Chapter 3: Methodology

This chapter provides a general summary of the research technique and methodologies used in the study. It covers data collection, data transformation, business analytics modeling, risk assessment.

3.1. Business Process

3.1.1. Comparison of Existing and Proposed Business Processes

3.1.1.1. Process Map (Activity Diagrams with Swimlanes)

3.1.1.1.1. Existing Business Process/es

3.1.1.1.2. Proposed Business Process/es

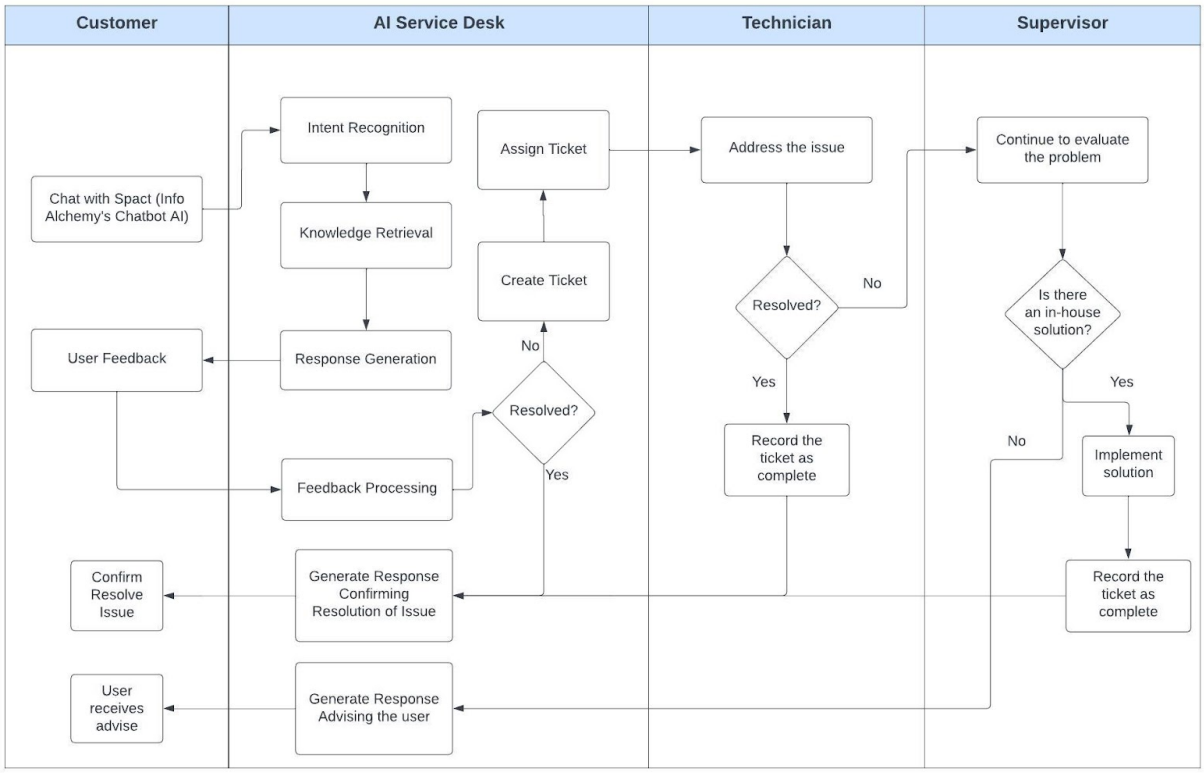


Figure 3.1.1.1.2.1 Proposed Business Process Flow: AI-driven Virtual Service Desk of Info Alchemy

Figure 3.1.1.1.2.1 shows the process map for the AI-driven virtual service desk of Info Alchemy outlines the workflow for customer requests and issue resolution. Using Spact, Info Alchemy's chatbot AI, customers initiate a chat session to report an issue or request for assistance. Spact uses intent recognition and knowledge retrieval to generate a response to the customer's request, with natural language generation. The customer provides feedback on whether the issue has been resolved or not. If not, a ticket is created and assigned to a technician, who addresses the issue and determines if it has been resolved or not. If the issue is unresolved, the ticket is escalated to the supervisor for evaluation. The supervisor determines if there is an in-house solution available and implements it, if possible. Overall, the process map provides a systematic and efficient approach to service operations management using a large language model and AI-driven virtual service desk.

3.2. Business Solutions

To streamline and enhance Info Alchemy's service operations management, we propose an AI-driven Virtual Service Desk based on the ITIL framework. ITIL provides best practices for delivering high-quality IT services. The Virtual Service Desk, powered by AI, efficiently handles user requests, incidents, and inquiries while ensuring a consistent approach. Adopting ITIL establishes structured and standardized service operations, delivering consistent services. It encompasses incident, problem, change management, and request fulfillment.

The AI-driven Virtual Service Desk complements ITIL, understanding natural language, identifying patterns, and providing accurate responses. Machine learning enables continuous improvement based on user interactions and feedback. This reduces resolution time, enhances self-service, and improves customer satisfaction. The AI-driven Virtual Service Desk aligns with automation and digital transformation trends in service operations. It reduces manual effort, minimizes errors, and accelerates issue resolution. It handles a large volume of requests simultaneously, ensuring quick and efficient responses anytime, anywhere.

This delivers a seamless and responsive service experience, enhancing customer satisfaction and loyalty. Implementing the AI-driven Virtual Service Desk based on ITIL optimizes service operations management for Info Alchemy. By leveraging AI, the organization improves service desk efficiency, effectiveness, and customer satisfaction. This ultimately contributes to Info Alchemy's success and growth in the digital era.

The AI-driven Virtual Service Desk offers a range of business solutions to enhance service operations management and drive customer satisfaction. By leveraging advanced AI technologies, the system provides a solution that

addresses key challenges faced by businesses in managing customer inquiries and support requests.

1. **Streamlined Service Operations**:

   By automating the handling of routine and repetitive customer inquiries, the Virtual Service Desk reduces the burden on human service agents, allowing them to focus on more complex and value-added tasks. This streamlines service operations, improves productivity, and enables businesses to handle a higher volume of customer interactions without compromising the quality of service. Additionally, the system provides a centralized platform for managing and tracking customer inquiries, facilitating efficient ticket management and resolution.

