Blog Website

Name: Pranav Unnikrishnan

Register Number: 24PMC141

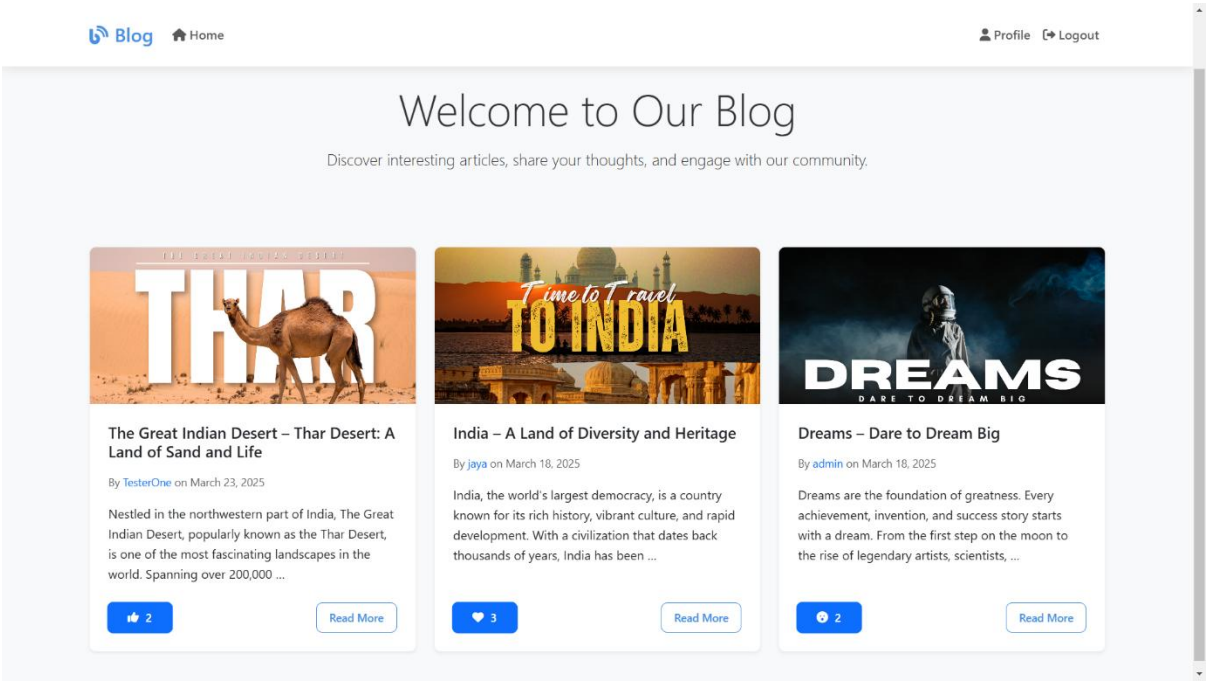
GitHub Repository: https://github.com/Pranav-un/Blogs-Django

Customization Details:

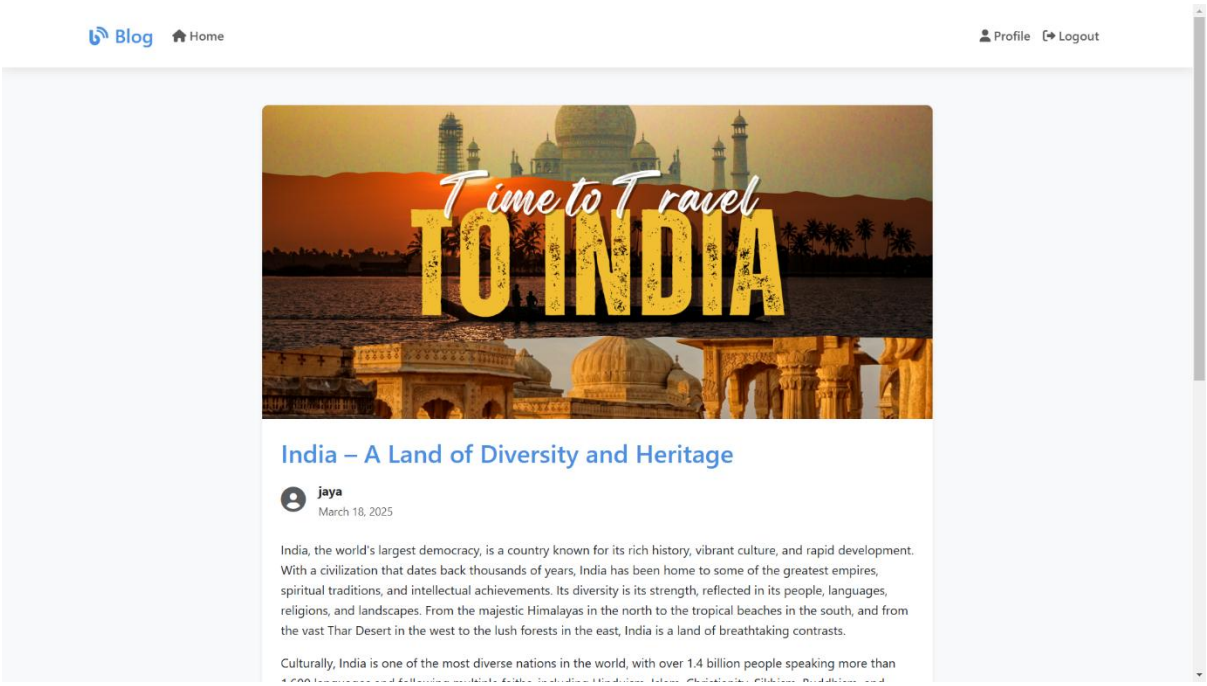
Additional Features Implemented:

1. Facebook-Style Reactions:
    - Integrated reaction buttons similar to Facebook (Like, Love, Wow) for each blog post.
    - Stored user reactions in the database and allowed users to change their reaction.
    - Displayed reaction counts dynamically next to each post, updating in real-time without page reload using AJAX.
2. Image Uploads:
    - Enabled users to upload images with their blog posts.
    - Implemented image validation (file type, size restrictions).
    - Stored uploaded images securely and displayed them properly within the post content.
3. User Profiles with Bio:
    - Added dedicated user profile pages.
    - Displayed user information and bio on all their blog posts for better personalization.
4. Comments on Posts:
    - Implemented a comment system where users can add their comments on blog posts.
    - Displayed all comments under each post in chronological order.

