Real-Time Web Application for Blogging

Name: Ankur Aggarwal
Net ID: aa10336

Note: This is a solo project

This application is a dynamic, real-time blogging platform using Spring Boot 3, Spring MVC, Thymeleaf, Spring Security 6, and Spring Data JPA with a MySQL database. It's designed for immediate content updates and interactive user experiences.

Key Features

- **User Authentication**: Secure registration and login using Spring Security 6.
- **Blog Post Management**: Users can create, read, update, and delete blog posts in real-time, managed efficiently via Spring Data JPA.
- **Comment System**: Interactive platform for users to comment on posts, enhancing community engagement.
- **Instant Updates**: Real-time reflection of new posts and comments, eliminating the need for page reloads.
- **Search Functionality**: Easy navigation and search options for finding specific blog content.
- **Responsive UI**: User-friendly interface designed with Thymeleaf, adaptable to various devices.

This application is a comprehensive solution for blogging, emphasizing user interaction, content management, and security. It's a showcase of integrating various Spring technologies to create a responsive and efficient web application.

Administrative Access and Database Connection

Admin Access to the Application
For administrative access to the blogging platform:

- **Username**: admin@abcd.com
- **Password**: admin

This admin account provides elevated privileges for managing the application, including user management and content moderation.

Amazon RDS Database Connection
The application's database is hosted on Amazon RDS, designed for scalability and reliability. Contrary to typical configurations, this database is set up to accept connections from any user who possesses the below-mentioned credentials.

Connection Details:

- **Username**: admin
- **Password**: 6beu2GVwoTNugoD2exPa
- **Endpoint**: blog-application.chwwtfwl4m1j.us-east-2.rds.amazonaws.com

By using these credentials, users can connect to the database, facilitating direct data management and interaction.

Installation of the application

Video timestamps

1. IntelliJ IDEA Installation (0:00 to 1:45): Demonstrates how to install and set up IntelliJ IDEA for Java development.
2. Eclipse Installation (1:45 to 7:06): Guides through the installation process of the application on Eclipse, including setting up required extensions and dependencies.
3. Blog Application Demo (7:07 to 8:25): Showcases the blog application's features and user interface.

Video link
https://www.dropbox.com/scl/fi/a1tm29spbd31hggj4qux4/Video-demo.mov?rlkey=gjkds8qza7sm9g3ggxkn05uz7&dl=0

I have also uploaded this video to Brightspace

These instructions will get you a copy of the project up and running on your local machine for development and testing purposes.

Prerequisites

- Java JDK 11 or later
- Maven 3.6 or later
- IntelliJ IDEA (Recommended) or Eclipse IDE (2023-12 version)

Running the Application in IntelliJ IDEA (Highly recommended)

1. Open IntelliJ IDEA.
2. Clone the repository or open the downloaded project.
3. Navigate to the main application file: **src/main/java/com/ankurnow/springbootblogwebapp/SpringbootBlogWebappApplication.java**.
4. Right-click on the file and select 'Run SpringbootBlogWebappApplication'.

Setting Up and Running the Application in Eclipse

Downloading the Project

1. Download the project source code from NYU Brightspace.
2. Extract the downloaded project to a known location on your computer.

Eclipse Setup

1. Make sure you have Eclipse IDE version 2023-12 installed.
2. Install the Spring Tools 4 (aka Spring Tool Suite 4) extension for Eclipse:
    - Go to 'Help' > 'Eclipse Marketplace'.
    - Search for 'Spring Tools 4' and click 'Install'.
    - Follow the prompts to complete the installation.
3. Restart Eclipse after installing the extension.

Installing Lombok

Lombok is a required dependency for this project:

1. Download the Lombok jar (version 1.18.30) from Maven Central.
2. Run the jar file using **java -jar lombok-1.18.30.jar**.
3. In the installer, either auto-detect your Eclipse installations or manually specify their location.
4. Select your installations and press the 'Install/Update' button.
5. Restart Eclipse after the installation to ensure Lombok is correctly configured.

Project Setup in Eclipse

1. Open Eclipse and import the downloaded project:
    - Go to 'File' > 'Import' > 'Existing Maven Projects'.
    - Browse to the location where you extracted the project.
    - Ensure that the **pom.xml** file and the **src** folder are selected for import.
2. Replace the existing **pom.xml** and **src** folder in your Eclipse environment with the ones from the downloaded project.
3. Ensure that Maven dependencies are correctly resolved. Right-click on the project and select 'Maven' > 'Update Project'.

Running the Application

1. Navigate to **src/main/java/com/ankurnow/springbootblogwebapp/SpringbootBlogWebappApplication.java** in the Project Explorer.
2. Right-click on the file and select 'Run as' > 'Java Application' to start the application.