COLLEGE CODE: 8203

COLLEGE: A.V.C COLLEGE OF ENGINEERING

DEPARTMENT: INFORMATION TECHNOLOGY

STUDENT NM-ID: 355D90D88AE7822824F49869175ED7E6

ROLL NO: 23IT101

DATE:08/09/2025

Completed the project named as Phase1

TECHNOLOGY PROJECT NAME: Blogging Platform

SUBMITTED BY,

NAME:SONA SRI V

MOBILE NO: 7845869297

Problem Statement:

Current blogging platforms often lack flexibility, customization, and developer control. Many provide limited markdown support, restricted API usage, and complex content management. This creates challenges for students, developers, and organizations who want a simple, secure, and API-driven blogging solution.

The proposed **Blogging Platform** will provide:

- Secure **user authentication and authorization**.
- Easy **blog creation, editing, and deletion** with markdown support.
- **CRUD operations** via REST APIs.
- Commenting support for community engagement.
- Scalable backend using **Node.js, Express, and MongoDB**.

Users and Stakeholders:

Users:

- **Bloggers** – register/login, create blogs with Node.js-powered APIs.
- **Readers** – view and comment on posts stored in MongoDB.
- **Admins** – monitor blogs using protected Node.js routes.

Stakeholders:

- **Students & Developers** – to practice full-stack development with Node.js.
- **Educational Institutions** – internal blogging/knowledge sharing.
- **Organizations** – employee-driven communication.

- **General Public** – open blogging and discussion platform.

User Stories:

- **As a blogger:** I want to **register/login securely** so that I can manage my blog posts.
- **As a blogger:** I want to **create blog posts with Markdown support** so that I can format my content professionally.
- **As a blogger:** I want to **edit or delete my blogs** so that I can update or remove content as needed.
- **As a reader:** I want to **view all blogs** so that I can read and explore different topics.
- **As a reader:** I want to **comment on blogs** so that I can provide feedback or ask questions.
- **As a registered user:** I want to **save drafts** so that I can continue writing later.
- **As an admin:** I want to **moderate blogs and comments** so that the platform remains safe and appropriate.

Stakeholder Stories:

- **Project Owner:** I want to **ensure the platform is secure and functional** so that users have a reliable experience.
- **Developer:** I want to **build the backend using Node.js, Express, MongoDB, and JWT** so that the system is scalable and secure.
- **UI/UX Designer:** I want to **create intuitive wireframes and layouts** so that users can navigate easily.

- **Content Moderator:** I want to **review and approve posts and comments** so that inappropriate content is prevented.
- **Educational Institution:** I want to **use the platform for knowledge sharing** so that students and faculty can share blogs internally.
- **Organization:** I want to **track and maintain blogs for internal communication** so that employees can stay updated.
- **Sponsor/Investor:** I want to **see a professional, functional platform** so that my investment is well-utilized.

MVP (Minimum Viable Product) Features:

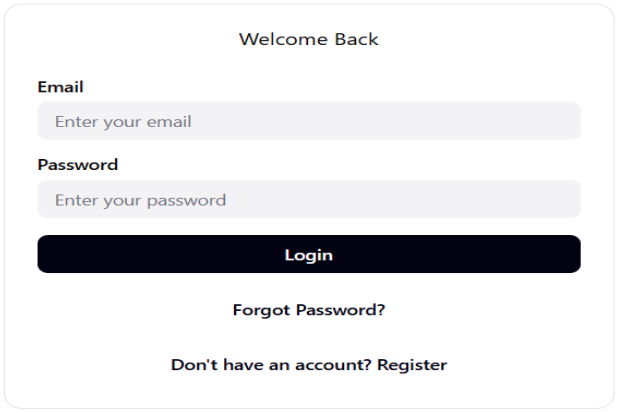
MVP Features for Blogging Platform

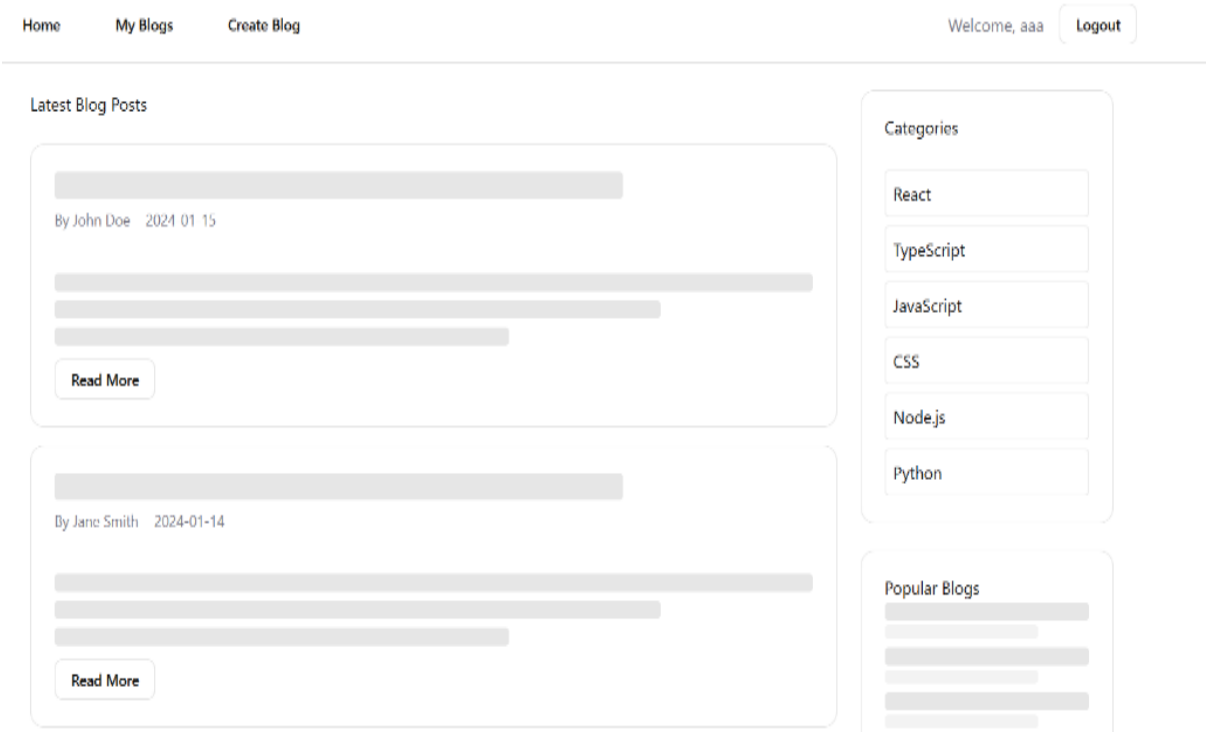
- **Node.js & Express.js Backend:** Provides a scalable, non-blocking server for handling user requests.
- **MongoDB Database with Mongoose:** Efficiently stores blogs, users, and comments.
- **JWT Authentication:** Ensures secure login and protects blog creation/editing routes.
- **CRUD Operations for Blogs:** Users can **Create, Read, Update, and Delete** blogs via REST APIs.
- **Markdown Support:** Bloggers can format their content professionally with headings, bold/italic text, links, and lists.
- **Comment System:** Readers can comment on blogs using a nested schema or separate collection.

