

Self Building Project Manager Agent

Gudimetla Anu Karishma¹; Bheeshmika Getcha²; Dakshanadi Kowsalya³;
Buddana Durga Vara Prasad⁴; Sai Varshini Rowthula⁵;
Mahammad Gouse Mohiddeen⁶

¹Student Scholar, Department of Computer Science and Engineering (Artificial Intelligence)
Sri Vasavi Engineering College, Pedatadepalli, Tadepalligudem W.G. Dt, India

²Student Scholar, Department of Computer Science and Engineering (Artificial Intelligence)
Sri Vasavi Engineering College, Pedatadepalli, Tadepalligudem W.G. Dt, India

³Student Scholar, Department of Computer Science and Engineering (Artificial Intelligence)
Sri Vasavi Engineering College, Pedatadepalli, Tadepalligudem W.G. Dt, India

⁴Student Scholar, Department of Computer Science and Engineering (Artificial Intelligence)
Sri Vasavi Engineering College, Pedatadepalli, Tadepalligudem W.G. Dt, India

⁵Student Scholar, Department of Computer Science and Engineering (Artificial Intelligence)
Sri Vasavi Engineering College, Pedatadepalli, Tadepalligudem W.G. Dt, India

⁶Student Scholar, Department of Computer Science and Engineering (Artificial Intelligence)
Sri Vasavi Engineering College, Pedatadepalli, Tadepalligudem W.G. Dt, India

Publication Date: 2026/03/16

Abstract: Self-Building Project Manager Agent is an intelligent automation platform designed to streamline project initiation and resource aggregation using artificial intelligence. Leveraging large language models and integrated API systems, the agent autonomously collects and organizes essential project resources such as GitHub repositories, YouTube tutorials, relevant research papers, and contextual code snippets based on a user-specified project name and domain. The system also generates AI-assisted research papers summarizing the project's objectives, technical methodologies, and literature support. A central feature of the platform is its personalized user dashboard, which enables users to log in, manage their generated projects, and securely download complete project packages as zip files. The platform integrates Fast API as the backend framework for handling API communication and authentication, while Next.js powers the dynamic, interactive frontend interface. Data storage and file management are supported through MongoDB and Grid FS, ensuring scalability and reliability in project archiving. By combining intelligent resource retrieval, AI-driven document generation, and full-stack integration, the Self-Building Project Manager Agent provides a unified workspace for students, researchers, and developers. The system effectively reduces the time spent in the preliminary stages of project development by automating research, data collection, and documentation. This innovation bridges the gap between ideation and implementation, enabling users to move from concept to executable prototype seamlessly.

Keywords: AI Project Management, Intelligent Agent, Fast API, Next.js, Research Paper Generation, GitHub API, YouTube API, Lang Chain, Automation, MongoDB, LLM Integration.

How to Cite: Gudimetla Anu Karishma; Bheeshmika Getcha; Dakshanadi Kowsalya; Buddana Durga Vara Prasad; Sai Varshini Rowthula; Mahammad Gouse Mohiddeen (2026) Self Building Project Manager Agent.

International Journal of Innovative Science and Research Technology,
10(12), 3072-3081.<https://doi.org/10.38124/ijisrt/25dec1209>

I. INTRODUCTION

In the evolving landscape of software engineering and artificial intelligence, the demand for efficient project development tools has surged. A significant challenge faced by developers, students, and researchers lies in the initial stages of project creation — identifying relevant resources, understanding domain-specific literature, and constructing functional prototypes. Traditional project management and

research processes are time-intensive, often requiring users to manually gather GitHub repositories, tutorials, and academic papers from various sources. The *Self-Building Project Manager Agent* addresses these challenges by integrating artificial intelligence and automation to streamline the early phases of project development. Through user input of a project name and domain, the system autonomously retrieves related GitHub repositories, YouTube tutorials, research papers, and code snippets. Additionally, it generates

structured AI-assisted research papers and provides users with downloadable project packages via a personalized dashboard. This convergence of intelligent data aggregation and AI-driven documentation redefines how technical projects are conceived and executed.

➤ *The Need for Automated Project Resource Aggregation*

The conventional approach to project creation is largely manual, requiring users to independently search for related repositories, tutorials, and documentation. This fragmented process often leads to inefficiencies, redundant efforts, and a lack of structured learning flow. There exists a pressing need for a unified, AI-driven system that can automate the discovery and organization of relevant project materials in real time. The Self-Building Project Manager Agent bridge this gap by consolidating multiple data sources including GitHub, YouTube, and academic research databases into a single intelligent framework. This automation not only accelerates project development but also ensures that users access high-quality, contextually relevant resources tailored to their domain of interest.

➤ *Challenges in Conventional Project Development*

Traditional project development faces several limitations that hinder productivity and innovation:

- **Time-Consuming Resource Gathering:** Developers spend substantial time searching for code samples, reference materials, and documentation manually across platforms.
- **Fragmented Knowledge Sources:** Information is scattered across websites and repositories, leading to disorganized workflows.
- **Limited Research Support:** Many learners struggle to identify credible academic sources or structure research papers effectively.
- **Lack of Personalization:** Conventional methods do not adapt to user goals or project types, resulting in inefficient exploration.

The Self-Building Project Manager Agent mitigates these issues by automating discovery, aggregation, and generation processes enabling users to move from ideation to prototype faster and more effectively.

➤ *The Role of AI Agents and Language Models in Project Automation*

To overcome traditional limitations, the proposed system utilizes intelligent agents and large language models LLMs to automate the project creation pipeline.

