

Project Synopsis

Project Title: **Blog & Content Publishing Platform**

1. Introduction

In today's digital era, blogs serve as a powerful medium for sharing knowledge, opinions, and information. Individuals and organizations use

blogs to publish articles, tutorials, and updates to a wide audience. Managing blog content manually, however, can be time-consuming and

inefficient, especially when multiple users, posts, likes, and comments are involved.

The **Blog & Content Publishing Platform** is a web-based system designed to allow users to create, publish, and manage blog posts in a structured and organized manner. The platform provides role-based access, enabling admins to manage content while users publish and interact with blogs.

Key Features:

- Easy-to-use interface
- Secure and reliable operations
- Scalable for future enhancements
- Suitable for small to medium-sized blogging platforms

2. Problem Statement

Existing Challenges:

- Manual blog management is time-consuming
- No centralized system for managing users and posts
- Difficulty in tracking likes, comments, and post status
- Lack of admin control over published content
- Poor organization of blog data

Core Problems Addressed:

- Lack of proper content publishing workflow
- No role-based access (Admin/User)
- Absence of a structured database for blogs
- Difficulty managing user interactions (likes, comments)

3. Project Objectives

- Allow users to create and publish blog posts
- Enable admins to manage blogs and users
- Store blogs, users, likes, and comments in a structured database
- Provide a simple, user-friendly interface
- Ensure data security and controlled access
- Facilitate future expansion (comments, categories, analytics)

4. System Overview

The Blog & Content Publishing Platform is a web-based application developed using advanced Java technologies.

High-Level Workflow:

1. User Registration / Login

2. User creates a blog post
3. Blog is stored in the database
4. Admin reviews and manages blogs
5. Users can view, like, and comment on blogs

5. Users of the System

5.1 Admin

- Manage users
- Approve or delete blogs
- View all blog posts
- Monitor user activity

5.2 User

- Register and login
- Create blog posts
- View blogs
- Like and comment on blogs
- View own blog history

6. Functional Requirements

6.1 User Management

- User registration
- User login/logout
- Role-based access control (Admin/User)

6.2 Blog Management

- Create new blog posts
- Edit own blog posts
- Delete own blog posts
- View all published blogs

6.3 Like Management

- Users can like a blog
- One user can like a blog only once
- Total likes displayed per blog

6.4 Comment Management (Optional / Future)

- Users can comment on blogs
- Admin can remove inappropriate comments

7. Non-Functional Requirements

- Easy-to-use UI
- Fast response time
- Secure data storage
- Proper error handling
- Scalable architecture

8. Technology Stack

Frontend:

- HTML
- CSS
- JavaScript
- (Optional) Thymeleaf / React (if UI is enhanced later)

Backend:

- Java (Spring Boot)
- Spring Web (REST APIs)
- Spring Data JPA (Hibernate ORM)

Database:

- PostgreSQL

Server:

- Embedded Apache Tomcat (Spring Boot)

API Testing Tool:

- Postman

Build & Dependency Management:

- Maven

9. Database Design

9.1 Users Table:

Column Name	Data Type	Description
user_id	INT (PK, AUTO_INCREMENT)	Unique user ID
name	VARCHAR(100)	User name
email	VARCHAR(100)	User email
password	VARCHAR(255)	Encrypted password
role	VARCHAR(20)	Admin / User
created_at	TIMESTAMP	Account creation time

9.2 Blogs Table:

Column Name	Data Type	Description
blog_id	INT (PK, AUTO_INCREMENT)	Unique blog ID
user_id	INT (FK)	Author of the blog
title	VARCHAR(200)	Blog title
content	TEXT	Blog content
status	VARCHAR(20)	Published / Draft
created_at	TIMESTAMP	Creation date
updated_at	TIMESTAMP	Last updated date