2. **24/7 Availability**:

   The AI-driven Virtual Service Desk operates round the clock, offering businesses the advantage of providing uninterrupted customer support. Customers can access the virtual service desk at any time, eliminating the constraints of traditional support hours. This continuous availability enhances customer satisfaction by addressing their needs promptly and reducing response times.

3.3. Business Requirements Overview

3.3.1. Functional Requirements

- The AI-driven virtual service desk should be able to understand and respond to user queries and requests in natural language.
- The system should be able to leverage a large language model to generate relevant and accurate responses to user queries.
- The virtual service desk should be able to identify and classify user requests according to their respective service operation management processes.
- The system should be able to integrate with existing service management tools, such as ticketing and monitoring systems, to automate the processing of user requests.

- The virtual service desk should be able to learn and adapt to user behavior over time to improve the accuracy and effectiveness of its responses.
- The system should be able to handle multiple user requests simultaneously and maintain context across multiple interactions to provide more personalized and relevant responses.
- The virtual service desk should be able to provide proactive support, such as system alerts and notifications, to identify and prevent potential issues before they occur.
- The system should be able to provide clear and concise reporting and analytics on user interactions, ticket resolution times, and other key performance indicators to improve service operation processes over time.

3.3.2. Non-Functional Requirements

Performance:

- The system should respond to user queries within a reasonable time frame.
- The response time for generating AI-driven suggestions or recommendations should be fast to maintain smooth conversations.

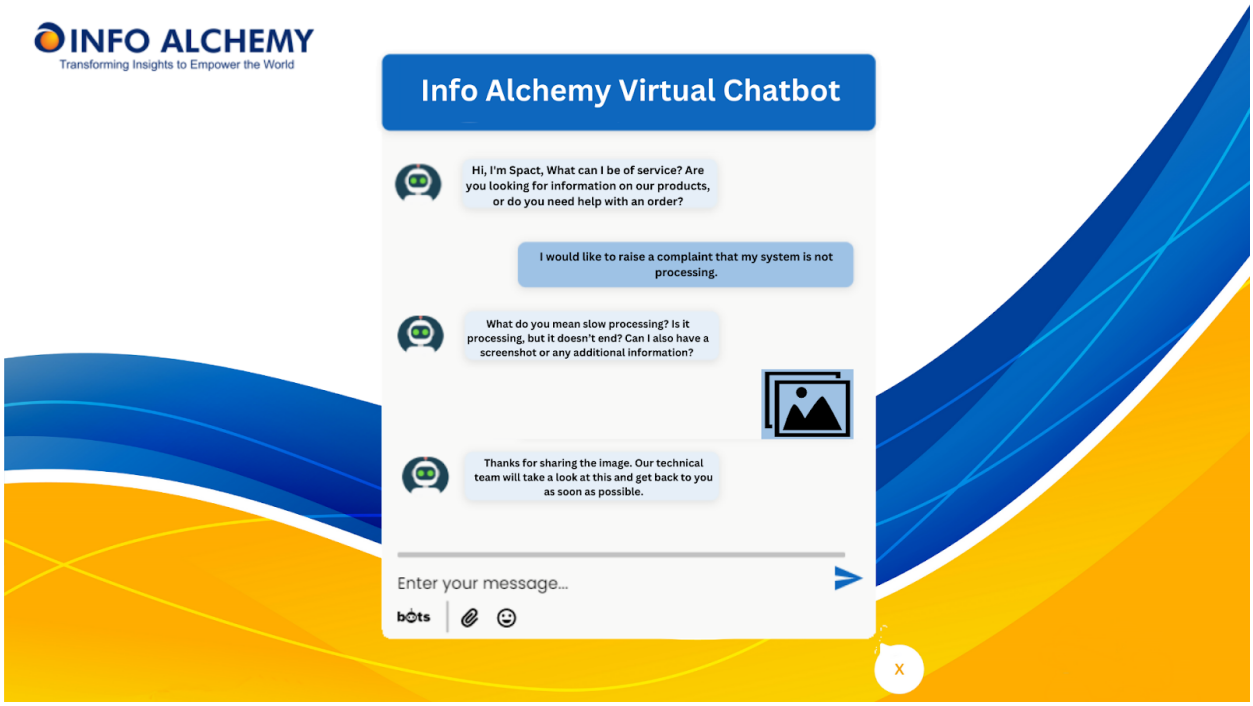
Usability:

- The user interface should be intuitive, user-friendly, and visually appealing to provide a positive user experience.

Maintainability:

- The system should be designed with modularity and extensibility in mind, making it easy to maintain and add new features or integrations.
- Code documentation and best practices should be followed to facilitate future development, debugging, and troubleshooting

3.3.3. Mock-ups


INFO ALCHEMY
Transforming Insights to Empower the World
Info Alchemy Virtual Chatbot
Hi, I'm Spact, What can I be of service? Are you looking for information on our products, or do you need help with an order?
I would like to raise a complaint that my system is not processing.
What do you mean slow processing? Is it processing, but it doesn't end? Can I also have a screenshot or any additional information?
Thanks for sharing the image. Our technical team will take a look at this and get back to you as soon as possible.
Enter your message...
bots
X

