COLLEGE CODE : 9111

COLLEGE NAME : SRM Madurai College for Engineering and Technology

DEPARTMENT : Computer Science and Engineering

STUDENT NM-ID : E8833FF2599686597AC4E10467D37FFD

ROLL NO : 911123104047

DATE : 15/09/2025

Completed the project named as Phase 1

TECHNOLOGY PROJECT NAME : IBM-FE-Blog Site with Comment Section

SUBMITTED BY,

NAME : SANJAY KARTHI R P

MOBILE NO : 9360463606

Project Charter: Blog Site with Comment Section

Phase 1: Project Initiation & Requirements Definition

This document outlines the foundational elements for the development of a professional-grade blog platform. The primary objective is to create a robust and intuitive system that facilitates content creation and enhances user engagement through a dynamic comment section. This phase focuses on a comprehensive understanding of the project's scope, identifying key stakeholders, and defining the core requirements necessary for the successful delivery of a Minimum Viable Product (MVP).

1.0 Problem Statement

The digital landscape is saturated with blog platforms that often present significant barriers to entry for new authors or lack the essential community-building features that drive reader engagement. This project addresses this gap by developing a streamlined, user-centric platform that simplifies the content publication process for authors and provides a seamless, interactive experience for readers. The core problem is to build a highly functional, reliable, and scalable blog site that serves as a central hub for sharing content and fostering community dialogue, without the complexities or limitations of existing solutions.

2.0 Users & Stakeholders

The success of this project hinges on meeting the needs of all parties involved. This project defines two primary user personas and a clear set of project stakeholders.

2.1 User Personas

- **Content Creators (Authors):** Individuals or entities responsible for generating and publishing blog posts. Their primary needs include a frictionless content management system, a reliable editor, and the ability to monitor and manage engagement on their published work.
- **Content Consumers (Readers):** The target audience for the content. Their requirements center on ease of discovery, an excellent reading experience, and the ability to interact with the content and other users through a dedicated comment system.

2.2 Project Stakeholders

- **Project Sponsor/Product Owner:** The ultimate decision-maker responsible for defining the project's vision, prioritizing features, and ensuring the final product aligns with strategic business objectives.
- **Development Team:** The technical personnel responsible for architecting, coding, and deploying the application. Their focus is on technical feasibility, scalability, and code quality.
- **Design & UX Team:** Responsible for creating a visually appealing and intuitive user interface and experience that ensures maximum usability for both authors and readers.
- **Quality Assurance (QA) Team:** Tasked with verifying that the application meets all functional and non-functional requirements and is free of defects prior to launch.

3.0 User Stories

The following user stories provide a detailed, functional blueprint for the application's features, written from the perspective of the end-user.

3.1 Author Stories

- **Content Creation:** As an author, I can **create, edit, and publish** new blog posts with a title, body, and metadata so that I can share my original content with a broad audience.
- **Content Management:** As an author, I can **access a dashboard to view and manage** all my published posts so that I can easily track and modify my work.
- **Engagement Tracking:** As an author, I can **view all comments associated with my posts** so that I can understand reader feedback and engage in the conversation.

3.2 Reader Stories

- **Content Discovery:** As a reader, I can **browse a list of all blog posts** on the homepage so that I can discover content relevant to my interests.
- **Content Consumption:** As a reader, I can **access and read a full blog post** on a dedicated page so that I can consume the entire article without distraction.
- **Engagement:** As a reader, I can **submit a comment on a blog post** with my name and a message so that I can share my thoughts and contribute to the discussion.
- **Community Viewing:** As a reader, I can **see a chronological list of all comments** on a post so that I can understand the ongoing conversation.

4.0 Minimum Viable Product (MVP) Scope

The MVP represents the foundational feature set required for the initial launch. It is designed to be a fully functional product that addresses the core problem statement while allowing for future iterations and enhancements.

4.1 Core Functionality

- **Post Management System:** A backend system enabling authors to create, read, update, and delete (CRUD) blog posts. This includes support for a post title and body.
- **Public-Facing Blog Interface:** A homepage displaying a list of all published posts and individual pages for each post.
- **Comment System:** The ability for any user to leave a comment on a post. Comments will be displayed under the post content.
- **Data Persistence:** A database to securely store blog posts and their associated comments.

5.0 Technical Specifications: API Endpoint List

The application will be built on a RESTful API architecture. The following endpoints will serve as the communication layer between the frontend and the backend services.

5.1 Posts

- `GET /api/posts`: Retrieves a paginated list of all blog posts, sorted by publication date.
- `GET /api/posts/{postId}`: Fetches the details of a single blog post.
- `POST /api/posts`: Creates a new blog post. Requires authentication and a valid JSON payload (`title`, `body`).

- `PUT /api/posts/{postId}`: Updates an existing blog post. Requires authentication and ownership verification.
- `DELETE /api/posts/{postId}`: Deletes a blog post. Requires authentication and ownership verification.

5.2 Comments

- `GET /api/posts/{postId}/comments`: Retrieves a list of all comments for a specific post.
- `POST /api/posts/{postId}/comments`: Submits a new comment on a specific post. Requires a valid JSON payload (`author_name`, `comment_body`).

6.0 Acceptance Criteria

Each feature must meet the following criteria to be considered complete and ready for deployment.

6.1 User-Facing Criteria

- **Homepage Display:** The homepage must load within 3 seconds and display a list of all posts with their respective titles and a brief excerpt.
- **Post Viewing:** Clicking a post's title must navigate the user to a dedicated page that fully renders the post content and its associated comments.
- **Comment Submission:** A reader must be able to submit a comment with their name and message. Upon successful submission, the comment must appear at the end of the comment list without requiring a page refresh.

6.2 Author-Facing Criteria

- **Post Creation:** The post creation form must include validation for required fields (title, body) and provide a confirmation message upon successful submission. The new post must be immediately viewable on the homepage.
- **Post Editing:** The editing interface must be pre-populated with the existing post data. Any saved changes must be reflected across all public-facing views of the post.
- **Post Deletion:** An author must be able to permanently delete their post. A confirmation dialog must be presented to prevent accidental deletion, and upon confirmation, the post must be removed from the site.

This detailed plan provides a solid foundation for the subsequent phases of design, development, and testing. It ensures all team members have a shared understanding of the project's goals and requirements, thereby mitigating risks and streamlining the development process.