Development and Evaluation of an AI-Powered Content Creation SaaS Platform Using the PERN Stack

Prof.Nagraj Kamble[1] , Prof.S.M.Kale[2] , Manasi Mahaling Swami[3]
[1,2,3]Department of Information Technlogy M.S.Bidve Engineering College, Dr. Babasaheb Ambedkar Technological University, Latur, India
Email Id: [1]Nagraj.kamble@gmail.com ,[2]smkale14jan@gmail.com ,[3]Manasiswami27@gmail.com

***Abstract*—** The rapid growth of digital media has increased the demand for intelligent platforms that simplify content creation and design tasks. This research paper presents the development of an AI-powered content creation web application built using the PERN stack (PostgreSQL, Express.js, React.js, and Node.js). The proposed system provides multiple AI-driven features, including background removal from images, AI-based image generation, blog title generation, and automated article creation, using modern external APIs.

The backend of the application is implemented using Node.js and Express.js, while PostgreSQL is used for efficient data storage and management. The frontend is developed using React.js, enabling a responsive and interactive user interface. For AI functionalities, the system integrates the Gemini API for natural language processing tasks such as blog title and article generation, and the ClipDrop API for image processing operations including background removal and image generation.

The application allows users to input text prompts or upload images, which are then processed through the respective AI APIs to generate high-quality outputs in real time. This approach reduces manual effort and eliminates the need for advanced design or writing skills. Experimental results indicate that the platform delivers accurate image processing and contextually relevant textual content while maintaining performance and scalability.

The proposed system demonstrates how combining the PERN stack with powerful AI APIs can create a scalable, efficient, and user-friendly content creation platform. This research highlights the practical application of AI-driven automation in modern web development and digital content generation.

Keywords **-** Artificial Intelligence, PERN Stack, React.js, Node.js, PostgreSQL, Gemini API, ClipDrop API, Background Removal, Image Generation, Natural Language Processing, Content Generation, Web Application

I. INTRODUCTION

The increasing use of digital platforms for marketing, blogging, and online communication has created a strong demand for tools that can efficiently generate high-quality visual and textual content. Traditionally, tasks such as image editing, background removal, graphic design, and content writing required specialized software and professional skills. However, with recent advancements in Artificial Intelligence (AI), many of these tasks can now be automated and simplified through intelligent web applications [1], [2].

Artificial Intelligence has played a crucial role in transforming content creation by enabling machines to understand user inputs, process images, and generate human-like text using natural language processing and deep learning techniques [3], [4]. Modern AI-powered platforms allow users to create blog titles, articles, and visual content with minimal effort, thereby increasing productivity and reducing time consumption. This capability has become particularly beneficial for bloggers, digital marketers, students, and small businesses who require fast and cost-effective content solutions [5].

This research focuses on the development of an AI-based content creation platform implemented using the PERN stack, which includes PostgreSQL, Express.js, React.js, and Node.js. The PERN stack provides a scalable and efficient architecture for building full-stack web applications with a robust backend and a dynamic frontend. PostgreSQL ensures reliable and consistent data storage, while React.js enables the creation of responsive and user-friendly interfaces [6], [7].

To incorporate intelligent features, the system integrates the Gemini API for natural language processing tasks such as blog title generation and automated article creation. Additionally, the ClipDrop API is utilized for image processing operations, including background removal and AI-based image generation. The integration of these APIs with a PERN stack–based architecture enables real-time AI-driven responses while maintaining system performance and usability [8], [9].

The primary objective of this research is to develop a unified web application that simplifies digital content creation by combining multiple AI-driven functionalities into a single platform. This work demonstrates the practical application of AI in modern web development and highlights how API-based AI services can be effectively integrated into full-stack applications to enhance digital creativity and productivity [10].

II. RELATED WORK AND EXISTING SYSTEMS

With the rapid advancement of Artificial Intelligence, numerous platforms and tools have been developed to support automated content creation and image processing. Existing systems generally focus on either visual design automation or text-based content generation, with limited integration between the two domains. Popular design platforms such as Canva provide template-based graphic design tools along with basic AI-powered features like background removal. However, these platforms are largely template-driven and often require premium subscriptions to access advanced AI functionalities, limiting flexibility and customization for users [11].

AI-based writing tools such as Jasper, Copy.ai, and Writesonic have gained significant popularity for generating blog titles, articles, and marketing content. These platforms leverage natural language processing and deep learning techniques to generate human-like text based on user prompts. While they are effective in automating text generation, they do not support integrated image editing or visual content creation, thereby restricting their use in comprehensive digital content workflows [12], [13].

