

Social Publishing Platform

Please implement a **Social Publishing Platform** using Laravel 10. The application should have the following features:

User Registration/Authentication:

- Users should be able to **register** an account and log in using Laravel's built-in authentication system.
- Only authenticated users can use the system(create, view, and interact with posts).

Post Creation and Management:

- Users should be able to **Create, Read, Update, and Delete** their own posts.
- Each post should include:
 - **Title**
 - **Content** (text body)
 - **Creation Date and Time**
 - The **Author's Name** (auto-assigned based on the logged-in user).
- Posts must be assigned to one or more **predefined categories** (many-to-many relationship). For example, categories might include "Technology," "Health," or "Lifestyle."
- Only the author of a post can **edit or delete** their own post.

Categories:

- Categories are **predefined** and cannot be edited or added by users.
- Posts can belong to multiple categories (Many-to-Many relationship), and categories can include multiple posts.
- Users should be able to filter posts in the main feed by category.

Comments:

- Users should be able to **comment** on posts.
- Only registered users can add or delete their own comments.
- Comments must display the author's name and the date/time they were posted.

User Profile:

- Each user should have a **profile page** where their own posts are displayed.
- A profile page should only show posts created by the profile owner, ordered by the most recent.
- Visitors to the profile should be able to see the author's posts along with the assigned categories.

Main Feed:

- The main feed should display posts from all users, including the post's title, content preview, author's name, categories, and the number of comments.
- Users should be able to filter posts by category and see posts from all categories by default.

Search Functionality:

- Users should be able to search posts by **keywords** found in the post's title or content.
- The search results should include posts that match the search term in their title or body content.

Additional Requirements:

- **Input Validation:** All user inputs should be validated before processing. Ensure posts have a valid title, content, and assigned categories.
 - **Middleware for Access Control:** Use Laravel's Middleware to ensure:
 - Only the author of a post can **edit or delete** their post.
 - Only the author of a comment can delete their comment.
 - **XSS Prevention:** All user inputs and displayed data should be sanitized to prevent cross-site scripting (XSS) attacks.
-

To Achieve This:

- Follow **Laravel's MVC design pattern** for organizing the code.
 - Use **Eloquent ORM** to manage the relationships between users, posts, comments, and categories.
 - Utilize the **Blade templating engine** for rendering views and integrate **Tailwind CSS** for frontend styling, though you may use any other CSS framework of your choice if preferred.
-

Deliverables:

Once you have completed the task, provide the entire codebase, including:

- **Migrations** for setting up the database tables for users, posts, comments, and categories.
 - **Seeders** for preloading sample data into the database, including users, categories, posts, and comments to test functionality and integration.
 - **README.md** with installation and usage instructions, including:
 - Instructions to clone the repository.
 - Steps for installing dependencies.
 - Directions for setting up the environment.
 - Commands to run database migrations and seeders.
 - Instructions for serving the application and accessing its features.
-

Key Features:

- Users can create posts with titles, content, and assign multiple predefined categories.
- Users can view and filter posts by category in the main feed.
- Users have profiles displaying their own posts.
- Users can post comments on others' posts and delete their own comments.