A Secure And Scalable Blogging Platform Using Web Technologies

Mini Project

Submitted in partial fulfillment of the requirement for the award of Degree of Bachelor of Technology in Computer Science and Engineering (Data Science)
Submitted to:

RAJIV GANDHI PROUDYOGIKI VISHWAVIDYALAYA, BHOPAL (M.P.)

Submitted by:

Aditya Pratap Singh Sisodiya - 0808DS231008
Ansh Dubey – 0808DS231018
Harsh Sahu – 0808DS231052

Under the Supervision of:
Dr. Kalyani Tiwari

IPS ACADEMY,

INDORE INSTITUTE OF

ENGINEERING & SCIENCE

(A UGC Autonomous Institute Affiliated to RGPV)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

(DATA SCIENCE)

Session:2025-26

IPS ACADEMY, INDORE

INSTITUTE OF ENGINEERING & SCIENCE

(A UGC Autonomous Institute Affiliated to RGPV)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

(Data Science)

2025-26

Mini Project

"BLOGZE : A Secure And Scalable Blogging Platform Using Web Technologies"

For the partial fulfillment for the award of the ***Bachelor of Technology in Computer Science and Engineering (Data Science)*** *Degree by Rajiv Gandhi Proudyogiki Vishwavidhyalaya, Bhopal.*

Guided by: -

Dr. Girish Gagote

Submitted by:-

Aditya Pratap Singh Sisodiya - 0808DS231008
Ansh Dubey - 0808DS231018
Harsh Sahu – 0808DS231052

IPS ACADEMY,

INDORE INSTITUTE OF

ENGINEERING & SCIENCE

(A UGC Autonomous Institute Affiliated to RGPV)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

(Data Science)

2025-26

CERTIFICATE

This is to certify that Mini Project entitled

“BLOGZE : A Secure And Scalable Blogging Platform Using Web Technologies ”

Has been successfully completed by the following students
Aditya Pratap Singh Sisodiya , Ansh Dubey , Harsh Sahu

In partial fulfillment for the award of the **Bachelor of Technology in Computer Science and Engineering (Data Science***)* *Degree by Rajiv Gandhi Proudyogiki Vishwavidhyalaya, Bhopal during the academic year2025-26 under our guidance.*

Dr. Girish Gagote	Dr. Kalyani Tiwari	Dr. Neeraj Shrivastava
Assistant Professor	Branch Coordinator	HOD

Dr. Archana Keerti Chowdhary
Principal

Acknowledgement

We would like to express my heartfelt thanks to our guide, **Dr. Girish Gagote ,** CSE(DS), for her guidance, support, and encouragement during the course of our study for B.Tech. (CSE (Data Science) at IPS Academy, Institute of Engineering & Science, Indore. Without her endless effort, knowledge, patience, and answers to my numerous questions, this Project would have never been possible. It has been great honor and pleasure for me to do Project under her supervision.

Our gratitude will not be complete without mention of **Dr. Archana Keerti Chowdhary, Principal, IPS Academy, Institute of Engineering & Science, Dr. Neeraj Shrivastava, HOD CSE, IPS Academy, Institute of Engineering & Science and Dr. Kalyani Tiwari, Branch Coordinator CSE(DS), IPS Academy, Institute of Engineering & Science** for the encouragement and giving me the opportunity for this project work.

We also thank my friends who have spread their valuable time for discussion/suggestion on theoretical aspects of this report. We want to acknowledge the contribution of our parents and our family members, for their constant motivation and inspiration.

Finally We thank the almighty God who has been my guardian and a source of strength and hope in this period.

Aditya Pratap Singh Sisodiya – 0808DS231008
Ansh Dubey – 0808DS231018
Harsh Sahu – 0808DS231052

CONTENTS

LIST OF FIGURES

ABSTRACT

Blogze is a full-stack web application designed as a secure, scalable, and user-friendly blogging platform. Modern content-creation tools often suffer from complexity, performance issues, or lack of customization. Blogze addresses these challenges by providing a lightweight blogging solution built using HTML, CSS, JavaScript, and an SQL database, offering a seamless experience for writing, editing, publishing, and managing blog posts.

The system incorporates secure user authentication, session handling, password hashing, and role-based access, ensuring that only authorized users can create or modify content. The backend is designed around modular REST APIs, enabling scalability and easy integration with additional services. The SQL database provides structured data storage for users, blogs, and metadata.

The platform supports CRUD operations for blogs, responsive UI, and efficient rendering mechanisms for content. It is designed with extensibility in mind, allowing future integration of analytics dashboards, comment systems, media uploads, and AI-based SEO or content recommendations.

Overall, Blogze demonstrates the practical application of web engineering principles, secure programming practices, and full-stack development workflows. It provides a foundation for scalable content-management systems and showcases modern web-development architecture suitable for academic and real-world applications.

CHAPTER 1: INTRODUCTION

1.1 Overview

In the digital era, content creation and online publishing have become central to communication, learning, and information sharing. Individuals, professionals, and organizations rely heavily on blogging platforms to express ideas, educate audiences, market products, and share expertise. Over the years, blogging has evolved from simple text-based journals to complex platforms featuring real-time analytics, multimedia support, search engine optimization tools, and social engagement mechanisms.

Despite the availability of many commercial platforms, there is still a significant need for simple, secure, scalable, and customizable blogging systems that can be adapted for academic, personal, or organizational use. Many existing blogging solutions are overly complex, dependent on heavy content-management frameworks, or restricted by commercial limitations. Furthermore, most provide limited control over the backend, which prevents learners and developers from understanding or modifying their internal workflows.

Blogze addresses this gap by delivering a full-stack web application that demonstrates how a modern blogging system is designed, implemented, and deployed using core web technologies: HTML, CSS, JavaScript, and SQL. Blogze includes user authentication, secure login, blog creation and editing tools, structured content storage, and a responsive interface. It is engineered for modularity, making it easy for students, developers, and researchers to understand each component of the platform clearly.

