Documentations of Online job portal + admin panel (php Laravel)

Project Name: Online Job 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 Job Portal** is a web application developed using Laravel that connects job seekers with employers. It allows users to create profiles, apply for jobs, and manage applications, while administrators can oversee job postings, user accounts, and site activities.

Features

User Interface:

1. **User Registration and Login**
   o Job seekers can register using their email and password.
   o Employers can create accounts to post job openings.
2. **Profile Management**
   o Users can create and update their profiles, including resumes and personal information.
3. **Job Search**
   o Users can search for jobs based on various filters (e.g., location, category, salary).
4. **Job Applications**
   o Job seekers can apply for jobs directly through the portal and track their application status.
5. **Saved Jobs**
   o Users can save jobs for later viewing and easy application.

Admin Panel:

1. **Admin Login**

- o Secure login for administrators to manage the job portal.
2. **Manage Job Postings**
   - o Add, edit, or delete job postings from various employers.
   - o Approve or reject job listings before they go live.
3. **User Management**
   - o View and manage job seeker and employer accounts.
   - o Approve or deactivate user registrations.
4. **Application Management**
   - o View and manage applications submitted by job seekers.
   - o Communicate with users regarding their application status.
5. **Generate Reports**
   - o Generate reports on job postings, applications, and user activity.

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**
   - o Download the zip file containing the source code of the Online Job Portal.
2. **Unzip and Place in Root Directory**
   - o 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**
   - o Navigate to the project directory in the terminal and run:

     ```bash
     Copy code
     composer install
     ```

4. **Create and Import Database**
   - o Open phpMyAdmin (`http://localhost/phpmyadmin`).
   - o Create a new database (e.g., `job_portal_db`).
   - o Import the SQL file into the database.
5. **Configure Environment Settings**
   - o Rename `.env.example` to `.env` and update the database credentials.
6. **Generate Application Key**
   - o In the terminal, run:

```bash
bash
Copy code
php artisan key:generate
```

7. **Access the Application**
   - User Interface URL: `http://localhost/job_portal/public/`
   - Admin Panel URL: `http://localhost/job_portal/public/admin`
8. **Login Credentials**
   - Admin Login:
     - **Username:** Admin
     - **Password:** Admin

UI/UX Wireframes

- **Homepage:** Overview of job listings, search functionality, and user registration/login links.
- **Job Listings Page:** Display of available jobs with filtering options.
- **Job Details Page:** Detailed view of the job with application buttons.
- **User Dashboard:** Overview of user applications, saved jobs, and profile management.
- **Admin Dashboard:** Overview of job postings, applications, 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. **Job Management**
   - Allow administrators to create, update, and delete job postings.

3. **Application Process**
   - o Handle the entire application lifecycle, including tracking and notifications.
4. **Reporting**
   - o Generate and display reports based on job postings and user activity.

Session Management

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

PHP Functions

1. **OnLoad Functions**
   - o Implement functions to optimize the performance of Laravel routes and controllers.
2. **Database Connectivity**
   - o Use Laravel's Eloquent ORM for secure database interactions.
3. **Utility Functions**
   - o Implement utility functions for common tasks such as input validation and data processing.