Grenze International Journal of Engineering and Technology, January Issue

Agentic AI Project Companion: A Framework for Consultative, Adaptive, and Proactive Research and Development

Sanket Dangle[1], Dev Mehta[2], Sarat Alane[3] and Dr. Chitra Bhole[4]
[1-4]K. J. Somaiya Institute of Technology/ Department of Computer Engineering, Mumbai, India
Email: sanket.dangle@somaiya.edu, dev.m@somaiya.edu, s.alane@somaiya.edu, cbhole@somaiya.edu

***Abstract*— The need for quick prototyping and the accelerating growth of open-source projects have driven a demand for smarter, more adaptive project management solutions. The paper summarizes the conceptual framework and one possible implementation of an Agentic AI Project Companion. At the core, is a consultative agent community in which expert AI agents work together to assist a user from initial concept through to a fully fleshed-out, documented project. The review looks at the system's most important phases, such as Project Initiation, Structuring and Planning, and Execution, and Adaptation. The system's innovative method of employing an army of agents in activities such as problem refinement, survey of literature, and timeline generation, presents an intriguing contrast to more conventional, static project management software. The paper also explains how the adaptive user skill layer and built-in research center dramatically improve the learning and documentation process of the user. Lastly, the paper suggests a methodology to validate the system's effectiveness and correctness based on of important performance indicators like project completion rates, user satisfaction, and quality of generated project artifacts.**

***Index Terms*— Multi-Agent Systems, agentic communication, shared memory, collaborative approach.**

I. INTRODUCTION

The contemporary development paradigm calls for a high velocity of innovation, with people and small groups being challenged to take abstract concepts and bring them into concrete projects. This is often hampered by the absence of systematic guidance, inability to effectively synthesize pertinent information, and the natural complexities of project management. Conventional tools, though helpful, tend to remain passive information repositories instead of active, intelligent collaborators. They act as electronic logbooks with manual input and little in the form of smart advice, anticipation, or proactive problem-solving (Nenni et al., 2024). This produces multiple key pain points as:

The Cold Start Problem: Getting a nascent idea into a project plan format is a significant hurdle [1]. Users, especially new ones, are struggling to define scope, set key tasks, and develop viable timelines.

Reactive vs. Proactive Support: There are moments when developers and researchers hit a problem. Locating the problem, tracking down solutions, and implementing them costs them with time and interfere with the creative process. A proactive support is not provided by current tools [1][2].

Grenze ID: 01.GIJET.12.1.135

Static Planning in a Dynamic World: Project plans are usually rigid and become outdated in no time as the project grows further. Adapting to this growth requires a full manual overhaul of the plan.
Siloed Workflows: The planning, coding, research, documentation workflows are processed in standalone applications, which creates context switching, loss of information, and disconnected project history [3].
To conquer these challenges, this paper proposes the "Agentic AI Project Companion", a departure from traditional human-AI collaboration and how it can propel projects from conceptualization to delivery. It is not only a tool but an active, intelligent companion. At its core lies an advisory space of domain-specific AI agents working in concert to guide and learn from the user's needs throughout the life cycle of a project.
Drawing inspiration from the principles of agentic AI systems that can perceive, reason, plan, and act autonomously to achieve goals - the Companion is designed to be:
Consultive: Various agents of the environment exchange information with one another, draw upon memories of one another and operate based on the consultations to work towards a more common objective [4][5].
Adaptive: The system also incorporates an "Adaptive User Skill Layer" that ranks the user's expertise based on their questions and code quality. This enables the agents to adapt itself to user's comprehension and requirements [2][3][6].
Integrated: It integrates planning for projects, monitoring execution, research, and documentation into one cohesive, integrated environment, removing the context switching friction from working with separate tools.

II. Literature Review

The conception of the Agentic AI Project Companion is grounded in several converging fields of research: agentic AI and multi-agent systems, AI's evolving role in project management, automated research assistance, and the dynamics of human-AI collaboration.

A. The Rise of Agentic AI amd Multi Agent Systems (MAS)

Agentic AI is a quantum improvement over generative AI, going far beyond content generation to self-sustaining, goal-oriented task performance [7]. They are programmed to decompose complicated issues into workable sub-tasks and perform them, sometimes utilizing an array of software "tools" without manual intervention. Tools such as LangGraph, AutoGen, and CrewAI have appeared to make building these systems easier, allowing developers to coordinate various specialized agents into a cooperative "crew" (DigitalOcean, 2025) [8].
A multi-agent system (MAS) is composed of multiple autonomous agents that co-operate in order to tackle problems that are beyond individual agents' capabilities [2][7]. This cooperative framework is at the core of the proposed Companion. Studies have shown the success of MAS in compound areas such as distributed team management (Cherednichenko et al., 2021) [9] and Agile software development (Cinkusz & Chudziak, 2025) [3], where agents may emulate jobs such as "Project Manager" or "QA Engineer". The system proposed here takes this a step further in that it builds a consultative loop, where agents evaluate and improve upon one another's work, reflecting the collaborative deliberation of an expert human group.

B. AI in Project Management

AI implementation in project management has continued to expand, with the tools now capable of features such as predictive risk management, task scheduling, and intelligent data analysis (Lark, 2025). Research has demonstrated that AI has the potential to revolutionize project efficiency through enhanced resource and cost management (Haghsheno, 2021) [10]. Eg. Gomes and Thomas (2024) discovered that GPT-4 can surpass other models in task speed, completion accuracy, and scalability for project management tasks [11].
Yet, a main gap found in the literature is the "lack of a unified framework" and the continued existence of issues concerning data quality, skill gaps among users, and the general-purpose nature of most AI models (Nenni et al., 2024; Gomes & Thomas, 2024) [11]. Most tools are still reactive, not having deep, domain-specific contextual awareness, and failing to support well the essential first stage of project ideation and structuring. The Agentic AI Project Companion addresses this gap directly by offering an integrated, proactive, and context-aware system that directs the user from the start of their project path.

C. Automated Research and Documentation

One of the key roles of the Companion is to help with the research and documentation part of a project. Scientific literature analysis using LLMs is an area that's advancing very fast. Tools such as "LitLLMs" are being built to create literature reviews automatically by breaking down the task into retrieval and synthesis

(LitLLMs, n.d.). Other work centers on employing LLMs to pull out structured information from scientific documents (Li et al., 2025) [12] and even assisting in replicating findings in research.
Though powerful, these tools tend to be isolated from the development process. One acknowledged limitation is "hallucinated references" and the inability to evaluate the quality and appropriateness of generated text [12][13]. The Companion incorporates the Literature_Survey_Agent into the project process. It does not generate the review for the user but instead offers a formalized analysis of work that has already been conducted - sketching out methodologies and noting gaps in research - thereby enabling the user to produce a better-quality review themselves and reducing the dangers of uninformed AI production [14][15].

D. Human-AI Collaboration and Adaptation

Successful human-AI collaboration is not merely delegating tasks to an AI but designing a synergistic partnership. Success depends on balancing human ingenuity with AI support and trust through openness and dependability [16]. The research indicates that AI can be a great partner in software engineering, enhancing developer decision-making and code quality.
One of the innovations of the proposed system is the Adaptive User Skill Layer. The idea of adaptive user interfaces (AUIs), adjusting their layout and information density according to the needs of the user, is widely known [17][18]. Based on the user's query sophistication and code quality, the Companion can shift between its communication style to behave like a patient tutor to a beginner or a high-level peer to an expert. Such dynamic adjustment makes sure that the collaboration is effective and interesting for the user no matter what his/her level of initial skill is [17][19].

III. METHODOLOGY

The Agentic AI Project Companion is built upon a foundation of interconnected concepts and a modular architecture. The system's intelligence emerges from the synergy between its specialized agents, the data flowing between them, and its adaptive interface to the user [20].

A. Key Concepts and Components

The system's centre is a team of customized AI agents developed with LangGraph. The agents have individual purposes but partake of a shared state and may call upon one another for input and validation. The system's backbone is a team of highly specialized AI agents constructed with LangGraph which enables the development of cyclical, stateful workflows, critical to the consultative loops among the agents. Each agent must play a unique role, yet they hold a shared state and may call upon one another for input and verification [20]. Fig. 1 illustrates the proposed architecture for the system which is described below:
Literature Survey Agent - This agent is the research wing of the system. It possesses mechanisms to browse and read academic databases (such as arXiv and IEEE Xplore), web search APIs, and a local user-uploaded document vector database. It gives summaries, detects methodologies, and identifies research holes [21].
Flowchart Agent - A visually orientated agent which accepts the problem statement refined and comes up with the logical structure of the project. Produces visual representations (e.g., through Graphviz) that diagram the system's components and data flow, with guarantees of logical consistency [22][23].
Timeline Agent - The project scheduler breaks down the flowchart into a list of doable steps. Referring to the Literature Agent, it makes time estimates from similar projects, producing a GANTT chart [22][23].
Progress Tracker Agent - The project tracker. This agent has a GitHub API utility (PyGithub) to monitor the user's repository. It analyses commit messages to update the status of tasks automatically in the project database and is able to identify phases of inactivity or straying from the plan [23].
Critic Agent - A cross-functional agent that evaluates the outputs of other agents. For instance, it verifies whether the plan of the Timeline Agent aligns with the Flowchart Agent's design. The internal peer review guarantees the quality and consistency of the system's advice.
Adaptive User Skill Layer - It is not an individual agent but a guiding logic that alters the action of all agents [17]. The system keeps a dynamic user profile based on:
Sophistication of Questions: Examining the technicality and complexity of the user's queries in the chat.
Code Quality & Commit Patterns: Examining code structure, commenting, and purposefulness of code committed to the GitHub repository [22].
Interaction Frequency: Monitor the frequency with which the user needs help or clarification. From this profile, the system adapts its response. For a beginner-flagged user, answers are longer, providing links to root concepts, and explicit code examples.

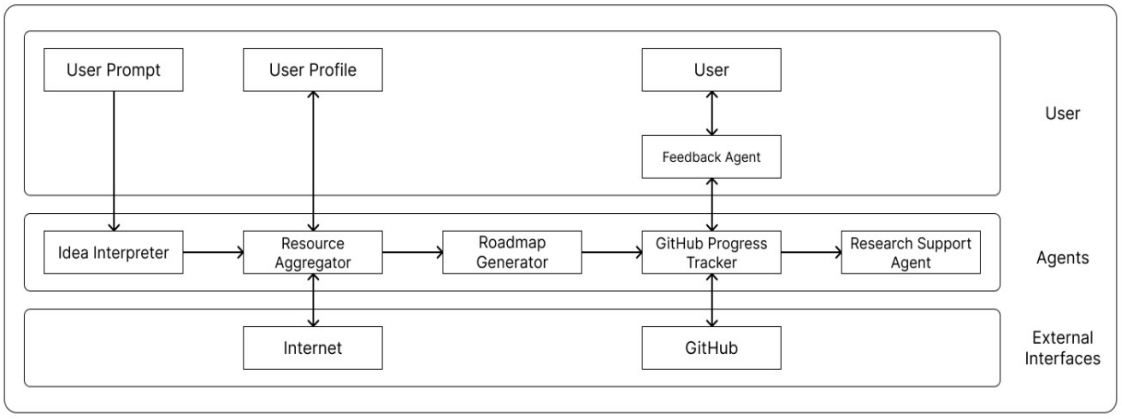

Figure 1. Proposed System Architecture

For an expert-flagged user, answers are brief, utilizing technical terminology, with more high-level strategic commentary in nature, acting more like a peer colleague [22].
The technology stack used for this can be grouped as below –

AI & Orchestration
LangGraphs can be used for constructing the central multi-agent graph, to facilitate sophisticated, consultative communication loops. Large Language Models (GPT-4/Gemini) can be utilized as the individual agent's reasoning engine, performing tasks such as planning, summarization, and analysis.

Backend
FastAPI can be used as the primary API server, receiving requests from the frontend and initiating the LangGraph workflows, whereas PostgreSQL as the main relational database to store structured data such as user accounts, project plans, task lists, and citation metadata [24]. Additionally, ChromaDB can be used as a vector database to store document embeddings, allowing the Literature Survey Agent to carry out semantic searches [25][26][27].

Frontend
Streamlit can be utilized to create the entire user-facing application, including the dashboard, chat interface, and research hub, whereas Pyplot or Graphviz can be incorporated to provide visual representations of the outputs of the Flowchart Agent and Timeline Agent [10][24].

B. Project Development Roadmap

The development of the Agentic AI Project Companion will follow a phased approach as outlined below -
The first phase is all about creating the project's base infrastructure, including configuration of FastAPI backend server, and creating the initial PostgreSQL database schema for users, projects, and tasks. UI & Authentication [24][25]. Create the basic Streamlit app structure and create a secure login system utilizing streamlit-authenticator, tightly coupled with the user table in the database. Construct and thoroughly test the API connectors that will be the agents' "tools." This comprises a wrapper for the GitHub API (PyGithub) and wrappers for academic search (e.g., the arxiv library) and general web search.
Next, the emphasis comes to constructing the intelligence of the system. Roll out the main logic for the Planner Agent and the Literature Survey Agent. This means writing the core prompts and declaring their main functions. Create a simple, linear flow that relates a user input to the Planner Agent then the Researcher Agent. The outputs of this chain will be recorded into the PostgreSQL database. Recreate the basic chain into a more complex, cyclical graph with LangGraph. This stage includes the design of the state graph, the execution of the router logic, and construction of the initial consultative loop (for example, Planner - Critic - Planner). Integration of all the agents into the graph will be done.
The next step is focuses on developing the user interface aspects and integrating them with the backend agentic system. Develop the backend functionality of PDF upload and summarization. Simulataneously, developing the frontend UI for uploading documents and observing the managed list of citations, integrating the dashboard with the PostgreSQL database to show real-time project status, pending work, is to be done.

The last phase is focused on making the system reliable, robust, and ready for deployment and academic reporting. Perform end-to-end thorough testing of the entire prototype. Fix bugs throughout the stack and execute immediate tuning to enhance the reliability and consistency of the output of the agents. Prepare and conduct formal evaluation case studies. Gather and analyse logs and user feedback in order to evaluate the performance of the system against pre-defined metrics.

C. System Working

The interaction of the user with the Agentic AI Project Companion is an unbroken process through four distinct yet interrelated stages, all executed by an interactive user interface.

The Dashboard - This is the home page of the project. It gives a global, read-only view of the health of the project, such as a completion progress bar, a list of upcoming tasks, and any notification from the agents. It is feed directly by the content in PostgreSQL, updated real-time by the Progress Tracker Agent [28].

The Chatbot Interface - It is the main interactive part. The user messages trigger the LangGraph agent workflows. The interface sends streaming responses, displaying the "thoughts" of the agents while working (e.g., "Literature Survey Agent" is looking for papers on collaborative filtering."). This makes the system seem to be transparent and alive [29][30].

The Research Hub - This is the project's own library. PDFs can be uploaded by users, which will be summarized automatically and included in a central, searchable list of citations. The text is also embedded and saved in ChromaDB, enabling users to pose semantic questions about their research library

The four phases of system working is given below –

The first phase starts with a user entering an initial, high-level concept in the chat interface, e.g., "I would like to create an app that suggests books." The Problem Refiner Agent takes initiative, asking insightful questions to discern purpose ("Are you thinking of recommendations based on user ratings or book content?"). Behind the scenes, it references the Literature Survey Agent to anchor its questions on typical methods so that the refinement process can be streamlined and with knowledge. This step is completed once the user confirms an accepted problem statement.

With a defined objective, the system proceeds to formulate a specific plan. The Flowchart Agent conceptualizes the project architecture, creating a diagram of the application's logic. The flowchart is then handed over to the Timeline Agent, which decomposes it into an elaborate GANTT chart of tasks and milestones, with realistic time expectations by consulting the Literature Survey Agent regarding the time required for similar, documented projects. The output is an unambiguous roadmap that the user may follow.

This is the actual development phase. The user starts coding and commits their work to a linked GitHub repository. Progress Tracker Agent tracks this action. A commit message such as "feat: implemented genre-based matching algorithm" would automatically mark the corresponding task as "complete" on the dashboard. If it senses that a user is getting stuck (e.g., no activity on a critical task for a few days), it will automatically notify the Literature Survey Agent to locate and send relevant resources to the user through the chat interface. If the user strays from the plan, the system will politely ask if scope has shifted and will volunteer to start a re-planning cycle.

At any point, the user may utilize the system as a research assistant. In response to the query, "How do I do the literature review in this?", the Literature_Survey_Agent does not produce the review but identifies major papers and offers a systematic analysis with synopses of alternative methodologies and recurrent limitations raised by previous researchers. This equips the user with the analytical building blocks required to write an informed and critical review themselves, supported by the citation management tools in the Research Hub.

IV. RESULTS AND EVALUATION

This section provides the framework and measures against which its performance, accuracy, and usefulness will be assessed when the prototype is complete. The assessment will be multi-faceted with quantitative measures blended with qualitative user feedback accomplished by a series of controlled case studies. The section also utilises existing evaluation ideas utilised by previous researchers in this field [31][32].

A. Evaluation Framework

The analysis will be framed to respond to three central questions:

- Efficiency: Does the Companion shorten time and effort to transition from an idea to a finished project?
- Quality: Does the Companion enhance the quality of the project's deliverables? (e.g., plan, code)
- Usability: Is the system easy to use, reliable, and beneficial for the user?

TABLE I. CONTAMINATION RATES BY LLMS [12]

Evaluation Datasets	In-house Textbooks (%)	In-house Journals (%)
SciRIFF	1.1	0.7
SciAssess	11.0	1.1

TABLE II. MULTI-AGENT SYSTEMS' RESOURCE UTILIZATION [9]

Case Study	Agents Quantity	Memory Usage	CPU Usage (in %)
1	8	268 MB	0.03 – 1
2	1000	1.2 GB	5 - 10
3	1000000	4.0 GB	32 - 67

B. Quantitative Metrics

The correctness and efficiency of the system will be gauged from data recorded in the PostgreSQL database and from the GitHub repositories [4][6][31].

Planning Efficiency

- Time-to-Plan - Capture the time from the user's first request to their confirmation of the final project plan and timeline produced by the agents. Compare this with the time for the control group.
- Plan Completeness - The task list that is automatically generated will be compared by impartial reviewers in terms of completeness and logical consistency to those plans designed manually by the control group [7][33].

Execution and Proactive Support

- Task Completion Speed - Monitor the task completion speed (tracked by Progress Tracker Agent).
- Blocker Resolution Time - When the system actively detects a user is blocked, measure the time from when the suggested resource was delivered until the next relevant commit.
- Deviation from Plan - Measure the number of tasks accomplished that are outside of the initial plan, and measure the effectiveness of the system in the initiation of re-planning cycles [6][35].

Research and Documentation Quality

- Literature Review Quality - The experimental group users' literature reviews will be marked by domain experts and contrasted with the control group for the extent of sources covered and the critical examination of gaps [35][37].
- Citation Accuracy - Quantify the accuracy and the formatting of citations automatically exported from the Research Hub [37].

C. Identifying Correctness

The correctness of the output of the system will be established using a mix of automated verification and human-in-the-loop validation.

Logical Coherence: The Critic Agent offers an initial level of internal consistency checking. Outputs will additionally be examined by human professionals to verify that the suggestions are logically consistent [37].

Factual Grounding: Summaries and analyses provided by the Literature Survey Agent will be compared with the source documents to assess factual accuracy and detect any cases of hallucination [34].

Table I, for instance, illustrates the error rate in comprehending Scientific Journals and Publications by intelligent systems [12].

Code Viability: Architectural ideas or code snippets will be assessed on if they are executable, are following best practices, and are applicable to the problem of the user [18][35].

Through integration of these metrics, a comprehensive evaluation of the effectiveness of Agentic AI Project Companion and its ability to alter the research and development process can be constructed.

Further, Resource Utilization plays an important role in communication among Multi-Agent Systems. Table II demonstrates experiments by Cherednichenko et al [9] who considered various case studies by using multiple agents, clearly showing that resource utilization increases gradually with number of agents.

V. CONCLUSION

The Agentic AI Project Companion is a major conceptual and technological innovation over conventional project management software. Shifting the paradigm from passive monitoring to active, smart collaboration, the system solves key inefficiencies in the contemporary research and development cycle. The central innovations -

a consultative multi-agent environment, anticipatory and context-sensitive guidance, and an adaptively responding user skill layer, are intended to form a synergistic relationship between the human user and the AI. The system does not supplant human creativity but enhances it, taking on the mechanical and organizational drudgery of project management such that users are free to engage in higher-level creative and strategic thought. The comprehensive project plan, based on a solid and up-to-date technology stack, establishes a clear path for implementation. The suggested evaluation framework, integrating strict quantitative measures with thoughtful qualitative comments, will enable extensive evaluation of the system's effect on project efficiency, output quality, and user experience. Although the construction of such a system is challenging, in terms of making agentic workflows reliable and human-AI collaboration seamless, the value added is immense. Agentic AI Project Companion has the capability to democratize software research and development to both experts and novices and enable them to convert grand ideas into properly designed, executed, and documented projects with increased speed and assurance. In the end, this project is a map to the future of intelligent tools that will enable tomorrow's innovators.

There will be further work in the future to improve the agentic behaviors, integrate more tools, and test the system on a broader set of educational and industrial applications. In addition to technical advancements, one key direction in the future is scaling up the system to accommodate team and multi-user collaboration, requiring agents to deal with sophisticated interactions, enable intelligent task delegation according to team members' capabilities, and keep track of a consistent project state among various contributors.

REFERENCES

[1] M. E. Nenni, F. De Felice, C. De Luca, and A. Forcina, "How artificial intelligence will transform project management in the age of digitization: a systematic literature review," Management Review Quarterly, Apr. 2024, doi: 10.1007/s11301-024-00418-z.

[2] L. Wang et al., "A survey on large language model based autonomous agents," Frontiers of Computer Science, vol. 18, no. 6, Mar. 2024, doi: 10.1007/s11704-024-40231-1.

[3] K. Cinkusz and J. Chudziak, "Agile Software Management with Cognitive Multi-Agent Systems," Proceedings of the 14th International Conference on Agents and Artificial Intelligence, pp. 385–392, Jan. 2025, doi: 10.5220/0013153000003890.

[4] D. Maldonado, E. Cruz, J. A. Torres, P. J. Cruz, and S. Del Pilar Gamboa Benitez, "Multi-agent Systems: A survey about its components, framework and workflow," IEEE Access, vol. 12, pp. 80950–80975, Jan. 2024, doi: 10.1109/access.2024.3409051.

[5] P. A. Olujimi, P. A. Owolawi, R. C. Mogase, and E. Van Wyk, "Agentic AI Frameworks in SMMEs: A Systematic Literature Review of Ecosystemic Interconnected Agents," AI, vol. 6, no. 6, p. 123, Jun. 2025, doi: 10.3390/ai6060123.

[6] I. Taboada, A. Daneshpajouh, N. Toledo, and T. De Vass, "Artificial Intelligence Enabled Project Management: A Systematic Literature Review," Applied Sciences, vol. 13, no. 8, p. 5014, Apr. 2023, doi: 10.3390/app13085014.

[7] J. K. Das, I. Elegbe, Lord Coffie, R. Khadka, L. Chen, and Y. Ji, "AI-Powered IT Project Management: Analyzing the effectiveness of advanced project management tools to ensure project efficiency," SoutheastCon, pp. 1554–1559, Mar. 2025, doi: 10.1109/southeastcon56624.2025.10971718.

[8] K. B. Siddiqa et al., "AI-Driven Project Management Systems: Enhancing IT Project Efficiency Through MIS Integration," ICPIDS-2024, pp. 114–119, Dec. 2024, doi: 10.1109/icpids65698.2024.00027.

[9] O. Cherednichenko, O. Matveiev, O. Yanholenko, R. Maneva, and National Technical University "Kharkiv Polytechnic Institute," "Multi-Agent modeling of project management processes in distributed teams," 2021. [Online]. Available: http://ceur-ws.org/Vol-2851/paper12.pdf

[10] S. Haghsheno, "AI-driven project management in software engineering," IJSDR.org, Jan. 01, 2021. https://ijsdr.org/viewpaperforall.php?paper=IJSDR2101050

[11] R. F. Gomes and L. Thomas, "DIP AI-Driven Architecture for Enhanced Project Management Using Large Language Models," IEEE ICTBIG-2024, pp. 1–8, Dec. 2024, doi: 10.1109/ictbig64922.2024.10911129.

[12] S. Li et al., "SCILITLLM: How to Adapt LLMs for Scientific Literature Understanding," arXiv (Cornell University), Aug. 2024, doi: 10.48550/arxiv.2408.15545.

[13] H. Ghanem and C. Cruz, "Fine-tuning or prompting on LLMs: evaluating knowledge graph construction task," Frontiers in Big Data, vol. 8, Jun. 2025, doi: 10.3389/fdata.2025.1505877.

[14] A. Chinmulgund, R. Khatwani, P. Tapas, P. Shah, and R. Sekhar, "Anthropomorphism of AI based chatbots by users during communication," 2023-CONIT, pp. 1–6, Jun. 2023, doi: 10.1109/conit59222.2023.10205689.

[15] P. Dhyani, S. Nautiyal, A. Negi, S. Dhyani, and P. Chaudhary, "Automated API Docs Generator using Generative AI," 2020 IEEE International Students' Conference on Electrical,Electronics and Computer Science (SCEECS), vol. 33, pp. 1–6, Feb. 2024, doi: 10.1109/sceecs61402.2024.10482119.

[16] S. K. Vishnumolakala, S. C. C, N. P. Subheesh, P. Kumar, and R. Kumar, "AI-Based Research Companion (ARC): an innovative tool for fostering research activities in undergraduate engineering education," 2022 IEEE Global Engineering Education Conference (EDUCON), pp. 1–5, May 2024, doi: 10.1109/educon60312.2024.10578646.

[17] V. Viradia, V. K. Mishra, and S. Parakh, “Agentic AI for Medicare Annual Wellness Visits: Improving Preventive Care,” BioSMART Vol-6, pp. 1–6, May 2025, doi: 10.1109/biosmart66413.2025.11046123.
[18] N. F. M. Ozman, “Systematic literature review on the rise of agentic AI in enterprise operations,” International Journal of Frontiers in Science and Technology Research, vol. 8, no. 2, pp. 001–015, Apr. 2025, doi: 10.53294/ijfstr.2025.8.2.0025.
[19] S. Spillias et al., “Human-AI collaboration to identify literature for evidence synthesis,” Cell Reports Sustainability, vol. 1, no. 7, p. 100132, Jul. 2024, doi: 10.1016/j.crsus.2024.100132.
[20] R. J. Tong et al., “IEEE AI Standards for Agentic Systems,” IEE 2023-CAI, pp. 1603–1609, May 2025, doi: 10.1109/cai64502.2025.00269.
[21] N. Kanataria, K. P. Patel, H. N. Patel, P. Goel, K. Patel, and D. Garg, “RAG-Enhanced Large Language Model for Intelligent Assistance from Web-Scraped Data,” 2022 7th International Conference on Communication and Electronics Systems (ICCES), pp. 1043–1048, Dec. 2024, doi: 10.1109/icces63552.2024.10859894.
[22] A. M. Felicetti, A. Cimino, A. Mazzoleni, and S. Ammirato, “Artificial intelligence and project management: An empirical investigation on the appropriation of generative Chatbots by project managers,” Journal of Innovation & Knowledge, vol. 9, no. 3, p. 100545, Jul. 2024, doi: 10.1016/j.jik.2024.100545.
[23] A. Bharati and C. Sandbrink, “The implementation of artificial intelligence in project management,” REAL CORP 2024 Proceedings/Tagungsband, p. 627, 2024, doi: 10.48494/REALCORP2024.0078.
[24] T. Samarah, M. Almiani, A. Mughaid, S. AlZu’bi, and A. Al-Rahayfeh, “Intelligent Strategic Decision-Making for optimized project Management,” 2022 International Conference on Decision Aid Sciences and Applications (DASA), pp. 1–5, Dec. 2024, doi: 10.1109/dasa63652.2024.10836504.
[25] M. K. K, S. MehtaS, A. H. L, S. N, G. Jadhav, and A. Mitra, “Neuromorphic-Driven Agentic AI for Autonomous Decision-Making Systems,” ICMNWC-2024, pp. 1–8, Dec. 2024, doi: 10.1109/icmnwc63764.2024.10872131.
[26] H. Wei, Z. Zhang, S. He, T. Xia, S. Pan, and F. Liu, “PlanGenLLMS: A Modern Survey of LLM Planning Capabilities,” arXiv (Cornell University), Feb. 2025, doi: 10.48550/arxiv.2502.11221.
[27] N. Rakshitha, P. Mohan, M. B. Shanthi, D. N. Disha, and S. Rao, “Advances in Natural Language Processing and Deep Learning for Document Summarization,” 2023-ICIICS, pp. 1–6, Nov. 2023, doi: 10.1109/iciics59993.2023.10421343.
[28] X. Ma and J. Wang, “WIP: Active Learning through Prompt Engineering and Agentic AI Simulation-A Pilot Project in Computer Networks Education,” 2021 IEEE Frontiers in Education Conference (FIE), pp. 1–5, Oct. 2024, doi: 10.1109/fie61694.2024.10892925.
[29] S. Hosseini and H. Seilani, “The Role of Agentic AI in Shaping a Smart Future: A Systematic review,” Array, p. 100399, May 2025, doi: 10.1016/j.array.2025.100399.
[30] W. Wysocki and C. Orłowski, “A multi-agent model for planning hybrid software processes,” Procedia Computer Science, vol. 159, pp. 1688–1697, Jan. 2019, doi: 10.1016/j.procs.2019.09.339.
[31] Agents and Multi-agent Systems: Technologies and Applications 2024. 2025. doi: 10.1007/978-981-97-6469-3.
[32] Z. Alliata, T. Singhal, and A.-M. Bozagiu, “The AI Scrum Master: Using Large Language models (LLMs) to automate agile project management tasks,” in Lecture notes in business information processing, 2025, pp. 110–122. doi: 10.1007/978-3-031-72781-8_12.
[33] D. Tong, B. Jin, Y. Tao, H. Ren, A. Y. M. A. Islam, and L. Bao, “Exploring the role of human-AI collaboration in solving scientific problems,” Physical Review Physics Education Research, vol. 21, no. 1, May 2025, doi: 10.1103/physrevphyseducres.21.010149.
[34] A. Przegalinska, T. Triantoro, A. Kovbasiuk, L. Ciechanowski, R. B. Freeman, and K. Sowa, “Collaborative AI in the workplace: Enhancing organizational performance through resource-based and task-technology fit perspectives,” International Journal of Information Management, vol. 81, p. 102853, Nov. 2024, doi: 10.1016/j.ijinfomgt.2024.102853.
[35] T. Singh, “Human-AI Collaboration in Project Management,” Digital Commons at Harrisburg University. https://digitalcommons.harrisburgu.edu/dandt/30
[36] S. S. Binyamin and S. B. Slama, “Multi-Agent systems for resource allocation and scheduling in a smart grid,” Sensors, vol. 22, no. 21, p. 8099, Oct. 2022, doi: 10.3390/s22218099.
[37] S. Karakama, N. Matsunami, and M. Ito, “Task decomposition and role sharing for real-time Human-AI swarm collaboration,” International Journal of Smart Computing and Artificial Intelligence, vol. 5, no. 1, pp. 33–50, Jan. 2021, doi: 10.52731/ijscai.v5.i1.637.
[38] H. Li, Y. Hao, Y. Zhai, Z. Qian, and University of California, Riverside, USA, “Enhancing static analysis for practical bug detection: an LLM-Integrated approach,” journal-article, 2024. [Online]. Available: https://www.cs.ucr.edu/~zhiyunq/pub/oopsla24_llift.pdf