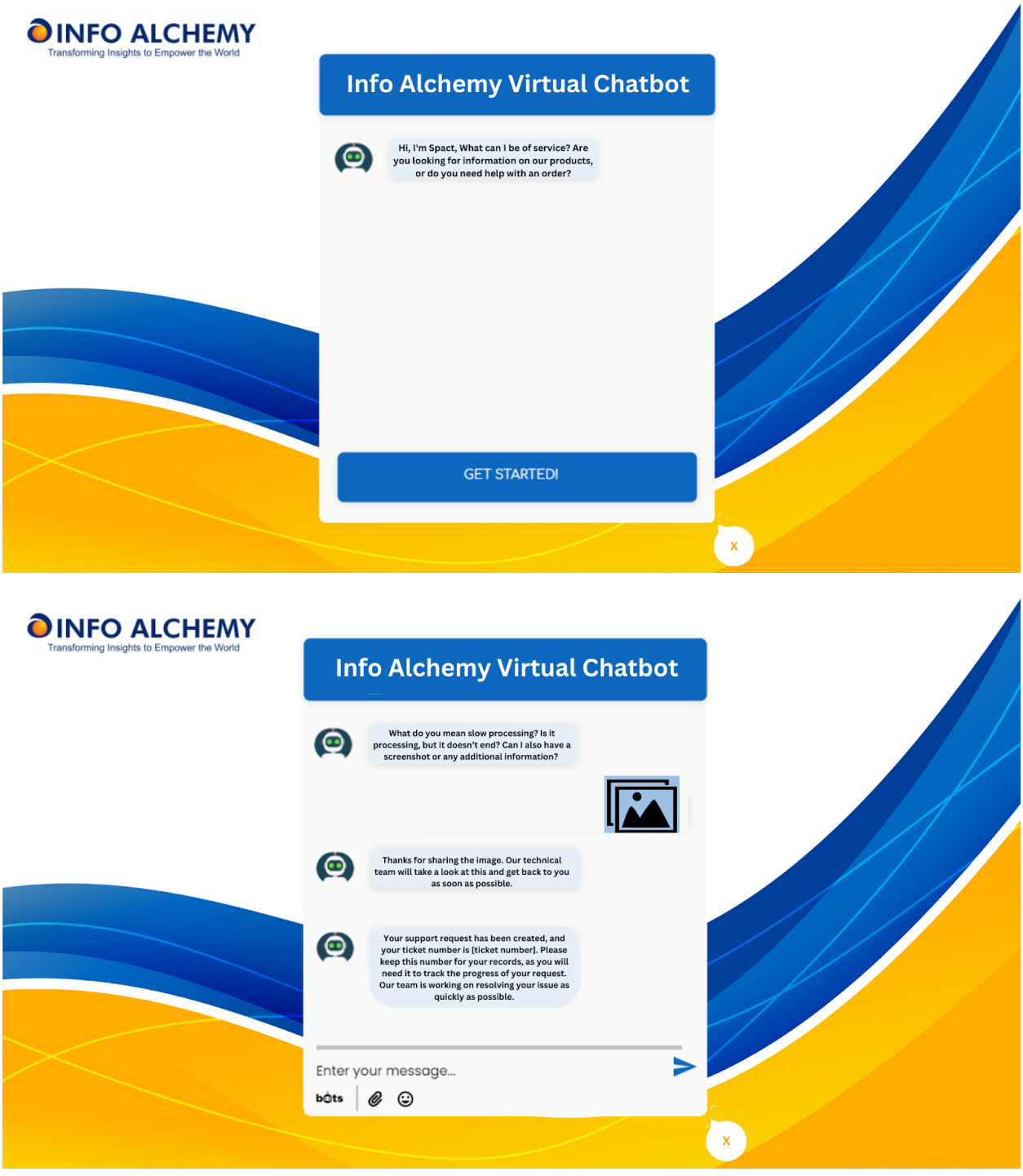


Figure 3.3.3.1 Info Alchemy Virtual Chatbot

Figure 3.3.3.1 The chatbot is the primary feature of our system, providing an intuitive and responsive way for users to interact with our ticketing system. Users can engage with

SPACT, our Smart Personal Assistant Chatbot Ticketing System, by initiating a chat and typing their queries or concerns.

The chatbot also has the ability to create tickets automatically, generating tracking numbers and opening support cases within our system for user issues. Additionally, users can attach files such as screenshots, videos, and documents to their chats to provide additional context or support.

Another unique aspect of our chatbot is its ability to display emotions and personality through a variety of emojis, adding a human touch to the support experience. This helps users feel more comfortable and engaged with the system, leading to better overall satisfaction and higher resolution rates.

Overall, the chatbot feature of our system provides an efficient and user-friendly way for users to interact with our ticketing system, reducing the time and effort required to file and resolve issues.

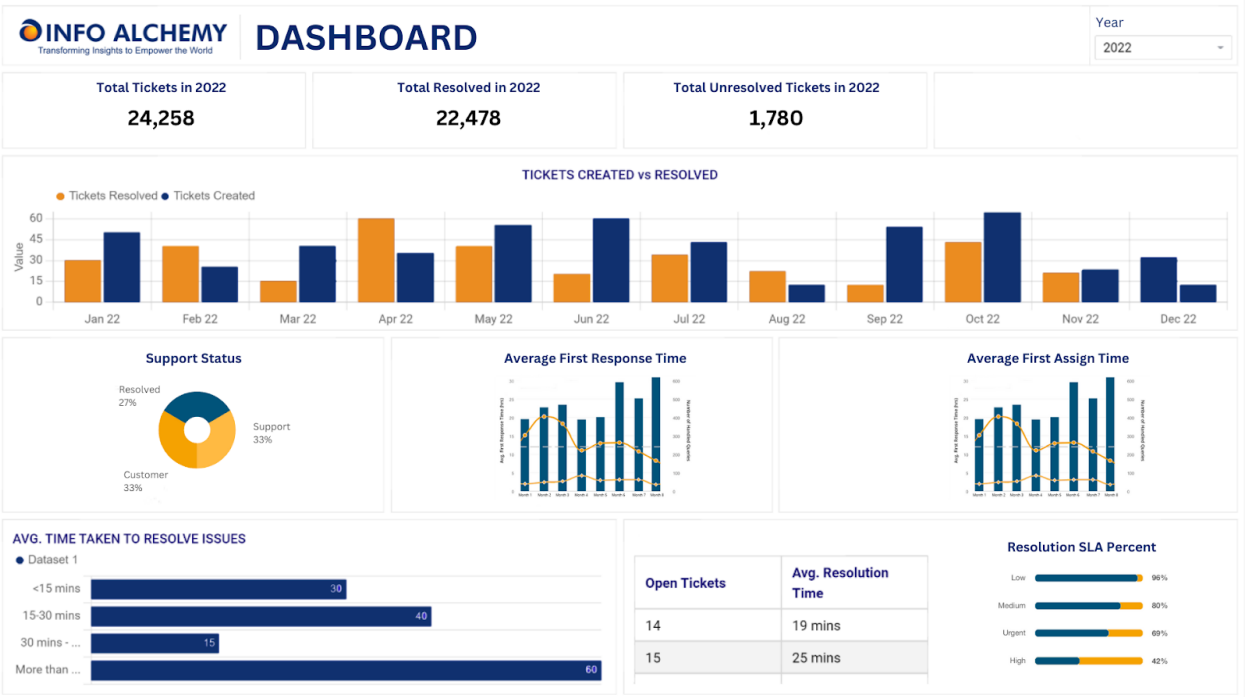


Figure 3.3.3.2. Dashboard

Figure 3.3.3.2. The system dashboard provides an overview of the ticketing system performance. It displays the total number of tickets created and resolved, as well as the number

of unresolved tickets for the current year (2022). The dashboard also allows the user to compare the number of created tickets versus the total tickets resolved.