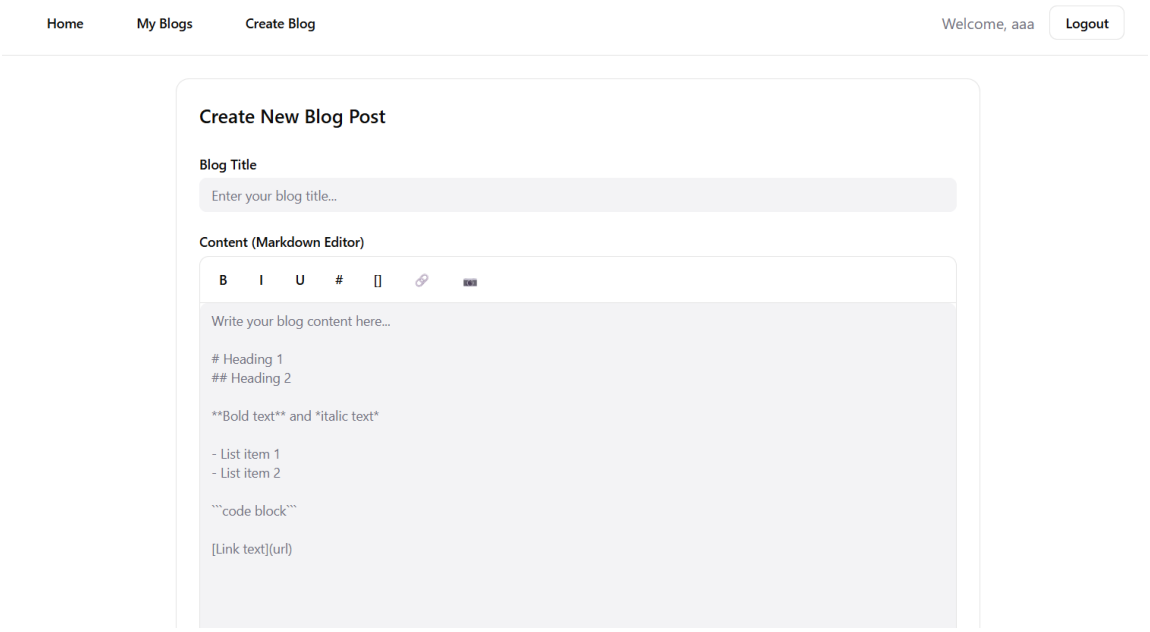
- **Draft Saving Feature:** Users can save blogs as drafts and continue editing later.
- **Responsive Design:** Platform works well on both desktop and mobile devices.

Wireframes & API Endpoint List:

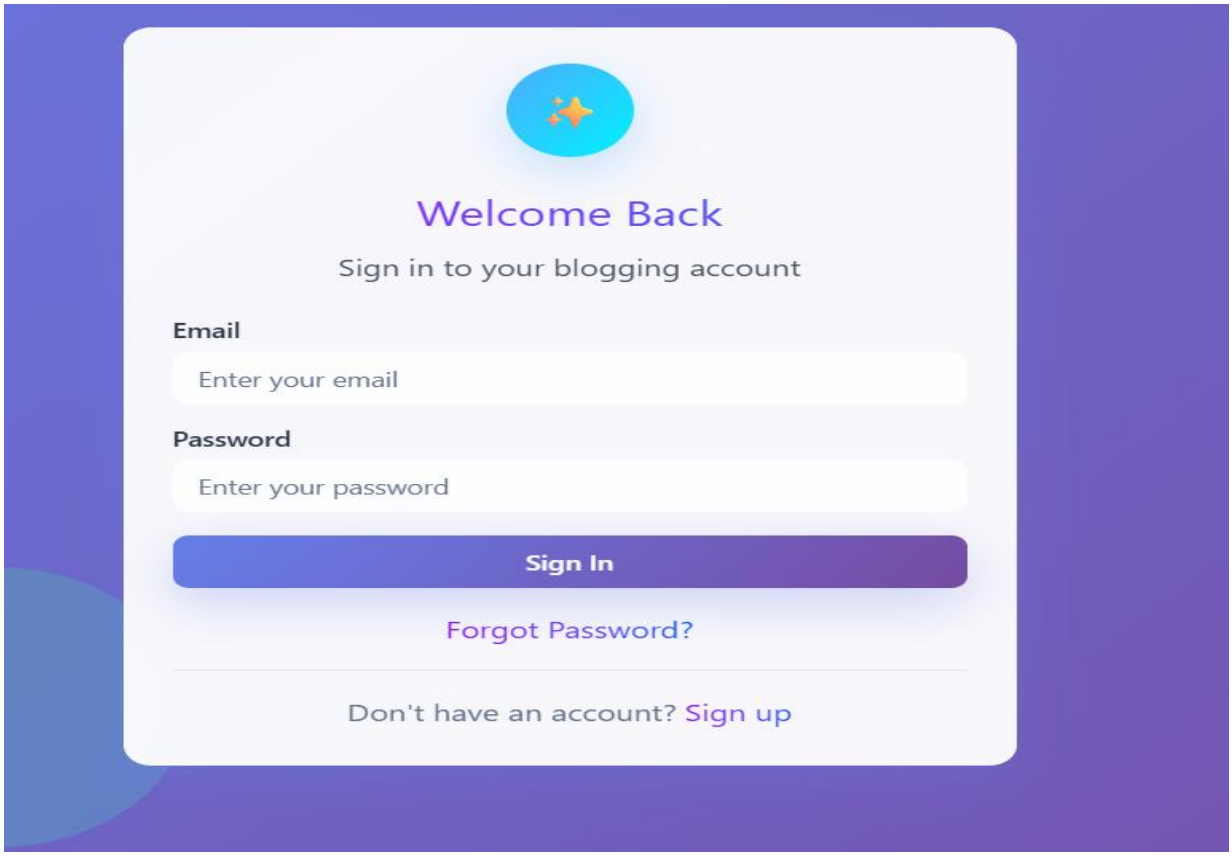
Mid-Fidelity (Mid-Fi) Wireframes:

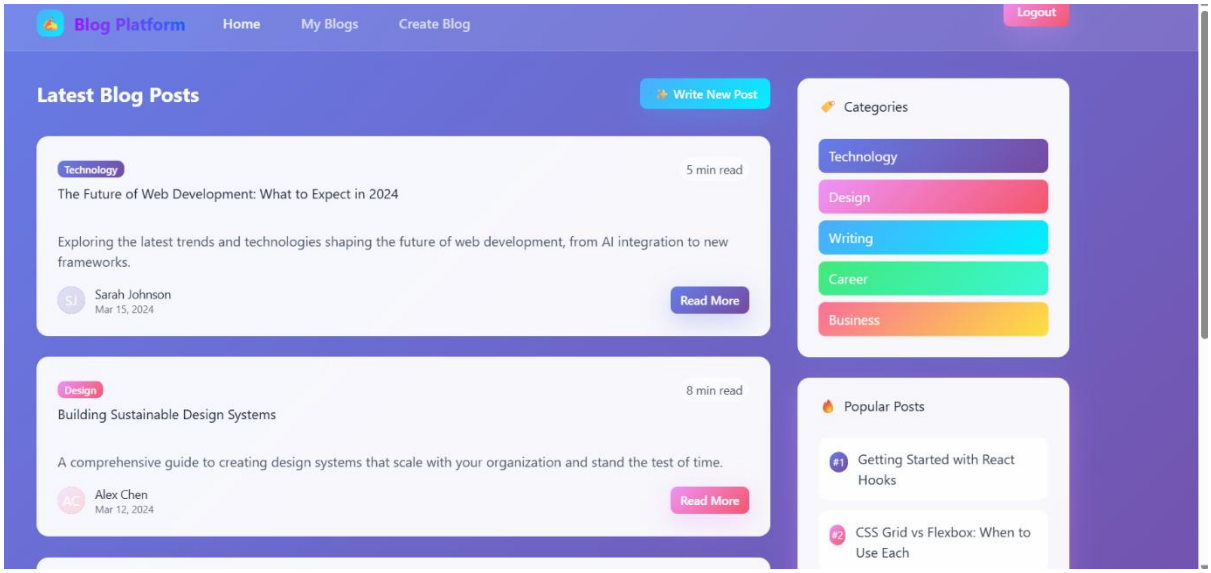


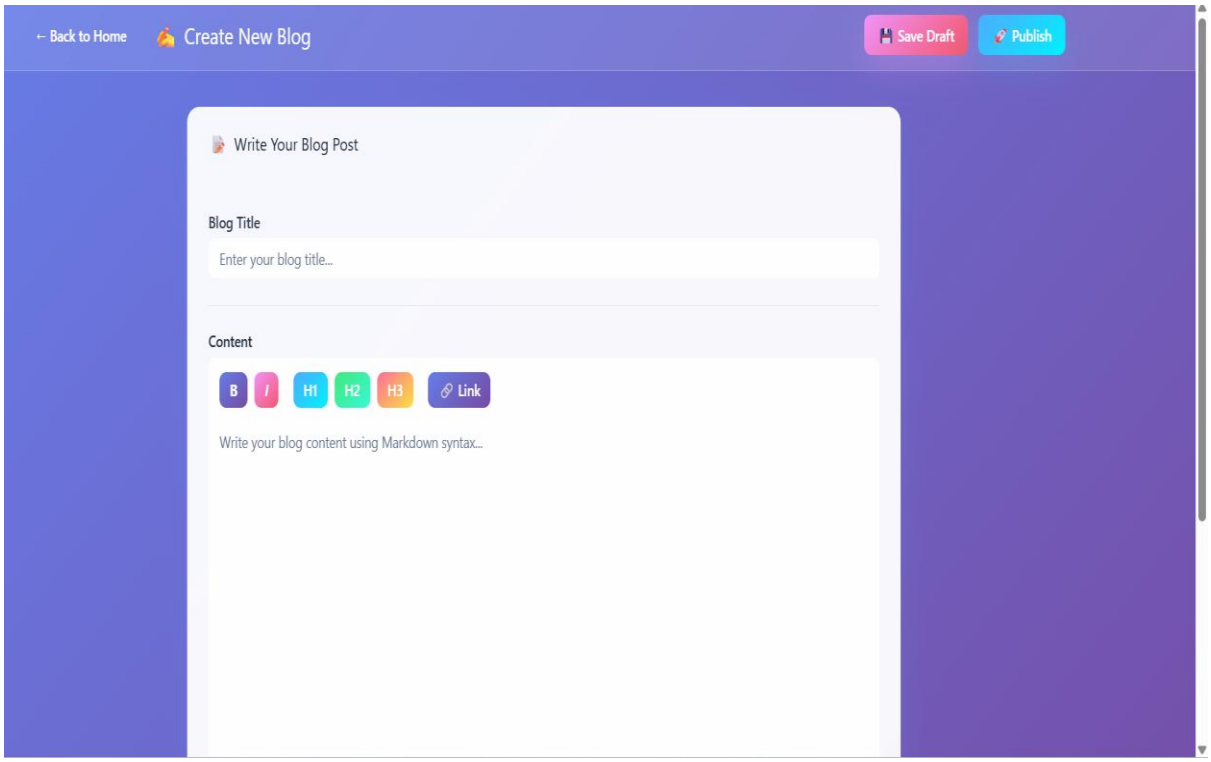




High-Fidelity (Hi-Fi) Wireframes:




Blog Platform
Home
My Blogs
Create Blog
Logout
Latest Blog Posts
Write New Post
Technology
5 min read
The Future of Web Development: What to Expect in 2024
Exploring the latest trends and technologies shaping the future of web development, from AI integration to new frameworks.
Sarah Johnson
Mar 15, 2024
Read More
Design
8 min read
Building Sustainable Design Systems
A comprehensive guide to creating design systems that scale with your organization and stand the test of time.
Alex Chen
Mar 12, 2024
Read More
Categories
Technology
Design
Writing
Career
Business
Popular Posts
#1
Getting Started with React Hooks
#2
CSS Grid vs Flexbox: When to Use Each


← Back to Home
Create New Blog
Save Draft
Publish
Write Your Blog Post
Blog Title
Enter your blog title...
Content
B
I
H1
H2
H3
Link
Write your blog content using Markdown syntax...

API Endpoint List:

Method	Endpoint	Description
POST	/api/auth/register	User registration
POST	/api/auth/login	User login, JWT returned
GET	/api/blogs	Get all blogs
POST	/api/blogs	Create new blog (Node.js + MongoDB)
PUT	/api/blogs/:id	Update blog
DELETE	/api/blogs/:id	Delete blog
POST	/api/blogs/:id/comments	Add comment
GET	/api/blogs/:id/comments	Get comments for a blog

Acceptance Criteria:

User Authentication

- Users must **register and login successfully**.
- JWT token must be returned on login to access protected routes.

Blog Management

- Users (authors) can **create, edit, and delete their own blogs**.
- Blogs must support **Markdown formatting** for headings, lists, and links.
- Drafts can be **saved and resumed later**.

Blog Viewing

- Readers can **view all blogs**.
- Individual blogs show **title, author, date, content, and comments**.

Comments

- Users can **add comments** to blogs.
- Comments are displayed in chronological order below each blog.

Authorization & Security

- Only **authenticated authors** can edit or delete their blogs.
- Protected routes cannot be accessed without a valid **JWT token**.

Responsiveness

- Platform must work on **desktop and mobile devices**.

Admin Moderation

- Admin can **moderate blogs and comments** to prevent inappropriate content.