

# Tribhuvan University

**Institute of Science and Technology**

**WriteOn: “The Ultimate Blogging Experience” Project Report**

## Submitted to

**Department of Computer Science and Information Technology**

**National College of Computer Studies**

*In partial fulfillment of the requirements for Bachelor in Computer Science and Information Technology*

## Submitted By:

**……………………**

**Under the Supervision of Sachita Maharjan**



**Tribhuvan University**

**Institute of Science and Technology**

**National College of Computer Studies**

# SUPERVISOR’S RECOMMENDATION

We hereby recommend that this project prepared under our supervision by

**Sachita Maharjan** entitled “**WriteOn: The Ultimate Blogging Experience**” in partial fulfillment of the requirements for a degree of Bachelor’s in Computer Science and Information technology is recommended for the final evaluation.

**SIGNATURE**

**Sachita Maharjan SUPERVISOR**

**National College of Computer**

# Abstract

WriteOn is a blogging platform that makes it easy and fun for people to share their thoughts and experiences. With its simple design, bloggers of all levels can easily create, organize, and share their content. The platform offers plenty of customization options, so users can make their blogs truly unique.

WriteOn includes features like social sharing, comment management, and analytics to help bloggers grow their audience and engage with readers. Built with the latest web technology, WriteOn ensures a smooth and reliable blogging experience, making it a great choice for anyone looking to start or improve their blogging journey.

# Acknowledgement

Bringing this study to life has been both rewarding and challenging. Throughout this journey, many people have stepped in at the right moments, offering their help and support, and they deserve special thanks.

We are deeply grateful to the department for their guidance and continuous support. Their valuable information and supervision played a crucial role in helping us complete this project. We also want to express our heartfelt thanks to everyone who contributed, directly or indirectly, to make this study possible.

Lastly, we sincerely appreciate those who take the time to read this project and hope it will benefit them now and in the future.

|  |  |
| --- | --- |
| **Table of Contents** |  |
| SUPERVISOR’S RECOMMENDATION | i |
| Abstract | ii |
| Acknowledgement | iii |
| Table of Contents | iv |
| List of Figures | vi |
| List of Tables | vii |
| Chapter 1: Introduction | 1 |
| 1.1. Introduction | 1 |
| 1.2. Problem Statement | 1 |
| 1.3. Objectives | 1 |
| 1.4. Scope and Limitation | 2 |
| 1.5. Report Organization | 3 |
| 1.5.1. Introduction | 3 |
| 1.5.2. Background Study | 3 |
| 1.5.3. System Analysis and Design | 3 |
| 1.5.4. Implementation and Testing | 3 |
| 1.5.5. Conclusion and Future Recommendations | 3 |
| Chapter 2: Background Study | 4 |
| 2.1. Background Study | 4 |
| 2.1.1. Study of existing system | 4 |
| 2.1.2. Literature Review | 5 |
| Chapter 3: System Analysis and Design | 7 |
| 3.1. System Analysis | 7 |
| 3.1.1. Requirement Identification | 8 |
| 3.1.1.1. Functional Requirements | 8 |
| 3.1.1.2. Non-Functional Requirements | 10 |

* + 1. [Feasibility Study 11](#_TOC_250023)
       1. [Technical Feasibility 11](#_TOC_250022)
       2. [Operational Feasibility 13](#_TOC_250021)
       3. [Economic Feasibility 14](#_TOC_250020)
       4. [Schedule Feasibility 15](#_TOC_250019)
    2. [Data Modeling 15](#_TOC_250018)
    3. [Process Modeling 17](#_TOC_250017)
    4. [Sequence Diagram 19](#_TOC_250015)
    5. [Activity Diagram 20](#_TOC_250014)
  1. [System Design 21](#_TOC_250013)
     1. [Architectural Design 21](#_TOC_250012)
     2. [Database Schema Design 22](#_TOC_250011)

[Chapter 4: Implementation and Testing 23](#_TOC_250010)

* 1. [Implementation 23](#_TOC_250009)
     1. [Tools used 23](#_TOC_250008)
  2. [Testing 24](#_TOC_250007)
     1. [Test Case 24](#_TOC_250006)

[Chapter 5: Conclusion and Future Recommendation 29](#_TOC_250005)

* 1. [Lesson Learnt / Outcome 29](#_TOC_250004)
  2. [Conclusion 29](#_TOC_250003)
  3. [Future Recommendations 29](#_TOC_250002)

[References 30](#_TOC_250001)

[Appendices 31](#_TOC_250000)

# List of Figures

Figure 1: Use Case Diagram of WriteOn 9

Figure 2: Timeline of WriteOn 14

Figure 3: E-R Diagram of WriteOn 14

Figure 4:Data Flow Diagram of WriteOn 16

Figure 5: Sequence Diagram of WriteOn 17

Figure 6: Activity Diagram of WriteOn 19

Figure 7: Basic Database Schema of WriteOn 22

Figure 8: Home Page of WriteOn 31

Figure 9: Contact Page of WriteOn 32

Figure 10: User Home Page of WriteOn 32

Figure 11: User Post page of WriteOn 33

Figure 12:User Profile Page of WriteOn 34

Figure 13: Quick Blog View of WriteOn 35

Figure 14: Blog Preview Page of WriteOn 36

Figure 15: User Stats Page of WriteOn 37

Figure 16: Login Page of WriteOn 38

Figure 17: Admin Dashboard of WriteOn 39

Figure 18: Available User of WriteOn 40

|  |  |  |
| --- | --- | --- |
|  | **List of Tables** |  |
| Table 1: Test Case 1 |  | 24 |
| Table 2: Test Case 2 |  | 25 |
| Table 3: Test Case 3 |  | 26 |
| Table 4: Test case 4 |  | 27 |
| Table 5: Test Case 5 |  | 28 |

# Chapter 1: Introduction

## Introduction

Blogging is a way to share experiences, and information online by writing posts. It's become a popular way for people and businesses to connect with others, share their ideas, and provide helpful information. Blogs can be about anything, from personal stories to career tips, and they give writers a chance to interact with readers. Plus, many bloggers can even earn money from their content!

Starting a blog is easy, with many simple tools and platforms available to help anyone get started, regardless of their technical skills. Whether you want to share your passions, offer helpful tips, or build a community, blogging provides endless opportunities for creativity and connection.

## Problem Statement

Many people face challenges when starting a blog. They might struggle with choosing the right platform, creating engaging content, or growing their audience. Lack of technical skills and time constraints can also make the process seem overwhelming.

Concerns about finding the right tools for managing a blog and keeping personal data secure are common. However, these challenges can be easily overcome with the right platform. With an easy-to-use and secure platform, anyone can enjoy blogging and share their ideas, stories, and thoughts with the world.

## Objectives

The main objective of WriteOn is:

* To is to offer valuable insights and tips for individuals looking to start or improve their blogging journey.

## Scope and Limitation

The scope of the WriteOn site might include:

* Offer practical tips for starting and managing a successful blog.
* Guide readers in selecting the right blogging platform and creating engaging content.
* Share strategies for building an online audience and boosting blog visibility.
* Cover important topics like blog security, privacy, and essential blogging tools.
* Focus on enhancing blog management and improving the overall user experience.
* Explore the integration of various tools and features to streamline the blogging process, such as payment options for monetization (if applicable).

Some limitations of the WriteOn site might include:

* The blog will not cover other types of content creation, such as video blogs or podcasts.
* Some articles may be focused on specific blogging platforms, which may not apply to all users.
* It will provide general advice, but not personalized support or detailed technical troubleshooting.

.

## Report Organization

### Introduction

This chapter introduces the concept of the project. It describes the problem the existing problems faced and our objectives to tackle it. It also briefs about the project objective, scope, and limitations.

### Background Study

This chapter focuses on study of existing systems and review of different articles studied. It also explores the base ideology for building the system.

### System Analysis and Design

This chapter covers the project's requirements collecting, feasibility assessment, and design. Diagrams, functionality analysis, a technique for obtaining requirements, and a process model are all included.

### Implementation and Testing

This chapter aims to explain how the project was carried out, the software and tools that were used, and the various testing methods that the project experienced.

### Conclusion and Future Recommendations

This chapter discusses the project's potential outcomes, as well as the conclusion and future recommendations.

# Chapter 2: Background Study

## Background Study

WriteOn is a blog platform which offer a simple and happy experience for anyone looking to share their thoughts, stories, and thoughts online. It provides a user-friendly interface that makes it easy to create, manage, and publish blog posts. The platform focuses on making blogging accessible to everyone, whether you're a beginner or an experienced writer.

Through WriteOn, users can personalize their blogs, engage with readers through comments, and share their content on social media. The platform also includes tools for tracking blog performance, helping bloggers understand their audience and improve their content over time. With a focus on ease of use and customization, WriteOn aims to help people express themselves online with confidence and simplicity.

### Study of existing system

In our research of existing platforms, we also examined Blogger as a reference for developing WriteOn. Blogger is a widely used and established blogging platform known for its ease of use, customization options, and user engagement.

Blogger’s success stems from its simple interface, which enables users to easily create, publish, and manage their blog content. It provides various templates and customization tools, allowing bloggers to design their sites to match their personal style. Additionally, Blogger integrates with Google services like Google Analytics, helping users monitor their blog’s performance and understand audience behaviour.

The simplicity and accessibility of Blogger played a key role in shaping the design and functionality of WriteOn. Our goal is to offer an equally intuitive, flexible, and seamlessly integrated blogging experience that empowers users to express themselves and build their online presence without needing technical expertise.

### Literature Review

Guillaume Thevenot (June 7, 2007) published a paper highlighting that blogging is one of the most popular tools in social media. On these platforms, people share their thoughts, experiences, and perspectives through various formats such as text, images, and videos. Blogging often starts with one person posting an article, which readers then comment on, creating a conversation. Thevenot mentions that approximately 120,000 new blogs are created daily worldwide, which equates to about 1.4 blogs per second. Many of these blogs are related to travel. The paper also discusses different types of blogs, including individual, collaborative, corporate, and traditional media blogs. [1]

Dou Shen, Jian-tao Sun, Qiang Yang, and Zheng Chen (2006) wrote an article on extracting hidden friendships from blog data. Blogs are a valuable source of data that provide rich resources for social community mining. This process involves analyzing large amounts of user-generated content from social media and mobile apps to identify patterns, often for advertising or research purposes. The authors define latent friends as individuals who share similar topics in their blogs. The paper outlines three methods for detecting these hidden friendships: the cosine similarity-based method, the topic model-based method, and the two-level similarity-based method. [2]

Child, J. T., Haridakis, P. M., and Petronio, S. (2012) published a paper discussing privacy rules and management in blogging. The authors studied different stages of blogging, focusing on how it functions before and after blogs are posted. They categorized various privacy rules and explored why bloggers might delete previously posted content. The paper collected data on bloggers' activities to understand their reasons for deleting content before it is shared. The study highlights the role of privacy rules, privacy management, and the motivations for deleting posts after blogging. [3]

# Chapter 3: System Analysis and Design

## System Analysis

Systems development is typically carried out in two phases: system analysis and design. This chapter focuses on analyzing the research data and providing a clear view of the entire process. We use diagrams to model the data, helping us visualize the system's design and specifications in an object-oriented.

This project is on a small scale and has well-defined requirements with a focus on flexibility and adaptability to changing market trends. For this type of project, we considered both the Agile development model, specifically Rational Unified Process (RUP) model

We have chosen to use the Rational Unified Process (RUP) methodology for this e-commerce project because it offers a structured yet flexible approach to development. RUP focuses on iterative development, allowing continuous refinement and adaptation of the system based on feedback. This makes it ideal for projects with clear requirements and the need for effective risk management throughout the development lifecycle. By applying RUP, we aim to create a reliable, scalable, and well-documented e-commerce platform that caters to user needs and stays in line with market trends.

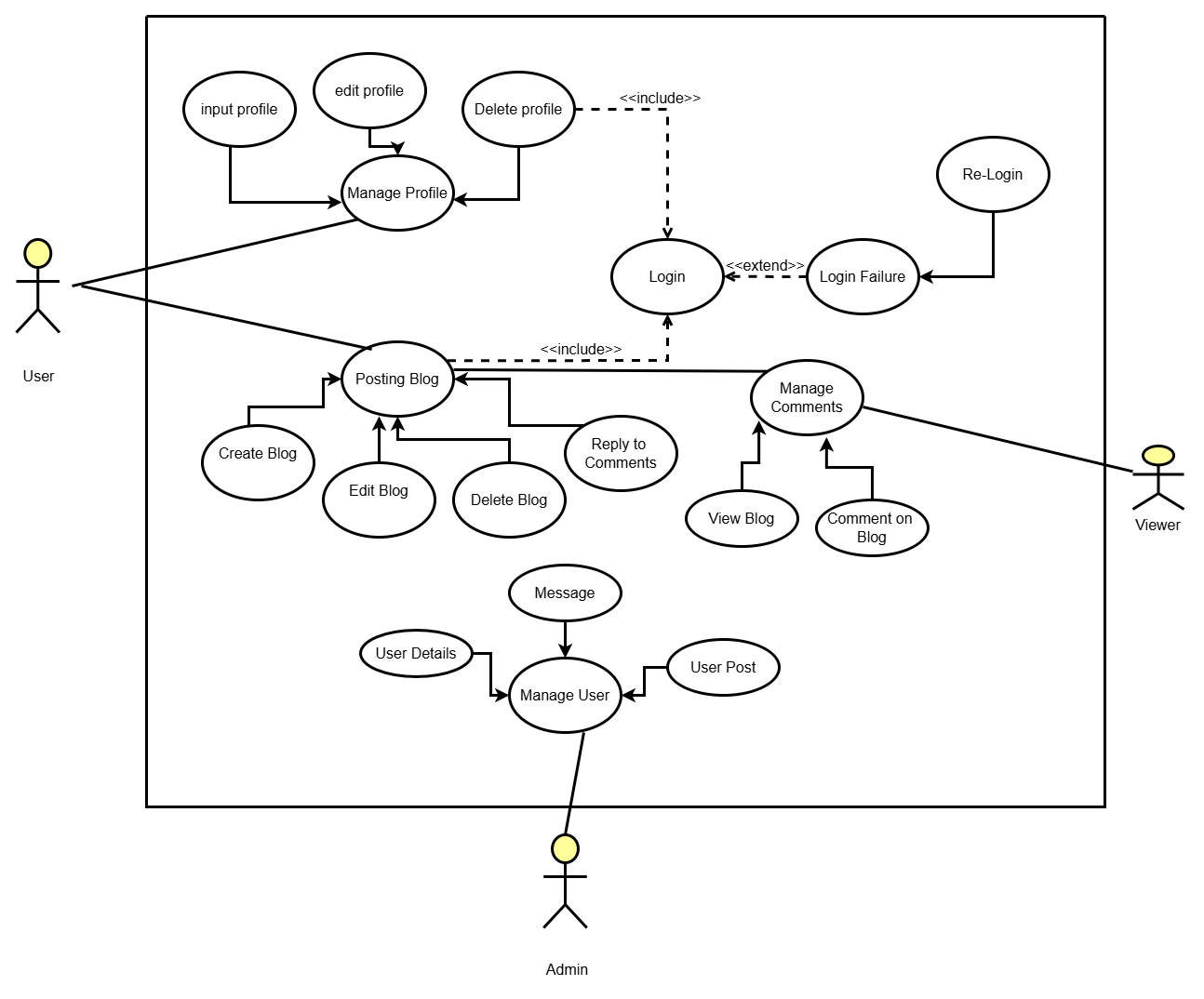
### Requirement Identification

Requirements of the system are identified through personal research of visiting various blogging sites.

### Functional Requirements

Functional requirements for WriteOn e-commerce platform:

* + - * 1. User registration and login - Users should be able to create an account and log in to the platform to access the full functionality of the site.
        2. Blog Creation and Management - Users should be able to create, edit, and delete blog posts with ease and should be able to categorize their blog posts and add tags for better organization.
        3. Content Browsing and Search - Visitors and users should be able to browse the blog by categories, tags, or recent posts.
        4. Commenting System:- Users should be able to leave comments on blog posts and Users should be able to reply to comments on individual blog posts
        5. User Profile Management- Users should have a profile page where they can view their own blog posts and edit personal information.
        6. Social Sharing - Users should be able to share their blog posts via social media platforms (e.g., Facebook, Twitter, LinkedIn).
        7. Analytics and Performance Tracking- Users should be able to view basic statistics of their blog posts, such as the number of views, comments, and shares.
        8. Responsive design- The platform should be responsive and compatible with different devices such as Desktop, Tablets and Mobile to provide a seamless user experience.
        9. Admin User Management - The admin should be able to view and manage user accounts, including viewing user details such as name, email, registration date, and account status.

1. 

*Figure 1: Use Case Diagram of WriteOn*

The use case diagram for WriteOn blog platform outlines the key user interactions. It includes actors such as Users and Administrators, with use cases like creating posts, commenting, liking, sharing, and managing user accounts. The diagram shows how users interact with the platform and how administrators manage content. It helps visualize the system’s functionality and user interactions, providing a clear understanding of its behaviour.

### Non-Functional Requirements

Non-Functional requirements for WriteOn blog platform:

* + - * 1. Performance - The platform should run smoothly, even when many users are online at the same time, ensuring no noticeable slowdown during high traffic periods
        2. Scalability - As the business grows, the platform should easily handle more users and transactions without needing major changes or causing performance issues.
        3. Availability - The platform should be up and running around the clock, with only minimal interruptions for necessary maintenance or updates to ensure a reliable experience for users
        4. Ease of Use - The system should be straightforward and user-friendly, making it easy for both administrators and regular users to navigate and use all features efficiently.
        5. Compliance- The platform must comply with all relevant laws, regulations, and industry standards to protect both the business and its users.
        6. User Experience- The platform should offer a positive and engaging user experience, with intuitive navigation, a clean and responsive design, and an overall satisfying interface that encourages users to return.
        7. Mobile Compatibility:- The platform should work seamlessly on mobile devices, offering a responsive design that ensures a smooth and enjoyable experience whether users are on a phone, tablet, or computer.

### Feasibility Study

A feasibility study evaluates the operational, technical, and financial aspects of the proposed project. Its purpose is to conduct an initial review to determine if the project is viable enough to move forward to the detailed analysis phase. For system analysts, the feasibility study is a crucial tool for deciding whether to recommend continuing with the project or halting it.

It ensures that the project is practical, achievable, and worth pursuing based on the available resources and expected outcomes

### Technical Feasibility

The technical feasibility of the WriteOn blog platform focused on determining whether the project could be successfully built and deployed using available technology and resources.

**Technology and Tools:** WriteOn is built using reliable and widely available web technologies, ensuring a smooth development process. The platform leverages modern frameworks and tools to provide essential blogging features, including user registration, content creation, and management. Customization options allow users to personalize their blogs to suit their preferences.

**Payment Integration:** The platform supports multiple payment options for premium features, including the integration of secure online payment gateways (esewa). This ensures smooth and safe transactions, which is a critical component of the platform.

In summary, WriteOn has a high level of technical feasibility. Even though payment integration is not yet finalized, all necessary technology and resources are in place to build and deploy the platform successfully

### Operational Feasibility

Operational feasibility assesses whether the WriteOn blog platform can be effectively implemented and integrated into current operations and whether it will be well-received by its intended users.

**Target Users:** WriteOn is designed primarily for writers, bloggers, and content creators. The platform aims to meet their needs by offering an easy-to-use, efficient way to publish and manage blog content. Given the increasing demand for user-friendly blogging solutions, WriteOn is expected to attract and be adopted by a wide range of users.

**Integration with Existing Operations:** WriteOn is designed to integrate smoothly with existing web hosting and content management operations. This ensures that users can seamlessly manage their blogs and publish content without disrupting their workflow. The platform can also be adapted to support potential future integrations, such as analytics tools or other third-party services, to enhance user experience.

**Support and Maintenance:** The platform will be maintained by a dedicated team responsible for ensuring smooth operation and addressing any technical issues. This team will also handle regular updates and improvements to keep WriteOn aligned with user needs and market trends.