Blogze blends principles from UI/UX design, database engineering, web security, and RESTful API architecture to create a reliable content-publishing system. It showcases how foundational technologies can be combined efficiently to build scalable web platforms without relying on heavy frameworks. This project demonstrates the essential components of a professional-grade blogging system, including:

- User registration and authentication
- Role-based access and content authorization
- Post creation, editing, deletion, and management
- Secure storage and retrieval of structured data
- Responsive frontend layout
- Scalable backend logic

By designing Blogze from scratch, the project provides hands-on exposure to full-stack development while reinforcing best practices related to security, modularity, performance, and maintainability.

1.2 Literature Survey

A significant amount of research and development has been conducted in the area of blogging systems and content-management platforms. Leading platforms like WordPress, Medium, Blogger, Wix, and open-source CMS frameworks provide powerful publishing capabilities. However, these systems often use extensive libraries, plugins, or specialized programming languages, making them less accessible as learning tools for students seeking to understand fundamental web-development concepts

1.2.1 Traditional Blogging Platforms

WordPress, one of the most widely adopted blogging systems, uses PHP-based architecture and MySQL. Its robustness lies in extensibility through plugins and themes. However, this same extensibility also introduces vulnerabilities, plugin dependencies, slow performance under load, and heavy server-side resources.

Medium provides a minimalist writing interface but restricts user control over data, layouts, and backend logic. Its closed ecosystem makes customization impossible for educational or experimental purposes.

Blogger, owned by Google, provides free hosting but offers limited backend access and minimal database interaction opportunities. These limitations reduce its value for learners building their own systems.

These platforms demonstrate the importance of features such as:

- CRUD operations
- Rich text-support
- Secure user authentication
- Scalable architecture
- Database integration

Blogze draws inspiration from these systems but focuses on simplicity, accessibility, performance, and educational value, allowing students to observe and build every layer of the stack.

1.2.2 Web Technologies in Modern Blogging Systems

Modern blogging systems increasingly use JavaScript ecosystems due to their speed, scalability, and widespread adoption. JavaScript offers front-end interactivity and can also be used on the server side through platforms like Node.js.

CSS frameworks, responsive design principles, and client-side rendering have made blog interfaces more engaging. SQL databases remain a preferred choice for storing blog metadata due to their rigid structure, ACID compliance, and easy query management.

The industry trend emphasizes:

- REST APIs for communication
- Session management and token-based login
- Responsive layouts
- Security best practices

- Lightweight front-end rendering

Blogze aligns with these trends by using clean JavaScript logic, secure SQL storage, and modular components.

1.2.3 Security in Web Applications

Security remains a core challenge in blogging systems because they allow public interaction and require authentication. Common threats include:

- SQL injection
- Cross-site scripting (XSS)
- Broken authentication
- Data leakage
- Weak password storage
-

Research strongly highlights the importance of:

- Password hashing (bcrypt or similar)
- Input validation
- Session management
- Sanitizing user inputs
- Parameterized SQL queries

Blogze incorporates these mechanisms to ensure safe content creation and controlled access to the platform.

1.2.4 Scalable Architecture for Blogging Platforms

Academic literature and industry reports emphasize that content systems must be scalable and optimized for performance. Efficient caching, fast rendering, query optimization, and modular APIs significantly improve performance. Although Blogze is a lightweight academic model, it is architected with reusability and scalability in mind. The platform’s modular code structure enables easy enhancement, such as:

- Adding comment and like systems
- Integrating media uploads
- Implementing AI-driven SEO tools
- Deploying cloud-based database options

Thus, Blogze stands as a foundational system from which advanced blogging features can be built.

1.3 Summary of Chapter 1

This chapter establishes the foundation for the Blogze project by outlining the motivations, research background, technological landscape, and need for an accessible blogging platform. It introduces Blogze as a modern full-stack web application engineered for learning and practical use. The literature survey emphasizes the limitations of existing platforms, the importance of security, and the advantages of using lightweight web technologies.

.

CHAPTER 2: PROBLEM IDENTIFICATION & SCOPE

2.1 Problem Domain

The rapid growth of digital content creation has created a high demand for blogging platforms that are both powerful and easy to use. Despite the availability of many established blogging systems, several challenges still exist that affect students, developers, and small organizations when building or using blogging platforms.

Most existing blogging platforms such as WordPress, Blogger, and Medium are either too complex or too restrictive. Complex systems rely heavily on plugins, themes, and frameworks that hide core functionality from learners. On the other hand, highly restrictive platforms do not allow access to databases, backend logic, or internal workflows.

The major problems identified in the problem domain include:

- Lack of Backend Control
  Users cannot view or customize backend logic in commercial platforms, making learning difficult.
- Security Vulnerabilities
  Many popular blogging platforms suffer from frequent security flaws due to weak plugins, outdated dependencies, and unpatched vulnerabilities.
- Overdependence on Plugins
  Systems like WordPress require multiple plugins for basic features, increasing complexity and reducing performance.
- Scalability Issues
  Lightweight systems may fail under heavy load due to unoptimized architecture and database design.
- Steep Learning Curve
  Beginners struggle to understand the complete workflow because of heavy frameworks and CMS architectures.
- Limited Customization
  Many platforms limit layout control and backend access.
- Commercial Limitations
  Free versions restrict features such as custom domains, database access, and monetization tools.

Thus, the problem is to provide a secure, scalable, and understandable blogging platform that is simple for academic usage yet capable of real-world deployment.

2.2 Solution Domain

The solution proposed to overcome the identified challenges is the development of **BLOGZE**, a full-stack web-based blogging system that is based on simple, lightweight, and modular web technologies.

Blogze focuses on solving the problems by:

- Using **pure web technologies** (HTML, CSS, JavaScript, SQL)

- Avoiding heavy CMS frameworks
- Implementing secure authentication
- Providing structured backend logic
- Allowing full database interaction
- Offering a responsive user interface

The system allows users to:

- Register and log in securely
- Create, edit, and delete blog posts
- Store content in a structured relational database
- Retrieve and display posts dynamically
- Apply role-based access control
- Experience responsive UI design
- Learn full-stack development concepts

The solution is designed with flexibility so that future features such as comments, likes, analytics, and image uploads can be added easily.

