreated | timestamp |

| topics | |
|---|---|
| topicId | string pk |
| title | string |
| content | string |
| learningOutcomes | string |
| links | string |
| dateCreated | timestamp |

The ERD ensures a robust database structure, supporting efficient data retrieval and management across Learnify's functionalities. The relationships facilitate seamless interaction between Students, Tutors, and Admins.

# Data Flow Diagram



The DFD highlights the efficient handling of user interactions, course content, and administrative operations within the Learnify platform. Each component ensures smooth operation and data security.

## Final Summary

Learnify is a comprehensive online learning platform designed to empower tutors and students while streamlining the learning experience. With features like role-based access, detailed course management, and automated certificate issuance, Learnify fosters an engaging and efficient environment for education. The robust backend, supported by technologies like Node.js, MongoDB, and JWT authentication, ensures scalability, security, and smooth operation.

We are committed to continuous improvement and innovation, with future enhancements planned to make learning even more interactive and personalized.

---

### Thank You!

We deeply appreciate your interest in Learnify. Whether you are a student, tutor, or developer, your engagement is vital to the success of this platform. Together, we can create a brighter future for education. Thank you for your support!