### Economic Feasibility

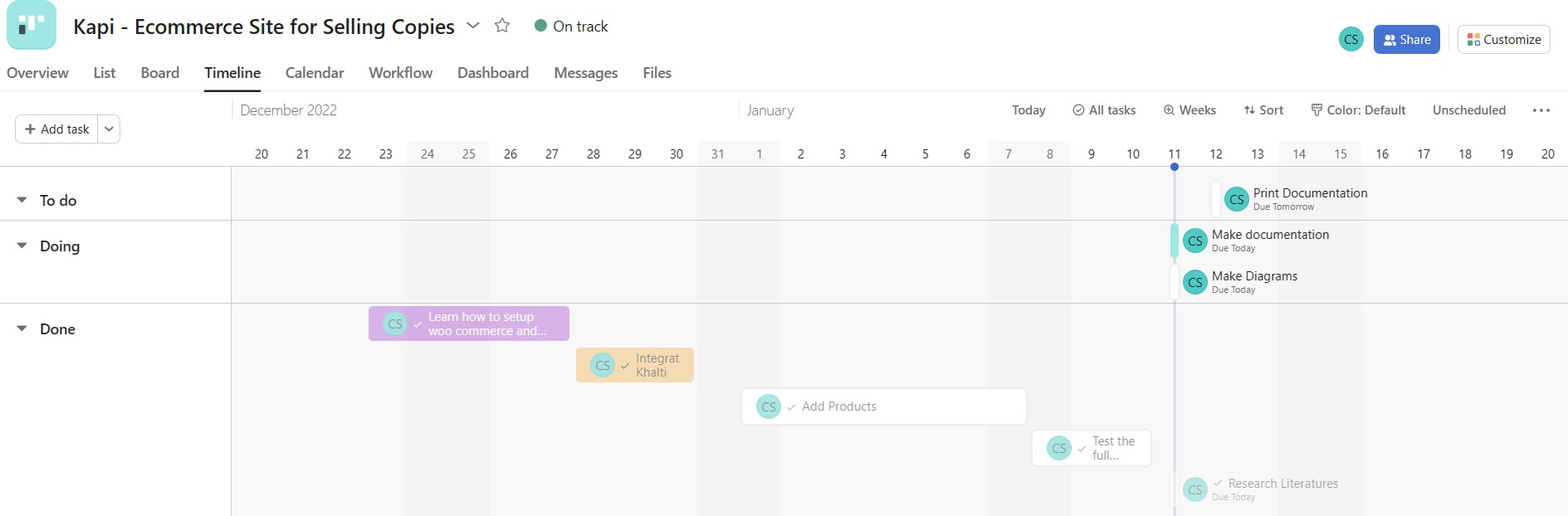
Economic feasibility evaluates whether the WriteOn blog platform is financially viable, ensuring that the benefits outweigh the costs of its development and operation.

**Development Costs:** WriteOn is built using the MERN stack (MongoDB, Express.js, React, and Node.js), which is open-source and free to use. This choice minimizes software licensing costs. However, there will be expenses related to developing and customizing the platform to meet user needs, such as designing unique features and enhancing the user experience. Additionally, hosting services and domain registration will incur recurring costs.

**Operational Costs:** The platform requires ongoing maintenance, including server management, software updates, and security enhancements. Customer support and content moderation will also add to operational expenses. Despite these costs, the platform is expected to generate revenue through premium subscriptions, ads, or additional services, helping to balance operational costs.

**Revenue Generation:** WriteOn is designed to attract bloggers and content creators, offering them a robust and flexible platform for managing their content. Revenue streams may include premium features, subscription plans, and ad placements. The growing demand for modern, user-friendly blogging platforms positions WriteOn well for significant user adoption and revenue growth.

### Schedule Feasibility

****

*Figure 2: Timeline of WriteOn*

Schedule feasibility assesses whether the WriteOn platform can be developed and launched within the desired time frame.

Given the project's scope and the use of the MERN stack, the development is expected to take approximately 1–2 months, including phases for design, development, testing, and deployment. The timeline accounts for potential challenges and allows for iterative feedback.

With a well-organized development process and clear milestones, the platform can be delivered on time without compromising quality

### Data Modeling



*Figure 3: E-R Diagram of WriteOn*

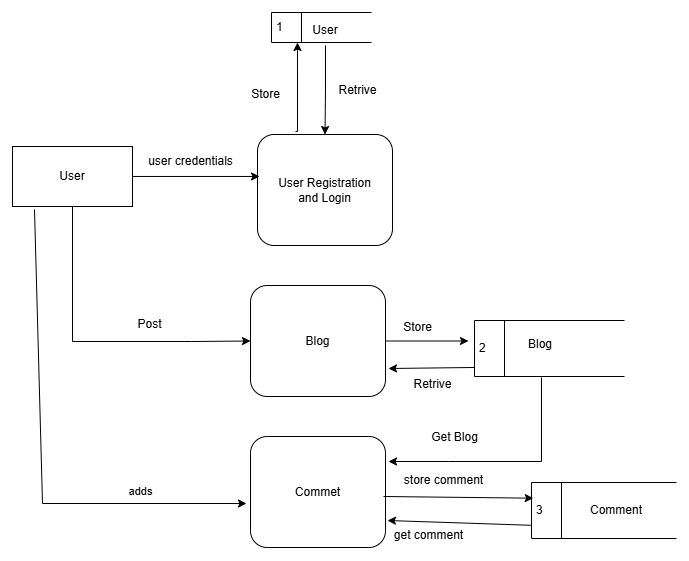
The ER diagram of **WriteOn** provides a comprehensive view of the entities and relationships that form the blog platform. It highlights the data structure and demonstrates how different entities are interconnected. The main entities in the **WriteOn** ER diagram are: **User**, **Blog**, **Comment**, **Message**, and **Administrator**. Each entity is represented by a rectangle, with the entity name and its attributes listed inside

* A User can create multiple Blogs, while a Blog is associated with one User. This is represented by a 1:N relationship between the User and Blog entities.
* A Blog can have multiple Comments, but each Comment belongs to only one Blog. This is a 1:N relationship between the Blog and Comment entities
* A User can write multiple Comments, but each Comment is written by only one User. This is a 1:N relationship between the User and Comment entities.
* An Administrator has the ability to manage multiple Blogs, but a Blog can be managed by one Administrator. This is a 1:N relationship between the Administrator and Blog entities.

There also exists foreign keys between entities. The major foreign keys in the ER diagram of WriteOn are:

* User\_ID becomes the foreign key in both the Blog and Comment entities, linking the User to the content they create and comment on.
* Blog\_ID becomes the foreign key in the Comment entity, linking each comment to a specific blog.
* Sender\_ID and Receiver\_ID become the foreign keys in the Message entity, linking the User entities to each message they send and receive.

### Process Modeling



*Figure 4: Data Flow Diagram of WriteOn*

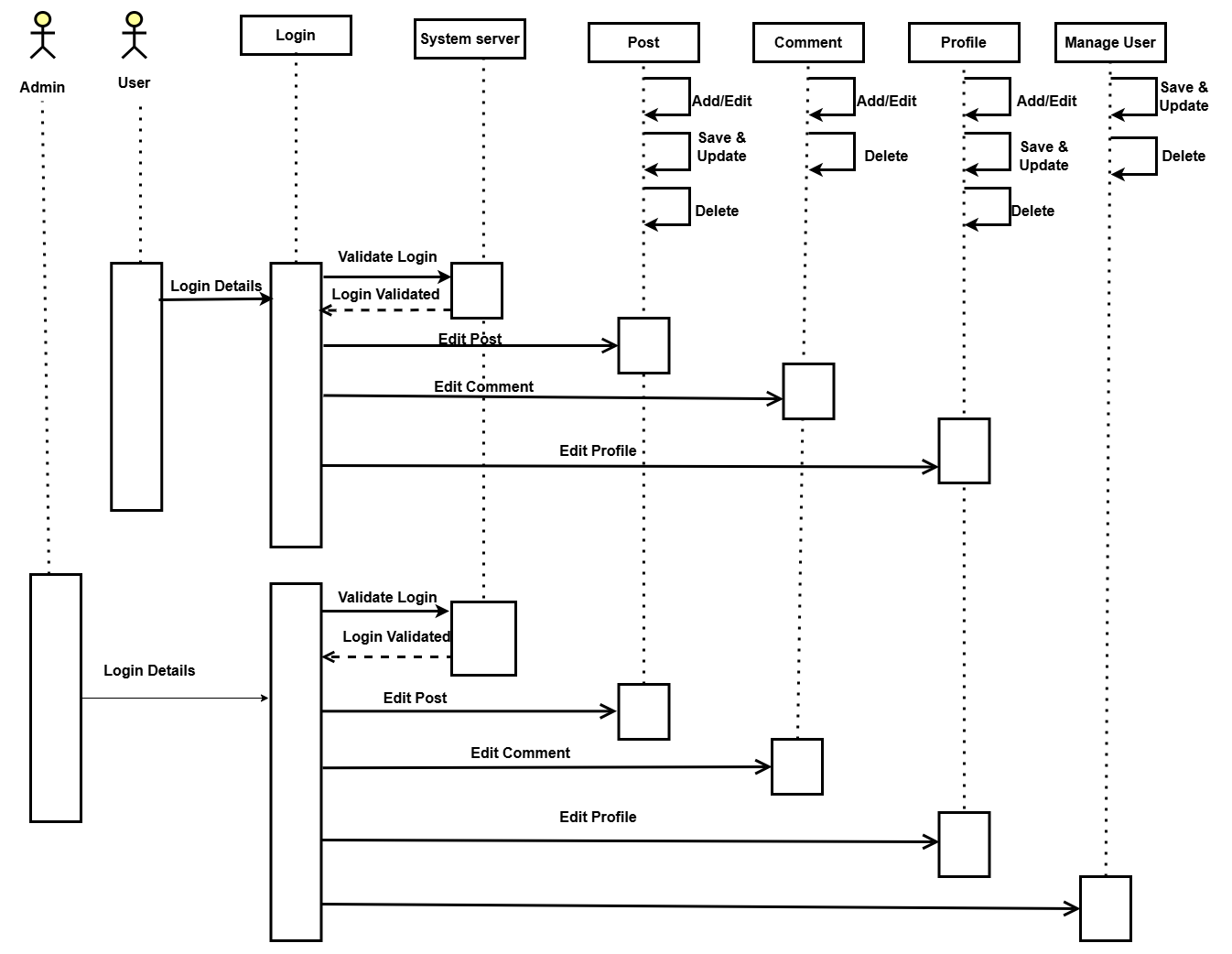
A data flow diagram (DFD) is a way of modeling the flow of data within a system, showing how data is transformed as it moves through the system. The DFD of WriteOn provides an overview of the main processes and data flows within the blogging platform.