In the field of image processing, tools such as Remove.bg and ClipDrop provide efficient background removal and AI-driven image manipulation capabilities. These systems utilize deep learning–based image segmentation models to accurately separate foreground objects from complex backgrounds. Although they deliver high-quality visual results, these tools function as standalone services and lack integrated text generation, content management, or full-stack application support [14], [15].

Recent research studies have explored the application of generative AI models for automated content creation, highlighting improvements in contextual understanding, output quality, and response time. Several works emphasize the growing role of cloud-based AI services and API-driven architectures in enabling scalable and flexible AI applications. However, many existing solutions rely on monolithic system designs or closed third-party platforms, offering limited control over system architecture, data management, and customization for developers [16], [17].

The review of existing systems reveals a significant research gap in the development of a unified platform that combines both AI-driven text generation and image processing within a scalable and modular web application architecture. This research addresses the identified gap by integrating the Gemini API for natural language processing and the ClipDrop API for image processing within a PERN stack–based SaaS platform. The proposed system provides an efficient, extensible, and developer-friendly solution for modern digital content creation.

III. PROPOSED SYSTEM AND METHODOLOGY

1) A. Proposed System

The proposed system is an AI-powered content creation web application designed to provide multiple automated functionalities, including background removal, AI-based image generation, blog title generation, and automated article creation within a unified platform. The system is developed using the PERN stack, which ensures scalability, reliability, and efficient performance for modern web applications [6], [10].

The frontend of the application is implemented using React.js to provide a responsive and interactive user interface. Users can upload images or submit textual prompts through the web interface. The backend is developed using Node.js and Express.js, which are responsible for handling API requests, business logic, authentication, and communication with external AI services. PostgreSQL is utilized as the database management system to store user data, generated content, and application metadata in a structured and secure manner [7].

To enable intelligent automation, the system integrates external AI services through APIs. The Gemini API is employed for natural language processing tasks such as blog title generation and article creation, while the ClipDrop API is used for image processing tasks, including background removal and AI-based image generation. By leveraging API-driven AI services, the system delivers accurate and real-time results without requiring computationally expensive model training on the server side [8], [9].

Overall, the proposed system aims to reduce manual effort, enhance content creation efficiency, and provide an all-in-one AI-powered solution for users requiring both textual and visual digital content.

2) B. System Architecture

The proposed system follows a client–server architecture model. The client side, developed using React.js, interacts with the server through RESTful APIs. The server side, implemented using Node.js and Express.js, processes user requests, manages application logic, and communicates with external AI services. PostgreSQL acts as the persistent storage layer for maintaining application data and generated outputs [6].

The operational flow of the system is as follows. Initially, the user submits a text prompt or uploads an image through the frontend interface. The request is then forwarded to the Node.js–Express backend, which determines the type of operation requested. For text-based content generation, the backend communicates with the Gemini API, whereas for image processing tasks, requests are forwarded to the ClipDrop API. The AI-generated responses are received by the backend, optionally stored in the PostgreSQL database, and finally returned to the frontend for display to the user.

This modular and layered architecture ensures flexibility, scalability, and ease of maintenance, making the system suitable for future extensions and large-scale deployment [10].

3) C. Methodology

The methodology adopted for the development of the proposed system consists of several structured stages. Initially, a requirement analysis phase was conducted to identify the core features required for the platform, including text generation and image processing functionalities. Based on these requirements, the frontend was designed and developed using React.js to ensure intuitive user interaction and efficient input handling.

Subsequently, the backend was implemented using Node.js and Express.js to support RESTful communication, API integration, and secure data handling. External AI services were integrated using the Gemini API for natural language processing and the ClipDrop API for image-based operations. PostgreSQL was used to manage and store user-related data and generated content securely.

The system was then tested and validated under various input scenarios to evaluate accuracy, response time, and overall performance. Finally, the application was deployed on a suitable hosting platform to ensure accessibility, scalability, and real-world usability.

IV. EXPERIMENTAL SETUP AND DATA COLLECTION

The experimental setup for the proposed AI-powered content creation platform was designed to evaluate the performance and effectiveness of automated text and image generation functionalities. The primary objective of the experimentation was to analyze the system's ability to process user inputs accurately and generate high-quality textual and visual content in real time with minimal manual intervention.

The experiments were conducted on the developed web application implemented using the PERN stack. The frontend interface was accessed through a modern web browser, while the backend server handled request processing, API communication, and data management. The system interacted with external AI services through secure RESTful APIs to perform natural language processing and image processing tasks.

