ISSN 1816-6075 (Print), 1818-0523 (Online)
Journal of System and Management Sciences
Vol. 14 (2024) No. 8, pp. 99-114
DOI:10.33168/JSMS.2024.0807

Automating Cloud Multi-Staging Management with ITIL Continuous Improvement Practices

Stewart Sulivinio, Ahmad Nurul Fajar

Information System Management Department, BINUS Graduate Program – Master of Information Systems Management, Bina Nusantara University, Jakarta 11480, Indonesia

stewart.sulivinio@binus.ac.id; afajar@binus.edu

Abstract. This paper presents an ITIL-based approach to automate Information Technology Service Management (ITSM) process for multi-staging environments hosted in cloud service provider. The improvement is based on ITIL (Information Technology Infrastructure Library) Continual Service Improvement (CSI) model. The Continual Service Improvement model provided a structured framework for evaluating and enhancing existing ITSM for a company's staging platforms. By implementing new workflows and API Integrations between ITSM platform and Cloud Services, manual and repetitive processes were automated. The results show a reduction of resolution time by over 90% with a high accuracy. This study demonstrates the potential of using ITIL CSI to optimize ITSM for cloud environments. Further work is needed to incorporate additional cloud providers into the automated workflows. Emerging technologies such as Artificial Intelligence can be implemented to advance ITSM data preparation and automation capabilities.

Keywords: Orchestration, ITSM, ITIL, Cloud Computing, Support

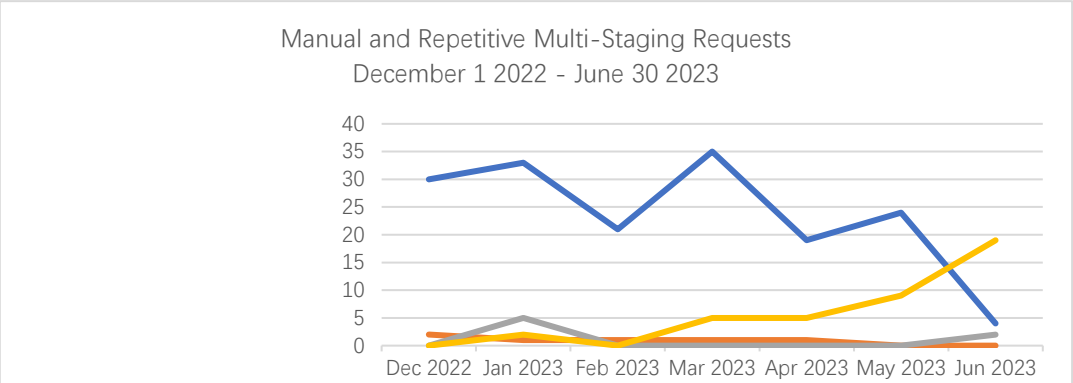
1. Introduction

PT XYZ is an e-commerce company in Indonesia that sells products from its mobile application and website. PT XYZ uses a cloud staging environment with multiple Amazon EC2 (Elastic Cloud Compute) Instances from Amazon Web Services to support the development and testing process of its application and website. Amazon Web Service is a cloud computing provider that enables users to host and use computation resources from the internet (Amazon Web Services, 2022). The usage of cloud computing is suitable for companies in Indonesia especially PT XYZ. Based on the reports by PricewaterhouseCoopers Indonesia, 80% of the total large enterprise respondents already use cloud technology, 11% are planning to use it in 2021 and 7% plan to use it in the next three years (PricewaterhouseCoopers, 2021).

In PT XYZ, the Infrastructure Team is solely responsible for provisioning and maintaining the infrastructure and instances in Amazon Web Services due to company policy that limits access to AWS Console. Currently, the Infrastructure Team in PT XYZ provides ITSM (Information Technology Service Management) by utilizing JIRA. ITSM consists of one or many IT Services. Service is a way for an organization to facilitate users in achieving their goals (Axelos, 2019). IT Service is Service provided by IT service providers which combine information technology, human resource, and processes (Irfandhi et al., 2016). Hence, ITSM (Information Technology Service Management) is a method that consists of planning, development, release, and support for IT Service Providers to provide IT Services to their users (Atlassian, 2022).