In addition to the ticketing statistics, the dashboard shows important support metrics, including the support status and average time to provide first responses, initial assignment times, time taken to resolve issues, and the current number of open tickets. The average resolution time and SLA (Service Level Agreement) attainment percentage are also visible on the dashboard.

3.3.4. System Generated Forms and/or Reports

Forms:

- **Ticket Submission Form** - Allows users to submit new support tickets for their issues
- **Escalation Request Form** - Allows users to request escalation for their ticket if not resolved within a certain time
- **Feedback Form** - Allows users to rate their support experience and provide feedback for improvements

Reports:

- **Ticket Status Report** - Provides an overview of the current status of all open and closed support tickets
- **Service Desk Performance Report** - Analyzes the performance of the system by measuring key metrics such as average resolution time and SLA attainment percentage
- **Resolution Time Report** - Highlights the average time taken to resolve tickets based on various factors such as ticket priority, severity, and category
- **Escalation Report** - Summarizes the number of escalation requests received and the actions taken to address them
- **User Satisfaction Report** - Displays the results of user feedback responses, highlighting areas for improvement and satisfaction levels

These System Generated Forms and Reports provide valuable insights into the performance and effectiveness of the AI-driven Virtual Service Desk System. This allows service

operation management teams to identify areas for improvement and make data-driven decisions to improve service delivery.

3.4. Project Risk and Feasibility Studies (for Service Management and Business Analytics)

3.4.1. Risk Management

The researchers have identified potential risks that could impact the project's overall progress and assessed the impact of those risks. They have also developed mitigation strategies to prevent these risks from causing project failure.

LEGEND		
Rating	**Level**	**Description**
1	Low	A risk that, if it occurs, will cause a slight impact in creating the system
2	Medium	A risk that, if it occurs, will cause a minor impact in creating the system
3	High	A risk that, if it occurs, will cause a moderate impact in creating the system
4	Extremely High	A risk that, if it occurs, will cause a major impact in creating the system

Table 3.4.1.1 Risk Severity Levels

Table 3.4.1.1 shows the level of severity of the identified risks. This was used by the researchers to categorize risks based on their potential impact in this capstone project. This helps in prioritizing risks and allocating resources based on their severity level. By focusing on the most critical risks first, the project team can minimize the potential negative impact on the project's timeline, budget, and scope.

3.4.2 Risk Assessment and Mitigation

ID	IDENTIFIED RISK	LEVEL OF SEVERITY	MITIGATION ACTION
1	System failure or data corruption	High	Implement data backups and recovery plans
2	Integration issues with existing systems	Low	Conduct testing and validation during the integration phase
3	Bugs during development	Medium	Implement a bug tracking and resolution process
4	Complexity of implementing AI algorithms	High	Simplify the AI algorithms by using existing libraries and frameworks.
5	Internet connectivity issues	Low	Monitor network connectivity and address the issues promptly
6	Performance issues with the AI system	High	Monitor system performance and scalability as they arise
7	Malfunctioning system functions	High	Have a responsive technical support team for quick measure and issue resolution.
8	Privacy and ethical considerations	Extremely High	Develop a clear and comprehensive privacy policy, adhering to ethical guidelines, and conducting regular audits of privacy and ethical practices.
9	Researchers' knowledge	High	Provide continuous training and development opportunities
10	Data quality and accuracy	Extremely High	Define data quality metrics and conduct regular audits
11	Data security	Extremely High	Use data encryption to protect sensitive data while it is stored or transmitted.
12	Legal and regulatory compliance	Extremely High	Ensure compliance with all relevant laws and regulations,such as data protection and privacy laws.

Table 3.4.2.1 Risk Assessment and Mitigation

Table 3.4.1.2 shows the potential risks that could arise while conducting the capstone project. The researchers utilized a risk assessment matrix to identify and prioritize the

most critical risks that need to be addressed. This will be helpful in managing and mitigating potential risks during the capstone project's development.

3.4.3. Technological Feasibility

This section aims to assess the technological feasibility of the system by identifying the key technological challenges we expect to encounter. We expect to encounter the following challenges:

a. **Language Model Selection**: One of the primary technological challenges we face is selecting an appropriate large language model to power the Virtual Service Desk.
b. **Training Data and Knowledge Base**: Building an effective Virtual Service Desk requires a comprehensive and accurate knowledge base. Acquiring and organizing the training data is a significant challenge. We need to assess the feasibility of gathering relevant information, including existing documentation, FAQs, and other resources, and transforming it into a suitable format for training the language model.
c. **Data Storage and Privacy:** The Virtual Service Desk will generate and store user interactions, feedback, and potentially sensitive information. Ensuring secure and compliant data storage is a critical technological challenge. We need to evaluate suitable data storage solutions, including databases or cloud storage services, while adhering to Info Alchemy's data privacy and security policies.

3.4.4. Organizational/Cultural Feasibility

In this section, we assess the organizational and cultural feasibility of our capstone project. By evaluating the organization's readiness and change management strategies, we aim to ensure a smooth integration of the virtual service desk within Info Alchemy's operations.

Implementing a virtual service desk powered by AI may bring significant changes to existing processes, roles, and responsibilities within Info Alchemy. Assessing the organization's change readiness and developing a comprehensive change management plan will help mitigate potential resistance to change. This

includes communication strategies, training programs, and providing ongoing support to employees during the transition period.