2.3 Need & Scope

Need of the Project

The increasing dependency on digital platforms has created a need for:

- Educational-level full-stack projects
- Secure login systems
- Scalable database-driven platforms
- Hands-on development practice
- Understanding client-server architecture

Blogze fulfills the following academic and practical needs:

- Helps students learn web technologies in real-world scenarios
- Provides understanding of backend security concepts
- Encourages modular software development
- Promotes database interaction knowledge
- Improves UI/UX design capabilities

Scope of the Project

In Scope:

- User registration and login
- Blog post creation and modification
- Secure session handling
- Database storage
- Responsive user interface
- Input validation and security enforcement

Out of Scope (can be added in future):

- Social media integration
- Cloud deployment
- Monetization system
- SEO automation
- Multimedia uploads
- Comment system
- AI content optimization

CHAPTER 3: SOFTWARE ENGINEERING APPROACH

3.1 Software Model Used

Agile Development Model.

3.1.1 Description

The Agile Model follows an iterative and incremental development process. Development is divided into cycles called iterations where features are developed, tested, and reviewed continuously.

Each iteration includes:

- Planning
- Design
- Coding
- Testing
- Review

This model allows change during development and promotes continuous improvement.

3.1.2 Reason for Use

Agile is selected because:

- Allows flexibility
- Easy debugging
- Faster development
- User feedback incorporation
- Supports changes without failure
- Iterative improvement

3.2 Platform Specification

3.2.1 Hardware Specification

Component	Minimum Requirement
Processor	Intel i3 or above
RAM	4GB

Component	Minimum Requirement
Hard Disk	100GB
Display	14-inch monitor
Keyboard	Standard
Network	Internet Connection

3.2.2 Software Specifications

Software	Description
OS	Windows / Linux / macOS
Frontend	HTML, CSS, JavaScript
Backend	JavaScript (Node.js / PHP optional)
Database	MySQL / SQLite
Tools	VS Code
Browser	Chrome / Firefox

CHAPTER 4: DESIGNS

4.1 Use Case Diagram

Actors:

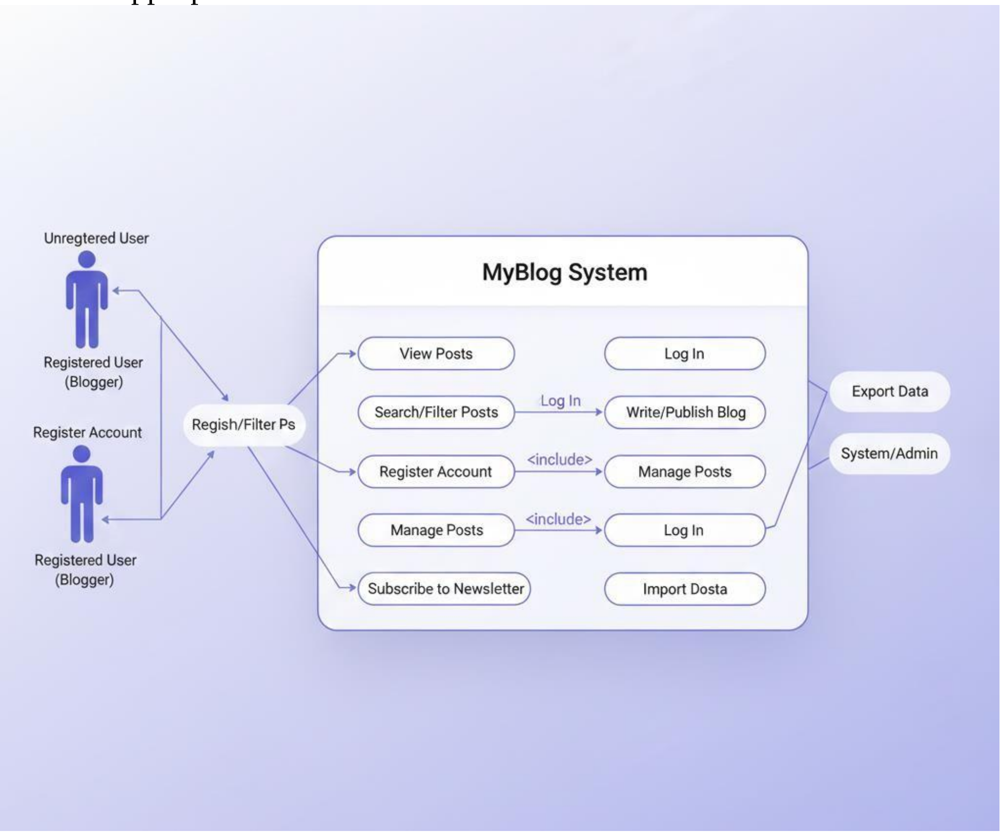
- Admin
- User

Use Cases:

- Register
- Login
- Write Blog
- Edit Blog
- Delete Blog
- View Blogs
- Logout

Admin Additional:

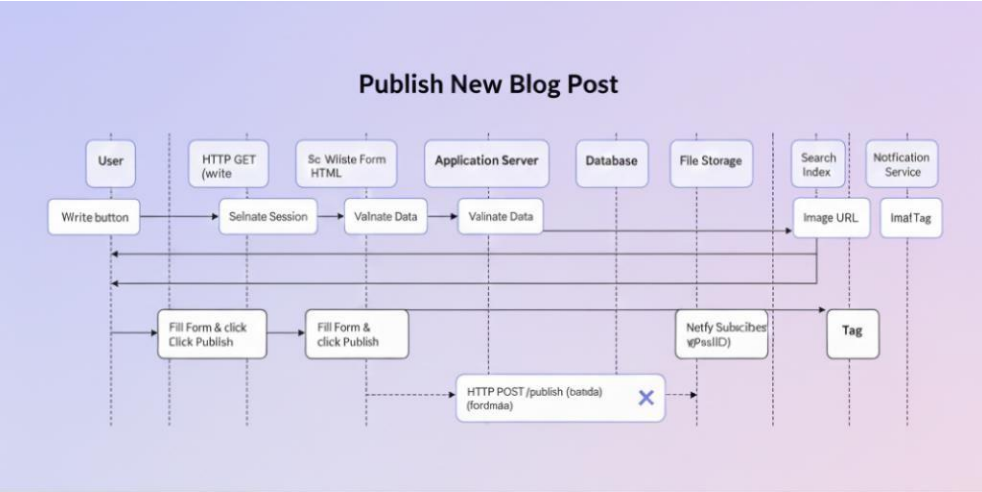
- Manage Users
- Delete Inappropriate Content