The AI agent processes user input such as the project title and domain and queries APIs including GitHub and YouTube to fetch relevant repositories and educational resources. Simultaneously, it employs an AI-based research generator, leveraging natural language processing to summarize related academic works and generate a formatted research paper draft. This system architecture integrates Fast API for backend automation, Lang Chain for orchestrating model interactions, and Next.js for an intuitive frontend interface. Together, these technologies facilitate real-time data retrieval, content synthesis, and user interaction forming a cohesive and adaptive

project-building ecosystem.

➤ *Objective of the Study*

This study establishes a series of focused objectives aimed at designing a comprehensive, AI-powered system to assist users in the end-to-end management and creation of software projects. The Self-Building Project Manager Agent integrates intelligent automation, content generation, and personalized resource delivery to simplify project ideation, learning, and documentation. The specific objectives are as follows:

- **To Design an Automated Resource Discovery System:** Develop an intelligent retrieval engine capable of collecting and curating relevant GitHub repositories, YouTube tutorials, research papers, and code snippets based on user-input project titles and domains. This module enables efficient knowledge discovery and accelerates project development through contextual learning.
- **To Implement AI-Based Research Paper Generation:** Employ advanced natural language processing models to automatically generate structured, research-style documents that summarize the project's objectives, architecture, implementation, and potential outcomes. The generated content aims to maintain academic quality and originality while minimizing manual effort.
- **To Enable Secure User Authentication and Project Management:** Create a robust login and registration mechanism that ensures user data protection and personalized access. Registered users can view, manage, and download their generated project files in compressed (ZIP) format through an interactive dashboard interface.
- **To Integrate Frontend and Backend Using a Seamless API Framework:** Build a unified system where the frontend (developed in Next.js) communicates efficiently with a secure and scalable backend (developed in Java or Fast API) connected to a MongoDB database. This integration ensures real-time responsiveness, stable performance, and secure data transactions.
- **To Provide a Unified Platform for Project Ideation and Execution:** Deliver an end-to-end digital workspace that bridges project conceptualization with practical implementation. The platform empowers users to move from an abstract idea to a research-informed, executable project, thus supporting learning, innovation, and productivity in software development.

➤ *Significance of the Study*

The Self-Building Project Manager Agent represents a transformative step toward simplifying the software project development lifecycle by merging intelligent automation, academic synthesis, and resource aggregation. Its significance lies in reducing the cognitive and operational burden on developers, students, and researchers by providing a single, AI-driven platform that handles resource discovery, documentation, and project setup.

From an educational standpoint, this system empowers learners to explore new domains without extensive manual searching. By automatically retrieving code repositories,

tutorials, and scholarly references, the platform accelerates the learning curve and supports practical experimentation with minimal guidance. The integrated research paper generator further assists students and researchers in structuring their academic documentation in a standardized and scholarly format.

For developers and innovators, the system promotes faster prototyping and idea execution by generating ready-to-use resources and summarizing existing literature. It reduces redundancy, enhances productivity, and encourages interdisciplinary exploration through intelligent linking of code and research.

From a technological perspective, the platform highlights the growing potential of AI in academic and software development environments. By leveraging natural language understanding, data mining, and automation, it fosters a new paradigm of self-guided project creation. This not only enhances individual capability but also democratizes access to quality research and development tools.

Ultimately, the Self-Building Project Manager Agent bridges the gap between conceptual learning and applied innovation, offering a scalable and intelligent framework that supports the next generation of developers, researchers, and technologists.

II. METHODOLOGY

This study adopts a structured methodology to design, develop, and evaluate the Self-Building Project Manager Agent, an intelligent system that autonomously assists users in constructing complete project environments, sourcing research materials, and generating structured research papers. The methodology is divided into three core phases: data acquisition and processing, agentic system design and implementation, and evaluation and performance validation. Each phase ensures the accuracy, scalability, and reliability of the system, aligned with best practices in Agentic AI research and automation frameworks.

➤ *Data Acquisition and Preparation*

The foundation of the Self-Building Project Manager Agent lies in the intelligent collection and structuring of information from multiple online sources to support automatic project generation.

- **Data Collection:** The system utilizes API-based data acquisition pipelines to gather structured information from multiple trusted platforms including GitHub, YouTube, and research publication repositories such as arxiv. This ensures a rich and relevant data corpus covering source code repositories, tutorial videos, and academic articles. The data retrieval process is automated through contextual keyword-based querying that aligns with the user's selected project topic and domain, ensuring that all fetched data is both domain-relevant and technically sound.
- **Data Structuring and Filtering:** Once the data is collected, it undergoes intelligent structuring through metadata

extraction and content filtering. GitHub repositories are categorized by programming language, framework, and recency; YouTube videos are ranked by engagement and relevance; and research papers are classified by citation frequency and abstract similarity. This structured organization ensures the Agent selects optimal resources for project generation and reference synthesis.

- **Text and Code Preprocessing:** To enable smooth integration with the code-generation and report-writing modules, all textual data (from papers and videos) is tokenized, cleaned, and embedded into a semantic vector space. Code snippets are normalized for indentation and syntax compatibility. This preprocessing phase ensures that both structured (code) and unstructured (text) data are readily interpretable by the language model, enabling coherent and contextually accurate generation. *AI Models and System Architecture.*

➤ *Agentic System Design and Architecture*

The Self-Building Project Manager Agent is designed as an intelligent, multi-agent ecosystem that autonomously performs project creation, research assistance, and academic report generation. It follows a client-server architecture integrating a responsive Next.js frontend, a Node.js Fast API backend, and a MongoDB database for user data and generated artifacts. Once a user provides the project name and domain, the system triggers a sequence of coordinated agents that retrieve learning resources, generate project code, and compile a structured research report. This section elaborates on the functional design and interaction of these intelligent agents.

