Information Systems Student Research Journal

Innovation Snapshot DOI: ISSN:

Beacon: Pioneering the Future of Artificial Intelligence in Information Technology Service Management

Elliot Miller
The University of Alabama
elliotmmiller21@outlook.com

Sam Riddle
The University of Alabama

Andrew Hudson
The University of Alabama

Overview

"Beacon" is a comprehensive Information Technology (IT) automation solution that utilizes artificial intelligence (AI) to revolutionize the management and maintenance of enterprise-level infrastructure, including servers, networks, and data storage systems. The solution functions as an IT Service Management (ITSM) platform, designed to automate and streamline core IT support processes. The platform was conceived and developed as a submission to the AI-Driven IT Support and Routine Task Automation Challenge for the Association of Information Systems Student Chapter Leadership Conference 2025. The project won first place in the competition. By integrating AI and advanced automation techniques, Beacon minimizes human error, reduces operational overhead, and supports the integrity of critical systems.

The Vision

The goal of the Beacon project was to create a self-contained system that could proactively and autonomously address common IT challenges such as system downtime, network performance issues, and security vulnerabilities. The project vision and requirements included the need for the resulting system to serve upwards of 30,000 employees daily.

At a high level, the requirements of this project were as follows:

- Diagnose and propose resolutions to common IT issues
- Streamline service management and ticketing processes
- Automate regular operational tasks such as backups and monitoring
- Learn from past incidents to improve future performance regularly
- Comply with modern corporate IT security standards such as NIST SP 800-53

To deliver a system that exceeded these requirements and demonstrated the future of AI in ITSM solutions, the Beacon team aimed to move beyond reactive problem-solving, instead focusing on developing a tool that actively works to prevent problems and improve system health with minimal human intervention.

What is Beacon?

Beacon's capabilities are realized through the successful integration of five key technological components, each highlighting an important attribute of a self-sufficient ITSM solution:

- Natural Language Processing (NLP)
- Automated Ticketing Systems
- Routine Task Management

- Self-Healing Mechanisms
- Security

Beacon's sophisticated NLP capability is the heart of the user's experience. A conversational AI agent, powered by the popular ChatGPT-4o Large Language Model (LLM), is engineered to understand and respond to user queries in a conversational style, while simultaneously providing accurate, context-aware responses. It also employs sentiment analysis to identify and prioritize urgent issues automatically, ensuring that critical problems receive the immediate attention of the IT support team.

Beacon communicates user needs to the IT support team via integration with an ITSM system. As part of the chat assistant functionality, the assistant automatically creates support tickets within the system. Using AI to generate IT tickets reduces human error, improving the overall efficiency and accuracy of the ticketing process. The ticketing includes summarizing user issues concisely, categorizing support tickets based on user sentiment, and tracking relevant metadata such as ticket status and creation date. Furthermore, if users need to view or update the status of their case, the chat assistant can recall and manage tickets that users have created, including executing pre-defined workflows for ticket escalation and resolution. Beacon's support ticket capability ensures that issues are reported and tracked immediately by relevant stakeholders, enabling accurate issue prioritization and resource allocation while also improving the user experience.

Beacon is also designed to handle some user issues proactively, when permitted to do so. This feature was designed with the understanding that most user issues encountered are minor and can be resolved with minimal effort, such as rebooting systems, investigating network bandwidth issues, and remediating memory leaks. Corporations hesitate to adopt AI due to LLMs possibly generating unsafe code or commands, increasing security risks (Rapoport et al., 2025). Therefore, Beacon is designed to avoid executing code or shell commands automatically without explicit user consent. In addition to executing user-approved commands, Beacon provides a comprehensive framework for managing routine tasks essential for maintaining system health, such as software updates, data backups, and log file analysis.

Finally, Beacon's most innovative feature is its ability to learn from successful solutions, both positive and negative. Beacon uses an internal AI engine to recommend solutions to consumers, and this engine is configured to retrain itself nightly, allowing Beacon to learn what solutions work, improving its future recommendations. Combined with its ability to resolve many issues automatically, its ability to learn from past issues improves its diagnostic accuracy and resolution efficiency over time, minimizing maintenance needs and maximizing effectiveness.

The Process

Beacon's development was guided by the principle of building a robust, scalable, and cost-effective system using a modern and tailorable technical stack. Our design choices optimize for rapid prototyping and future deployment, with a strong emphasis on security, compliance, and performance.

