⊛ ChatGPT

Online Learning System: Course and Quiz Management

Course Creation

- **Creators:** Both Admins and Instructors can create new courses.
- **Required fields:** Each course requires a title, description, and status (Public or Private). An optional cover image can be uploaded (defaults to "default_course.jpg" if omitted).
- **Hierarchy:** Courses are the top-level container. Each course can have multiple chapters, each chapter can have multiple topics, and each topic can include lessons and associated content.

Course Structure

- **Hierarchy:** The content is organized as Courses → Chapters → Topics → Content. For example, a course might have "Introduction" and "Advanced Concepts" chapters, each with several topics and lesson materials.
- **Nested organization:** This hierarchical model ensures a clear learning path. Instructors build this structure via the UI by adding chapters under courses and topics under chapters.

Course Management Features

- **Admins:** Can view all courses, add new courses, edit existing courses, and remove or toggle the visibility of courses (Public/Private). Admins can also manage all course content (chapters, topics, lessons) and view details of any course, including its chapters and topics.
- **Instructors:** Can create and manage only their own courses. They can add chapters to their courses, create topics within those chapters, and add/edit lesson content. Instructors can toggle their course visibility (Public/Private) and update course details.
- **Course Content Management:** Both roles use form-based interfaces to input course data. Input validation (title length, required fields) ensures data integrity. Uploaded images (cover images) are stored securely, and default images are used if none is provided.

Quiz Creation and Structure

- **Creators:** Both Admins and Instructors can create quizzes associated with their courses or topics. A quiz is typically tied to a specific course or a course topic.
- **Required fields:** Each quiz has a title and optional description. You can attach the quiz to a course or specific chapter/topic. Other fields may include a passing score, time limit, and status (Public/Private). An optional cover image or banner can be set (with a default placeholder if none is provided).
- **Structure:** Quizzes consist of multiple questions. Each question includes the question text and a list of answer options. Questions can be multiple-choice (single or multiple correct answers) or other types (e.g., true/false, short answer) if implemented. The basic data model is hierarchical: each Quiz

has many Questions, and each Question has many Options (answers). A "correct" flag identifies the right option(s).

- **Multilingual fields:** For each question and answer option, text is provided in all supported languages: English, Amharic, Tigrinya, and Oromo. In practice, this can be implemented by having separate database columns (e.g. `question_en`, `question_am`, `question_ti`, `question_om`) for each language [1]. This allows the system to display the question and options in the learner's chosen language.

Quiz Management Features

- **Admins:** Can view and manage all quizzes in the system. This includes adding new quizzes, editing existing ones (changing title, questions, etc.), deleting quizzes, and toggling quiz status (Public/Private). Admins can also manage the content of a quiz by adding or modifying its questions and answer options. They can view quiz details, including all questions and answers, and see student attempt data for any quiz.
- **Instructors:** Can create and manage only their own quizzes. They use the quiz creation form to add questions and answers (with fields for each language). Instructors can publish quizzes to make them available, edit or remove questions, and toggle the quiz's visibility. They can also view performance statistics for students who have taken their quizzes (scores, correct/incorrect answers).
- **Quiz Content Editing:** The UI provides forms for adding questions and options. Each question form includes fields for the text in English, Amharic, Tigrinya, and Oromo. Answer options similarly have multilingual fields. Instructors mark which option(s) are correct. Changes are saved to the database, and content is updated immediately.

Multilingual Quiz Support

- **Languages:** The system fully supports content in English, Amharic, Tigrinya, and Oromo. All quiz text (titles, questions, answers) must be entered in each of these languages.
- **Data storage:** For a fixed set of languages, the simplest design is to include separate columns in the database for each language [1]. For example, a `questions` table might have columns `text_en`, `text_am`, `text_ti`, and `text_om`. Likewise, an `options` table has language-specific text columns. This ensures fast lookups and straightforward queries when rendering a quiz in a particular language.
- **Alternative translation table:** If more languages might be added in the future, one can instead use a generic translations table. For example, a `translations` table with columns `(language_id, string_id, value)`, where each piece of text is referenced by a string ID and language code [2]. When presenting the quiz, the system queries this table with the chosen language. This dynamic approach accommodates an open-ended number of languages, though for our fixed four languages, separate columns are efficient.
- **User experience:** When a student takes a quiz, they select their preferred language (or it defaults to their account setting). The front-end then displays the quiz title, all questions, and answer options in that language. This multilingual support ensures learners can comfortably read and respond in their native language.