3.4.4. Economic Feasibility

Cost:

Item	Estimated Cost per Unit	Quantity	Purpose
Desktop / Laptop	₱0	4	Provide the necessary computing power and versatility to perform a wide range of tasks such as document creation, software development, data analysis, and communication. They are essential tools for productivity and enable efficient work.
Visual Studio Code	₱0	1	A code editor that helps developers write, edit, and debug code. It offers features such as syntax highlighting, code completion, and debugging tools, enhancing developers' productivity and facilitating software development.
GitHub	₱0	1	A web-based platform for version control and collaboration in software development projects. It allows teams to manage source code, track changes, collaborate on code, and streamline the development process, improving productivity and code quality.
Jira Software	₱0	1	A project management tool that helps teams plan, track, and manage their work. It provides features like issue tracking, task management, and agile project management, enabling teams to stay organized, collaborate effectively, and deliver projects

			on time.
Zoom / Google Meet	₱0	1	Video conferencing tools that facilitate remote communication and collaboration. They enable teams to hold virtual meetings, share screens, and collaborate in real-time, making it easier to connect and work together, regardless of geographical locations.
Uizard / Canva	₱0	1	Design tools that simplify the creation of visual content. Uizard focuses on generating code from design prototypes, while Canva provides a user-friendly interface for designing graphics and visual assets. These tools help streamline design workflows, enhance creativity, and save time in creating visually appealing content and visual assets.

Benefits:

Item	Benefits
Desktop / Laptop	● Increased productivity and efficiency in performing various tasks. ● Versatility for different work requirements. ● Facilitates collaboration and communication among team members.
Visual Studio Code	● Increased productivity and efficiency in coding tasks. ● Time-saving features and tools that enhance the development process. ● Access to a large community and a wide range of extensions for additional functionality.
GitHub	● Improved version control and code collaboration, reducing errors and conflicts. ● Enhanced project management capabilities and visibility. ● Streamlined development process, leading to improved productivity and code quality.

Jira Software	● Improved project planning, tracking, and management. ● Enhanced collaboration and communication among team members. ● Increased efficiency and on-time delivery of projects.
Zoom / Google Meet	● Enables remote collaboration and reduces the need for in-person meetings. ● Facilitates real-time communication and screen sharing, enhancing collaboration and decision-making. ● Time and cost savings associated with reduced travel and meeting expenses.
Uizard / Canva	● It can simplify the creation of visual content by offering efficient features and workflows. ● Can save time and work more efficiently in creating high-quality designs. ● Offer a wide range of design elements, templates, and customization options, allowing users to express their unique visual styles. ● Enable collaboration among team members, facilitating real-time collaboration and feedback on design projects.

Figure 3.4.4.1 Cost Benefits Analysis

The Cost Benefits Analysis for the AI-driven Virtual Service Desk project will assess the financial implications of implementing the system. It will evaluate the costs involved in development, operations, and integration, while also identifying potential benefits and returns on investment. The analysis aims to provide a concise overview of the project's financial viability.

3.5. Development Methodology

3.6. Software Quality Assurance Plans (include plans for testing Functional Requirements, Non-Functional Requirements, User-Acceptance Test, and System Stress Test)

3.6.1. Unit Test (Functional Requirements and Non-Functional Requirements)

In this section, we will delve into the unit testing phase of our project. Unit testing is a fundamental component of the software development process that focuses on verifying the individual units or components of the system in isolation. The purpose of this phase is to ensure that each unit of the system performs as expected and meets the defined requirements. In this section, we will outline the criteria and scenarios that will be used to conduct comprehensive unit testing for our system.

3.6.1.1. Unit Test Criteria

In this section, we will outline the unit test criteria for our system. These criteria will guide the development of effective unit tests to thoroughly evaluate the functionality and behavior of individual units within our system. The following unit test criteria will be employed:

Unit Test Criteria	Description
Isolation	Each unit test will isolate the specific unit under test to ensure that it can be evaluated independently of other components or dependencies.
Repeatability	Each unit test will be designed to produce consistent results when executed multiple times. This repeatability ensures that the behavior of the units remains stable and predictable over time.
Readability and Maintainability	We will prioritize writing unit tests that are easy to read, understand, and maintain. Following best practices for test organization, using meaningful test case names, providing clear documentation, and adhering to coding standards will enhance the readability and

	maintainability of our unit
Assertions	Each unit test will include assertions that validate the expected behavior and outcomes of the units being tested. These assertions will check if the actual results match the expected results, ensuring that the functional requirements are met.

3.6.1.2. Unit Test Scenario

In this section, we will define several unit test scenarios to validate the behavior and functionality of individual units within our system. These unit test scenarios aim to cover various aspects of the system and simulate different user interactions to ensure its correctness and robustness.

Scenario	Description	Test Steps
Natural Language Understanding	Input a user query and assess if the virtual service desk accurately understands the intent and extracts relevant information.	● Provide a user query containing specific keywords or phrases related to a service request. ● Verify that the virtual service desk correctly identifies the intent and extracts the necessary details from the query.
Response Generation	Evaluate the virtual service desk's ability to generate relevant and accurate responses based on user queries.	● Input different user queries covering various topics or service-related questions. ● Verify that the virtual service desk generates appropriate responses that address the user's query accurately and provide relevant information.

Request Identification and Classification	Test the system's capability to identify and classify user requests according to service operation management processes.	● Input different user requests representing various service categories or operations. ● Validate that the virtual service desk correctly identifies and classifies each request into the corresponding service operation management process.
Question Probing	Assess the system's capability to probe and clarify user questions when necessary	● Input user queries that require further clarification or additional information. ● Evaluate if the virtual service desk recognizes the need for clarification and responds with probing questions to gather the necessary details. ● Verify that the system's probing questions are clear, contextually relevant, and help the user provide the required information for accurate assistance.
Reporting and Analytics	Verify the system's ability to generate clear and concise reporting and analytics on user interactions, ticket resolution times, and other key performance indicators.	● Monitor and record user interactions and system responses. ● Assess the accuracy and completeness of the generated reports and analytics, ensuring they provide valuable insights for improving service operation processes.

3.6.2. Stress Test

In this section, we will focus on conducting stress tests to evaluate the performance and stability of our system under extreme workload conditions. The primary objective of stress testing is to assess the system's behavior, responsiveness, and resource utilization when subjected to high levels of concurrent user activity, heavy data loads, or resource-intensive operations. By simulating scenarios that exceed the system's normal operational capacity, we can identify any performance bottlenecks, weaknesses, or potential failures that may arise in real-world usage scenarios.