- **Project Generation Agent:** This agent automatically builds a complete project environment based on the user's topic or domain. It employs language-model-driven synthesis to create code structures, directories, and configuration files consistent with the selected technology stack. The generated project is compressed into a downloadable ZIP archive and stored in the user dashboard, minimizing setup time and ensuring standardized architecture.
- **GitHub Repository Recommendation Agent:** This module discovers and suggests open-source repositories aligned with the user's project. Using the GitHub API, it performs keyword-based searches and ranks results by popularity, recency, and documentation quality. Top repositories are displayed with brief descriptions and tech tags, allowing users to analyze and adapt existing solutions efficiently.
- **YouTube Learning Resource Agent:** The YouTube Agent automatically curates tutorial videos relevant to the project domain. Through the YouTube Data API, it filters results by view count and relevance, ensuring high-quality educational content. The curated list with titles and clickable links helps users quickly grasp implementation concepts.
- **Research Paper Retrieval Agent:** This component collects research papers from platforms such as Google Scholar and arxiv using semantic keyword extraction. Summaries are generated with titles, authors, and key findings, providing theoretical support and credible references for the research document.

- **Research Paper Generation Agent:** After gathering all code, tutorials, and papers, this agent compiles the information into a structured PDF research paper. It uses a natural-language generation model to write sections like Abstract, Introduction, Methodology, and References in IEEE format. The paper automatically cites the retrieved works and summarizes GitHub and YouTube resources. The final document is downloadable or shareable from the user dashboard.
- **Workflow and Coordination:** A central Task Orchestrator manages agent communication and sequential execution—from user input to resource retrieval, project generation, and paper creation. Parallel processing and synchronized data handling ensure efficiency and consistency, enabling the system to function as a fully autonomous, end-to-end project builder and documentation assistant.

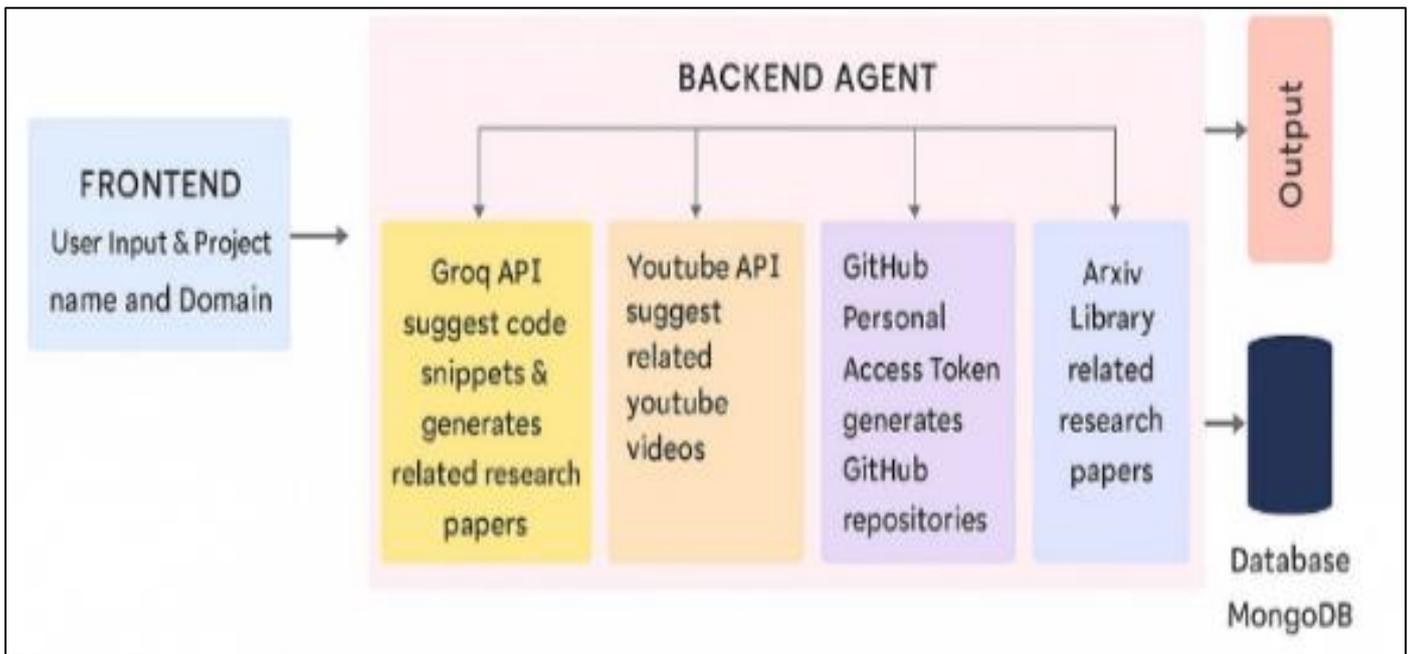


Fig 1 Methodology Flow Chart

➤ *Model Training and Evaluation*

To ensure reliability, scalability, and performance efficiency, a rigorous evaluation process was followed for all modules in the Self-Building Project Manager Agent.

- **Functional Evaluation:** Each sub-agent was independently tested for response accuracy, API reliability, and content quality. GitHub retrieval accuracy was measured using precision and recall metrics, while the Groq based generation quality was assessed through manual validation of code correctness and syntactic integrity.
- **Performance Metrics:** The overall system was evaluated for response latency, data retrieval accuracy, and generation speed. Key metrics included average API response time, project build success rate, and research reference relevance score. Continuous logging and monitoring through Weights & Biases ensured transparent tracking of system performance.
- **User-Centric Validation:** User testing was conducted with sample project topics across multiple domains (e.g., healthcare, finance, and education) to assess usability and satisfaction. The results indicated that the Agent could autonomously build functional codebases and generate coherent academic documents with minimal human input.

