IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 14, April 2026
Impact Factor: 8.2

Secure Generative AI-Based Intelligent Documentation System for Engineering Workflows

Pratham Chavhan[1], Pranav Sagne[2], Prajwal Mohitkar[3], Nayan Samruthwar[4], Ritesh Durge[5], Neha Nandanwar[6]
Student, Department of Computer Science[1–5]
Assistant Professor, Department of Computer Science[6]
G. H. Raisoni College of Engineering and Management, Nagpur (GHRCEMN), Maharashtra, India

Abstract: *Engineering workflows depend on technical documents such as manuals, standard operating procedures (SOPs), maintenance instructions, and safety guidelines for daily operations. However, conventional document management systems often struggle to understand user intent and provide relevant, context-based information, which can reduce efficiency and create risks in critical engineering environments. This paper presents a Secure Generative AI-Based Intelligent Documentation System for Engineering Workflows that combines Generative AI, semantic search, and Retrieval-Augmented Generation (RAG) to deliver accurate responses based on authorized documents. The proposed system allows users to interact through chat and voice, includes role-based access control (RBAC) to ensure secure access to documents, and uses AI-powered diagram generation to make complex technical information easier to understand. By combining intelligence, security, and user-friendly features, the system aims to improve workflow efficiency, support better decision-making, and enhance reliability and safety in engineering operations.*

Keywords: Generative AI, Engineering Workflows, Retrieval-Augmented Generation (RAG), Semantic Search, Role-Based Access Control (RBAC), Large Language Models (LLMs), Intelligent Documentation System, Technical Document Retrieval, AI-Powered Diagram Generation, Natural Language Processing (NLP).

I. INTRODUCTION

In today's engineering and industrial sectors, technical documents such as manuals, Standard Operating Procedures (SOPs), maintenance records, and safety guidelines play a critical role in supporting daily workflows. As organizations expand, managing and retrieving information from large volumes of complex documentation becomes increasingly difficult. Manual searching methods are often slow and prone to mistakes, while traditional keyword-based systems may fail to understand the actual meaning or intent behind user queries, often returning incomplete or irrelevant results. In safety-sensitive engineering environments, such limitations can lead to operational disruptions, safety concerns, and financial consequences. This creates a strong need for an intelligent solution capable of understanding natural language and delivering accurate, context-based information.

To overcome these challenges, the proposed Secure Generative AI-Based Intelligent Documentation System for Engineering Workflows combines Generative AI with advanced retrieval methods such as semantic search and Retrieval-Augmented Generation (RAG). The system is designed to generate responses only from verified and authorized documents, helping reduce errors and minimize hallucination issues. It also supports interactive communication through chat and voice interfaces, applies role-based access control to ensure secure and restricted access to documents, and includes AI-powered diagram generation to simplify the understanding of complex technical

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 14, April 2026
Impact Factor: 8.2

information. By bringing together intelligence, security, and usability, the proposed system aims to improve workflow efficiency, support informed decision-making, and strengthen reliability and safety in engineering operations.

II. LITERATURE REVIEW / RELATED WORK

Recent years have seen major progress in AI-driven approaches for document understanding and information retrieval. Patrick Lewis and colleagues introduced the concept of Retrieval-Augmented Generation in 2020, combining document retrieval with generative models to produce responses based on actual source data, improving accuracy and reducing misinformation. Building on this foundation, Jacob Devlin and later transformer-based research strengthened contextual language understanding, making it easier to process complex and technical documents more effectively.

Further advancements by Tom Brown and subsequent GPT-based models demonstrated the potential of Large Language Models to generate human-like, context-aware responses. However, these models also revealed challenges such as hallucination, which affects their dependability in safety-critical applications. To improve security in such systems, Ravi Sandhu and other researchers emphasized the role of Role-Based Access Control in protecting sensitive information through controlled and restricted access.

More recent studies from 2022 to 2025 have focused on combining semantic search, vector embeddings, and AI-powered retrieval systems to improve the relevance and precision of search outcomes. At the same time, developments in explainable AI and AI-driven visualization have enabled systems to produce diagrams and simplified representations that improve user understanding. Despite these improvements, many existing solutions still lack a unified framework that brings together secure access, reliable document-grounded responses, generative AI capabilities, and intuitive user interaction.

The proposed system addresses this gap by integrating LLMs, RAG, NLP, and RBAC into a secure and intelligent documentation platform specifically designed for engineering workflows.