In the WriteOn DFD, there are four main entities: Users, Administrator, Blog Authors, and Message System. These entities interact with the system in different ways, such as users can register, log in, read and comment on blogs, while the administrator manages blog content, users, and comments. The Message System handles the communication between users.

There are three main processes: User Registration and Login, Blog Management, and Message Management. These processes handle important tasks such as capturing user registration and login details, managing blog content, including creating and commenting on blogs, and facilitating communication between users through private messages.

There are three main data stores: User Database, Blog Database, and Message Database. These data stores hold essential information, including user details, blog content, comments, and messages exchanged between users. Data flows between the processes and data stores, illustrating how data is captured, transformed, and moved throughout the system

### Sequence Diagram



*Figure 5: Sequence Diagram of WriteOn*

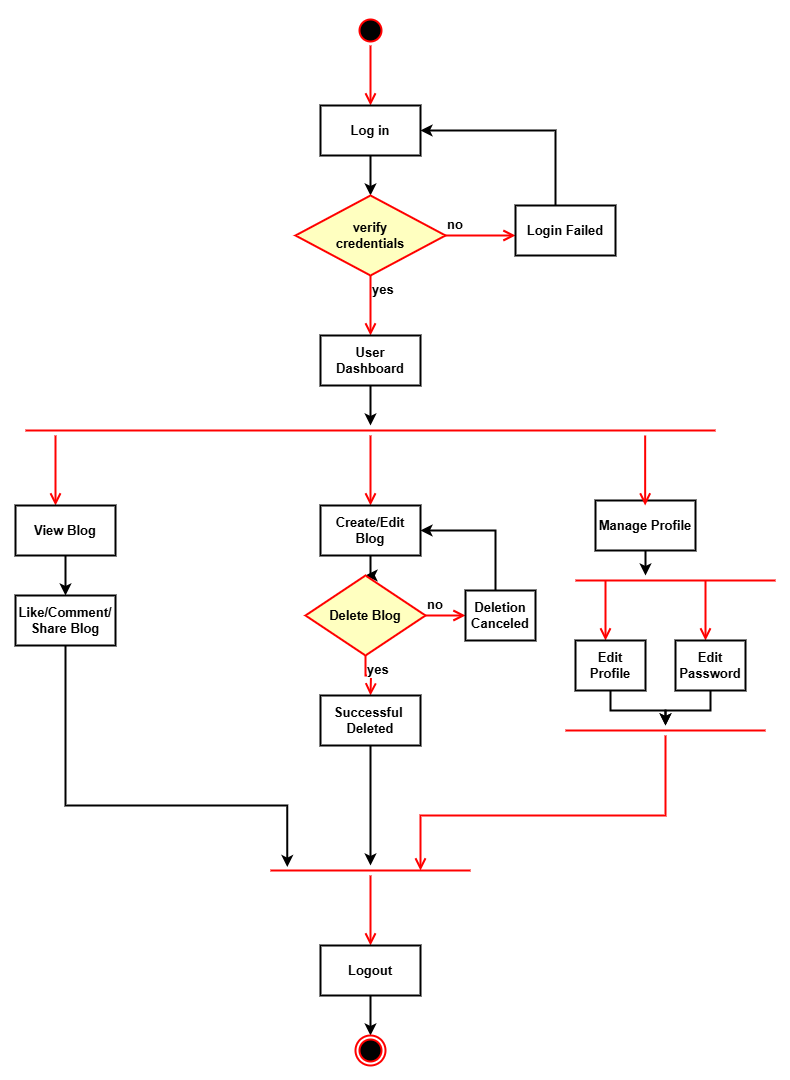
The sequence diagram for WriteOn visualizes the interactions between key components of the system, including the user, administrator, post, comment, profile, and system server. It outlines how messages are exchanged between these entities and the order in which actions occur.

The process begins when a user logs into the system and accesses their profile. They can then create a post, which is stored in the system. Once the post is created, other users can add comments to the post. The system server processes these actions and updates the database accordingly.

In the case of an administrator, they can manage users by reviewing user profiles, approving or removing posts, and moderating comments. They interact with the system server to update the platform, ensuring content is in line with the platform's guidelines.

The sequence diagram showcases the dynamic interactions between users and the system, from creating posts, adding comments, and managing user content, to admin oversight. It effectively captures the flow of control and data across the system and helps identify any potential issues or areas for optimization in the platform's functionality.

### Activity Diagram

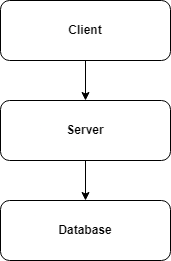
1. 

*Figure 6: Activity Diagram of WriteOn*

The activity diagram created for WriteOn models the flow of activities within the platform, providing a clear visualization of how tasks are coordinated and the control flow between them. This diagram helps to understand the behaviour of the system, identify potential issues, and communicate its functionality to those involved in the project. The activities depicted include users logging in, accessing their profiles, creating posts, and adding comments. Administrators manage user posts, moderate comments, and update content. This diagram serves as a valuable tool to demonstrate how the platform operates and how the different roles, such as users and administrators, interact with each other.

## System Design

### 3.2.1 Architectural Design

The WriteOn project follows a three-tier architecture, dividing the system into three layers: the presentation layer (user interface), application layer (business logic), and data layer (data storage). This separation ensures that each component is independent, making the system more scalable, maintainable, and flexible.

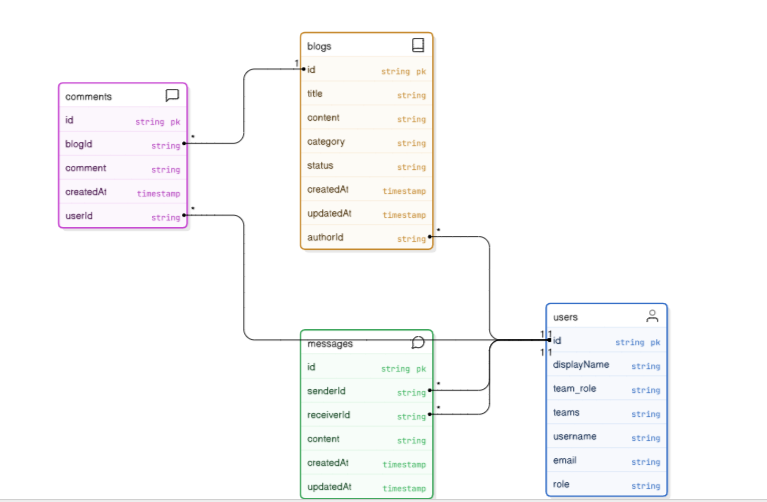
### Figure 3.9: Three-Tier Architecture of WriteOn

Three-Tier Architecture has the following layers.

* **Presentation Layer:** This layer is responsible for displaying information to users and presenting the available services on the website. It communicates with the other tiers by sending results to the browser and other components in the network.
* **Application Layer:** Also known as the middle or logic tier, this layer handles the business logic and controls the application's functionality by performing detailed processing. It operates on the backend and is managed by Node.js, which processes client requests and sends appropriate responses.
* **Data Layer:** This tier houses the database where information is stored and retrieved. In the WriteOn project, MongoDB serves as the database, keeping data independent from the application servers and business logic. Data is managed through MongoDB’s collection structure, ensuring efficient data storage and retrieval.

### Database Schema Design

Database schema design is a strategy for constructing a framework for data management. Just like in architecture, a solid database needs to have a blueprint to keep the project on track.



*Figure 7: Basic Database Schema of WritOn*

The WriteOn project uses MongoDB, part of the MERN stack, for its database. Unlike traditional relational databases, MongoDB stores data in collections, offering flexibility. Currently, we have collections like Users, Posts, Comments, and Admin Actions to manage user data, blog posts, comments, and admin actions. The schema is flexible, allowing easy scalability as the project grows

The above figure includes 9 table structures form the database.

# Chapter 4: Implementation and Testing

## Implementation

### Tools used.

* + - * **Platform:** Windows
      * **Backend Framework**: Node.js with Express.js
      * **Frontend**: ReactJS
      * **Database**: MongoDB
      * **Design and prototyping Tool**: Draw.io, eraser.io

### Implementation Detail of Module

The WriteOn project was implemented using the MERN stack. Node.js and Express.js were used for the backend, while MongoDB served as the database. React.js was used for building the frontend. The application’s design and functionality were custom-built to meet the project's requirements.

## Testing

System testing is a level of testing that validates the complete and fully integrated software product. The purpose of a system test is to evaluate the end-to-end system specifications. Usually, the software is only one element of a larger computer-based system.

### Test Case

**Project Name: WriteOn**

**Test Case ID:** 1

**Test Title:** Verify login with valid username and password.

**Test Designed Date:**  January 5, 2025

**Pre-conditions:** User has valid username and password.

**Dependencies:**

*Table 1: Test Case 1*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Description** | **Test Data** | **Expected Result** | **Actual Result** | **Status (Pass/Fail)** |
| 1. Navigate to login page    * Provide valid username    * Provide valid password    * Click login button | Email: test1@gmail.com  Password: test | User should be able to login | User is navigated to the dashboard with successful login | Pass |
| Provide invalid username and password. | Email: test1@gmail.com  Password: 00 | User login invalid | User login invalid | Pass |

**Post-condition:**

User is validated and the page is redirected to present the user dashboard.