3.6.2.1. Stress Test Criteria

Stress Test Criteria	Description
Load Capacity	Determine the maximum load that the system can handle without significant degradation in performance or stability. This criterion focuses on assessing the system's ability to handle high user concurrency and data processing demands.
Response Time	Measure the system's response time under stress conditions and ensure that it remains within acceptable limits. This criterion aims to evaluate the system's responsiveness and its ability to handle user requests promptly.
Throughput	Assess the system's throughput, which refers to the number of user requests processed per unit of time. This criterion helps evaluate the system's efficiency in handling a large volume of requests concurrently.
Error Handling	Evaluate how the system handles errors and exceptions under high-load conditions. This criterion examines whether the system gracefully handles errors, provides appropriate error messages, and recovers without compromising its stability.

3.6.2.2. Stress Test Scenario

Stress Test Scenario	Description
Peak User Load	Simulate a scenario where a significantly higher number of concurrent users interact with the system simultaneously. This scenario aims to stress the system's capacity and assess its ability to handle peak user loads without significant degradation in performance or response times.
Error Injection	Introduce deliberate errors, exceptions, or disruptions during stress testing to evaluate the system's error handling and recovery mechanisms. This scenario helps assess how the system handles unexpected situations, such as database failures, network interruptions, or service disruptions, without compromising its stability or data integrity.
Continuous User Interactions	Simulate a scenario where users continuously interact with the system by sending a stream of requests without significant time gaps. This scenario tests the system's ability to handle a continuous influx of user requests, maintaining consistent response times and ensuring smooth user experiences.

3.6.3. User-Acceptance Test

In this section, we will discuss the user-acceptance testing phase of our system. User-acceptance testing is a critical step in the software development lifecycle that helps ensure that the software meets the needs and expectations of its intended users. The purpose of this phase is to validate that the virtual service desk meets the requirements defined in the functional and non-functional specifications, as well as to gather feedback from end-users to improve the software before its final release. In this section, we will describe the user-acceptance test criteria, and the user-acceptance test scenarios that will be used to evaluate the virtual service desk's user-acceptance.

3.6.3.1. User-Acceptance Test Criteria

User-Acceptance Test Criteria	Description
Functional Requirements Validation	Ensure that the virtual service desk meets all the defined functional requirements outlined in the project documentation. This criterion verifies that the system performs the intended tasks and functions as expected by the end-users.
User Interface and User Experience	Evaluate the user interface design and overall user experience of the virtual service desk. This criterion focuses on assessing the system's usability, intuitiveness, and visual appeal to ensure a positive and engaging user interaction.
Response Accuracy and Relevance	Validate the accuracy and relevance of the system's responses to user queries and requests. This criterion ensures that the virtual service desk provides correct, meaningful, and contextually appropriate answers, contributing to a satisfactory user experience.
Error Handling and Recovery	Evaluate the system's error handling and recovery mechanisms. This criterion focuses on how well the virtual service desk handles errors, exceptions, or unexpected situations, providing clear error messages and recovering gracefully without data loss or system instability.
Performance and Responsiveness	Measure the performance and responsiveness of the virtual service desk under normal usage scenarios. This criterion assesses the system's speed, stability, and ability to handle user interactions in a timely manner, ensuring a smooth and efficient user experience.
Security and Data Privacy	Assess the security measures implemented within the virtual service desk to protect user data and ensure data privacy. This criterion verifies that the system adheres to security best practices and regulatory requirements, safeguarding sensitive information from unauthorized access or breaches.

3.6.3.2. User-Acceptance Test Scenario

User-Acceptance Test Scenario	Description
Query and Response Validation	Users will submit a variety of queries and assess the system's responses for accuracy and relevance. This scenario aims to validate that the virtual service desk can understand and provide satisfactory answers to user queries across different domains and topics.
User Interface Evaluation	Users will interact with the virtual service desk's user interface and evaluate its intuitiveness, ease of use, and visual appeal. This scenario focuses on assessing the user interface design, layout, navigation, and overall user experience of the system.
Error Handling and Recovery Evaluation	Users will intentionally trigger errors or exceptions within the system and evaluate how the virtual service desk handles them. This scenario aims to test the system's error handling mechanisms, ensuring that appropriate error messages are displayed, and the system recovers gracefully without compromising its stability or user experience.
Security and Data Privacy Validation	Users will review and validate the security measures implemented within the virtual service desk, such as user authentication, data encryption, and access controls. This scenario aims to assess the system's compliance with security standards and regulatory requirements, ensuring the protection of user data and maintaining data privacy.