1) A. Dataset Collection

The dataset used for experimentation was dynamic and user-driven, as the proposed system does not rely on a pre-existing static dataset. Instead, data was generated during runtime based on user interactions with the application. User inputs consisted of text prompts for blog title and article generation, as well as image uploads for background removal and AI-based image generation.

The backend server managed communication with external AI services, including the Gemini API for text-based content generation and the ClipDrop API for image processing operations. Secure API keys and environment variables were used to ensure safe and reliable communication with these services. Generated outputs were displayed on the frontend interface and optionally stored in the PostgreSQL database for further analysis.

Multiple input scenarios were tested to evaluate system consistency, response time, and output accuracy across different types of content requests.

2) B. Development Environment

The experimental evaluation was performed in a standard development environment consisting of a personal computer, a modern web browser, and a backend server capable of handling REST API requests. React.js was used to build the frontend interface, while Node.js and Express.js were employed for backend processing and API integration. PostgreSQL served as the database management system for storing application data and generated content.

For text-based functionalities, various keywords and prompts related to blog titles and article generation were provided to the system. The Gemini API processed these inputs and generated corresponding textual outputs. The quality of the generated text was evaluated based on relevance, coherence, and grammatical correctness.

For image-based functionalities, images with different background complexities were uploaded to the system. The ClipDrop API was used to perform background removal and AI-based image generation. The generated outputs were analyzed for background accuracy, edge clarity, and overall visual quality.

The collected data included user input prompts, AI-generated text outputs, processed images, response time, and success rate of API calls. This data was used to assess the reliability and efficiency of the proposed system.

3) C. Evaluation Criteria

The proposed system was evaluated using multiple qualitative and performance-related criteria. These criteria included the relevance and coherence of generated textual content, accuracy of image background removal, visual quality of AI-generated images, system response time, and overall usability of the platform. The evaluation focused on determining whether the generated outputs met user expectations and application objectives.

4) D. Experimental Procedure

During experimentation, each input prompt or image upload was provided to the system individually. The generated textual and visual outputs were observed and analyzed based on the predefined evaluation criteria. Multiple test cases were executed to evaluate consistency and performance under different input conditions.

The system's performance was assessed by examining how effectively user intent was translated into meaningful content outputs. Observations from the experiments were recorded to identify strengths and limitations of the proposed approach. The experimental setup ensured consistent testing conditions and provided a practical evaluation of the system's capability to automate digital content creation using artificial intelligence techniques.

V. RESULTS AND DISCUSSION

The results obtained from the experimental evaluation demonstrate the effectiveness and reliability of the proposed AI-powered content creation platform. The system was tested using multiple user inputs, including text prompts for blog title and article generation, as well as image uploads for background removal and AI-based image generation. The evaluation was performed under consistent testing conditions to analyze system behavior and output quality.

For text-based functionalities, the Gemini API generated blog titles and articles that were contextually relevant to the user prompts. The generated content exhibited good coherence, grammatical correctness, and semantic relevance. In most test cases, the system successfully captured user intent and produced meaningful textual output with minimal latency. This indicates that the platform is suitable for automated content generation tasks such as blogging and digital marketing.

For image-based functionalities, the ClipDrop API demonstrated high accuracy in background removal and AI-based image generation. Uploaded images with varying background complexity were processed effectively, with clear separation between foreground and background objects. The generated images maintained good visual quality, sharp edges, and minimal distortion, making them suitable for practical design and content creation use cases.

The system performance was also evaluated based on response time and reliability. The average response time for both text and image generation operations remained within acceptable limits, ensuring a smooth user experience. The use of API-driven AI

services allowed the system to deliver real-time outputs without significant delays. Additionally, the success rate of API calls was high, indicating stable communication between the application backend and external AI services.

Overall, the results confirm that the proposed system efficiently automates digital content creation tasks while maintaining performance, scalability, and usability. The integration of text and image AI services within a unified PERN stack–based architecture provides a practical and effective solution for users seeking comprehensive content creation tools.