**Test Case ID:** 2

**Test Title:** Blog Post Creation

**Test Designed Date:** January 5, 2025

**Pre-conditions:** User has opened the create post page.

**Dependencies:**

*Table 2: Test Case 2*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Description** | **Test Data** | **Expected Result** | **Actual Result** | **Status (Pass/Fail)** |
| User fills all the input fields. | Title: ReactJS | The post is created successfully. | The post is created successfully | Pass |
|  | Tags: Programming  Category: Tech, Programming |  |  |  |
| Click on the create post |  |  |  |  |
| button. |  |  |  |  |
|  |  |  |  |  |

**Post-condition:**

The user’s post has been created.

**Test Case ID:** 3

**Test Title:** Delete user’s post

**Test Designed Date:** January 5, 2025

**Pre-conditions:** UserPostmust be deleted.

**Dependencies:**

*Table 3: Test Case 3*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Description** | **Test Data** | **Expected Result** | **Actual Result** | **Status (Pass/Fail)** |
| Go to post page. | Do you want to delete the blog? | The post is removed. | The post is removed. | Pass |
| Click on the delete |  |  |  |  |
| button. |  |  |  |  |
| . |  |  |  |  |
|  |  |  |  |  |

**Post-condition:**

The post is removed.

**Test Case ID:** 4

**Test Title:** Like/Comment/Share Process

**Test Designed Date:** January 5, 2025

**Pre-conditions:** User should be logged in.

**Dependencies:**

*Table 4: Test case 4*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Description** | **Test Data** | **Expected Result** | **Actual Result** | **Status (Pass/Fail)** |
| Provide all the user login credentials. | Blog Post: Increment Like, Share, Comment | The like count should increment when the "Like" button is clicked and decrement when "Unlike" is clicked.  . | The like count should increment when the "Like" button is clicked and decrement when "Unlike" is clicked. | (Pass) |
| User tries to like/unlike, share and comment on a post |  | The comment should be successfully added to the post and displayed | The comment should be successfully added to the post and displayed |  |
| Click on Like, Comment and share button |  |  |  |  |

**Post-condition:**

The admin should review the post and the user details page of the user .

**Test Case ID:** 5

**Test Title:** Graph Data Accuracy

**Test Designed Date:** January 5, 2025

**Pre-conditions:** User has several blog posts with recorded likes, shares, and comments.

*Table 5: Test Case 5*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Description** | **Test Data** | **Expected Result** | **Actual Result** | **Status (Pass/Fail)** |
| Manually calculate the number of likes, shares, and comments from the database | Post1: Like:10, Shares:5, Comment:2  Post2:Like:2,Shares:0, Comment:1 | Data in the graph matches the manual count from the database | Data in the graph matches the manual count from the database | Pass |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**Post-condition:**

The graph data is up-to-date and visually represents the interaction trends.

# Chapter 5: Conclusion and Future Recommendation

## Lesson Learnt / Outcome

In the WriteOn project, we learned how to manage a project using the MERN stack (MongoDB, Express.js, React.js, Node.js). We built a full-stack application, which helped us understand both front-end and back-end development better. Creating system diagrams also helped us plan and visualize the workflow. This project gave us valuable experience and prepared us to handle similar projects in the future with confidence.

## Conclusion

WriteOn is a blogging platform built using the MERN stack (MongoDB, Express.js, React.js, Node.js). It allows users to easily create, manage, and share their blog posts. With features like user profiles, comments, and content management, WriteOn provides a seamless and interactive blogging experience. The platform demonstrates the power of the MERN stack in building a flexible and scalable content management system. Additionally, the project includes various system diagrams, such as use case, ER, activity, and sequence diagrams, which help illustrate the platform’s functionality and overall workflow.

## Future Recommendations

The project will be upgraded and extended with the following features:

* + - Enhance the design, implementation, and documentation to make it easier for anyone to use the project and improve its performance.
    - Make the web page lighter so that it loads smoothly even with slow internet connections
    - The web application will be brought to android application.
    - Integrate email and SMS notification for users.
    - Continuously evolve the system based on user feedback.
    - Add a payment system for premium features or content monetization..

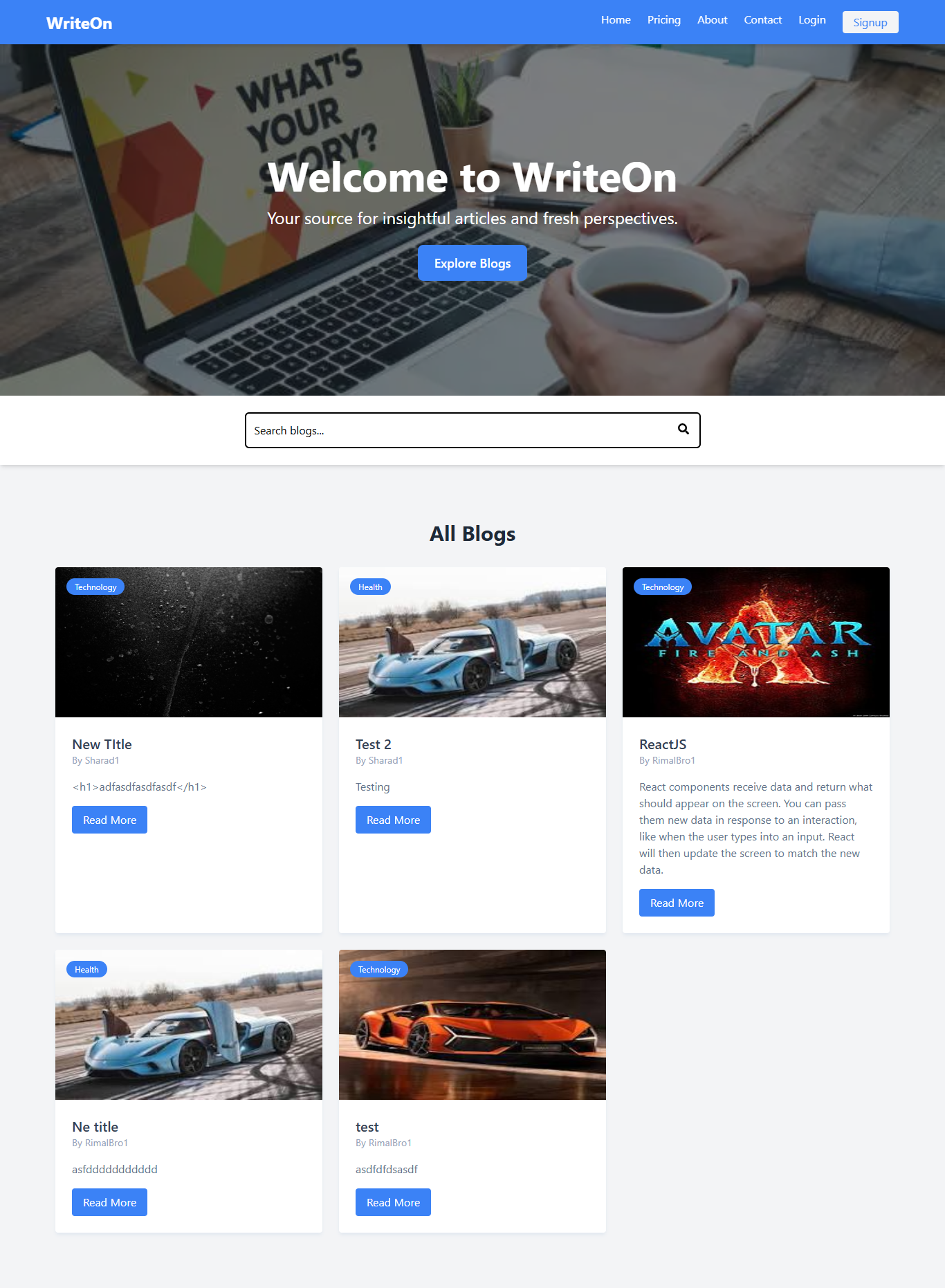
# References

|  |  |
| --- | --- |
| [1] | G. Thevenot, "Blogging as a Social Media," 2007. |
| [2] | J. S. Q. Y. Z. C. D Shen, "Latent friend mining from blog data," 2006. |
| [3] | P. M. H. a. S. P. J. T. Child, Blogging privacy rule orientations, privacy management, and content deletion practices, 2012. |