3.7. Deployment Plans (Infrastructure/Deployment where needed)

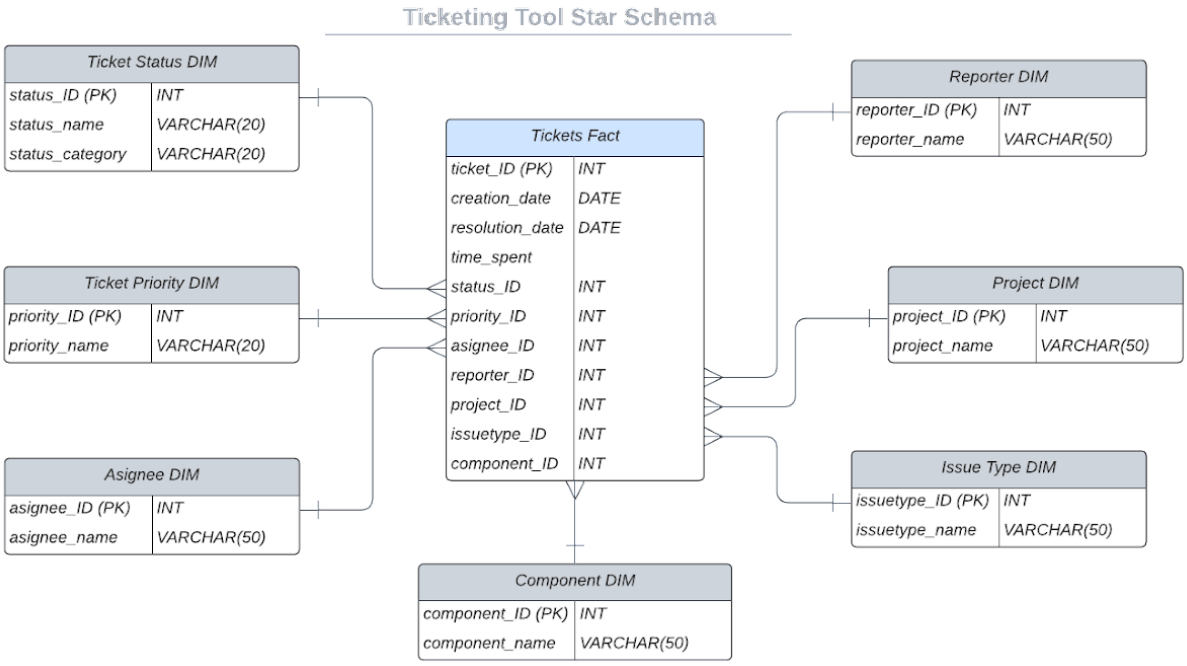
Deployment Plans	Description
Infrastructure Setup	Identify and procure the necessary hardware and software resources required for deploying the AI-driven virtual service desk solution. Set up a secure and scalable infrastructure environment, including servers, databases, networking components, and storage systems. Configure and optimize the infrastructure to ensure seamless integration and high performance of the deployed solution.
Data Preparation	Gather and preprocess the relevant data required for training and fine-tuning the large language model (LLM) powering the virtual service desk. Cleanse and organize the data to ensure accuracy, consistency, and relevance for effective training and subsequent deployment.
Model Development and Training	Develop and train the AI-driven large language model using the prepared data. Fine-tune the model specifically for service operations management processes, ensuring it understands the domain-specific terminology, context, and requirements of Info Alchemy. Validate and evaluate the model's performance through rigorous testing and validation processes.
Integration with Existing Systems	Identify the existing service operations management systems and tools used by Info Alchemy. Develop integration modules and APIs to connect the AI-driven virtual service desk solution with the existing systems. Ensure seamless data flow and interoperability between the virtual service desk and other relevant systems.
User Interface and Experience	Design and develop a user-friendly interface for accessing and interacting with the virtual service desk. Create intuitive workflows and user interfaces that enable easy navigation, input, and retrieval of information. Conduct usability testing and gather feedback from potential users to refine

	and optimize the user interface and experience.
Security and Privacy Measures	Implement robust security measures to protect sensitive data and ensure secure access to the virtual service desk. Incorporate encryption, access controls, and authentication mechanisms to safeguard data privacy and prevent unauthorized access. Conduct thorough security testing and vulnerability assessments to identify and mitigate any potential risks.
Pilot Testing and Rollout	Conduct pilot testing of the deployed solution with a subset of users to gather feedback, identify any issues, and refine the system. Address any identified issues and fine-tune the solution based on user feedback. Gradually roll out the AI-driven virtual service desk to a wider user base, monitoring its performance and gathering further feedback for continuous improvement.
Monitoring and Maintenance	Implement monitoring and logging mechanisms to track the performance and usage of the virtual service desk solution. Establish regular maintenance procedures, including updates, bug fixes, and performance optimizations. Continuously monitor user feedback, system logs, and key performance indicators to ensure the solution operates effectively and efficiently.

3.8. Data Gathering (with ETL)

In this section, To have an effective automation of service operations management processes using OpenAI, we'll collect essential data from multiple sources such as customer service tickets history, and knowledge base articles. The collected information will be transformed through an ETL process to be analyzed. Also ELT is used to make the acquired data in a format that can be analyzed. The acquired data will then be processed and cleaned during the ETL process to get rid of any errors. This process helps in ensuring the reliability of the data utilized for later analysis and modeling.

3.9. Presentation of Star Schema



3.9.1 Ticketing Tool Star Schema

Figure 3.9.1 illustrates the star schema for the ticketing system of the AI-driven virtual service desk. It consists of a central fact table, "Ticket Facts," which contains key information about each ticket, such as ticket ID, creation date, resolution date, and time spent. Surrounding the fact table are dimension tables representing different attributes of the tickets. These dimension tables include "Ticket Status," capturing the status ID, name; "Ticket Priority," storing priority ID and name; "Assignee," containing assignee ID and name; "Reporter," holding reporter ID and name; "Project," including project ID and name; "Issue Type," storing issue type ID and name; and "Component," representing component ID and name. This star schema allows for efficient analysis of ticketing data based on various dimensions, into ticket statuses, priorities, assignees, reporters, projects, issue types, and components.

3.10. Business Analytics Model and Testing

In this section, we'll use OpenAI to develop the application for the virtual service desk. On the data gathered in section 3.1, we will train the machine learning model. The virtual

service desk application will be tested to make sure it can efficiently handle client requests and automate service operations management procedures.

The machine learning model utilizes natural language processing (NLP) to understand and interpret customer requests, allowing the AI-driven virtual service desk to provide relevant responses. Testing is a crucial step to ensure that the application performs effectively in handling customer requests and automating service operations management procedures. It involves responsiveness, and ability to handle a variety of customer requests.

Furthermore, applying the pareto principle, also known as the 80/20 rule, to the business analytics model and testing process can help us optimize the performance and effectiveness of the AI-driven virtual service desk. For instance, by analyzing the data gathered in section 3.1, the researchers goal is to identify the top 20% of issues or causes responsible for 80% of the problems in service desk operations. By prioritizing these critical areas and allocating resources accordingly, the project aims to achieve significant improvements. Testing will be conducted to validate the AI-driven virtual service desk's effectiveness, measuring response times, and other relevant metrics.

3.11. Business Analytics Tools, Techniques, and Specific Applications

3.11.1 Business Analytics Tools

These tools provide functionalities for data collection. Data integration, data analysis and data visualization. They help the researchers to transform raw data into meaningful information and enable us to explore data, uncover patterns and identify trends. The researchers plan to use Microsoft Power Bi. Google Collab, Rstudio, Microsoft Excel.

3.11.2 Techniques

The business analytics techniques cover a set of methodologies and approaches which are used to extract insights from the data and uncover meaningful patterns and trends. The researchers will apply these techniques to analyze the data, make predictions, and support decision making. By employing various techniques, the researchers can gain a deeper understanding of the data. The techniques will consist of descriptive, diagnostic, predictive analytics and with natural language processing, and machine learning techniques.