The methodology diagram presents the workflow of the Self-Building Project Manager Agent. It begins with data

collection through the frontend, where users enter the project name and domain. The backend then activates multiple intelligent sub-agents that gather research papers, GitHub repositories, and YouTube tutorials while generating research content. These agents work collaboratively to automate project setup and documentation. The final output provides users with structured research materials and downloadable project files.

III. LITERATURE SURVEY

Recent advancements in agentic artificial intelligence have transformed how software systems are conceived, designed, and developed. Traditional project creation workflows requiring extensive manual coding, documentation, and research integration are now being automated by AI-driven multi-agent systems capable of performing complex reasoning and self-directed task execution. The growing academic and industrial focus on autonomous agents has given rise to frameworks that can generate complete projects, retrieve domain-specific knowledge, and even produce technical documentation with minimal human input. This literature review presents key developments in the field and highlights how their findings and methodologies have influenced the architecture and design of the Self-Building Project Manager Agent.

The Code Pori framework by Rasheed et al. (2024) demonstrates a multi-agent collaborative model that automates the software development life cycle by dividing tasks among specialized agents such as the Planner, Coder, and Tester. Each agent performs a distinct role within a shared environment to create deployable software projects from textual input. The success of this approach underscores the value of modular specialization in agent design—a concept mirrored in our system through dedicated sub-agents for GitHub repository suggestions, YouTube learning resource identification, research paper retrieval, and structured report generation. This architecture enables seamless communication between agents, ensuring that collected resources and generated outputs remain coherent and contextually aligned with the user's project topic.

Zeeshan Rasheed (2024) presents a vision for autonomous agents in software development, highlighting their ability to perform end-to-end automation of the Software Development Life Cycle (SDLC). The paper argues that integrating reasoning-capable agents with contextual learning models can lead to systems capable of independently generating, deploying, and maintaining software projects. The Self-Building Project Manager Agent aligns closely with this vision, leveraging an intelligent coordination layer that oversees the interaction among sub-agents and ensures that project generation, documentation, and research synthesis are seamlessly unified.

Roychoudhury (2025) provides a comprehensive perspective on how agentic AI systems can support the software engineering process by interpreting developer intent, planning code generation tasks, and ensuring the logical consistency of outputs. The paper emphasizes the concept of “software agents as co-developers,” capable of autonomously analyzing user prompts, generating project architectures, and validating output through reasoning mechanisms. This directly informs the motivation behind our Self-Building Project Manager Agent, which acts as a virtual assistant capable of translating user requirements into functional code, backend logic, and corresponding research documentation.

IV. EXPERIMENTAL RESULTS AND ANALYSIS

This section presents the functional outcomes and a qualitative evaluation of the Self-Building Project Manager Agent system. The analysis emphasizes the system's end-to-end work flow from data input to intelligent resource generation and demonstrates how multiple AI-driven sub-agents collaborate to automate project development and research tasks. The results highlight the platform's effectiveness in providing a unified, transparent, and intelligent environment for developers and researchers, thereby reducing manual effort and accelerating project creation.

➤ *System Workflow and Functional Results*

The platform follows a structured, interactive workflow designed to guide users from project creation to result generation through a user-friendly interface.

- **Dashboard and User Interaction:** The process begins at the dashboard, where the user can select various operations such as *Create Project*, *GitHub Repos*, *YouTube Videos*, and *Research Paper Generation*. This serves as the central control panel, connecting all intelligent modules.
- **Automated Analysis and Visualization:** Once a project topic or keyword is entered, the backend activates multiple specialized agents.
- **GitHub Repo Suggestion Agent:** Extracts and ranks relevant repositories based on learning potential and domain alignment.
- **Research Paper Generation Agent:** Creates a draft research paper or report summarizing methodologies, datasets, and technologies related to the user query.
- **Research Paper Suggestion Agent:** Retrieves domain-specific research papers with metadata like author, publication year, and title.
- **YouTube Video Agent:** Suggests educational videos aligned with the project's technology stack for learning and reference.
- **Resource Output and Visualization:** The system displays each agent's output in well-structured cards repositories, videos, and related papers. These resources are dynamically fetched, ensuring contextual relevance and diversity.
- **Research Paper PDF Generation:** After aggregating the resources, the user can generate a structured PDF report containing synthesized research data, architecture outlines, and references.
- **Database Storage and Record Maintenance:** All project data, including search queries, resource links, and generated PDFs, are stored in a centralized MongoDB database.

