

Project Idea

**Semester: Fall-2024**

**Student Name: Jannatul Jeor Bushra**

**Student ID: 0242310005341291**

**Batch: 40 Section: C**

**Course Code: SE221 Course Name: Object Oriented Design**

**Course Teacher Name: Akash Gosh**

**Designation: Lecturer**

**Submission Date: 08.12.24**

Personal Blog Application

**Project Title:**

Personal Blog Application using Java Spring Boot Named “**EchoWrite**”

**Overview:**

This project aims to develop a Personal Blog Application using the Java Spring Boot framework. The system will provide different functionalities for two types of users: Admin and Regular User. The admin will have control over blog posts, allowing them to create, edit, and delete posts, while the Regular User will be able to view blog posts after logging in. Both Admin and Regular User can log out at any time.

**Objectives:**

* To build a personal blog application where the admin has full control over blog posts.
* To provide user authentication and authorization for accessing posts.
* To design a simple and responsive web interface for users to interact with the blog.
* To integrate the application with a database for storing posts and user data.

**Technologies Used:**

* **Backend:** Java Spring Boot
* **Frontend:** Thymeleaf, HTML, CSS, JavaScript
* **Database:** MySQL
* **Security:** Spring Security (for user authentication and role-based access)
* **Build Tools:** Maven

**Features:**

**Admin Features:**

* **Create Post:** Admin can create new blog posts, including title, content, and tags.
* **Edit Post:** Admin can edit existing posts.
* **Delete Post:** Admin can delete any post from the system.
* **View All Posts:** Admin can view a list of all the blog posts.

**User Features:**

* **Read Posts:** Users can read any published blog posts after logging in.
* **User Authentication:** Users must log in to access blog posts (using credentials stored in the database).
* **Register & Login**
* **User Logout:** Both Admin and Users can log out of the system.

**Authentication and Authorization:**

* Admin and regular users will have separate access rights based on their roles.
* Only the Admin can perform create, edit, and delete operations on posts.
* Regular users can only read posts.

**Database:**

* A database will store user credentials (username, password), blog posts (title, content, creation date), and user roles (admin/user).
* The posts will be stored in a posts table, and user information will be stored in a users table.

**System Architecture:**

* **Spring Boot Backend:** Provides REST APIs for handling business logic, including user authentication, CRUD operations for posts, and role management.
* **Spring Security:** To handle user authentication, session management, and role-based authorization.
* **Database Layer:** Uses JPA and Hibernate for interacting with the MySQL/H2 database.
* **Frontend:** A simple user interface for reading posts and interacting with the system (using Thymeleaf or other templating engines).

**Database Schema:**

**Users Table:**

* id(Primary Key)
* address
* email
* full\_name
* password
* role (Admin/User)

**Posts Table:**

* id (Primary Key)
* post\_title
* post\_description
* content
* user\_id (Foreign Key to Users table)
* image\_name

**Project Phases:**

**Phase 1: Requirements Gathering and Design**

* Identify project requirements.
* Design the database schema.
* Create wireframes or mockups for the frontend interface.

**Phase 2: Backend Development**

* Implement user authentication with Spring Security.
* Develop CRUD operations for blog posts.
* Implement user role management (Admin and User).

**Phase 3: Frontend Development**

* Design the user interface for login, post reading, and displaying posts.
* Implement forms for creating, editing, and deleting posts (for admin).

**Phase 4: Testing**

* Unit testing for backend functionality (e.g., CRUD operations, security).
* Integration testing for user authentication and database integration.
* User acceptance testing (UAT) for the overall functionality of the application.

**Phase 5: Deployment**

* Deploy the application on a local server .
* Ensure proper configuration of the database and server settings.

**Expected Deliverables:**

* A fully functional personal blog application where users can read posts after logging in and admins can manage blog posts.
* A clean, documented codebase using Java Spring Boot.
* Database schema documentation.
* User and Admin manual (explaining how to use the system).
* Final project report detailing the development process, challenges, and solutions.

**Future Scope:**

The most required future scope could be built an encrypted Registration feature first.

1. **User Features:**
   * **Commenting & Likes:** Enable users to comment on and like posts.
   * **Profile Management:** Users can manage their profiles and bios.
   * **Subscriptions:** Users can subscribe to categories or authors for notifications.
2. **Admin Enhancements:**
   * **Category & Tag Management:** Admin can organize posts by categories and tags.
   * **Scheduling Posts:** Allow admin to schedule posts for future publishing.
   * **Analytics Dashboard:** Admin can monitor post views and user interactions.
3. **Security:**
   * **Two-Factor Authentication (2FA):** Add an extra layer of security for users and admins.
   * **Password Recovery:** Implement a password reset feature.
4. **Performance & Scalability:**
   * **Caching:** Use caching to speed up post loading and reduce database load.
   * **Scaling:** Implement load balancing for better performance.
5. **Mobile Application:**
   * **Responsive Design:** Make the blog mobile-friendly.
   * **Mobile App:** Develop native Android and iOS apps.
6. **Multilingual Support:**
   * Add internationalization (i18n) for multiple languages.
7. **Third-Party Integrations:**
   * **Social Sharing:** Allow sharing posts on social media.
   * **Email Notifications:** Notify users of new posts or comments.
8. **Cloud Integration:**
   * Store media (images/videos) in cloud services like AWS or Google Cloud.

**Conclusion:**

This Personal Blog Application aims to demonstrate the use of the Spring Boot framework for creating a user-authenticated blog system with role-based access control. The project will provide an opportunity to gain hands-on experience with backend development, security, and database management.