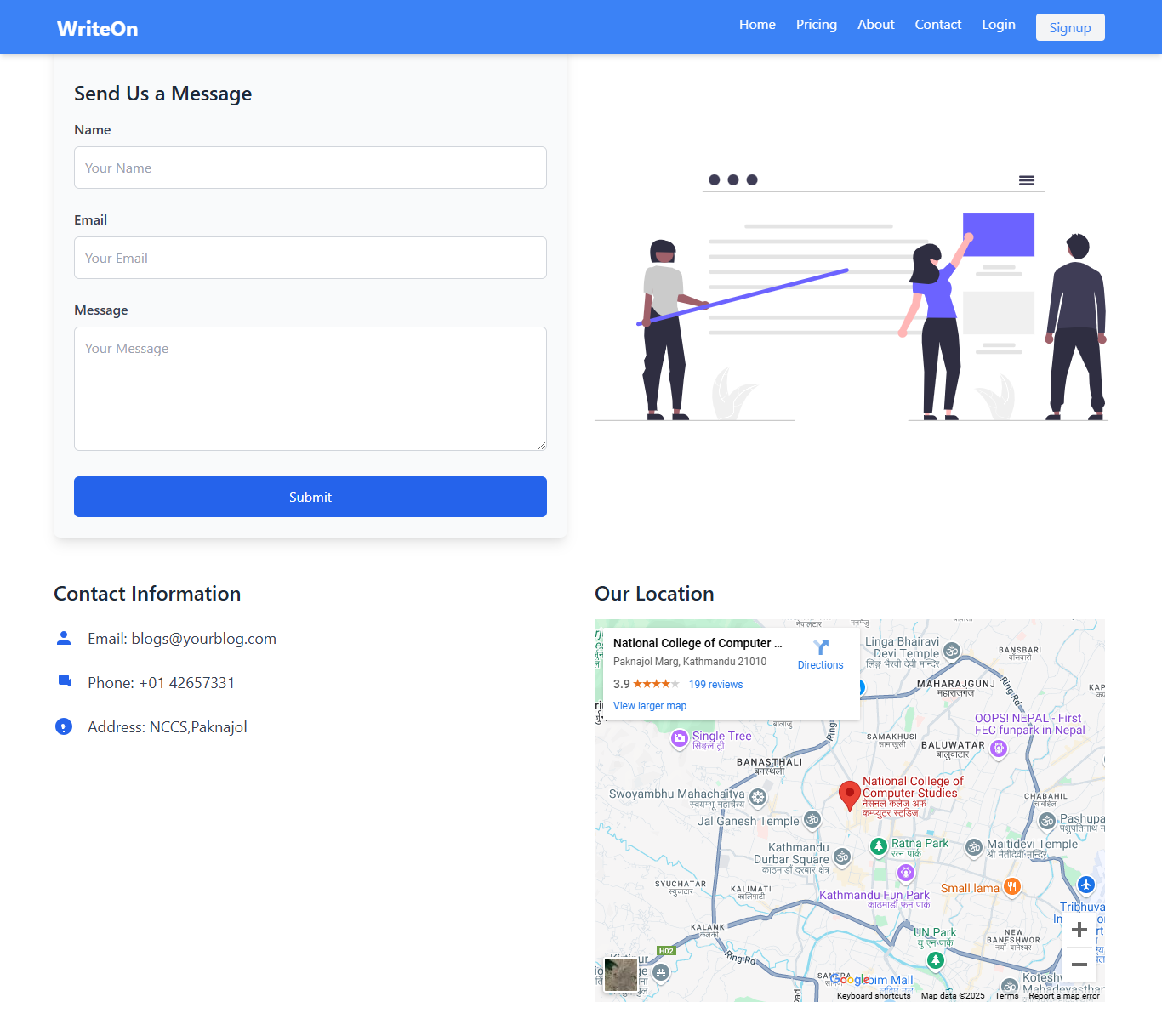
# 

# Appendices

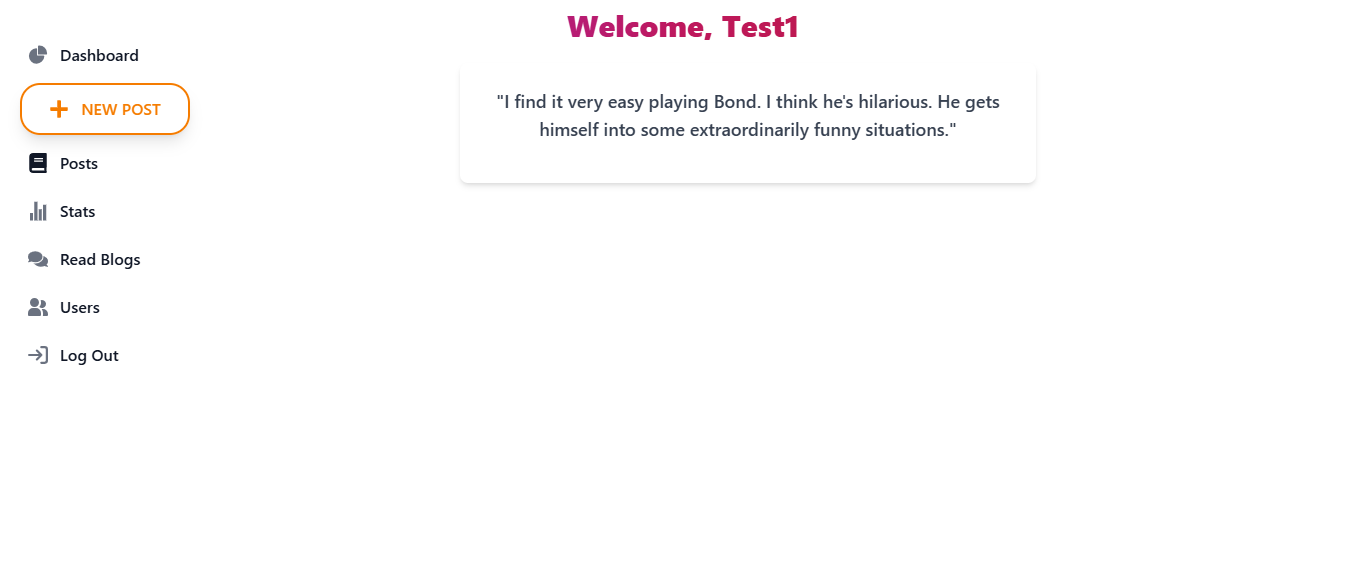
**Snapshots of Output:**

****

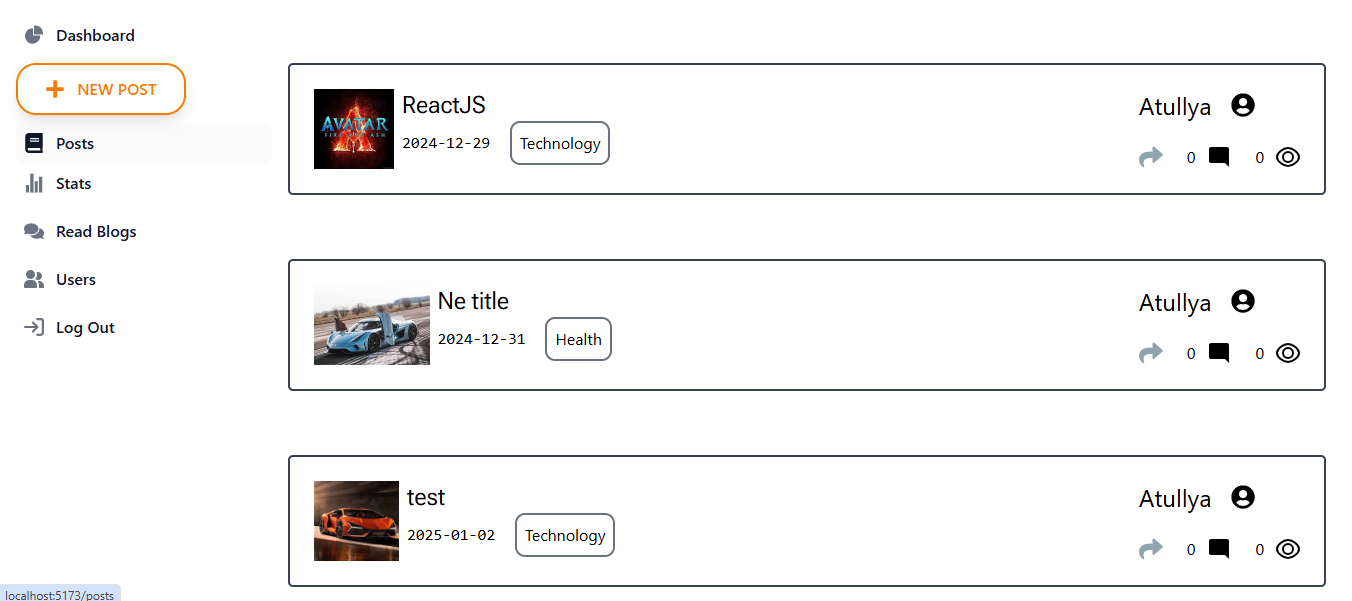
*Figure 8: Homepage of WriteOn*



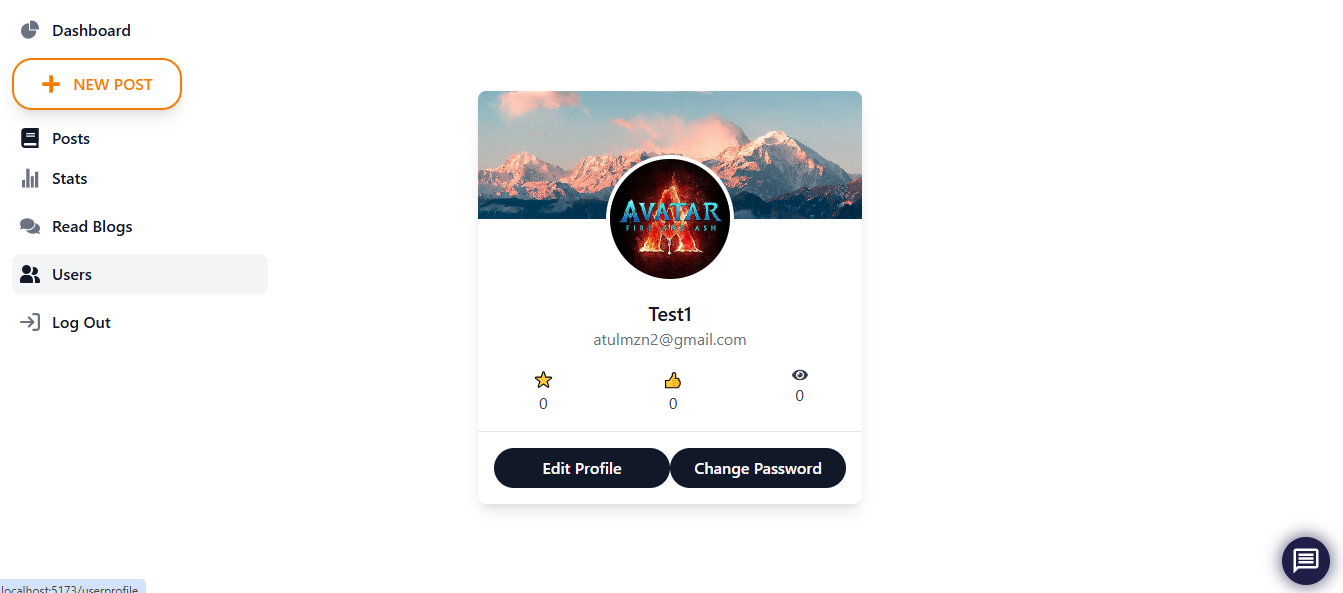
*Figure 9: Contact Page of WriteOn*

