শিক্ষা নিয়ে গড়বো দেশ তথ্য-প্রযুক্তির বাংলাদেশ

Bangabandhu Sheikh Mujibur Rahman Digital University, Bangladesh

Bangabandhu Sheikh Mujibur Rahman Digital University

Project Report

COURSE NO.-PROG302

Course Title: Advanced Programming with Python and Scripting Sessional

Project Title: Blog Application Using Django With Sentiment Analysis

Submitted By:

Mobashira Mehajabin Arpita
ID: 2101008
Rupu Rani Ghosh
ID: 2101010
Mehrin Farzana
ID:2101013
Department of IRE
Session :2021-2022
Bangabandhu Sheikh Mujibur Rahman Digital University, Bangladesh

Submitted To:

Farzana Akter
Assistant Professor
Department of IRE, BDU
Bangabandhu Sheikh Mujibur Rahman Digital University, Bangladesh

Date of Submission: 17 December 2024

Table of Contents

Introduction:

The Blog Application project aims to develop a feature-rich blogging platform using the Python Django framework. This platform is designed to serve as a hub for writers and readers to share ideas and interact effectively. Bloggers will have tools to create, edit, publish, and manage their posts, while readers can provide feedback through a seamless commenting system. A key innovation of this application is its ability to perform sentiment analysis on comments, offering valuable insights to authors about reader perceptions. The platform also ensures a responsive and user-friendly interface, making it accessible across various devices.

Objectives:

The primary objectives of this project are:

- To develop a robust blog application using Python Django, allowing users to create, manage, and publish blog posts.
- To provide a user-friendly front-end interface for submitting blog posts.
- To implement essential blog functionalities such as user authentication and commenting systems.
- To integrate sentiment analysis for user comments using Natural Language Processing (NLP) tools like Hugging Face Transformers.
- To design a responsive and intuitive user interface using modern front-end technologies, ensuring compatibility across devices.

Literature Review:

- Blog Management: Existing solutions like WordPress focus on ease of blog creation but lack integrated sentiment analysis for comments.
- Sentiment Analysis: Studies have shown the effectiveness of pre-trained transformer models, such as DistilBERT, for text classification tasks.
- Rich Text Editing: The django-ckeditor library is widely used in Django projects for enabling rich text editing.

Methodology:

1. Django Blog Platform Setup:
   - Develop user interfaces and basic CRUD functionalities for posts and comments.
   - Implemented user authentication functionalities, enabling users to register, log in, manage profiles, and securely interact with the application.

2. Sentiment Analysis Integration:
   - Integrated sentiment analysis for comments using a pre-trained sentiment-analysis pipeline from Hugging Face Transformers.
   - Processed user comments in real-time to classify them as positive or negative.

3. Frontend Development:
   - Utilized a pre-designed HTML template that incorporates Bootstrap for consistent and professional styling of web pages. Utilized HTML5, CSS3, and JavaScript for building the core structure and styling of web pages.

Technologies and Tools:

Backend Technologies:

- **Python:** The core programming language used for writing the business logic of the application.
- **Django:** The web framework used for the backend.
- **NLP Tools:** Pre-trained sentiment-analysis pipeline from Hugging Face Transformers.
- **Database:** To store User, Category, Blog, Comment, Reply.

Frontend Technologies:

- **HTML5, CSS3, JavaScript:** Core web technologies for rendering and styling the pages.
- **Bootstrap:** For designing a responsive user interface.
- **JavaScript:** For adding interactive features like comment submission without reloading the page.

Integrate Sentiment Analysis:

Library Installation:

Installed Hugging Face Transformers and PyTorch for implementing sentiment analysis.

Pipeline Initialization:
A pre-trained model (distilbert-base-uncased-finetuned-sst-2-english) is used for classifying comments as POSITIVE or NEGATIVE.

Sentiment Analysis Workflow:

- Sentiment analysis is applied to each comment text.
- Results are displayed alongside comments in the blog details view, helping authors assess user feedback.

Expected Outcomes:

1. A fully functional blog platform with core features like user authentication, commenting, and admin management.
2. Sentiment analysis for user comments, offering authors insights into reader feedback and improving engagement.
3. A responsive and intuitive user interface that provides a seamless browsing experience across all devices.

Core Functionalities of the Blog Platform:

1. **User Authentication:** Secure user registration, login, logout, and password reset functionalities.
2. **Admin Panel:** Manage posts, users, comments, and categories through Django's admin interface.
3. **Blog Post Management:** Authenticated users can create, edit, and delete their blog posts using a rich text editor
4. **Commenting System:** Readers can leave comments on blog posts, providing feedback and engaging in discussions.
5. **Responsive Design:** The application will be built with a responsive design to ensure it works smoothly across various devices, including desktops, tablets, and smartphones.
6. **Sentiment Analysis on Comments:** Analyze and display the sentiment of comments (e.g., "Positive," "Negative") to enhance interaction between authors and readers**.**

Testing:

Test Cases:

Blog Creation: Verify blogs are created with rich text formatting.

Comment Submission: Check comments are stored and displayed correctly.

Sentiment Analysis: Validate sentiment labels against sample comments.

Edge Cases: Handle empty or invalid comments.

Challenges:

Integration of pre-trained models with Django.

Future Work:

Extend sentiment analysis to support multi-language comments.

Add advanced NLP features like emotion detection or topic modeling.

Conclusion:

This project successfully delivered a comprehensive blog application built using Python Django. The platform not only met its objectives of providing core blogging functionalities but also elevated user engagement through the integration of sentiment analysis using Hugging Face Transformers. By leveraging modern front-end and back-end technologies, the application offers a secure, interactive, and user-friendly experience for bloggers and readers alike. The addition of a responsive design ensures accessibility for all users, regardless of their device. This project lays a strong foundation for further enhancements, such as real-time notifications, advanced analytics, and multimedia content support, to further enrich the blogging experience.

References:

[1] Repustate. (n.d.). *Real-time sentiment analysis*. Repustate Blog. Retrieved November 11, 2024, from https://www.repustate.com/blog/real-time-sentiment-analysis/

[2] TemplateMo. (n.d.). *Stand Blog - TemplateMo*. Retrieved December 17, 2024, from https://templatemo.com/tm-551-stand-blog

[3] Keerti, S. (n.d.). *Python-Django-Blog-Website*. GitHub. Retrieved November 11, 2024, from https://github.com/keerti1924/Python-Django-Blog-Website

[4] YouTube. (n.d.). *[Video title if available]*. Retrieved December 17, 2024, from https://youtu.be/GHt_AgcFt8Y?si=fd-S1JkPi4smyD6X