9.3 Likes Table:

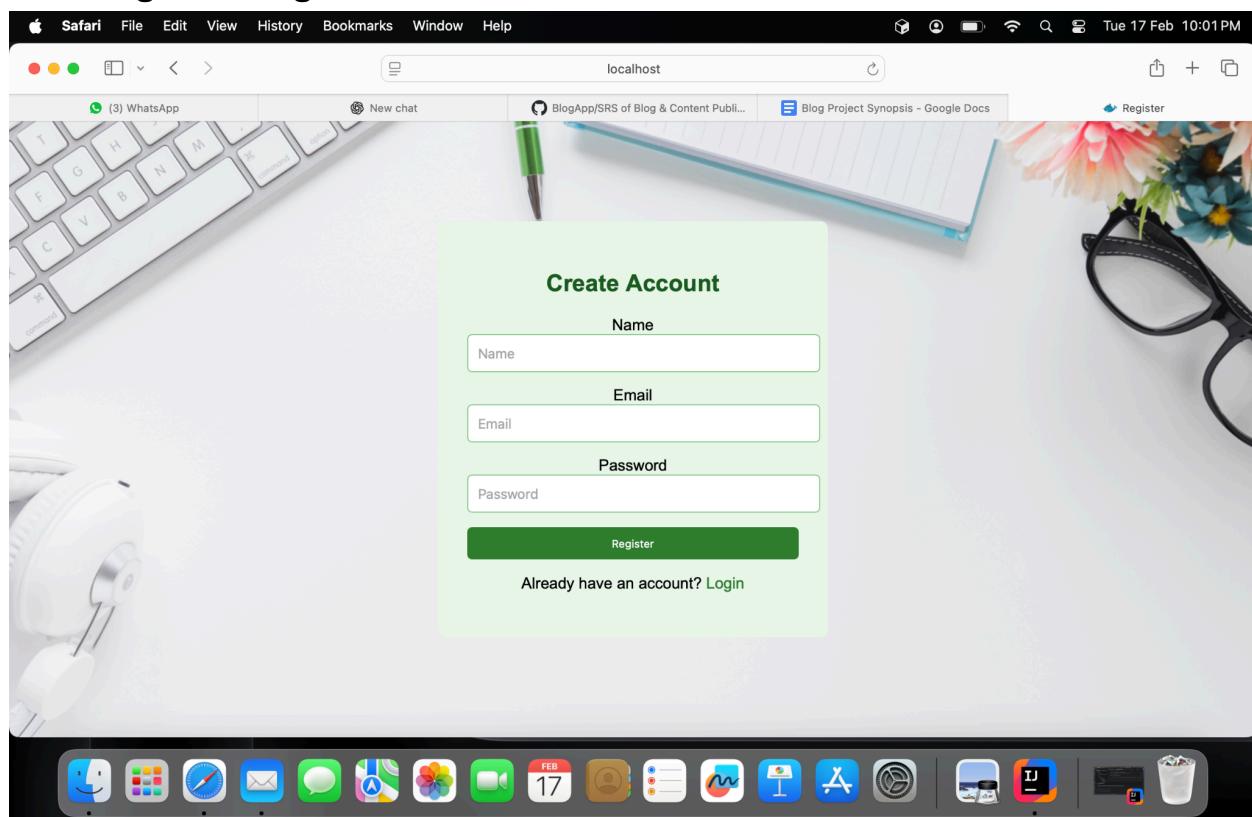
Column Name	Data Type	Description
like_id	INT (PK, AUTO_INCREMENT)	Unique like ID
blog_id	INT (FK)	Blog ID
user_id	INT (FK)	User who liked the blog
liked_at	TIMESTAMP	Like timestamp

9.4 Comments Table:

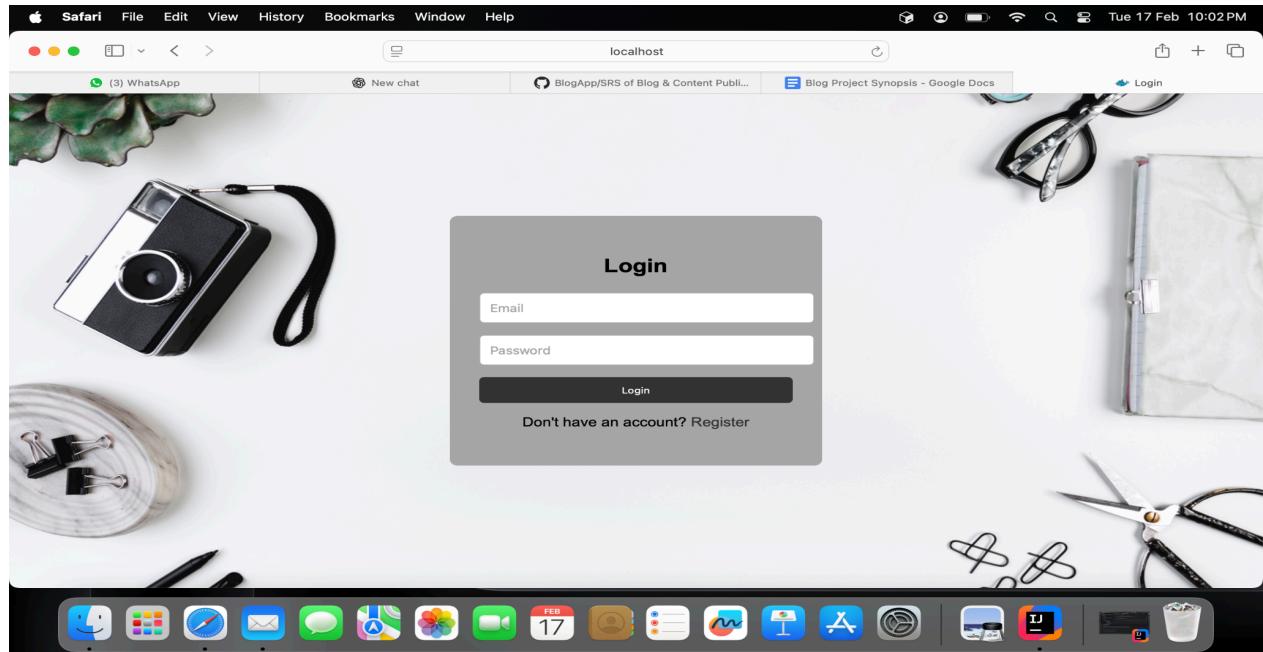
Column Name	Data Type	Description
comment_id	INT (PK, AUTO_INCREMENT)	Unique comment ID
blog_id	INT (FK)	Blog ID
user_id	INT (FK)	Commented user
comment_text	TEXT	Comment content
commented_at	TIMESTAMP	Comment timestamp