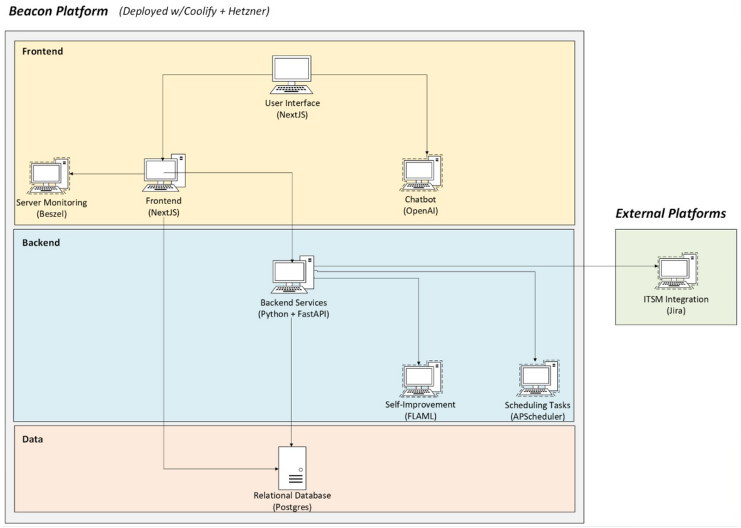

Figure 1. Beacon Architecture

For the hosting environment, Hetzner resources were paired with a self-hosted Coolify instance. Hetzner (https://www.hetzner.com/) is a low-cost Infrastructure as a Service (IaaS) platform that provides a cost-effective alternative to popular providers like AWS and Azure. By combining this with Coolify (https://coolify.io/), a Platform as a Service (PaaS) system, a wide variety of resources, including databases and web applications, can be spun up quickly at a low cost, while maintaining operational simplicity.

The frontend experience was built using NextJS, a React framework. NextJS was selected for its extensive feature set, which includes Static-Side Generation (SSG), Server-Side Rendering (SSR), and built-in routing and API capabilities. These features, along with the ease of prototyping production builds, made NextJS an ideal choice for a project requiring a highly intuitive and user-friendly interface (UI). Another feature of NextJS is NextAuth, which simplifies authentication and authorization.

The backend was developed in Python using the FastAPI framework. Python was the ideal choice due to its ability to interface with a wide variety of complex APIs, including those for task management and machine learning (ML). FastAPI was chosen for its high performance and ease of implementation, making it perfect for rapid prototyping.

To store data, Postgres was used. Its simultaneous support for relational databases, including different content types such as semi-structured data and vectors, made it ideal for AI-enabled applications that require complex data types to improve predictive capabilities. Furthermore, using an open-source platform was well-aligned with our low-cost focus.

To demo ITSM integrations, the team implemented Jira (https://www.atlassian.com/software/jira), one of the most popular IT project management tools on the market. Its comprehensive API documentation made it easy to demonstrate the prototyping and ticket-making capabilities.

For scheduling tasks, the APScheduler Python package was used. APScheduler abstracted away the intricacies of managing routine actions based on parameters specified in the Postgres database.

The self-improvement capabilities were implemented using FLAML, a lightweight AutoML (Automated Machine Learning) library. This library was configured to automatically select the most effective ML model for a given dataset. This approach safely leveraged the power of ML while being mindful of compliance concerns regarding data management (Wang et al., 2001).

Finally, to monitor the health of connected systems like servers and databases, a lightweight and security-focused tool called Beszel (https://beszel.dev/) was used. Beszel safely exposes endpoints for basic metrics like uptime and disk usage.

Reflections

Given the significant public attention AI has received over the past few years, the team made dozens of observations and reflections throughout this project. The team learned that the value of taking time to think through the requirements from all angles, rather than rushing into using the latest technological advances, was key to success. This is also the experience of senior IT architects and software developers in the industry (Hamid et al., 2022).

At the time of this writing, there have been several major innovations in the AI space that would have affected Beacon's design radically. For example, the evolution of Model Context Protocol (MCP) servers, which was in its infancy at the time of the competition, would greatly accelerate the development of a similar project today. Rather than develop a significant amount of custom functionality in one API and have the chat assistant decide which endpoint to call for a specific action, multiple pre-configured MCP servers could be implemented and called directly from the frontend. This architecture would streamline the deployment and scalability of the ITSM solution, making it more adaptable for various enterprise environments.