4.2 Sequence Diagram

Sample Flow – User Login:

1. User enters credentials
2. System validates
3. Database checks record
4. User authenticated
5. Dashboard loads

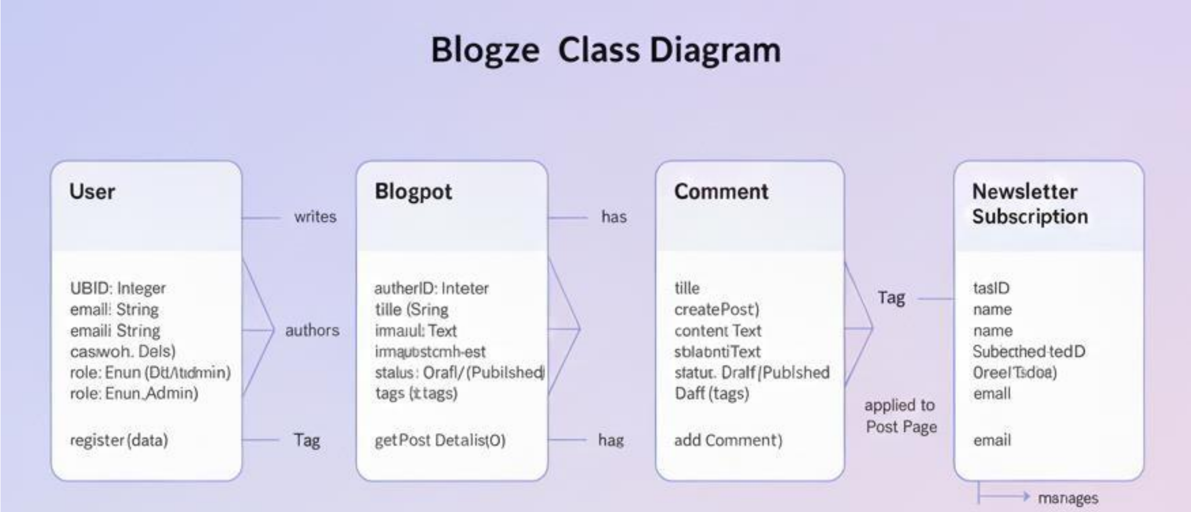


4.3 Class Diagram

Main Classes:

- User
- BlogPost
- Database
- Authentication
- SessionManager
- Admin

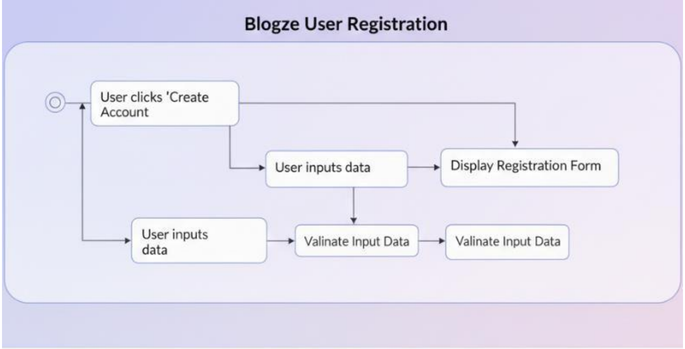
Relationships:
User → Writes → BlogPost
Authentication → Handles → User

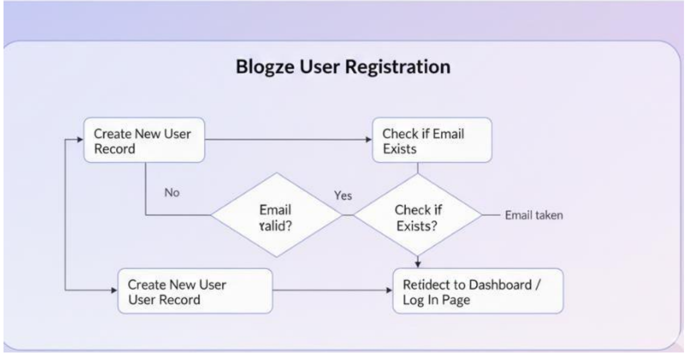


4.4 Activity Diagram

Example: Blog Posting Process

- Start
- Login
- Open Dashboard
- Write Blog
- Save Draft
- Publish
- End





CHAPTER 5: IMPLEMENTATION PHASE

5.1 Language Used & its Characteristics

HTML

- Used for structure
- Easy to learn
- Lightweight

CSS

- Used for styling
- Improves UI
- Responsive design

JavaScript

- Executes client-side logic
- Handles APIs
- Manages authentication

SQL

- Stores data
- Ensures consistency
- Fast queries

5.2 Modules Detail

The Blogze application is structured into several interconnected modules to manage user access, content creation, and display. The core functionality is divided into four primary modules: Authentication, Content Management, Public Interface, and Utility.

1. Authentication Module

This module handles all processes related to user identification and access control, ensuring only registered users can create content.

- Key Functions:
  - User Registration: Allows new users to create an account by providing basic details.
  - User Login/Logout: Manages session creation and termination for registered users.

2. Content Management Module

This is the core writer's dashboard, enabling registered users (Bloggers) to create, edit, and publish blog posts.

- Key Functions:
  - Post Creation: Provides a dedicated interface for drafting new blog posts.
  - Post Management: (Implied by the system's nature) Allows for viewing, editing, and deleting existing published posts.
  - Data Export/Import: Features for users to manage their blog data outside the application**.**

3. Public Interface Module

This module is responsible for displaying the application's main content, serving both registered and unregistered visitors.