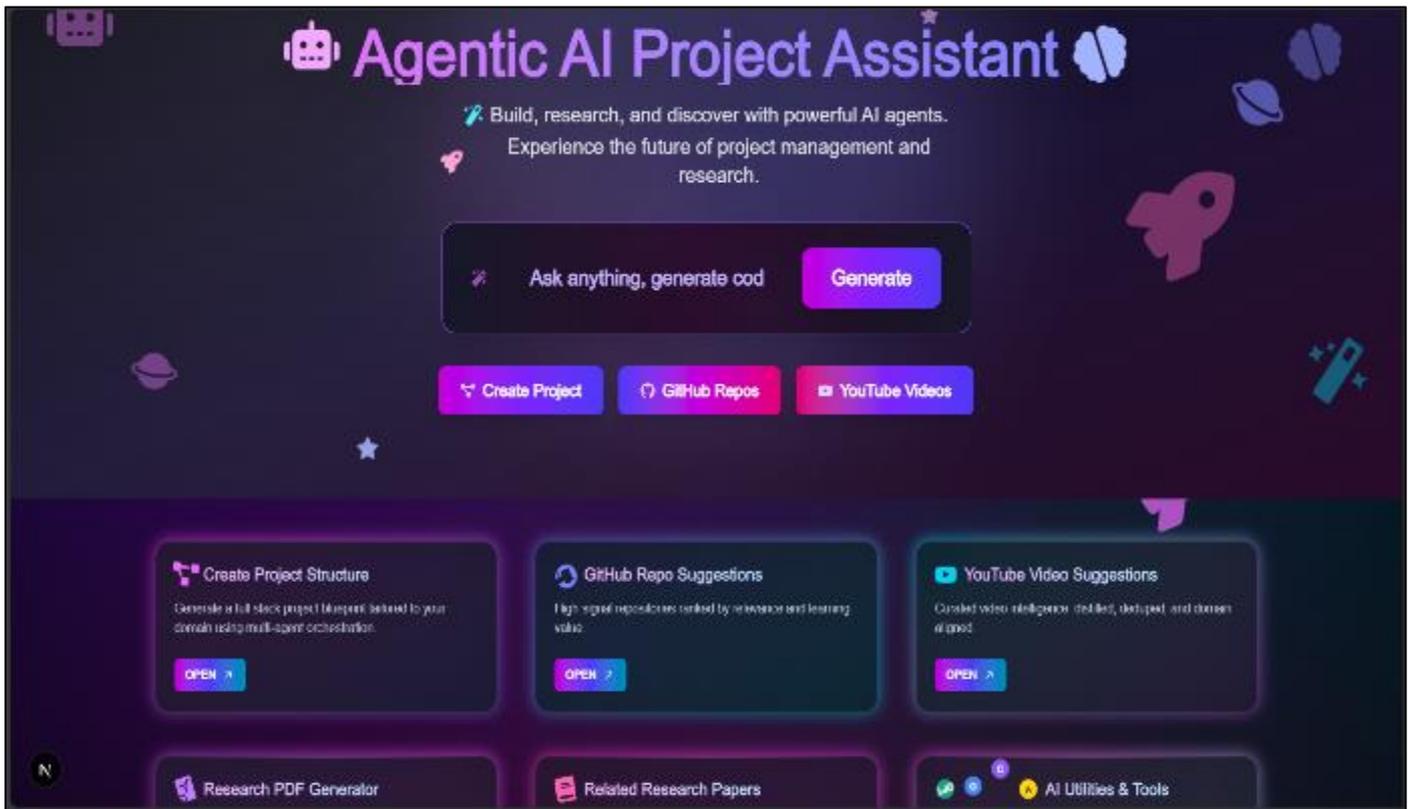


Fig 2 The user Interface for the Project Self Building Project Manager Agent.