**

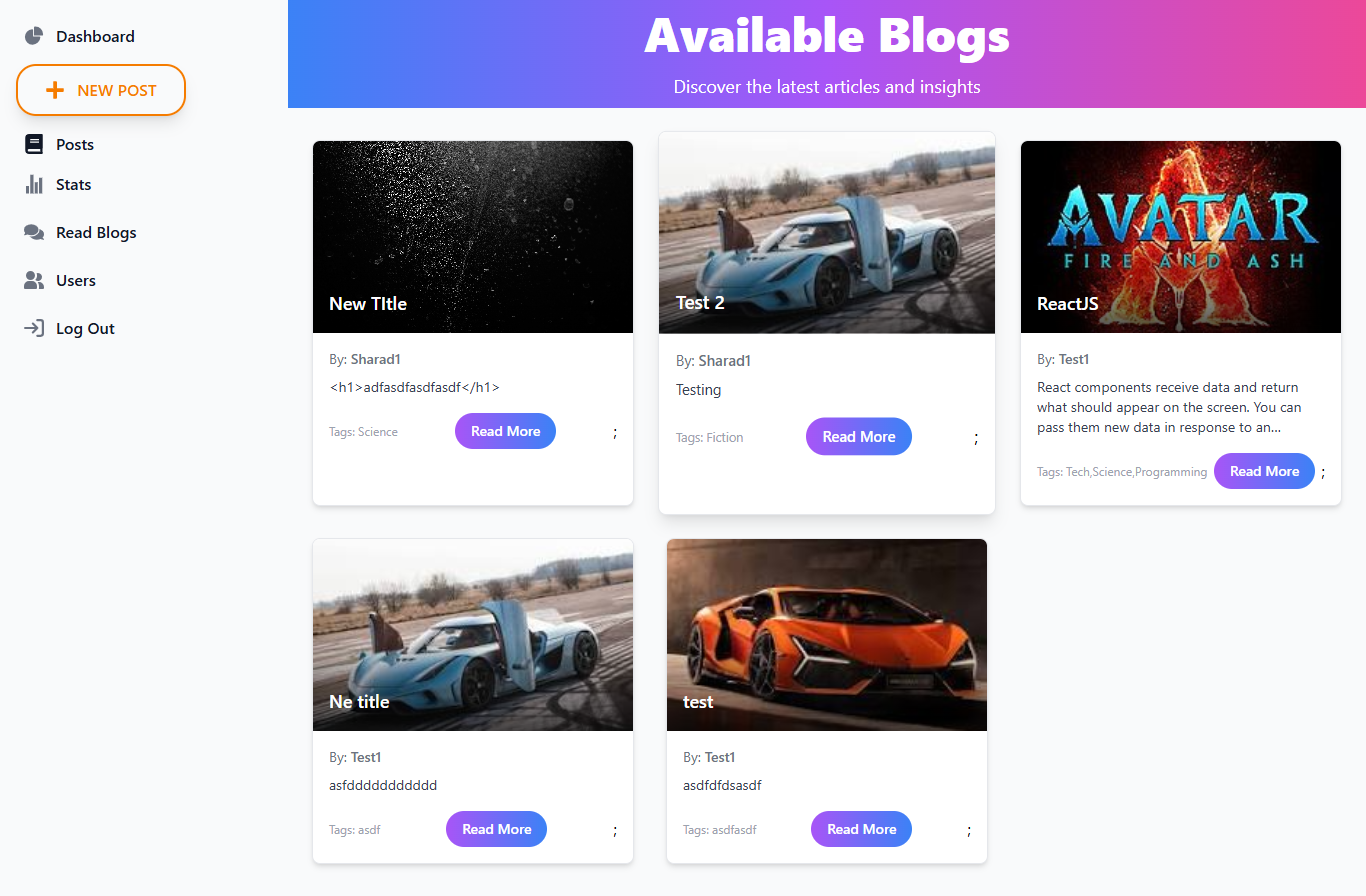
*Figure 10:User Home Page of WriteOn*



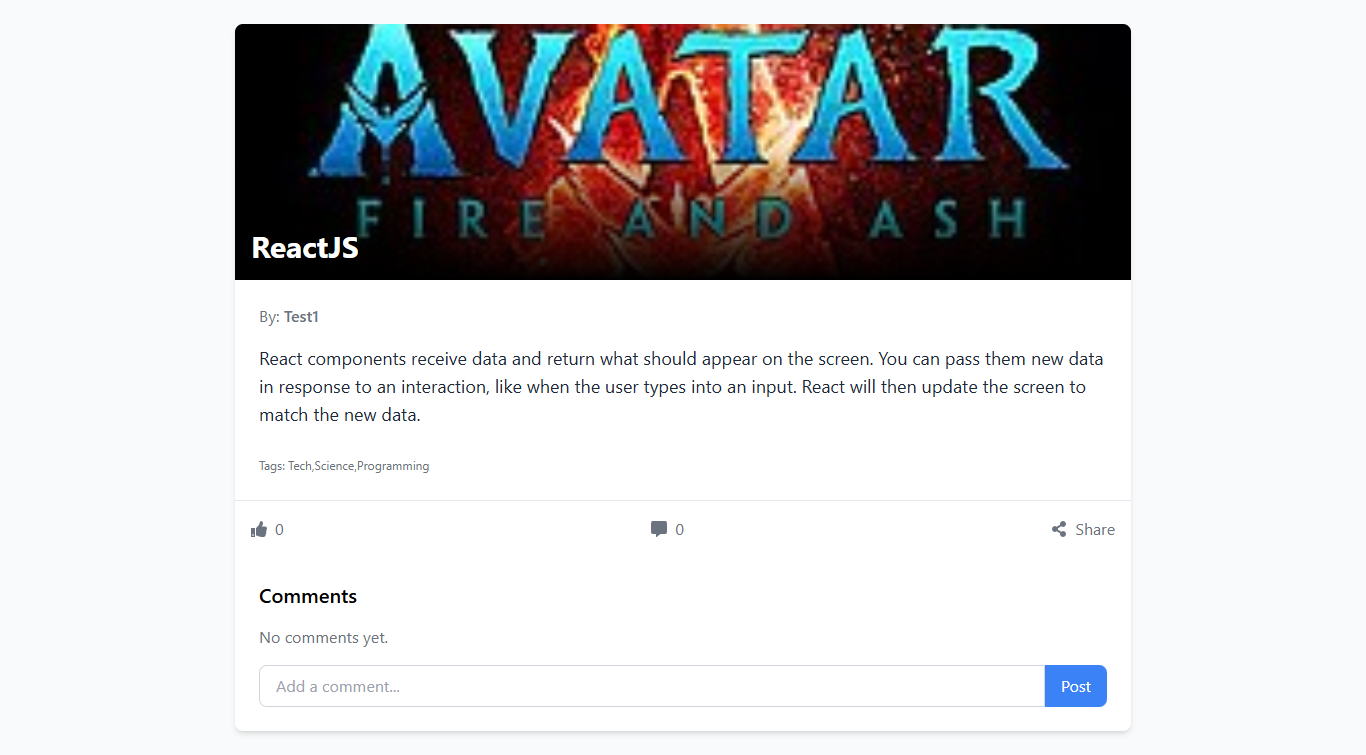
*Figure 11: User Post Page of WriteOn*



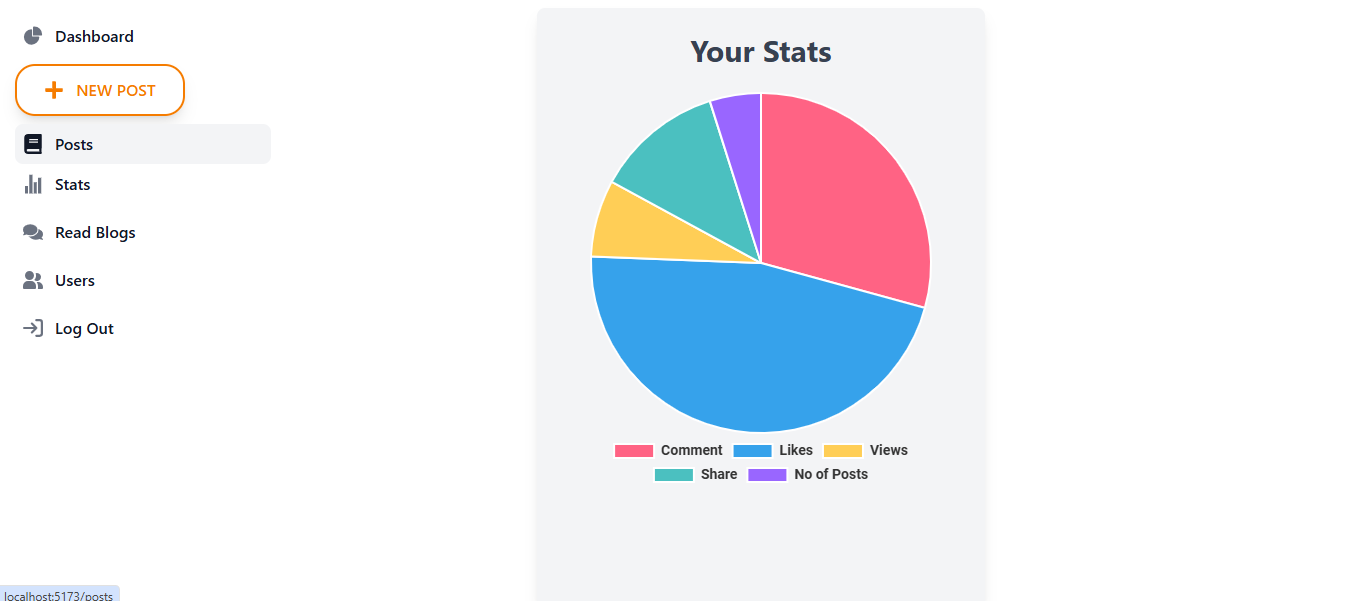
*Figure 12: User Profile of WriteOn*

**

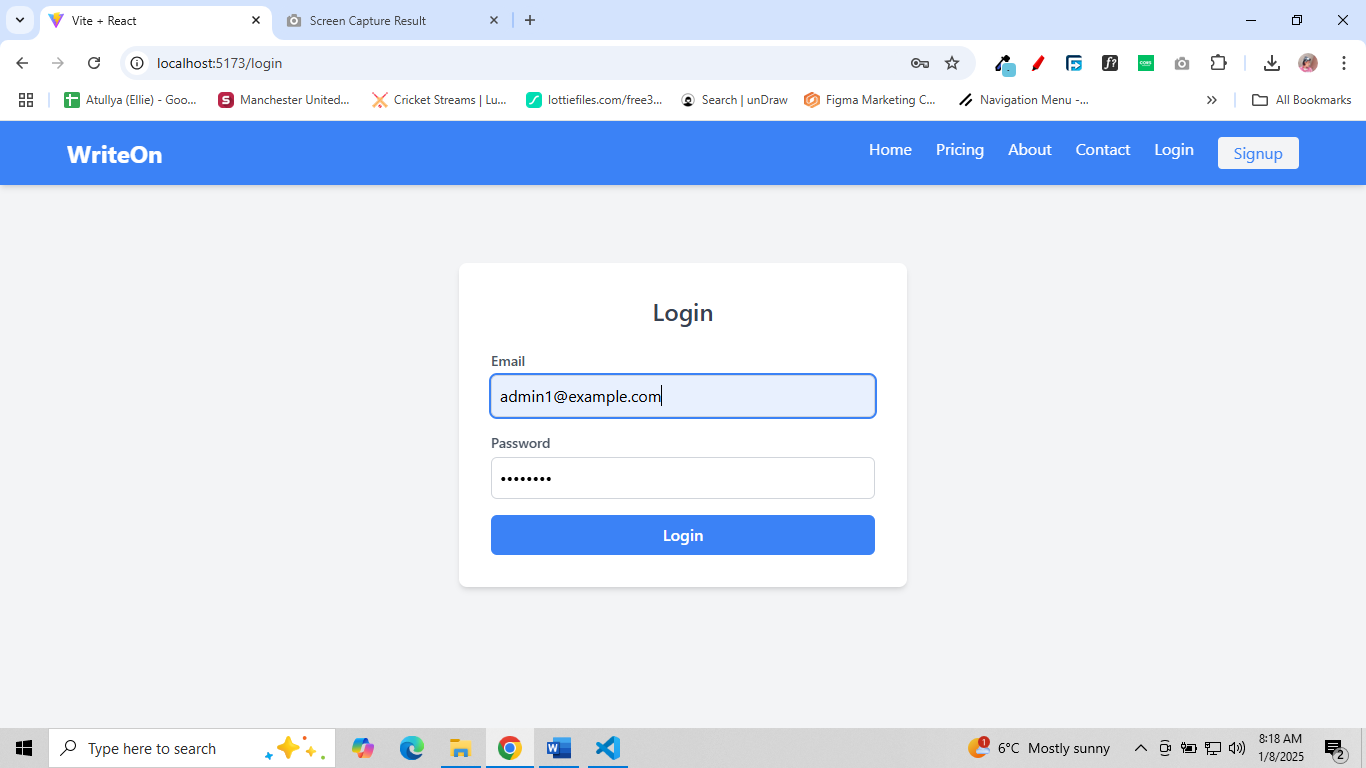
*Figure 13: Quick Blog View of WriteOn*



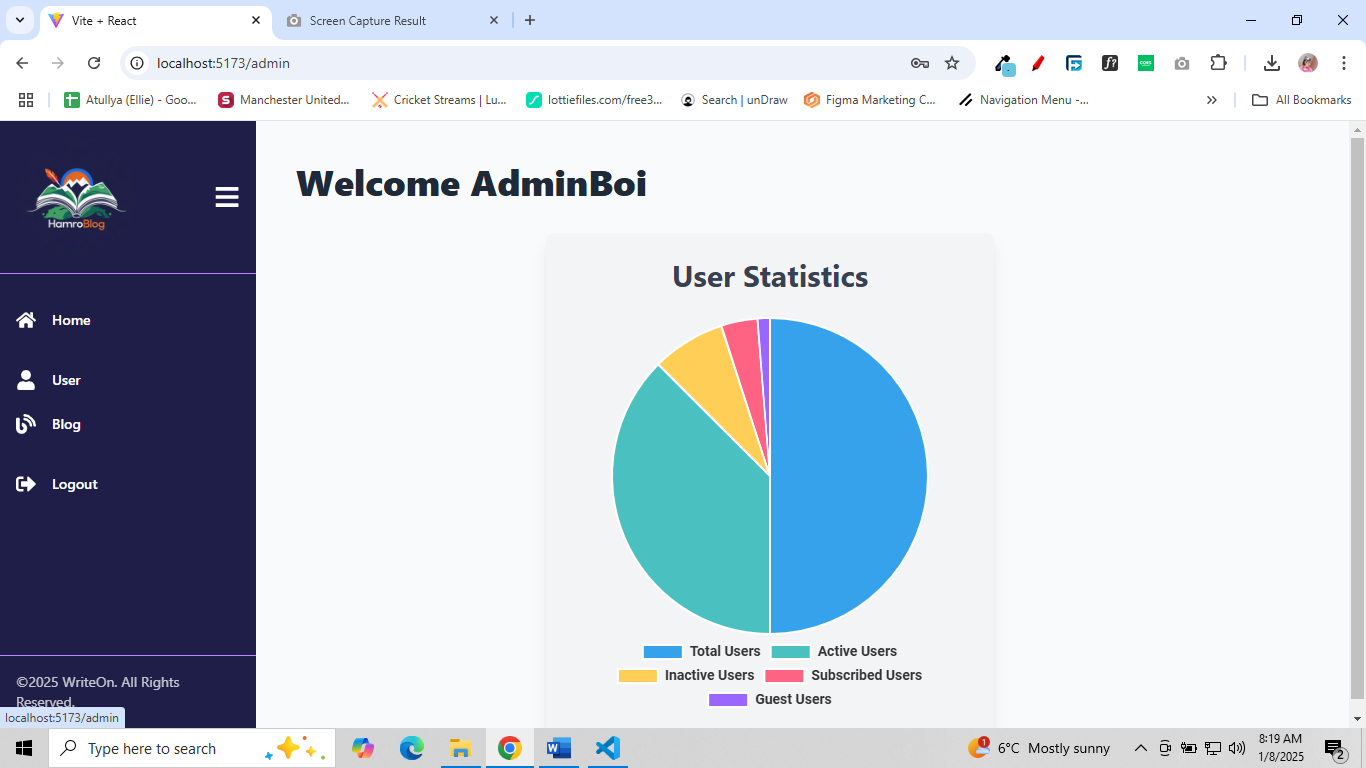
*Figure 14: Blog preview Page of WriteOn*



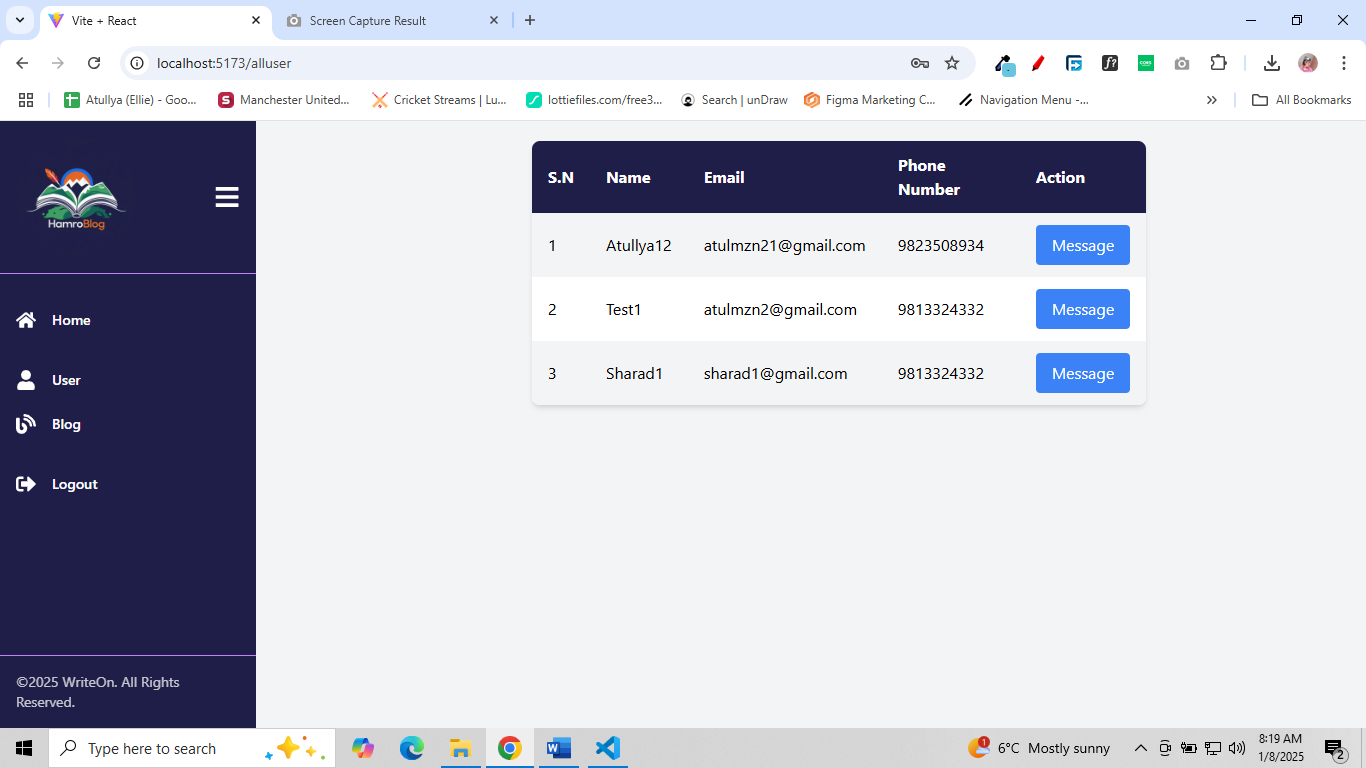
*Figure 15:User Stat Page of WriteOn*



*Figure 16: Login Page of WriteOn*



*Figure 17: Admin Dashboard of WriteOn*



*Figure 18: Available User of WriteOn*