A noteworthy design challenge encountered by the team was the implementation of FLAML as a self-managed ML recommendation engine due to a lack of quality data. In retrospect, with more time, a more complete open-source model for both NLP and resolution recommendation could have been implemented. For example, a conventional build for a project like this might have leveraged a popular open-source LLM like Mixtral-8x7b-Instruct-v0.1 made available through the HuggingFace platform (https://huggingface.co/mistralai/Mixtral-8x7B-Instruct-v0.1), possibly even leveraging the LangChain platform for request traceability and interoperability with other tools to evaluate model performance.

The self-healing component of Beacon's architecture is designed to address common system issues autonomously. This is achieved by the system diagnosing a problem and then executing a predefined or learned solution to fix it. Although the final implementation of this engine yielded sub-optimal results, the team was confident in its approach of training an internally managed model to propose IT solutions, rather than relying on a proprietary model like OpenAI's ChatGPT, given the strict compliance requirements of the modern corporate environment. Privacy remains a concern when using AI, and large corporations have started taking into consideration the questionable privacy practices of proprietary LLMs when conducting security audits. Using AI models locally, such as exhibited by the Beacon project, simplifies the process of demonstrating proper data management and access, which facilitates compliance with industry-specific data privacy and security regulations, such as GDPR and HIPAA. Although the final implementation utilized a proprietary NLP model to demonstrate AI capabilities, the team aimed to create a path for businesses to implement proper data management in their AI applications by separating system functionality into distinct logical components. This architectural approach isolates data processing from core AI models, which allows organizations to manage sensitive information on-premises while leveraging powerful cloud-based AI services.

In the months following the competition, social media fixated on individual model performance, with LLM releases like GPT-5, GPT-OSS, and Claude 4.1 Opus dominating headlines (Ponti et al., 2025). While individual model performance has continued to improve, it has been difficult to build applications on them, as the "best" model for achieving a particular task has changed frequently. Furthermore, the tools for interacting with the latest LLMs and other ML models continuously evolve, such as the recent invention of MCP. In light of these developments, the team decided to base Beacon's architecture on business requirements, not the most recent technological fad.

Finally, it would be remiss not to mention the excitement the AI field has generated for the IT profession, particularly for IT students and entry-level professionals. Currently, IT media is both "hype-filled" and "doom-and-gloom", hindering effective AI implementation and discouraging prospective professionals as entry-level jobs vanish and companies increasingly seek experienced employees. However, the history of technological innovation indicates that the job market will eventually adjust, with examples including the inventions of the lightbulb and the automobile. To ease students' entry into the job market, universities, employers, and governments must adapt to evolving skill demands and provide necessary learning opportunities. Meanwhile, students must proactively experiment with and integrate AI, as they will be responsible for spearheading AI innovation in their future careers.

More Information

More information about the authors:

- Elliot Miller: https://www.linkedin.com/in/elliot-m-miller/
- Sam Riddle: https://www.linkedin.com/in/samuel-w-riddle/

- Andrew Hudson: https://www.linkedin.com/in/andrew--hudson/

More information about the project:

- Original Requirements: https://communities.aisnet.org/aisstudentchapters/eventshome/competitions/2025competitions/aiitrta

Technical Resources:

- Frontend GitHub Repository: https://github.com/riddlemeS4m/ais-spring-2025-frontend
- Backend GitHub Repository: https://github.com/riddlemeS4m/ais-spring-2025-task-management-api

Declaration of AI Statement

Cursor, ChatGPT, and Gemini were all utilized in the creation of this prototype.

References

Hamid, M., Zeshan, F., Ahmad, A., Malik, S., Saleem, M., Tabassum, N., & Qasim, M. (2022). Analysis of software success through structural equation modeling. *Intelligent Automation & Soft Computing*, *31*(3), 1689-1701.

Ponti, E., Ali, J., Cohn, A., Dalton, J., & Cook, M. (2025). Expert Reaction to OpenAI's announcement of ChatGPT-5. Science Media Centre. https://www.sciencemediacentre.org/expert-reaction-to-openais-announcement-of-chatgpt-5

Rapoport, G., Bicanic, S., & Talabi, M. (2025). Survey: Generative AI's Uptake is Unprecedented Despite Roadblocks. Bain & Company. https://www.bain.com/insights/survey-generative-ai-uptake-is-unprecedented-despite-roadblocks

Wang, C., Wu, Q., Weimer, M., & Zhu, E. (2021). Flaml: A fast and lightweight automl library. *Proceedings of machine learning and systems*, 3, 434-447.