Table 1. Comparative Analysis of Approaches in Smart Contract Research

Author / Year	Focus Area	Methodology	Limitation / Gap Identified
Patrick Lewis (2020)	Knowledge-intensive NLP tasks	Retrieval-Augmented Generation	Limited focus on security and enterprise-level access control
Vladimir Karpukhin (2020)	Dense document retrieval	Dense Passage Retrieval (DPR)	Focused only on retrieval; lacks response generation capability
Angela Fan (2024)	RAG + LLM integration	Retrieval-augmented LLM frameworks	High computational complexity and resource cost
Gupta (2024)	Advanced RAG systems	Hybrid retrieval and generative models	Challenges in scalability and real-time performance
Gao (2023)	RAG evolution and architecture	Modular and advanced RAG pipelines	Strong dependency on quality of retrieved data
Karakurt (2025)	Enterprise knowledge systems	RAG + LLM for document automation	Lack of unified deployment standards
Wang (2025)	Optimized RAG systems	Multi-retriever + prompt optimization	Complexity in tuning and system integration

III. METHODOLOGY / PROPOSED WORKFLOW

The proposed system follows a structured process to provide accurate, secure, and context-aware retrieval of information from engineering documents:

Document Collection and Preprocessing –

Technical documents such as manuals, SOPs, and guides are collected and converted into machine-readable text, including extracting content from PDFs. The data is then cleaned and divided into smaller sections for efficient processing.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 14, April 2026
Impact Factor: 8.2

Embedding and Storage –

Each document segment is transformed into vector embeddings using NLP models and stored in a vector database. This allows the system to perform semantic search rather than relying only on keyword matching.

User Query Processing –

Users interact with the system through chat or voice interfaces. The input query is analyzed using NLP techniques to understand user intent and contextual meaning.

Semantic Retrieval (RAG) –

Based on the query, relevant document sections are retrieved from the database using similarity matching. This ensures the system selects meaningful and context-relevant information.

Response Generation (LLM) –

The retrieved content is provided to a Large Language Model, which generates a clear and accurate response grounded only in the selected documents, helping reduce hallucinations.

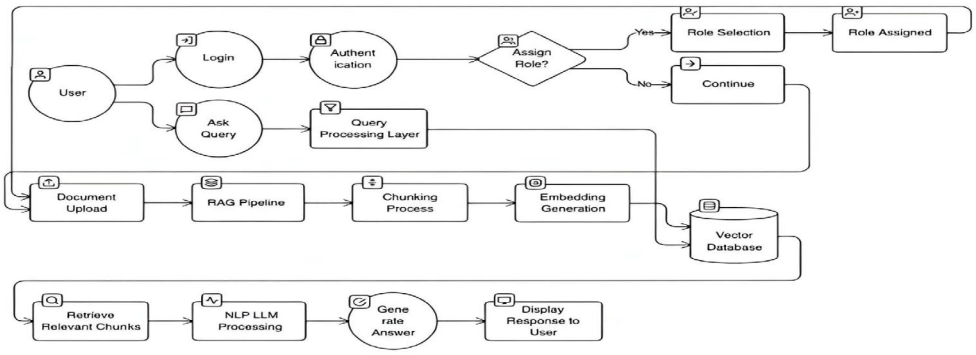
Security and Access Control (RBAC) –

Before retrieving documents, the system verifies user roles and permissions to ensure that only authorized information is accessed and displayed.

Visualization and Output –

The final response is presented in text form, and the system can also generate AI-based diagrams or simplified explanations to improve understanding of complex technical information.

Figure 1. Workflow for Smart Contract Generation



IV. IMPLEMENTATION

4.1 Dataset and Attribute Extraction

The implementation starts with collecting technical documents such as manuals, SOPs, maintenance guides, and safety instructions, primarily in PDF format. These documents are converted from unstructured content into machine-readable text using text extraction methods. After extraction, the data is preprocessed through cleaning, tokenization, and segmentation into smaller meaningful chunks to support efficient storage and retrieval.

Following this, attribute extraction is carried out using NLP techniques to identify important information such as equipment names, procedures, safety instructions, and technical parameters. Methods such as Named Entity Recognition (NER) help in extracting these key elements from the documents. The processed information is then transformed into vector embeddings and stored in a vector database to support semantic search. This structured dataset provides the basis for accurate and context-aware information retrieval within the proposed system.

4.2 Custom Parsing

The proposed system includes a document processing module designed to manage complex engineering documents such as manuals, SOPs, and technical guides. Unlike conventional approaches, the system preserves both the structure

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 14, April 2026 Impact Factor: 8.2

and contextual meaning of document content. It supports multiple formats, including PDF, Word files, and scanned documents through OCR, while extracting components such as headings, tables, and lists without losing document hierarchy. It also performs semantic chunking to divide content into meaningful sections for improved contextual understanding.

In addition, NLP techniques such as Named Entity Recognition (NER) are used to extract important information, including equipment details, procedures, and safety steps. Each document chunk is assigned metadata and protected through role-based access control (RBAC). The processed content is converted into vector embeddings and stored in a database to support efficient RAG-based retrieval, improving the accuracy, relevance, and reliability of the system.

Table 3. Parser Performance Comparison

Parser Type	Accuracy	Strengths	Weaknesses
Structural Parser	Medium	Preserves headings, tables, document format	Limited semantic understanding
Custom Parser (Proposed)	Very High	Combines structure + semantics + security (RBAC)	Higher development complexity

The results demonstrate that domain-specific customization is essential for extracting accurate data from documents. While the default parser is convenient, it fails in real-world scenarios, whereas the proposed custom parser provides robust and reliable performance.

4.3 Code Generation

The code generation module translates user queries into executable code snippets based on the requested task and available authorized documents. The frontend of the system is developed using Next.js and JavaScript, while the backend with Retrieval-Augmented Generation is implemented in Python to support efficient retrieval and response generation.

Groq-powered Large Language Models are used to generate accurate and context-aware code grounded in authorized documents. The system also supports Speech-to-Text (STT) and Text-to-Speech (TTS) features for voice-based interaction. For secure user management, Google Authentication and Firebase are integrated along with Role-Based Access Control to enforce controlled access permissions.

This approach improves efficiency, reduces manual effort, and ensures reliable and secure code generation within the proposed system.

4.4 User Interface

The system provides a user-friendly and interactive interface developed using the Next.js framework with JavaScript. It allows users to interact through both text and voice inputs, making it suitable for real-time engineering environments. The interface presents responses in a clear and structured format, including generated code, explanations, and document references.

The user interface is responsive and easy to navigate across different devices. Integration of Speech-to-Text (STT) and Text-to-Speech (TTS) improves accessibility, while Google Authentication and Firebase support secure login and session management. Overall, the interface improves user experience and enhances the usability of the proposed system.

V. COMPARATIVE FRAMEWORK

The proposed system is compared with traditional document management and existing AI-based systems based on key parameters such as accuracy, security, context understanding, and usability. Traditional systems rely on keyword-based search, which often provides irrelevant results and lacks contextual understanding. In contrast, AI-based approaches improve response generation but may suffer from hallucination and limited security controls.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 14, April 2026
Impact Factor: 8.2

Table 4. Comparison of Smart Contract Approaches

Feature / Step	Manual Coding Approach	Existing NLP Methods	Proposed Workflow (This Study)
Document Handling	Manual reading	Basic text extraction	Structured + semantic processing
Search Method	Keyword/manual search	NLP-based search	Semantic search (RAG)
Context Understanding	Low	Medium	High
Accuracy	Error-prone	Moderate	High
Response Generation	Manual coding	Auto-generated (may hallucinate)	Document-based (RAG grounded)

The proposed system combines RAG, LLMs, and RBAC, ensuring that responses are accurate, context-aware, and strictly based on authorized documents. It also supports voice interaction and code generation, improving usability and efficiency. Overall, the system provides a more reliable, secure, and intelligent solution for engineering workflows.

VI. FINDINGS AND DISCUSSION

The proposed system shows notable improvements over traditional and existing NLP-based methods for managing engineering documents. By integrating Retrieval-Augmented Generation, Large Language Models, and Role-Based Access Control, the system delivers accurate, context-aware, and secure responses based on authorized data sources. Semantic processing improves retrieval quality by reducing irrelevant results and minimizing hallucination.

The system also includes integration with Microsoft Teams to support collaboration, along with a separate AI assistant for intelligent query handling. Voice interaction through Speech-to-Text (STT) and Text-to-Speech (TTS) improves accessibility and supports real-time usability. In addition, features such as code generation help improve productivity and reduce manual effort. The implementation using Next.js for the frontend and Python for the RAG backend supports scalability and efficient performance.

However, the system may require significant computational resources and its performance depends on the quality of input documents and embeddings. Despite these challenges, the results indicate improvements in efficiency, reliability, collaboration, and decision-making, making the system suitable for safety-critical engineering environments.

Conclusion and Future Directions

The proposed system provides a secure and intelligent approach for managing engineering documents by integrating RAG, LLMs, and RBAC. It improves accuracy, contextual understanding, and security compared with traditional and existing NLP-based approaches. The system demonstrates strong performance through semantic retrieval, controlled access, and reliable response generation. Features such as an AI assistant, Microsoft Teams integration, voice interaction, and code generation further improve usability, collaboration, and operational efficiency. The use of Next.js for the frontend and Python for the RAG backend provides a scalable and responsive architecture suitable for practical engineering workflows.

Looking ahead, several directions can further enhance this work:

Improving real-time performance and system scalability

Enabling multimodal capabilities such as image and diagram understanding

Optimizing models to reduce computational cost

DOI: 10.48175/568

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 14, April 2026
Impact Factor: 8.2

Adding personalized responses based on user roles and preferences
Supporting multilingual interaction for wider accessibility
Developing advanced analytics dashboards for improved decision-making

Table 5. Future Directions for Smart Contract Automation

Area of Improvement	Current Status	Future Focus (2025–2030)
Accuracy & Retrieval	Good with RAG	Near real-time, highly precise retrieval
Scalability	Moderate system performance	Large-scale deployment with optimized models
Multimodal Support	Limited to text	Support for images, diagrams, and video inputs
Voice Interaction	Basic STT/TTS	Advanced natural voice conversations
Personalization	Limited user adaptation	AI-driven personalized responses

Traditional vs. Generative AI-Based Intelligent Documentation System for Engineering Workflows

Traditional Workflow →
Document Collection → Manual Reading → Keyword Search → Manual Interpretation → Result Output
Proposed Workflow →
Document Collection → Processing (OCR + Structure Extraction) → Semantic Chunking → Embedding & Storage → User Query → Semantic Retrieval (RAG) → LLM Response Generation → Secure Access (RBAC) → Output (Text/Voice/Code)

Advantages of the Proposed Workflow-
Provides high accuracy through semantic retrieval (RAG)
Ensures **c**ontext-aware responses using LLMs
Improves efficiency and reduces manual effort
Enhances security with RBAC and authentication
Supports code generation for faster development
Enables voice interaction (STT/TTS) for better usability
Allows collaboration through Teams integration
Reduces errors and hallucination by using document-grounded responsesReferences

REFERENCES

[1]. NAVIGATING RUPEE DEVALUATION: CAUSES, MACROECONOMIC IMPACTS, AND POLICY STRATEGIES FOR FINANCIAL STABILITY IN INDIA. *Lex Localis – Journal of Local Self-Government*, 23(10), 2025, ISSN: 1581-5374, E-ISSN: 1855-363X. https://doi.org/10.52152/801389. Available at: https://lex-localis.org/index.php/LexLocalis/article/view/801389 (Accessed: 15 August 2025)
[2]. Lewis, P. et al. (2020). Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. *Advances in Neural Information Processing Systems (NeurIPS)*.
[3]. Devlin, J. et al. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *NAACL-HLT*.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 14, April 2026
Impact Factor: 8.2

[4]. Brown, T. et al. (2020). Language Models are Few-Shot Learners. *Advances in Neural Information Processing Systems (NeurIPS)*.
[5]. Sandhu, R. et al. (1996). Role-Based Access Control Models. *IEEE Computer*, 29(2), 38–47.
[6]. Karpukhin, V. et al. (2020). Dense Passage Retrieval for Open-Domain Question Answering. *EMNLP*.
[7]. Fan, A. et al. (2024). Retrieval-Augmented Language Models. *arXiv preprint*.
[8]. Gupta, R. (2024). Hybrid Retrieval and Generative AI Systems. *International Journal of Intelligent Systems*.
[9]. Gao, Y. (2023). Evolution of Retrieval-Augmented Generation Architectures. *AI Review Journal*.
[10]. Karakurt, A. (2025). Enterprise Knowledge Systems Using RAG and LLMs. *Journal of Knowledge Management Systems*.
[11]. Wang, X. (2025). Optimized Multi-Retriever Architectures for RAG Systems. *Future Generation Computer Systems*