3.11.3 Specific Applications

The specific applications mentioned in this section demonstrate how data analysis and insights can. It highlights various ways in which business analytics techniques can be applied to enhance the functionality and performance of the AI-driven virtual service desk. Additionally a dashboard can serve as a visual representation of the analytics and insights derived from the data. The visual representation of the specific applications mentioned above can be seen in Figure 2.

Bibliography

Davenport, T. H., & Harris, J. G. (2007). Competing in analytics. Harvard Business Review, 85(1), 98-107.

Zeithaml, V. A., Bitner, M. J., & Gremler, D. D. (2006). Services marketing: Integrating customer focus across the firm. McGraw-Hill Irwin.

Kapoor, K. K., & Singh, A. K. (2016). A study of customer satisfaction with virtual help desk in Indian banking sector. Journal of Internet Banking and Commerce, 21(1), 1-14.

Fuchs, S. (2022). Improving Support Ticket Systems Using Machine Learning: A Literature Review. https://scholarspace.manoa.hawaii.edu/items/26872956-63d6-4028-af07-a8f26c2c6262

Kaur, H., & Phutela, A. (2018). Commentary upon descriptive data analytics. 2018 2nd International Conference on Inventive Systems and Control (ICISC). https://doi.org/10.1109/icisc.2018.8398884

Beers, B. (2023). What is Regression? Definition, Calculation, and Example. Investopedia. https://www.investopedia.com/terms/r/regression.asp

Kahneman, D., & Tversky, A. (1979). Prospect theory: An analysis of decision under risk. Econometrica, 47(2), 263–291. https://doi.org/10.230

Chen, Y., Li, J., Gao, L., & Huang, X. (2019). AI-driven customer service chatbots: Design, adoption, and evaluation. Information & Management, 56(3), 1-11. https://doi.org/10.1016/j.im.2018.09.002

Gartner. (2021). Gartner top strategic predictions for 2021 and beyond. https://www.gartner.com/smarterwithgartner/gartner-top-strategic-predictions-for-2021-and-beyond/

Kannan, S., Mohan, K., & Cho, Y. (2019). AI-powered chatbots: Accessibility and usefulness in the travel industry. Journal of Travel Research, 58(7), 1193-1207. https://doi.org/10.1177/0047287518800767

van der Aalst, W. M. P., Grigori, D., Guo, Y., & Zeng, L. (2020). AI-driven service desk: Challenges and opportunities. IEEE Intelligent Systems, 35(6), 30-37. https://doi.org/10.1109/MIS.2020.3027388

Burns, B., Grant, R., Oppenheimer, D., Brewer, E., & Wilkes, J. (2016). Borg, Omega, and Kubernetes. ACM Queue, 14(1), 32-53. https://doi.org/10.1145/2898445

Gartner. (2021). Predicts 2021: Cloud computing accelerates enterprise-wide adoption of

containerization.
https://www.gartner.com/en/documents/3990237/predicts-2021-cloud-computing-accelerates-enterprise-wid

Lu, W., Liu, X., Zhang, Y., & Wang, Q. (2018). A review of container orchestration platforms: Kubernetes and beyond. Journal of Grid Computing, 16(4), 549-565. https://doi.org/10.1007/s10723-018-9444-4

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). Language models are unsupervised multi task learners. OpenAI Blog, 1(8), 9. Link: https://d4mucfpksywv.cloudfront.net/better-language-models/language-models.pdf

Kavya, K., & Narayanan, V. R. (2021). A comprehensive review of chatbot frameworks for social media analytics. Journal of Ambient Intelligence and Humanized Computing, 12(11), 12119-12136. Link: https://www.sciencedirect.com/science/article/pii/S1868519321000190

Yang, L., Wu, Y., Wei, G., & Zhao, T. (2021). Auto-response generation for customer service using deep learning based on dialogue context. Applied Soft Computing, 106, 107312. Link: https://www.sciencedirect.com/science/article/pii/S1568494620306372

Wang, W., et al. (2017). Cybersecurity analytics with Splunk: A case study in a large financial institution. Proceedings of the 2017 International Conference on Big Data Analytics and Data Science, 105-111. https://dl.acm.org/doi/10.1145/3130514.3130523

Shi, L., Chen, M., Zou, X., & Chen, H. (2020). IT support ticket classification via ensemble learning methods. Journal of Ambient Intelligence and Humanized Computing, 11, 299-308. https://doi.org/10.1007/s12652-019-01524-1

Zhang, S., Liao, S., Lin, Y., & Liu, X. (2021). Intelligent classification of customer service messages in telecommunications industry based on improved decision tree forest algorithm. Telecommunication Systems, 76, 411-425. https://doi.org/10.1007/s11235-020-00700-w

Kariyawasam, K., Fernando, T., & Jayawardena, K. (2019). Email classification using decision tree ensembles. 2019 14th IEEE International Conference on Industrial and Information Systems (ICIIS), Kandy, Sri Lanka, pp. 42-47. https://doi.org/10.1109/ICIIS47387.2019.9025775

Bhattacharya, S., Banerjee, S., & Zhang, Y. (2021). AI-driven IT operations: Challenges, recent advances, and future directions. IEEE Communications Magazine, 59(1), 56-62.

AIOps Exchange. (2021). What is AIOps? Retrieved from https://www.aiops-exchange.org/what-is-aiops/

Eaton, N. (2021). The challenges of implementing AIOps. TechTarget. Retrieved from https://searchitoperations.techtarget.com/feature/The-challenges-of-implementing-AIOps

Gartner. (2019). AIOps platforms: The future of IT operations monitoring? Retrieved from https://www.gartner.com/smarterwithgartner/aiops-platforms-the-future-of-it-operations-monitoring/

Jain, R., Ansari, A., & Garg, S. (2020). Self-healing systems: Review of current research and open challenges. Journal of Network and Computer Applications, 151, 102536.

Nikolov, V. (2021). Self-healing systems: How AI helps IT take proactive action. IBM. Retrieved from https://www.ibm.com/cloud/blog/self-healing-systems-how-ai-helps-it-take-proactive-action

Raza, M. A., Chaudhry, S. A., & Malik, K. (2019). Self-healing systems: A comprehensive survey. ACM Computing Surveys (CSUR), 52(6), 1-41.