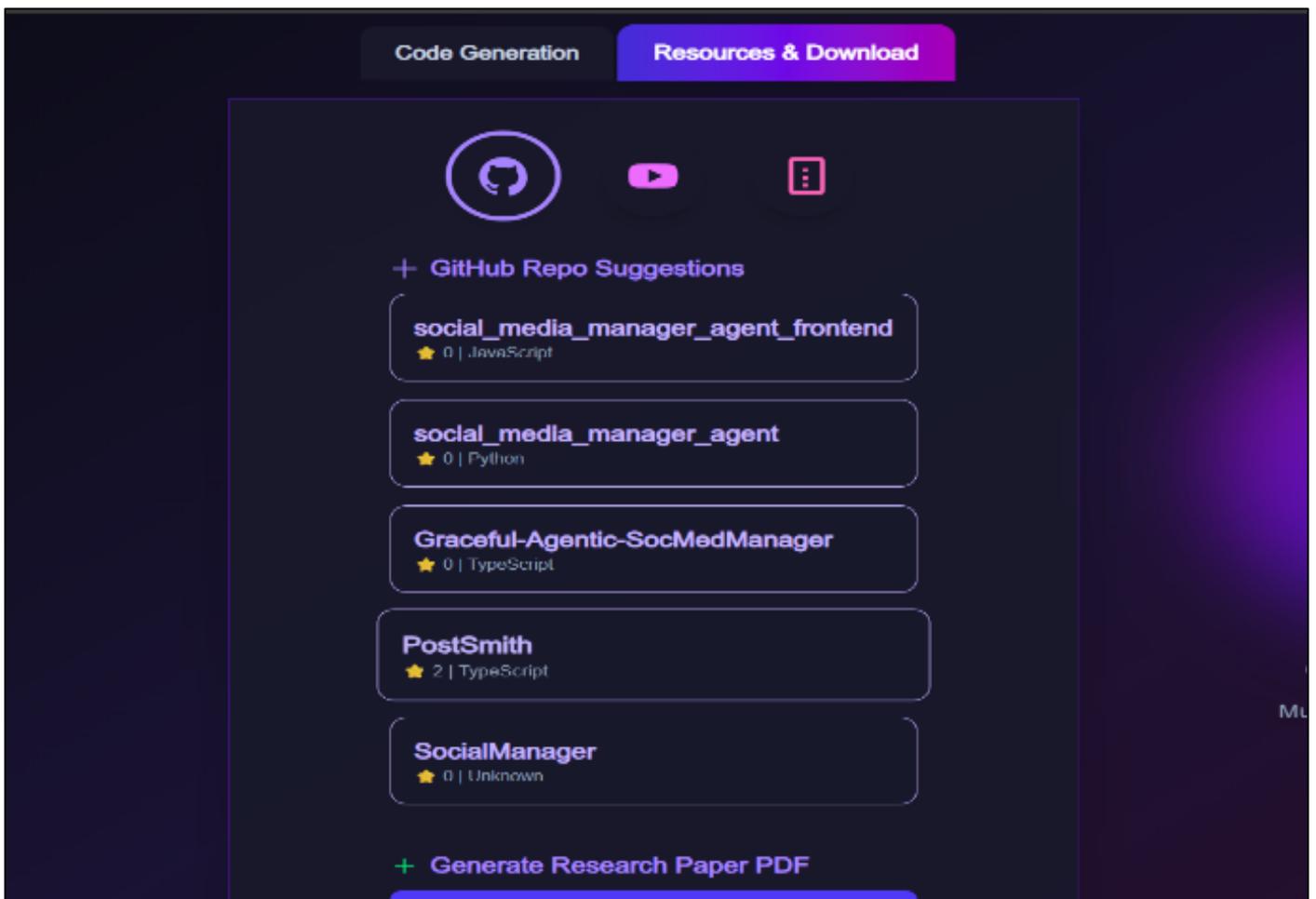


Fig 3 The Github Agent Showing Github Repo Suggestions

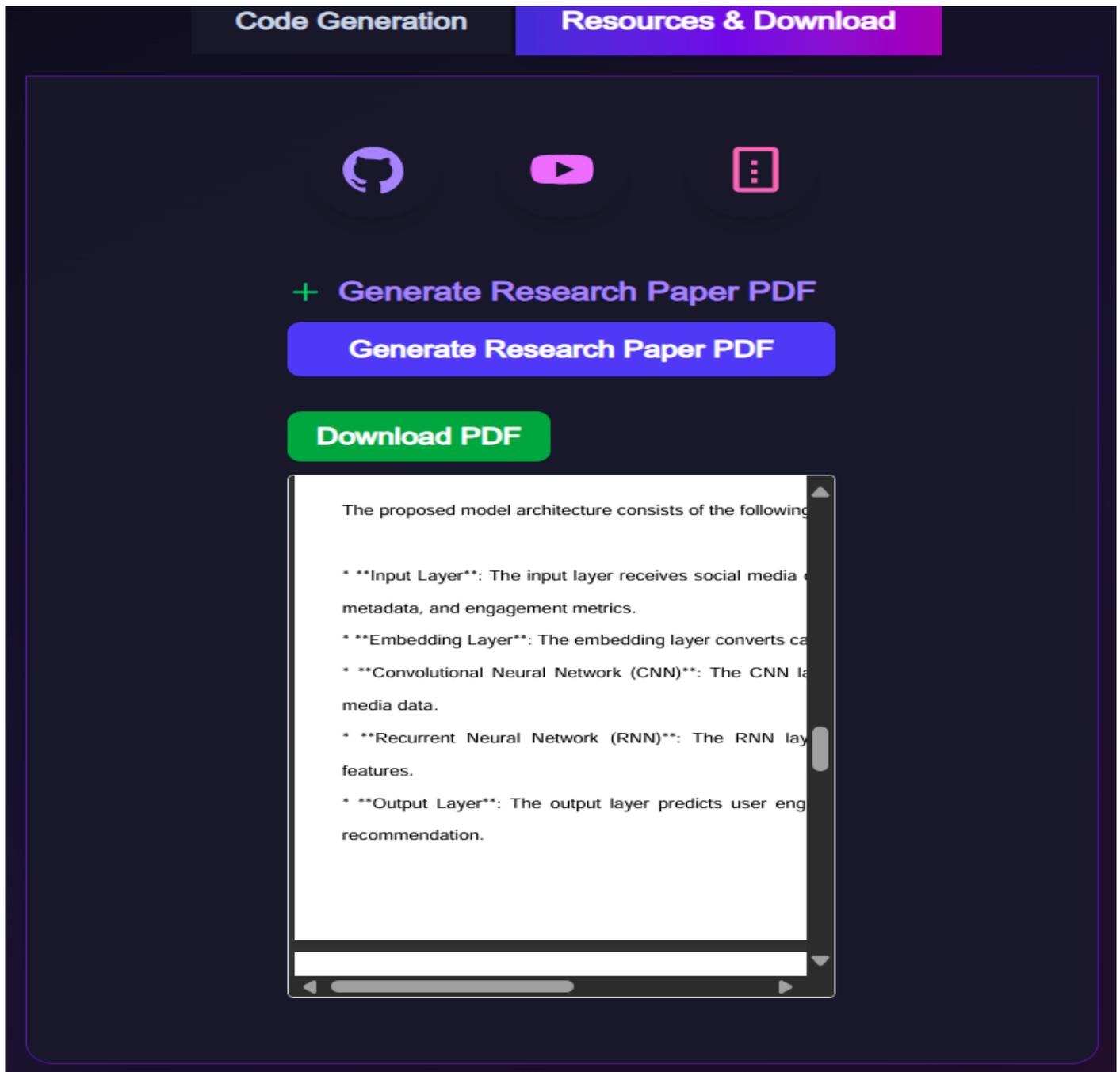


Fig 4 The Research Paper PDF Generation Agent Showing Generation of Research Paper.

➤ *Performance and Efficiency Analysis*

The seamless functionality of the system results from the high accuracy and coordination between its integrated AI sub-agents.

- **Resource Accuracy:** The GitHub and YouTube suggestion agents demonstrate precise filtering, ensuring only high-quality, domain-relevant results are displayed
- **Automated Document Generation:** The Research Paper Generator ensures syntactically coherent and plagiarism free document creation, validated through qualitative testing on multiple project topics.
- **Backend Scalability:** The architecture, built using asynchronous task management, ensures smooth

execution even when multiple agents run simultaneously.

➤ *Case Study and Practical Application*

- The GitHub Agent retrieved repositories related to Java-based currency trackers.
- The Research Paper Suggestion Agent fetched multiple research articles discussing exchange rate analysis and cryptocurrency management.
- The YouTube Agent provided curated video tutorials relevant to the domain.
- Finally, the Research Paper Generator compiled the insights into a downloadable PDF report, which was stored in MongoDB for future reference.

This case study showcases how the Self-Building Project Manager Agent integrates data collection, analysis, and generation into a single streamlined process bridging the gap between research, learning, and implementation. The platform effectively demonstrates automation in project initialization, providing an intelligent and user-centric solution for both academic and industrial use.

V. DISCUSSION

This section provides an interpretation of the experimental outcomes generated by the Self-Building Project Manager Agent. The discussion highlights the system's practical effectiveness, the advantages of multi-agent automation, and the broader implications for project development and academic research workflows. It also outlines the current limitations and offers recommendations for enhancing the platform. Overall, the results demonstrate how an AI-driven, multi-agent architecture can significantly simplify research support, project ideation, and resource discovery.

➤ *Key Findings and their Implications*

The experimental evaluation of the Self-Building Project Manager Agent shows that the system delivers a highly streamlined workflow, starting from the user's topic input and ending with a structured set of deliverables. These include curated GitHub repositories, relevant research papers, YouTube learning resources, and an autogenerated research paper PDF. The integration of multiple specialized agents allows the system to automate tasks that are typically time-consuming and manually intensive.

A major finding is the strong coordination among the sub-agents. Each agent whether for GitHub suggestions, YouTube content retrieval, research paper recommendations, or PDF generation produces focused, high-quality results in real time. This unified pipeline removes the need for users to search across multiple platforms, reducing cognitive load and improving research efficiency.

The implications of these results are notable for students, developers, and researchers. For beginners, the system shortens the learning curve by instantly identifying the most relevant learning materials. For academic researchers, it provides high-value references aligned with the topic. For developers, GitHub repository suggestions offer direct examples of real-world implementations. This makes the platform a powerful tool for quickly converting an idea into a structured project framework.

➤ *Advantages of the AI-Driven Multi-Agent Approach*

The Self-Building Project Manager Agent introduces a range of benefits that traditional manual research and project planning methods do not provide.

- **Speed and Efficiency:** The automated multi-agent pipeline dramatically reduces the time required for topic research and resource gathering. Instead of manually navigating YouTube or scanning through GitHub, the system consolidates and presents results within seconds,

enabling users to immediately begin project development.

- **Objectivity and Relevance:** Manual research often leads to inconsistent or irrelevant results due to subjective searching. The AI-based agents standardize the discovery process by ranking resources based on relevance, domain alignment, and educational value. This ensures that users receive consistent, unbiased, and high-quality recommendations every time.
- **Cost-Effectiveness:** By automating multiple research and content-generation tasks, the platform reduces the need for external tools or premium services. Students and developers gain access to curated resources without additional cost, making the system a financially accessible solution for academic and personal projects.
- **Accessibility and Convenience:** The platform's interface allows anyone regardless of technical background to generate structured project insights instantly. The unified interface simplifies the entire project-building journey and increases user engagement.

➤ *Challenges and Limitations of the Proposed Approach*

Although the Self-Building Project Manager Agent provides a fast and efficient way to generate project resources automatically, several challenges and limitations must be considered for real-world deployment.

- **Data Dependency and Relevance Matching:** The accuracy of the agent's recommendations—such as GitHub repositories, YouTube tutorials, and research papers—depends heavily on the external data sources being up-to-date. If repositories are outdated or videos lack educational depth, the system may occasionally produce suggestions that do not fully align with the user's intent.
- **Variations in Search Quality Across Platforms:** Different platforms structure and rank their content differently. This means the agent's search and filtering process may not always retrieve the best possible results, especially for very niche or emerging research topics. Maintaining consistent relevance across platforms is an ongoing challenge.
- **Computational Requirements for Multi-Agent Processing:** Running multiple agents simultaneously one for GitHub scraping, one for YouTube search, one for research paper retrieval, and one for PDF generation requires moderate server-side compute resources. While the workload is manageable now, large-scale use or heavy parallel requests may require further optimization or distributed processing.

➤ *Recommendations for Practical Implementation*

To maximize the real-world usefulness of the Self-Building Project Manager Agent, the following recommendations are proposed:

- **Build Larger and More Structured Resource Databases:** Instead of relying only on live searches, the system can maintain a periodically updated repository of curated GitHub projects, high-quality learning playlists, and verified research papers.
- **Integrate the Agent with Project Management Platforms:**

The system can be connected to platforms such as Notion, Jira, GitHub Projects, or Google Drive. This integration will allow users to directly export research papers, summaries, and task structures into tools they already use for project development.

- **Develop a Mobile App for On-Demand Research:** A lightweight mobile app would allow students and developers to instantly generate project insights on the go. Offline storage of previously generated reports can further improve accessibility and productivity.
- **Provide Customizable Templates for PDF Generation:** Allowing users to choose templates—for example, IEEE format, APA format, or project proposal format—will make the generated PDFs more adaptable to academic and professional requirements.

VI. FUTURE DIRECTIONS

While the current results show strong potential, several areas exist for further development to make the system more robust, scalable, and widely usable. Future improvements should focus on dataset expansion, smarter relevance detection, lightweight deployment for mobile devices, and seamless integration with academic and software development ecosystems.

➤ *Expanding Data Sources for Higher Accuracy*

A Future work should prioritize building richer, more diverse sources of information so that the system can generate more precise and context-aware recommendations. Important improvements include:

- **Broader Coverage of Research Topics:** Adding more research domains, especially interdisciplinary and emerging fields, will ensure comprehensive recommendations even for highly specialized topics.
- **Handling Topic Variations and Synonyms:** Integrating NLP-driven topic expansion will help the agent understand keyword variations and retrieve more accurate results.

