Documentations of Online exam web portal + admin panel (php Laravel)

Project Name: Online Exam Web Portal
Project Type: Web Application
Platform: PHP Laravel
Used Programming Languages: HTML, HTML5, CSS, CSS3, Bootstrap, JavaScript
Database: MySQL
Software Tools: Laravel, Composer, XAMPP, Visual Studio Code
Supported Browsers: Internet Explorer, Opera Mini, Google Chrome, Mozilla Firefox
Supported Operating Systems: Windows 7, Windows 8, Windows 10, Linux, Mac

Project Overview

The **Online Exam Web Portal** is a web application developed using Laravel, designed to facilitate online examinations for educational institutions. The system allows students to register, take exams, and view results, while administrators can manage exams, questions, and user accounts.

Features

User Interface:

- 1. User Registration and Login
 - Students can register using their email and password.
 - Existing users can log in to access their exam dashboard.

2. Exam Dashboard

- Students can view available exams and their schedules.
- Access to past exam results and performance analysis.
- 3. Take Exam
 - Users can take scheduled exams with real-time timer.
 - Automatic saving of progress and submission functionality.

4. Results and Feedback

- After completing exams, users can view their scores and feedback on performance.
- 5. Profile Management

• Users can update their profiles and change passwords.

Admin Panel:

- 1. Admin Login
 - Secure login for administrators to manage the portal.
- 2. Manage Exams
 - Create, edit, or delete exams and set schedules.
 - Upload exam-related documents and instructions.

3. Question Management

- Add, edit, or delete questions across various subjects and exams.
- 4. User Management
 - View and manage student accounts and their exam activities.
- 5. Generate Reports
 - Generate performance reports based on student results and exam metrics.

Software Requirements

- **PHP:** 7.2 or higher
- MySQL: 5.7 or higher
- Laravel: 8.x or higher
- Web Server: Apache or Nginx
- Database Management: phpMyAdmin or similar tool

How to Configure (Run) the Project

- 1. Download the Project Zip File
 - Download the zip file containing the source code of the Online Exam Web Portal.

2. Unzip and Place in Root Directory

• Unzip the project file and place the extracted files in the root directory of your web server (e.g., htdocs in XAMPP).

3. Install Dependencies

• Navigate to the project directory in the terminal and run:

```
bash
Copy code
composer install
```

4. Create and Import Database

- Open phpMyAdmin (http://localhost/phpmyadmin).
- Create a new database (e.g., exam_portal_db).
- Import the SQL file into the database.

5. Configure Environment Settings

• Rename .env.example to .env and update the database credentials.

6. Generate Application Key

• In the terminal, run:

bash Copy code php artisan key:generate

7. Access the Application

- o User Interface URL: http://localhost/exam_portal/public/
- o Admin Panel URL: http://localhost/exam_portal/public/admin

8. Login Credentials

- Admin Login:
 - Username: Admin
 - Password: Admin

UI/UX Wireframes

- Homepage: Overview of upcoming exams and registration/login links.
- **Exam Dashboard:** List of available exams with schedules and access buttons.
- **Exam Page:** Real-time timer, questions, and navigation between questions.
- **Results Page:** Display of scores, feedback, and detailed performance metrics.
- Admin Dashboard: Overview of user registrations, exams, and key metrics.

Error Handling Guidelines

- 1. Validation Errors
 - Display user-friendly messages for form validation errors (e.g., "Please enter a valid email address").

2. Database Errors

• Show generic error messages and log detailed errors for debugging purposes.

3. 404 Errors

- Display a custom 404 error page indicating that the requested resource was not found.
- 4. **500 Internal Server Errors**
 - Display a generic error message and log detailed errors for further analysis.

Business Logic

1. User Registration

• Validate user input and securely store user credentials in the database.

2. Exam Management

- Allow administrators to create, update, and delete exams and associated questions.
- 3. Exam Taking Process
 - Implement functionality for students to take exams with real-time monitoring.

4. **Reporting**

• Generate and display reports based on student performance and exam metrics.

Session Management

1. Session Initialization

- Use Laravel's built-in session management to handle user logins.
- 2. Session Protection
 - Ensure that only authenticated users can access admin functionalities.
- 3. Access Control
 - Define user roles and permissions to restrict access to specific features.

PHP Functions

1. OnLoad Functions

• Implement functions to optimize the performance of Laravel routes and controllers.

2. Database Connectivity

• Use Laravel's Eloquent ORM for secure database interactions.

3. Utility Functions

• Implement utility functions for common tasks such as input validation and data processing.