10:Frontend Module:

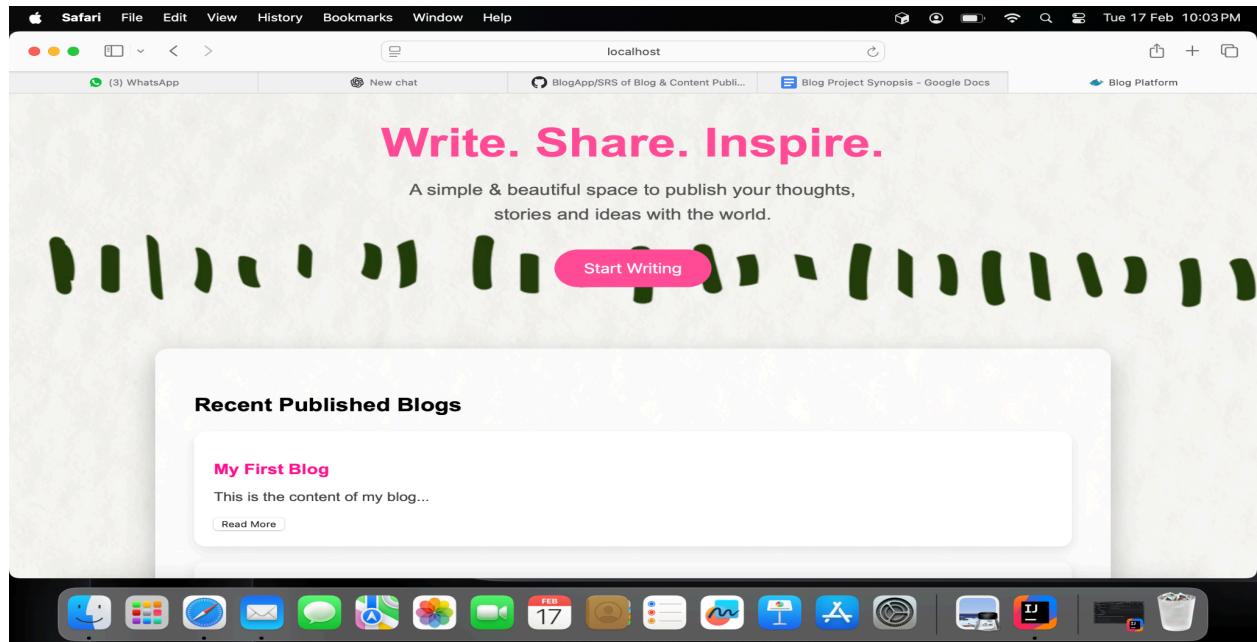
10.1 Register Page:



10.2 Login Page:



10.3 Home Page (Index Page):



10.4 Blog creation Page:

The screenshot shows a web browser window in Safari on a Mac OS X desktop. The title bar reads "Safari" and the address bar says "localhost". The main content area displays a "Create New Blog" form. At the top of the form is a section titled "Role of Data Science in Modern World". Below this is a text area containing the following text: "Data Science helps companies understand large amounts of data to make better decisions. It is used in healthcare, finance, marketing, and even social media. With tools like Python and machine learning, data scientists can predict trends and improve business strategies." At the bottom of the form are two buttons: a pink "Publish Blog" button and a grey "Back to Blogs" button. The Mac OS X Dock at the bottom of the screen is visible, showing icons for various applications like Mail, Safari, and Finder.