A. Generated Website Output

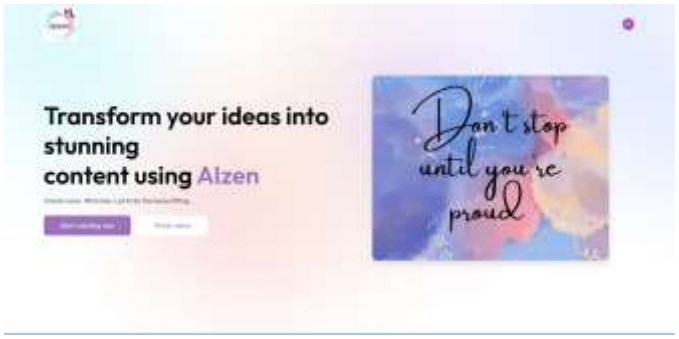

Fig. 5.1

Fig. 5.2

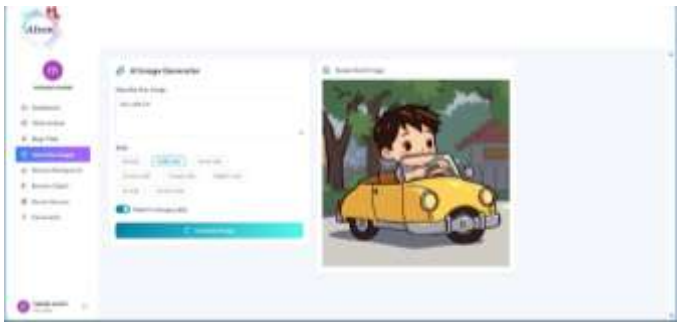

Fig. 5.3

Fig. 5.4

VI. CONCLUSION AND FUTURE SCOPE

1) *A. Conclusion*

This research paper presented the development and evaluation of an AI-powered content creation web application implemented using the PERN stack. The proposed system integrates the Gemini API for natural language processing and the ClipDrop API for image processing to provide automated features such as blog title generation, article creation, background removal, and AI-based image generation.

The experimental results demonstrate that the system is capable of generating high-quality textual and visual content with minimal user effort. The adoption of a modular and scalable architecture enables efficient interaction between the frontend, backend, database, and external AI services. By consolidating multiple AI-driven functionalities into a single platform, the proposed solution reduces dependency on multiple tools and enhances productivity for content creators, digital marketers, and users with limited technical expertise.

Overall, the study highlights the practical application of Artificial Intelligence in modern web development and confirms that API-driven AI integration can effectively support scalable and user-friendly digital content creation platforms.

2) *B. Future Scope*

Although the proposed system delivers effective performance, several enhancements can be incorporated in future work to improve functionality and scalability. Advanced user authentication mechanisms and role-based access control can be implemented to enable personalized content management. Additionally, the platform can be extended to support multi-language content generation using more advanced language models.

Future enhancements may also include domain-specific content optimization, real-time collaboration features, and the integration of additional AI services such as video generation and voice-based content creation. Performance optimization techniques, including caching, load balancing, and asynchronous processing, can further enhance system scalability. Furthermore, analytics and feedback modules can be introduced to evaluate content quality, system usage, and user engagement.

These future enhancements would make the system more robust, intelligent, and suitable for large-scale commercial and enterprise-level applications.

ACKNOWLEDGMENT

The authors would like to express their sincere gratitude to their project guide and faculty members for their valuable guidance, encouragement, and continuous support throughout the completion of this research work. Their insights and suggestions significantly contributed to improving the quality of this study.

The authors also acknowledge the institution for providing the necessary infrastructure and resources required for the successful implementation of the project. Special thanks are extended to friends and peers for their assistance, feedback, and motivation during the development and testing phases.

Finally, the authors would like to acknowledge the developers and communities of the Gemini API and ClipDrop API for providing powerful AI tools that enabled the implementation of intelligent content generation and image processing features in this research.

REFERENCES

[1] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, Pearson Education, 2021.

[2] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, MIT Press, 2016.

[3] T. Mitchell, *Machine Learning*, McGraw-Hill Education, 1997.

[4] M. Jurafsky and J. H. Martin, *Speech and Language Processing*, Pearson Education, 2019.

[5] Brown et al., "Language Models are Few-Shot Learners," *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[6] R. Fielding, "Architectural Styles and the Design of Network-based Software Architectures," Ph.D. Dissertation, University of California, Irvine, 2000.

[7] A. Banks and E. Porcello, *Learning React: Modern Patterns for Developing React Apps*, O'Reilly Media, 2020.

[8] Google, "Gemini API Documentation," Available: https://ai.google.dev

[9] ClipDrop, "ClipDrop API Documentation," Available: https://clipdrop.co/apis

[10] M. Armbrust et al., "A View of Cloud Computing," *Communications of the ACM*, vol. 53, no. 4, 2010.