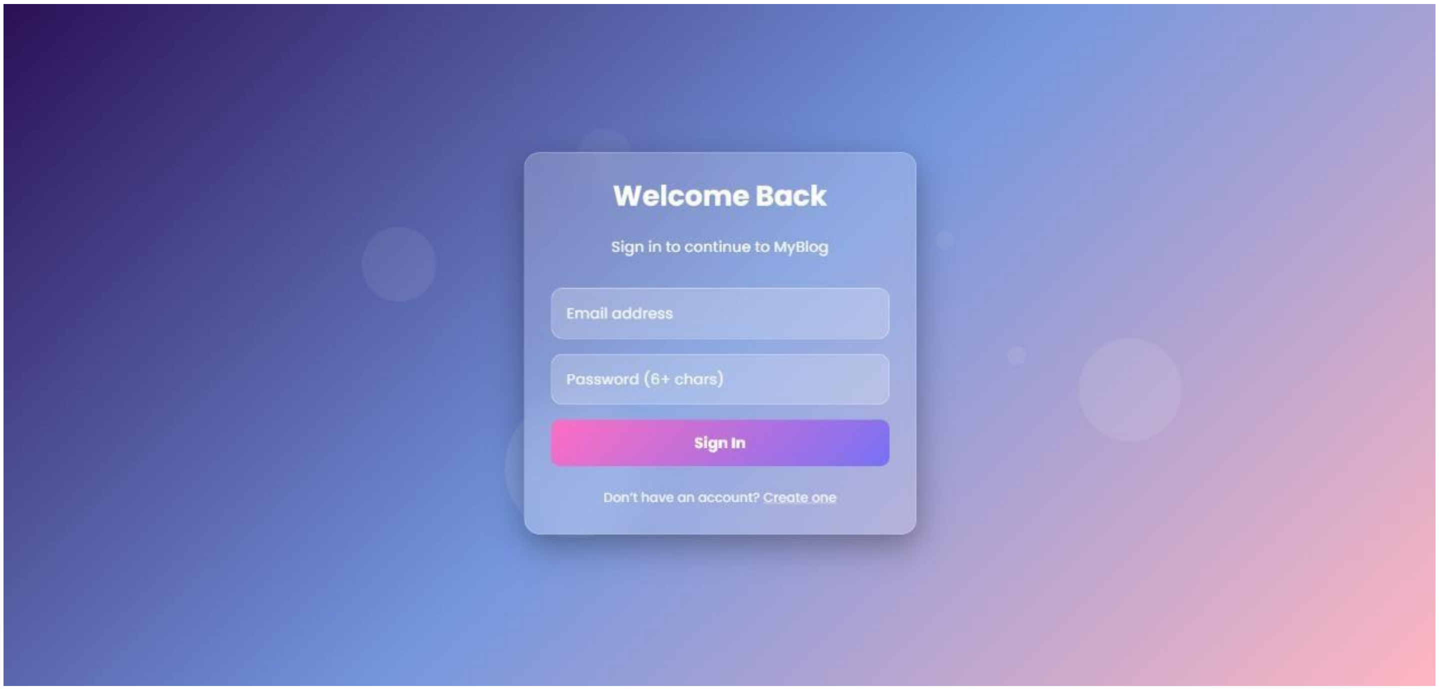
- Key Functions:
  - Featured Posts Display: Showcases the latest or most popular articles.
  - Navigation: Provides access to different sections (Home, Blog, Newsletter, Write).
  - Post Viewing: Handles the presentation of individual blog content.

4. Utility Module

This module covers auxiliary but essential functions that enhance user engagement and system maintenance.

- Key Functions:
  - Newsletter Subscription: Captures visitor emails for future updates.
  - System Footer: Provides branding, copyright, and quick navigational links.