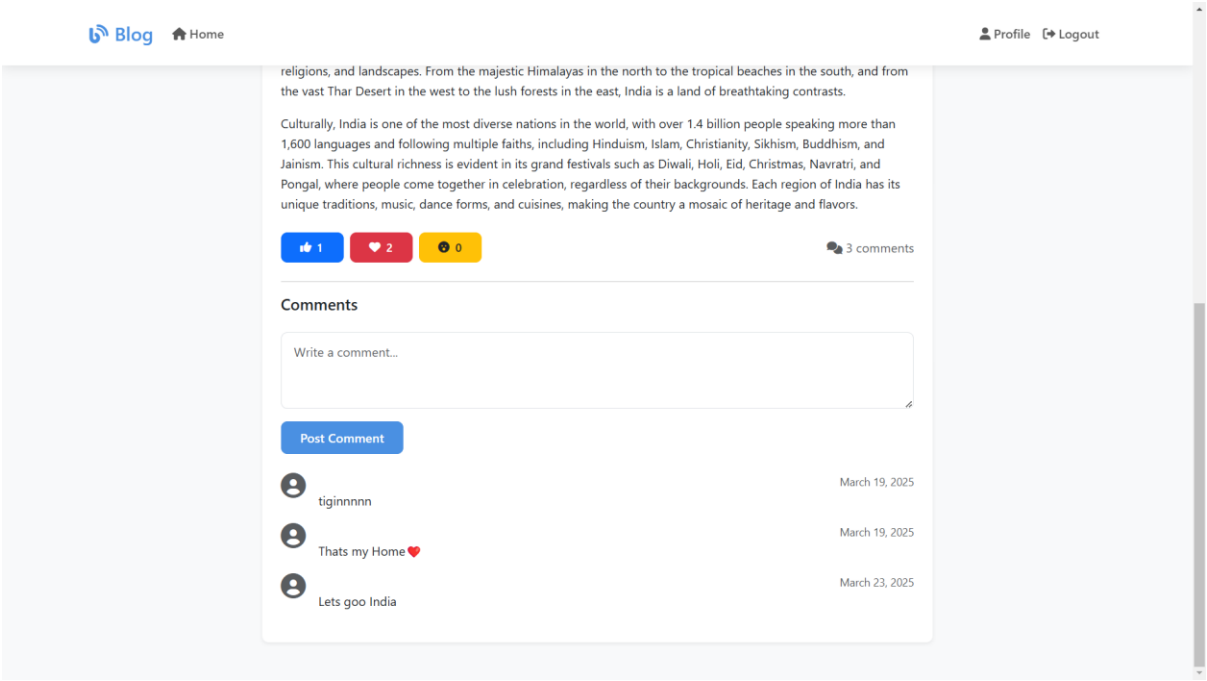
Screenshots



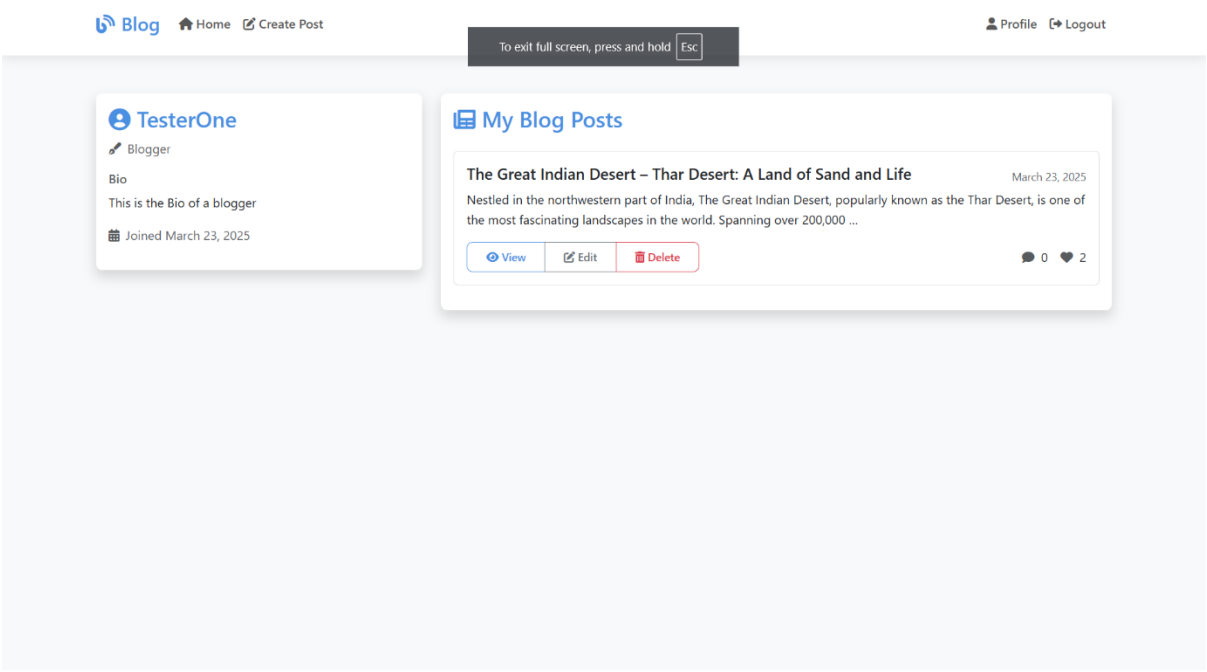
Home Page



Blog Details Page



Reaction and Like



User Profile

Challenges Faced

1. Debugging AI Code: Occasionally, AI-generated code contained minor errors or missed edge cases, leading to runtime issues or integration bugs that had to be debugged and fixed manually.
2. Understanding AI-Generated Code: Initially, it was challenging to fully understand the AI-suggested code snippets, especially when they introduced new patterns or methods unfamiliar to me.

Key Learnings:

1. AI as a Productivity Booster: Learned that AI tools like Cursor AI are great for increasing development speed and reducing boilerplate, but they still require strong foundational knowledge to apply them correctly.