iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 **Issue:** VI **Month of publication:** June 2025

DOI: https://doi.org/10.22214/ijraset.2025.68968

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue VI June 2025- Available at www.ijraset.com

Automated Code Documentation Generator using AI with React and Django

Gaurav Thopate[1], Prof. P. R. Jadhao[2]
[1]*Master of Computer Application, Trinity Academy of Engineering, Pune, India*
[2]*Associate professor, Trinity Academy of Engineering, Pune, India*

Abstract: ***This project proposes an AI-powered platform for generating automated software documentation. Developers often overlook comprehensive documentation due to time constraints or complexity. To address this, we built a system that enables users to upload source code, which is then processed by Google's Gemini LLM to produce detailed documentation. The application uses a React-based frontend for user interaction, Django REST Framework for backend services, and supports PDF generation and secure user authentication via JWT. The system aims to save time and improve code maintainability.***
Keywords: AI, React, Django, Software Documentation, Gemini API

I. INTRODUCTION

Documentation plays a vital role in software development, facilitating code understanding, reuse, and debugging. However, writing documentation is often neglected due to its tedious nature. Leveraging modern AI advancements, this system automates the generation of detailed documentation for uploaded code files. By integrating React, Django, and Google's Gemini API, this tool provides a seamless experience from code upload to documentation download.

II. LITERATURE SURVEY/BACKGROUND

Several open-source and proprietary tools exist for documentation generation, such as Doxygen and Sphinx. However, these tools are limited to specific languages and lack natural language explanations. With the advent of LLMs like OpenAI GPT and Google Gemini, AI can now understand and explain code in human-like language. Existing platforms have not yet combined file upload, AI-generated docs, and PDF export in a single, user-friendly interface.

III. PROPOSED WORK/SYSTEM

The system is divided into three main modules:

1) Frontend (React)

- User authentication via JWT
- File upload interface
- Documentation preview and download

2) Backend (Django REST)

- Secure user management
- API endpoints for file processing, PDF generation
- Integration with Gemini API to generate natural language documentation

3) AI Layer (Gemini API)

- Uses gemini-pro model to generate explanations
- Detects programming language and summarizes code functions and structure
- Returns markdown-formatted documentation for rendering

.

IV. RESULT AND DISCUSSION

Users can register, log in, and securely upload files. On processing, the documentation is presented with code summaries, method tables, and explanations. Markdown rendering is used for better formatting. Users can also download the output in PDF format. The system performs efficiently with Python, Java, and JavaScript files. Integration with Gemini provides better documentation quality than rule-based systems.

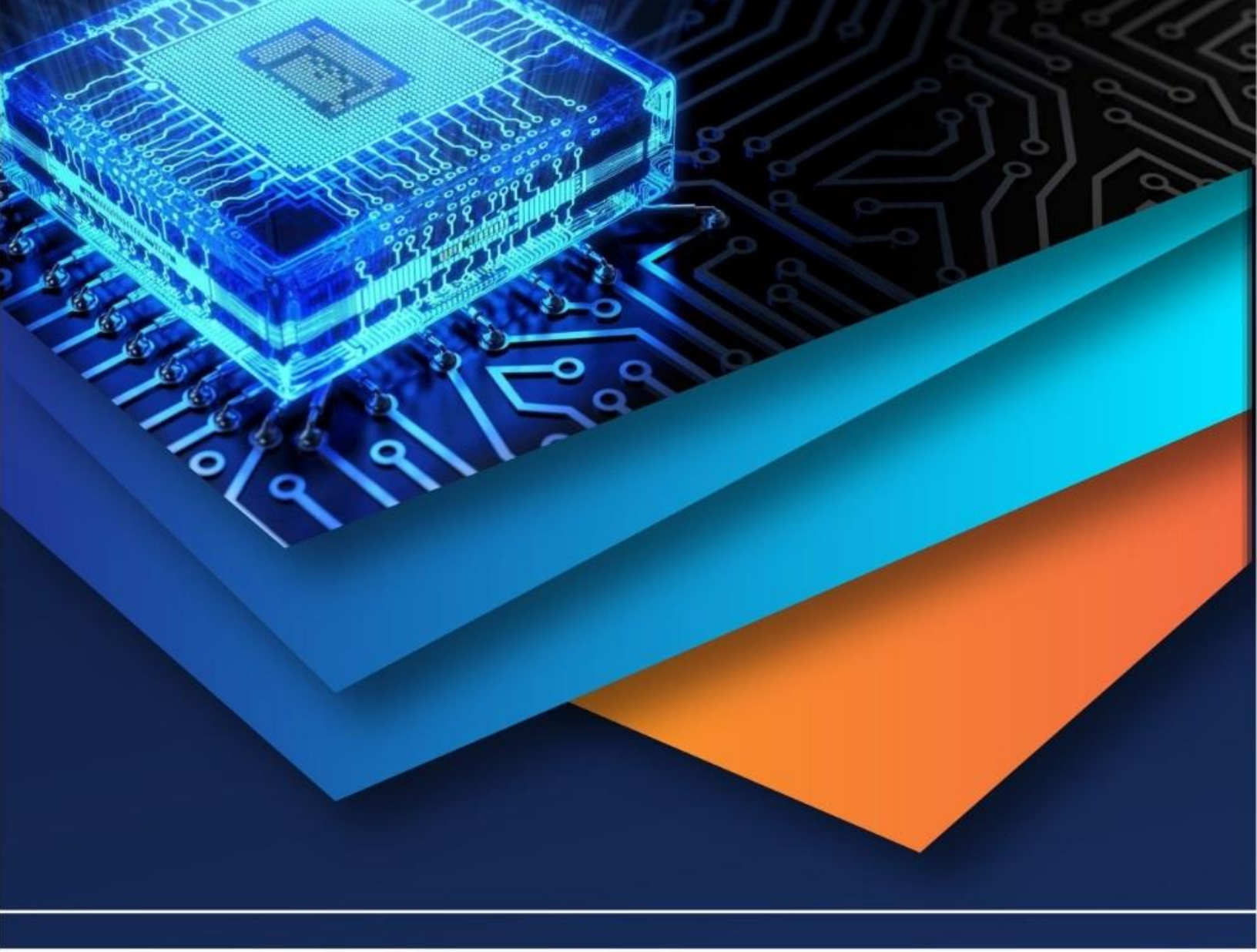
V. CONCLUSION

The Automated Code Documentation Generator successfully demonstrates how artificial intelligence can significantly simplify the often-overlooked process of software documentation. By integrating Google's Gemini API with a user-friendly React frontend and a Django REST backend, the system enables users to upload source code and receive structured, detailed documentation in real-time. This not only reduces the manual effort required but also improves code readability and maintainability.

The project supports multiple programming languages, handles file uploads, dynamically detects code structure, and outputs the documentation in both markdown and downloadable PDF formats. The user authentication system ensures secure access, while the intuitive interface improves accessibility for developers of all levels.

REFERENCES

[1] Google Cloud. (2024). Gemini API Documentation. Retrieved from https://ai.google.dev
[2] Django Software Foundation. (2024). Django Documentation (v4.2). Retrieved from https://docs.djangoproject.com
[3] Meta Platforms Inc. (2024). React – A JavaScript Library for Building User Interfaces. Retrieved from https://reactjs.org



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY