IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

An Intelligent Service Desk Framework For Incident Lifecycle Management And Support Analytics Using Agentic AI-Based Query Processing

Bimla S D[1], Chandan C[1], Kishan Kumar K R[1], Nidarsh M P[2]

[1] Students, Department of CSE–AIML, MIT Mysore

2Assistant Professor, Department of CSE–AIML, MIT Mysore

Abstract: Service desk systems are a critical component of organizational IT support, responsible for handling incident reporting, tracking, and resolution. Conventional service desk solutions depend largely on manual ticket handling and human-driven query resolution, which often results in delayed responses, operational overhead, and limited real-time insights. This paper presents the design and implementation of a conversational Service Desk Agent that automates essential service desk functions through structured backend logic and intent-based query processing. The proposed system enables users to create incidents, retrieve incident status, update records, and obtain analytical summaries such as total incidents, in-progress cases, and unresolved critical issues. A modular architecture is adopted, comprising a user interaction layer, backend services, and an agent-processing layer to ensure reliability and extensibility. The results indicate that the proposed approach effectively reduces manual effort while improving responsiveness and usability, making it suitable for practical service desk environments and future system enhancements.

Index Terms - Incident management, Conversational interface, Service desk automation, Support systems, Ticket analytics

I. INTRODUCTION

Service desk systems play a vital role in supporting organizational IT operations by acting as the primary interface between users and technical support teams. These systems manage incident reporting, service requests, and operational queries, ensuring continuity and reliability of IT services. As organizations increasingly depend on digital infrastructure, service desks are required to process a growing number of incidents efficiently while maintaining accuracy and timely responses.

Traditional service desk solutions rely mainly on manual or semi-automated workflows, such as email-based ticket submission and human-driven query handling. While functional, these approaches often result in delayed responses, inconsistent incident tracking, and increased workload for support teams. The absence of real-time analytical insights further limits the ability to monitor ongoing and critical incidents effectively. These challenges highlight the need for systems that can automate routine operations and provide structured access to incident information.

This paper presents the design and implementation of a conversational Service Desk Agent that automates essential service desk functions through intent-based query processing and backend integration. The proposed system supports incident creation, status retrieval, record updates, and analytical queries using a modular architecture. By reducing manual intervention and improving response efficiency, the system aims to enhance service desk operations while remaining scalable and adaptable for future integration and enhancement.

II. PROBLEM STATEMENT

Service desk operations in many organizations continue to rely on manual or semi-automated processes for incident handling and user support. Common tasks such as incident creation, status tracking, updates, and basic analytical queries often require human intervention, leading to delayed responses, increased workload, and inconsistent information delivery. Existing systems provide limited real-time insights into ongoing and critical incidents, making it difficult for users and support teams to quickly obtain accurate status updates. Additionally, the lack of an intuitive conversational interface forces users to depend on structured forms or direct communication with support staff, reducing efficiency. Therefore, there is a need for a reliable and automated service desk solution that can interpret user queries, manage incident information effectively, and provide timely, accurate responses while minimizing manual effort.

III. EXISTING SYSTEM

Existing service desk systems primarily operate through manual or semi-automated mechanisms such as email-based ticket submission, phone calls, and staff-managed ticketing tools. User requests are recorded and processed by human operators, who are responsible for categorizing incidents, tracking their progress, and providing updates. While these systems support basic incident management, they often lack real-time analytical capabilities and require significant human effort for routine queries such as checking incident status or counting ongoing issues. As the volume of support requests increases, these approaches become difficult to scale, leading to delayed responses, higher operational costs, and inconsistent user experience.

IV. PROPOSED SYSYTEM AND METHODOLOGY

The proposed system is an automated Service Desk Agent designed to improve incident management through a conversational interface. Users can interact using natural language to create incidents, retrieve status information, update records, and request analytical summaries. The system follows a modular, layered architecture consisting of a user interaction layer, a backend service layer developed using Python and FastAPI, and an agent-processing layer responsible for intent detection and task execution. Incident data is stored and managed using an in-memory database to support fast retrieval and updates. The methodology includes requirement analysis, system design, backend and agent implementation, and comprehensive testing using real service desk scenarios to validate accuracy, performance, and reliability. This approach ensures efficient service desk operations while maintaining scalability for future enhancements.

V. RESULTS AND DISCUSSION

The experimental evaluation of the Service Desk Agent demonstrates that the proposed system effectively automates key service desk operations while maintaining accuracy and reliability. The agent successfully handled incident creation, status retrieval, incident updates, and analytics-based queries through natural language interactions. Test results show that user queries were correctly interpreted and mapped to appropriate backend functions, resulting in accurate database operations and real-time responses. Analytical queries such as total incidents, incidents in progress, and unresolved critical incidents were processed efficiently, confirming the correctness of filtering and aggregation logic. The conversational responses remained clear and context-aware even for varied input formats, highlighting the robustness of the intent detection mechanism. Performance testing indicated stable and fast execution across multiple queries, with effective error handling for incomplete or ambiguous inputs. These results validate that the agentic architecture, combined with structured backend integration, provides a reliable and scalable solution for automated service desk management, aligning well with practical enterprise support requirements.

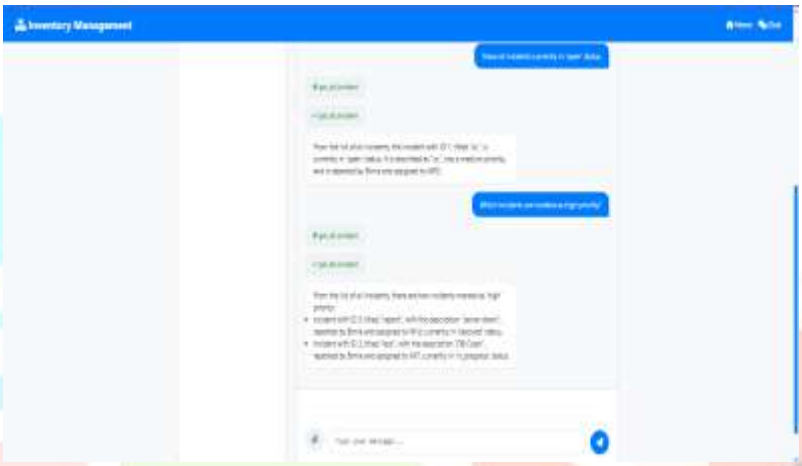

Fig 5.1: Service Desk Agent Chat Interface

VI. CONCLUSION AND FUTURE WORK

This project successfully designed and implemented an AI-enabled Service Desk Agent that automates incident management through a conversational interface. The system demonstrated accurate handling of incident creation, status tracking, updates, and analytics-based queries while maintaining reliable performance and clear user interaction. By integrating agentic logic with backend data processing, the solution reduces manual effort, improves response time, and enhances overall service desk efficiency. As future work, the system can be extended to support advanced features such as automated ticket escalation, email and notification alerts, integration with enterprise ITSM tools, and deployment on cloud platforms with larger databases. Incorporating learning-based models and multilingual support can further improve adaptability and user experience, making the solution suitable for real-world organizational environments.

REFERENCES

[1] R. Hernandez, S. Patel, and K. Wong, "TickIt: Leveraging Large Language Models for Automated Ticket Escalation," IEEE Transactions on Cloud Computing, 2025.

[2] N. Dutta and M. M. Sarker, "Agentic AI-Driven Technical Troubleshooting for Enterprise Systems: A Novel Weighted Retrieval-Augmented Generation Paradigm," IEEE Access, 2024.

[3] J. Park, L. Chen, and S. Rao, "DECO: Life-Cycle Management of Enterprise-Grade Chatbots," Proceedings of the IEEE International Conference on Enterprise AI Systems, 2024.

[4] A. Sharma, R. Iyer, and P. Kulkarni, "Enhancing Customer Support in Banking: Leveraging AI for Efficient Ticket Classification," Journal of Financial Technology and Analytics, 2024.

[5] M. Alvarez and T. Becker, "AI-Based Chatbots in Customer Service: A Systematic Review," Journal of Artificial Intelligence Research, 2025.