➤ *Enhancing Model Capabilities for Better Resource Discovery*

Several advanced AI techniques can improve how the system selects and ranks recommendations:

- **Semantic Similarity Matching:** Instead of keyword-based search, embedding-based semantic search will ensure that retrieved GitHub and research papers closely match the user's intent.
- **Dynamic Filtering Based on User Profile:** The system can adapt recommendations based on user preferences, academic level, project difficulty, or previously selected topics.

➤ *Development of Lightweight Models for Mobile and Edge Deployment*

A key area for future enhancement is creating a lightweight version of the Project Manager Agent that can operate efficiently on mobile and low-power devices. The

current system relies heavily on cloud processing to run multiple agents—such as the GitHub recommender, YouTube extractor, and research-paper retrieval module. To increase accessibility, future work should focus on model compression, faster search pipelines, and optimized storage so that core features can run directly on smartphones. This would allow users to generate project resources instantly, even with limited internet connectivity, making the system more practical for students, developers, and field professionals who require quick, on-the-go project insights.

➤ *Integration and Ethical Considerations*

For broader adoption, the system should be seamlessly integrated into existing digital ecosystems such as university LMS platforms, project management tools, and documentation systems. Creating a dedicated API will allow external applications to directly access features like topic analysis, resource recommendations, and PDF generation. As the system scales, ethical considerations become crucial. Future development must ensure strong data privacy for user queries and saved dashboards, secure handling of generated PDFs, and transparent AI behavior. Continuous monitoring for bias—from search results, ranking algorithms, or data sources—is essential to ensure fairness and reliability across all domains and project categories.

VII. CONCLUSION

This project demonstrates the potential of an autonomous Self-Building Project Manager Agent, capable of managing complex research workflows data processing, model execution, documentation generation, and result visualization without continuous human supervision. By integrating deep learning models, a multi-agent architecture, and a MongoDB-based knowledge store, the system shows how AI can automate the entire lifecycle of technical project development.

The core strength of this system lies in its ability to dynamically handle tasks such as dataset indexing, model execution, progress tracking, and automated research-paper generation. The agent design ensures that each module data extraction, analysis, evaluation, and reporting communicates seamlessly, resulting in a coordinated and highly efficient workflow. This eliminates repetitive manual steps, significantly reducing time spent on documentation and allowing users to focus more on decision-making rather than operational processes.

A key contribution of this project is the demonstration of how AI agents can standardize and automate research-oriented tasks with accuracy, speed, and reproducibility. The system ensures consistency in quality by validating intermediate outputs, maintaining structured storage of experiments, and generating clear visual explanations that enhance interpretability. By utilizing MongoDB to store experiment states, metrics, and generated reports, the project also highlights how scalable databases can support traceable and transparent AI-driven workflows.

The impact of this technology goes beyond simple automation. It can be integrated into academic research labs, industrial R&D teams, and enterprise-level engineering environments to streamline quality checks, accelerate experimentation, and maintain strong documentation standards. Automated report generation also supports compliance, reduces human error, and promotes data-driven decision-making key factors in modern AI-enabled ecosystems

Although the system performs effectively, there are several promising directions for future improvement. Expanding the agent's capability to handle multimodal inputs—including PDFs, images, and experimental logs—would broaden its applicability. Incorporating real-time collaboration with cloud platforms could make the agent accessible across teams, while developing lightweight deployment versions would enable use in resource-constrained settings. Addressing ethical considerations, such as secure data handling and prevention of model bias.

In conclusion, this project provides a strong foundation for autonomous research and project-management systems. By combining AI agents, deep learning modules, and automated documentation pipelines, it demonstrates a practical approach toward building intelligent tools that can significantly enhance productivity, reliability, and technical accuracy. With continued development, such systems have the potential to transform how research, engineering, and project management are conducted, paving the way for more efficient and innovation-driven workflows.

REFERENCES

- [1]. Wu, T., Liang, P. (2024). *Auto Gen: Enabling Next-Generation LLM Applications via Multi-Agent Collaboration*. arXiv:2308.08155.
- [2]. Park, J. S., Ouyang, L., Liu, H., et al. (2024). *Generative Agents: Interactive Simulacra of Human Behavior*. ACM Transactions on Intelligent Systems and Technology.
- [3]. Microsoft Research. (2023). *TaskMatrix.AI: A Unified Platform for Multimodal and Agent-based Automation*. arXiv:2303.16434.
- [4]. Peng, B., Galley, M., He, X., et al. (2023). *Instruction-Following Large Language Models and Autonomous Task Automation*. arXiv:2305.17888.
- [5]. OpenAI. (2024). *GPT-5 & Autonomous Reasoning Systems: Technical Overview*. OpenAI Research Release.
- [6]. Shreya, G., & Narayanan, A. (2023). *AI Workflow Automation Using Multi-Agent Systems*. Journal of Intelligent Systems, 32(4), 620–635.
- [7]. Zheng, Y., Liu, T., & Ma, X. (2024). *LLM-Powered Research Assistants: Automating Analysis, Paper Generation, and Experimentation*. IEEE Transactions on Knowledge and Data Engineering.
- [8]. Zhang, Q., Zhao, S., & Li, J. (2023). *Autonomous Coding and Documentation Generation Using Large Language Models*. IEEE Software, 40(6), 58–67.
- [9]. Jin, C., Wang, H., & Du, S. (2024). *Intelligent Project*

Management Using AI Agents and Workflow Orchestration. Proceedings of the 2024 International Conference on Artificial Intelligence and Applications.

- [10]. MongoDB Inc. (2024). *Using MongoDB for AI Workflows, Vector Storage, and Agent Memory*. MongoDB Technical Documentation.