Create New Blog

Role of Data Science in Modern World

Data Science helps companies understand large amounts of data to make better decisions. It is used in healthcare, finance, marketing, and even social media. With tools like Python and machine learning, data scientists can predict trends and improve business strategies.

Publish Blog

Back to Blogs

10.5 Blog Page (With Author name and other details):

The screenshot shows a blog post titled "Role of Data Science in Modern World" by Ayesha Shaikh, posted on 17/2/2026. The post discusses how Data Science helps companies understand large amounts of data to make better decisions, mentioning its applications in healthcare, finance, marketing, and social media. It notes that Python and machine learning tools are used for trend prediction and business strategy improvement. Below the post is a "Like" button. The browser's address bar shows "localhost". The top navigation bar includes "Safari", "File", "Edit", "View", "History", "Bookmarks", "Window", and "Help". The status bar at the bottom indicates "Tue 17 Feb 10:04 PM".

10.6 Admin dashboard page:

The screenshot shows an admin dashboard titled "Welcome, Admin!". It features a sidebar on the left labeled "Admin Panel" with "Dashboard" selected, and links for "Users", "Blogs", and "Comments". The main area displays three summary boxes: "Total Users" (150), "Total Blogs" (45), and "Pending Comments" (12). The browser's address bar shows "localhost". The top navigation bar includes "Safari", "File", "Edit", "View", "History", "Bookmarks", "Window", and "Help". The status bar at the bottom indicates "Tue 17 Feb 10:05 PM".

10.7 Admin Users Page:

The screenshot shows a web browser window in Safari with the URL `localhost`. The title bar indicates it's Tuesday, February 17, at 10:05 PM. The browser has three tabs open: WhatsApp, Blog Topics Ideas, and Blog Project Synopsis - Google Docs. The main content area is titled "Manage Users" and displays a table of user data. The table has columns for ID, Name, Email, Role, and Actions. The data shows three users: Alice (ID 1, USER), Bob (ID 2, ADMIN), and Charlie (ID 3, USER). Each row has "Edit" and "Delete" links in the Actions column. On the left, a sidebar titled "Admin Panel" lists Dashboard, Users (which is selected and highlighted in pink), Blogs, and Comments.

ID	Name	Email	Role	Actions
1	Alice	alice@mail.com	USER	Edit Delete
2	Bob	bob@mail.com	ADMIN	Edit Delete
3	Charlie	charlie@mail.com	USER	Edit Delete

10.8 Admin Blogs page:

The screenshot shows a web browser window in Safari with the URL `localhost`. The title bar indicates it's Tuesday, February 17, at 10:05 PM. The browser has three tabs open: WhatsApp, Blog Topics Ideas, and Blog Project Synopsis - Google Docs. The main content area is titled "Blog Management" and displays a table of blog posts. The table has columns for ID, Title, Author, Status, and Action. The data shows three blogs: "My First Blog" by Alice (Status PUBLISHED), "Spring Boot Tips" by Bob (Status DRAFT), and "JavaScript Tricks" by Charlie (Status PUBLISHED). Each row has "Edit" and "Delete" links in the Action column. At the top right of the content area, there are links for Dashboard, Users, and Comments. On the left, a sidebar titled "Blog Management" lists Blogs (which is selected and highlighted in blue).

ID	Title	Author	Status	Action
1	My First Blog	Alice	PUBLISHED	Edit Delete
2	Spring Boot Tips	Bob	DRAFT	Edit Delete
3	JavaScript Tricks	Charlie	PUBLISHED	Edit Delete

10.9 Comments Page:

ID	Blog	User	Comment	Action
1	My First Blog	Alice	Great post!	Delete
2	Spring Boot Tips	Bob	Very helpful.	Delete
3	JavaScript Tricks	Charlie	Nice article!	Delete

11. Workflow Summary

1. User registers and logs in
2. User creates a blog post
3. Blog is saved in the database
4. Admin manages blog visibility
5. Users view blogs
6. Users like blogs
7. Like count is updated

12. Security Considerations

- Passwords stored in encrypted format
- Session-based authentication
- Role-based access control
- SQL Injection prevention using Prepared Statements

13. Future Enhancements

- Category and tags system
- Search functionality
- Blog analytics
- Email notifications
- Admin dashboard
- User profile management

14. Conclusion

The **Blog & Content Publishing Platform** offers a simple, secure, and

effective solution for managing blog content and users. It demonstrates

the use of Spring Boot, database management, and web technologies in

a practical, real-world application.

The system is intuitive, scalable, and suitable for both academic projects

and real-world blogging platforms.