Quiz Scoring and Results

- **Auto-grading:** The system can automatically grade objective questions (e.g., multiple-choice). When a student submits a quiz, the backend checks the selected options against the correct flags and computes a score.
- **Results storage:** Each quiz attempt is recorded in a results table with references to the user, quiz, and each question's answer. Scores and timestamps are stored. This allows reporting and analytics.
- **Feedback:** After submission, students receive immediate feedback (score, which answers were correct). Instructors and admins can view aggregate results and statistics (average score, question difficulty).

Course Status

- **Visibility:** Both courses and quizzes have a status flag (Public or Private). A Public course or quiz is visible and accessible to all eligible students. A Private course/quiz is restricted to specific enrolled students or groups. Instructors and admins can toggle this status.
- **Access control:** If a course is private, its quizzes and content are only accessible by users explicitly granted access. Public courses' quizzes are available to any logged-in student.

User Interface

- **Design:** The platform uses a modern, responsive UI framework (e.g. Bootstrap) with a dark theme. The layout provides easy navigation: menus for Courses, Quizzes, and Profile.
- **Language switching:** A language selector (e.g., in the header) lets users switch the UI and content language among English, Amharic, Tigrinya, and Oromo. All interface labels (buttons, menus) are translated via a localization system (e.g., Laravel's localization feature 【9†source†L0-L3】), so the application text appears in the chosen language. Quiz content loads the appropriate language as described above.
- **Forms:** The course/quiz creation forms have fields for each supported language version of titles and descriptions. Rich text editors for content (e.g. CKEditor) support Unicode and input for non-Latin scripts (Amharic, etc.). Image upload fields (for cover images or question images) accept common formats.

Security Features

- **Role-based Access:** The system enforces permissions based on user roles. Admin-only routes protect site-wide actions (viewing all courses/quizzes, managing users). Instructors have access only to their content. Students cannot access creation or edit forms.
- **Authentication:** Users log in via a secure session-based authentication system. Passwords are hashed and sessions are stored securely.
- **Input validation:** All form inputs are validated and sanitized. For example, text fields are sanitized to prevent XSS, and uploaded images are validated for type/size.
- **Protected data:** Quiz answers and results are stored securely; only users with permission (the quiz taker, and the instructor/admin) can view a given result.

Using the System

- **As an Admin:**
- Log in with admin credentials.
- Navigate to the **Courses** section to add or manage courses (use the "Add Course" button).
- Go to the **Quizzes** section (or within a course) to add new quizzes. Use the quiz creation form to enter titles, upload images, and attach the quiz to a course/chapter.
- In the quiz editor, add questions and provide the text for each language (English, Amharic, Tigrinya, Oromo). Add answer options with correct flags. Save the quiz to publish it.

- Manage existing quizzes by editing or toggling their status. View any quiz's details and student results.

- **As an Instructor:**

- Log in with instructor credentials.
- Go to **My Courses** and create a new course (if needed), filling in title, description, and language-specific fields.
- Within your course, add chapters and topics. In each topic, you can attach content (lessons, videos, documents).
- Navigate to the **Quizzes** tab for a course or topic. Click "Add Quiz" to create a quiz for that section. Enter the quiz's title and description (with translations) and configure settings (status, time limit, etc.).
- Add questions: for each question, enter the prompt in each language and add multiple answer options (also multilingual). Mark the correct answer(s). Save the question and repeat for additional questions.

- Publish the quiz. Students enrolled in the course will see it. You can return later to edit questions or review quiz results.

- **As a Student:**

- Log in with student credentials.
- Browse the list of available **Public Courses** or any private courses you have access to.
- Enter a course to view its chapters and topics. The interface will display content in the language you've selected (or your profile's default language).
- When you reach a topic with an attached quiz, you will see a link or button to **Take Quiz**. Click it to begin.
- The quiz will present all questions in your chosen language (English, Amharic, Tigrinya, or Oromo). Select your answers and submit the quiz.
- After submission, you will see your score and which answers were correct. This result is also saved in your profile under completed quizzes or progress report.

The system provides a comprehensive online learning platform where courses and quizzes are structured hierarchically and fully localized. By requiring multilingual input for quiz content and offering a language-select UI, learners can study and be assessed in their native language (Amharic, Tigrigna, Oromo, or English) [1] [2] . Role-based access ensures Admins and Instructors manage content appropriately, while students have a seamless, multilingual learning experience.

Sources: The design of multilingual quiz support is based on best practices for database schemas and localization [1] [2] . Features such as role-based content management and hierarchical course organization are standard in modern LMS architectures.

---

[1] [2] mysql - database design for quiz with different languages - Stack Overflow
https://stackoverflow.com/questions/51022843/database-design-for-quiz-with-different-languages