Login page

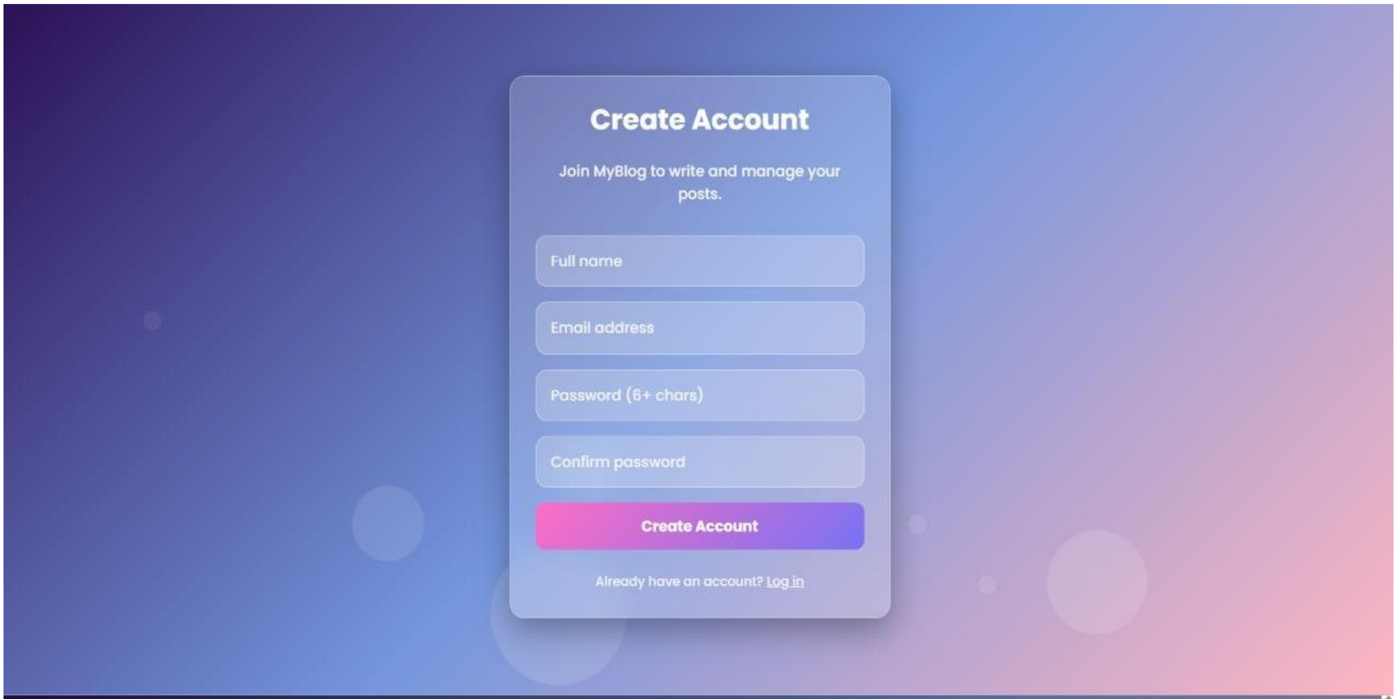


Welcome Back

Sign in to continue to MyBlog

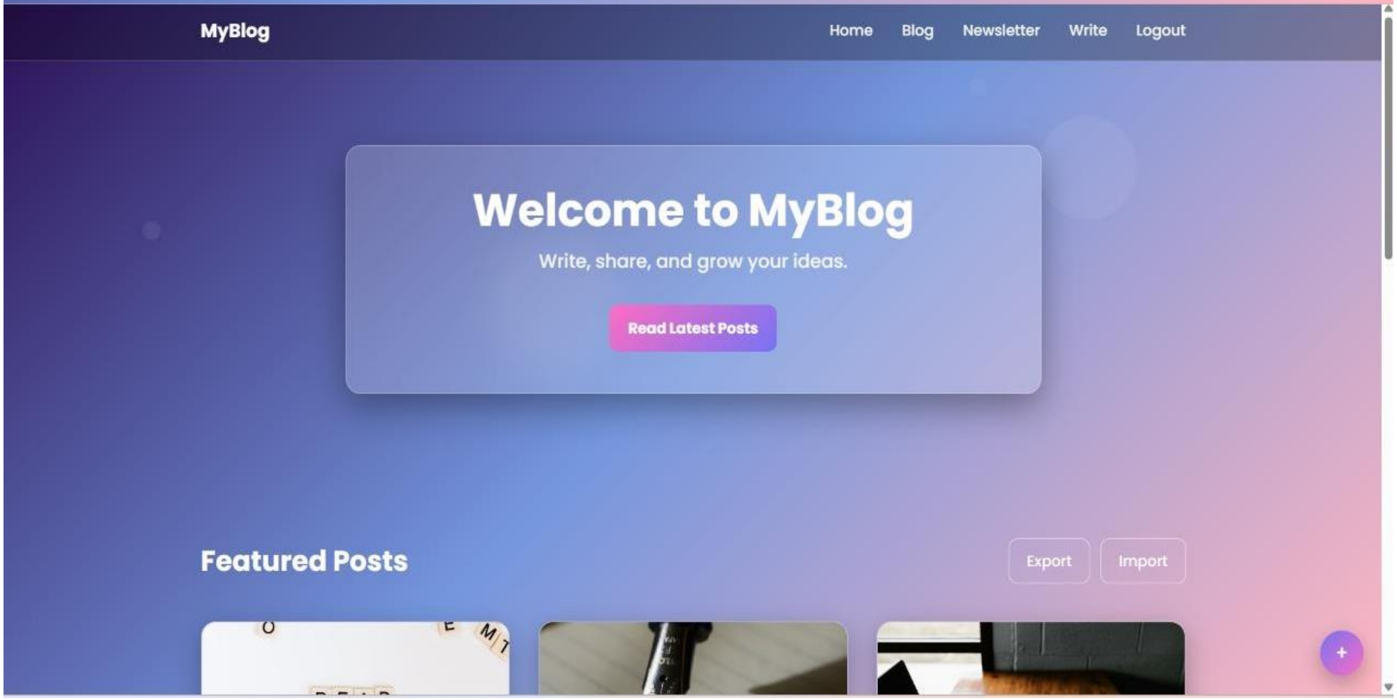
Don't have an account? Create one

Sign up page -



Create Account

Join MyBlog to write and manage your posts.

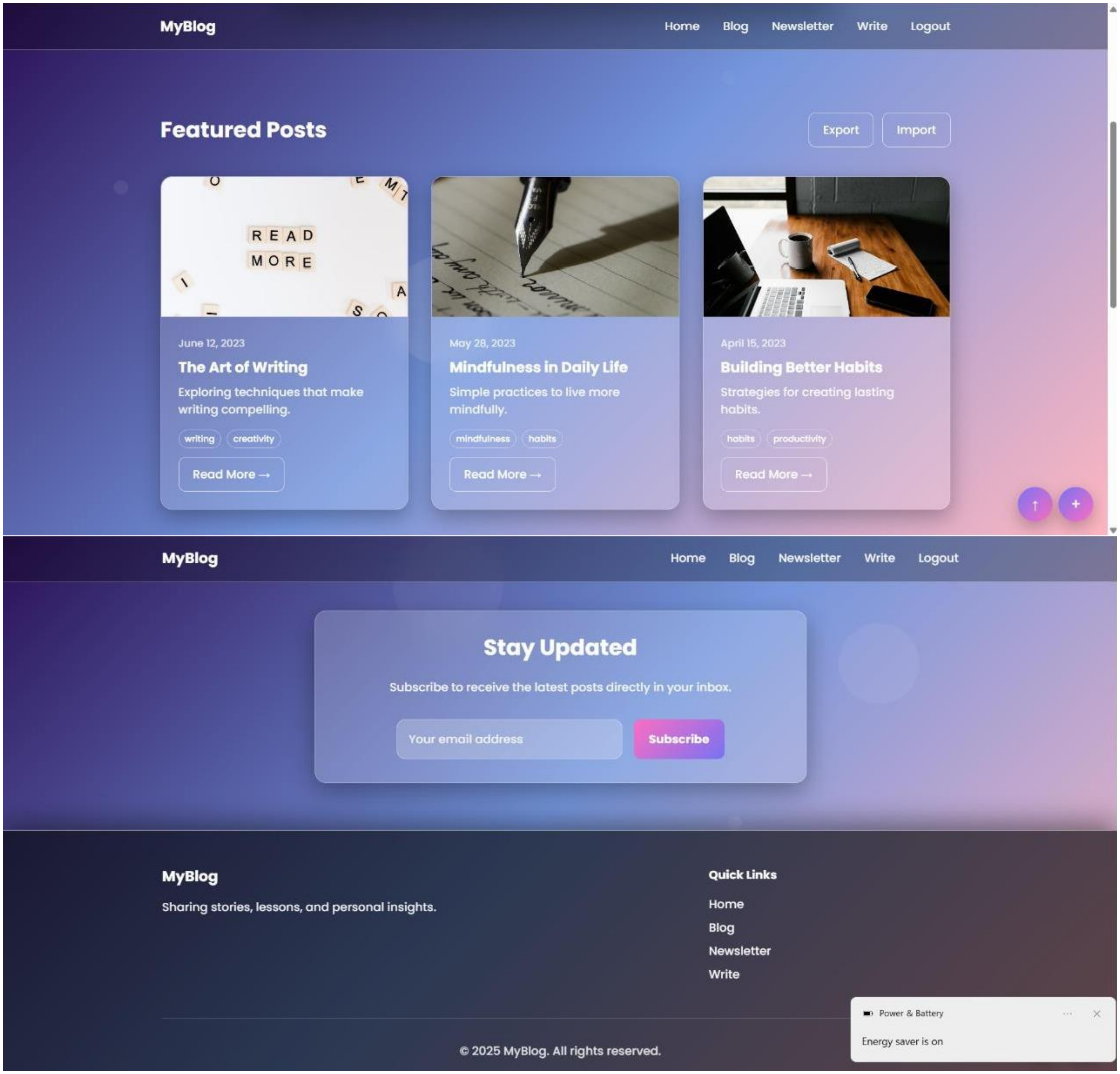


Welcome to MyBlog

Write, share, and grow your ideas.

Featured Posts

Home page –



MyBlog

Home Blog Newsletter Write Logout

Featured Posts

Export Import

June 12, 2023

The Art of Writing

Exploring techniques that make writing compelling.

writing creativity

Read More →

May 28, 2023

Mindfulness in Daily Life

Simple practices to live more mindfully.

mindfulness habits

Read More →

April 15, 2023

Building Better Habits

Strategies for creating lasting habits.

habits productivity

Read More →

↑ +

MyBlog

Home Blog Newsletter Write Logout

Stay Updated

Subscribe to receive the latest posts directly in your inbox.

Your email address Subscribe

MyBlog

Sharing stories, lessons, and personal insights.

Quick Links

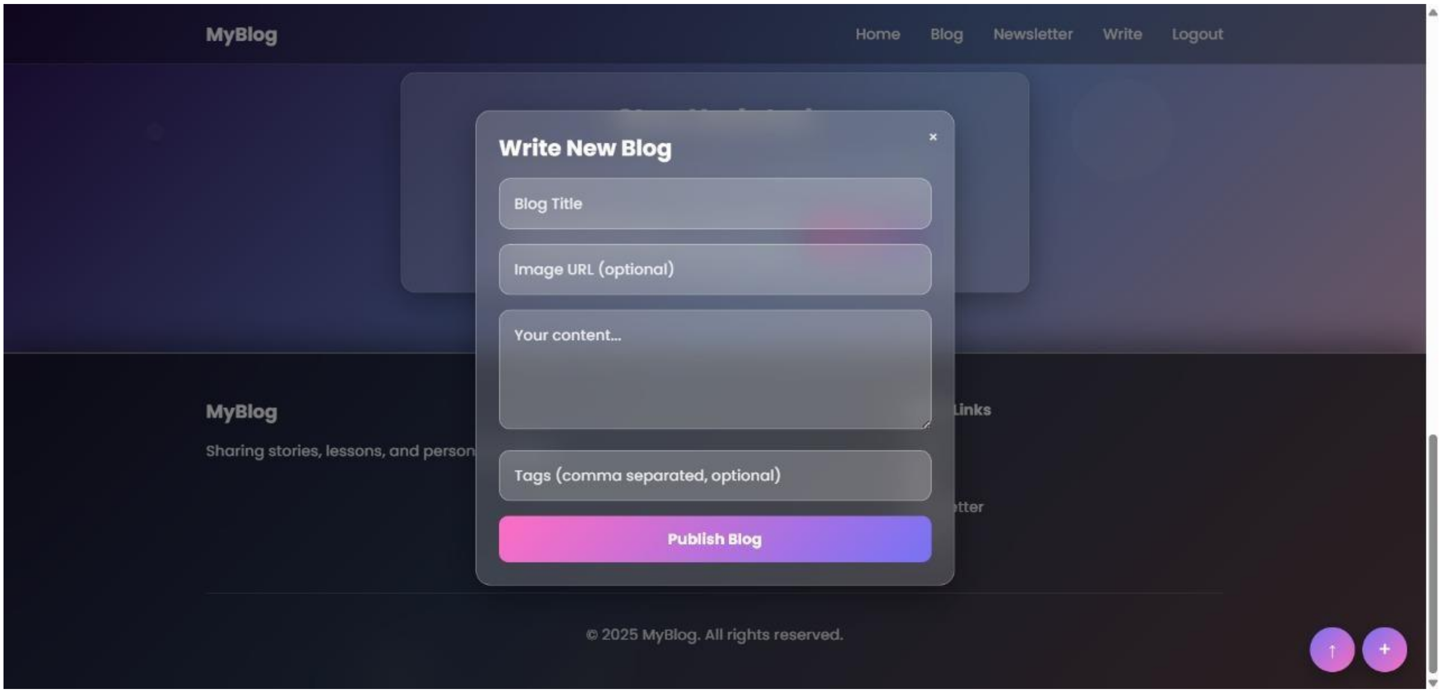
Home
Blog
Newsletter
Write

© 2025 MyBlog. All rights reserved.

Power & Battery

Energy saver is on

writer section –



6: TESTING METHOD

6.1 Testing Method

Testing is a critical phase in software development that ensures the system functions as expected, performs well under different conditions, and is secure from vulnerabilities. For BLOGZE, testing was conducted at different levels to verify correctness, security, performance, usability, and integration between modules.

Because Blogze is a full-stack web application, both **client-side and server-side testing** were carried out. The following testing methodologies were used to validate the system:

6.2 Test Cases

Test Case ID	Module	Test Description	Input	Expected Output	Result
TC01	Login	Valid login	Correct Email & Password	Login successful	Pass
TC02	Login	Invalid password	Wrong password	Error shown	Pass
TC03	Register	Duplicate user	Existing email	Registration denied	Pass
TC04	Blog	Create blog	Valid data	Post saved	Pass
TC05	Blog	Empty title	Null title	Error message	Pass
TC06	Security	SQL injection	' OR '1'='1	Blocked	Pass
TC07	Security	XSS attempt	<script>	Sanitized	Pass
TC08	Session	Logout	Click logout	Redirect to login	Pass
TC09	Admin	Delete post	Valid admin user	Post removed	Pass
TC10	Auth	Unauthorized edit	Access others' post	Denied	Pass

CHAPTER 7: CONCLUSIONS

CONCLUSION

The BLOGZE project successfully delivers a secure, scalable, and user-friendly blogging platform using core web technologies. The aim of developing a lightweight system that avoids heavy frameworks while still ensuring reliability and security has been achieved. Through this project, a complete understanding of how blogging platforms function internally has been gained

.

The system includes all essential features required for a modern blogging application such as user registration, secure authentication, blog creation, editing and deletion, structured database storage, and a responsive interface. These features provide users with a stable and efficient platform for content creation and management.

From a technical perspective, this project has strengthened knowledge in software design, full-stack development, database management, and security implementation. Concepts such as input validation, password encryption, session handling, and SQL security were practically applied to build a dependable system. The modular design makes the system easy to maintain and scalable for future improvement.

BLOGZE also serves as a strong educational model that demonstrates how web technologies like HTML, CSS, JavaScript, and SQL work together in real-world application development. It enables learners to study both client-side and server-side mechanisms in a structured way, supporting academic learning and skill development.

Although advanced features such as comment systems, media uploads, and analytics dashboards are not included, the current implementation meets all primary objectives of the project. The system performs efficiently under normal workload conditions and ensures user data protection.

In conclusion, BLOGZE is not only a functional blogging platform but also a complete learning framework for understanding modern web application development. With future enhancements, it has the potential to evolve into a production-level system suitable for real-world deployment.

CHAPTER 8: LIMITATIONS & FUTURE ENHANCEMENTS

8.1 Limitations of the Current System

Although BLOGZE has been implemented successfully as a secure and scalable blogging platform for academic and learning purposes, some limitations still exist due to project scope constraints, resource limitations, and time restrictions.

The major limitations of the current system are:

1. No Media Upload Support

Currently, users can only publish text-based blog posts. The system does not allow uploading images, videos, or audio files.

2. Limited Scalability Testing

The system has not been tested under extremely high concurrent user loads. Performance under enterprise-level traffic is not guaranteed.

3. Basic UI Design

The user interface is functional but simple. Advanced design features like animations and themes are not implemented.

4. No Comment System

Users cannot comment on posts or interact socially with authors.

5. No Notifications

Email or in-system notifications are not supported.

6. No SEO Optimization Tools

Features like sitemap generation, keyword analysis, and SEO suggestions are not built.

7. No Analytics Dashboard

The system does not include real-time statistics such as page views, reader analytics, or user engagement tracking.

8. No Backup & Restore System

Automatic database backup and recovery options are not implemented.

9. Single Language Support

The system currently supports only one language.

10. Deployment Limitation

The platform is tested locally and not deployed to cloud-based environments.

8.2 Future Enhancements

BLOGZE is designed using a modular structure, which allows for easy upgrades and feature expansion. Many advanced enhancements can be added in future versions:

1. Multimedia Content Support

Enable uploading and displaying of:

- Images
- Videos
- Audio files

2. Comment and Like System

Allow readers to:

- Comment on posts
- React with likes

- Share posts

3. Advanced Security Features

- Two-factor authentication
- CAPTCHA verification
- IP-based blocking
- Login attempt limits

4. Cloud Deployment

Deploy on:

- AWS
- Azure
- Google Cloud

5. SEO Automation

Add:

- Meta tags
- Sitemap generation
- Keyword analytics
- Search engine ranking tools

6. Admin Dashboard Analytics

Provide:

- Daily views

- Most read blogs
- User growth
- Engagement metrics

7. Backup and Recovery

Implement:

- Automatic backups
- Restore system
- Version control

8. AI-Based Enhancements (Optional / Advanced)

- AI grammar checking
- AI-based content suggestions
- Trend-based recommendations

9. Multi-Language Support

Enable international users to write blogs in any language.

10. Mobile Application

Develop:

- Android app
- iOS app

REFERENCES

Use these references. You can renumber if needed depending on formatting style required by your college:

1. Ian Sommerville, *Software Engineering*, Pearson Education
2. Pressman, R. S., *Software Engineering: A Practitioner's Approach*, McGraw Hill
3. W3Schools – Web Development Documentation
4. MDN Web Docs (Mozilla Developer Network)
5. MySQL Documentation
6. OWASP Web Application Security Project
7. GeeksforGeeks – Web Technologies Tutorials
8. HTML5 & CSS3 Design and Development – Wiley Publications
9. JavaScript: The Definitive Guide – O'Reilly Media
10. Database System Concepts – Silberschatz, Korth & Sudarshan
11. Node.js Official Documentation
12. Stack Overflow Developer Community
13. Agile Development Methodology – TutorialsPoint
14. IEEE Software Engineering Standards
15. REST API Design Guidelines – Microsoft Documentation