ITSM usage, specifically in Indonesian companies, shows promising growth due to its usage by large companies across multiple business sectors as stated by (Limanto et al., 2017). The Infrastructure Team in PT XYZ has provided its ITSM platform since July 2022 and is accessible from a website. The ITSM platform enables developers and product managers to request inquiries regarding PT XYZ multi-staging servers automatically. Each request will trigger a webhook, an automatic function that executes based on an event or parameters between HTTP API (Redhat, 2022). The webhook will trigger automation to PT XYZ multi-staging servers in Amazon Web Services.

While PT XYZ already provides automation to multiple requests, developers and product managers can inquire about requests manually if the current ITSM platform doesn't provide it. Developers and product managers can request manual requests by chat in a Discord Channel starting December 1, 2022. Between December 1, 2022, and June 30, 2023, there were 219 requests in the Discord Channel regarding repetitive multi-staging requests. The distribution of requests can be seen in



	Dec 2022	Jan 2023	Feb 2023	Mar 2023	Apr 2023	May 2023	Jun 2023
Turn On Staging Request Type	30	33	21	35	19	24	4
Recreate Multistaging Server	2	1	1	1	1	0	0
Duplicate Database	0	5	0	0	0	0	2
Register New Domain	0	2	0	5	5	9	19

Fig. 1: Manual and Repetitive Multi-Staging Request Distribution

The repetitive requests have the same pattern across all of the multi-staging servers. Each request also requires one member of the Infrastructure Team to leave their current work momentarily to process the request. Based on the pattern Fig. 1, manual and repetitive requests keep occurring between December 1, 2022, to June 30, 2023. The number of request frequencies in one of the categories also shows an increasing trend. Due to the high frequency of repetitive requests, an improvement to the current ITSM platform is very needed. Based on the problem, the main objective of this study is to improve the existing ITSM platform to include manual and repetitive multi-staging requests in PT XYZ. The included multi-staging requests will have automatic execution instead of manual execution. The improvement steps will follow the ITIL (Information Technology Infrastructure Library) Continual Service Improvement Model.

2. Literature Review

ITSM (Information Technology Service Management) has multiple benefits, such as a more mature business process, an increase in process metrics, better documentation and process control, an increase in organization performance and efficiency, improved competitiveness, an increase in user satisfaction, IT cost efficiency, better IS and business alignment, increasing internal communication efficiency, an increase of employee satisfaction, staff reduction, and grow of company profit (Serrano et al., 2021).

ITSM implementation has evolved over the years and has several phases (Meng et al., 2018). The first phase is the Ad-hoc manual ITSM, which is driven by the system administrator's tacit knowledge without any support tools. The second phase is Process-based Disciplined ITSM, which consists of process-driven ITSM implementations with tools based on popular ITSM frameworks, such as COBIT, ITIL, FitSM, and other frameworks. The third phase is Script-based Automated ITSM, which consists of automation techniques adaptation to existing ITSM processes by script or by another method of automation. In the future, there will be Cognitive ITSM, which will integrate Artificial Intelligence into ITSM processes.

One of the frameworks closely associated with ITSM is ITIL (Information Technology Infrastructure Library). The ITIL (Information Technology Infrastructure Library) is a framework for planning, designing, choosing, implementing, and continuous improvement for ITSM (Karkošková & Feuerlicht, 2014). Organizations can use the ITIL framework to design and implement new ITSM, as shown by (Lubis et al., 2020). In 2019, Axelos published ITIL 4 with two main components. The components are Service Value System, Four Dimension Model, and a Continual Service Improvement Model (Axelos, 2019). The Continual Service Improvement Model of ITIL 4 can be seen in Fig. 2:

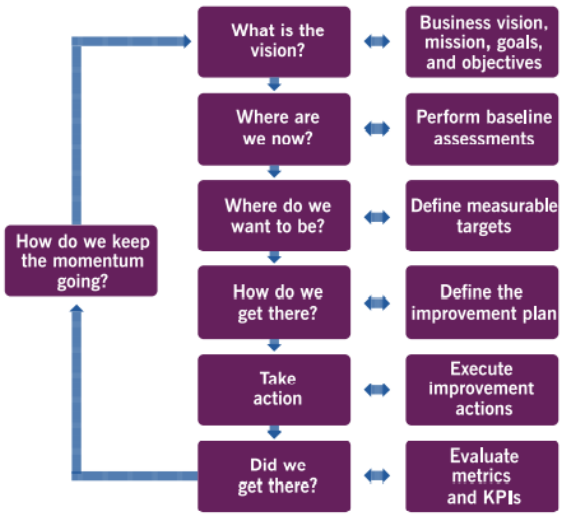


Fig. 2. ITIL Continual Service Improvement Model

The continual Service Improvement (CSI) Model can provide a framework to ensure the provided

ITSM is aligned with business objectives in an organization. Based on the implementation of the ITIL Continual Service Improvement Model, ITSM provided by an organization can be evaluated and improved as shown in (Lamichhane, 2019; Reiter & Miklosik, 2020). Implementation of the ITIL CSI model can also be evaluated as shown by (Heikkinen et al., 2020; Heikkinen & Jäntti, 2019).

In this study, the author will improve ITSM implementation to orchestrate requests in Amazon EC2 (Elastic Cloud Compute) Instances from Amazon Web Services. Cloud Computing can be defined as the utilization or usage of hardware and software from a Cloud Service Provider within the internet (Rashid & Chaturvedi, 2019). Cloud computing has multiple deployment models such as public, private, community, and hybrid (Bello et al., 2021). In this case, PT XYZ's multi-staging environment uses a public cloud provided by Amazon Web Service. Cloud Services are also comprised of 4 models Software as a Service (SaaS), Platform as a Service (PaaS), Infrastructure as a Service (IaaS), and Container as a Service (CaaS) (Alouffi et al., 2021).

ITSM has an implementation possibility in the cloud environment. Each cloud service provider has their own Service Level Agreement (SLA) to provide cloud service requests from their clients (Dhirani & Newe, 2020). Cloud resource management in an organization can also leverage ITSM from the strategy phase to the continual service improvement phase (Karkošková, 2018). Cloud service monitoring, for example, Amazon Cloudwatch, can be integrated to trigger an automatic ITSM ticket creation, as shown by (Pathak & Khandelwal, 2018).

The literature review process found multiple types of research that use the ITIL Continual Service Improvement model to create or improve existing ITSM in the organization. Several literatures also review ITSM in a cloud environment, such as resource management, SLA (service level agreement) review, and utilizing cloud alerts to create ITSM tickets. However, limited research has focused on using the ITIL Continual Service Improvement model to automate ITSM processes in a cloud environment.

3. Research Methods

The main objective of this study is to improve the existing ITSM platform to include manual and repetitive multi-staging requests in PT XYZ. The first step in this study is identifying the problems in the existing ITSM platform for multi-staging supports. Based on the data provided in Fig. 1, here are several request types that need manual and repetitive processes due to not being included in the current multi-staging ITSM. Due to the high frequency of manual and repetitive requests, the current multi-staging ITSM needs further improvement to provide manual and repetitive requests.

After problem identification, the next step is to procure all the supporting data to support the identified problems. The data is from chat scrapping results of all chats in the Infra-Support Discord Channel between December 1, 2022, to June 30, 2023. Chat scrapping process uses the “Discord Chat Exporter – Save discord chats” browser extension. All of the scrapped chat data are saved to disk with a .csv file extension.

The next step is the support data cleansing and processing. Due to a lack of understanding of chat processing tools and the diverse ways of typing from each developer and product manager, the author processed all the chat data manually. In this step, the author will categorize each chat based on the use cases and exclude other requests outside multi-staging scope.

After the chat support data processing, the next step is the improvement processes. The improvement processes consist of ITIL Continual Service Improvement model steps such as what is the vision, where are we now, where do we want to be, how do we get there, take action, did we get there, and how do we keep the momentum going.

The last step is ITSM menu release and evaluation. The improved ITSM can be accessed for all developers and product managers in PT XYZ. The author will then monitor the performance for each request types to ensure that all request types work as intended after the release. The complete research

methods can be seen in Fig. 3:

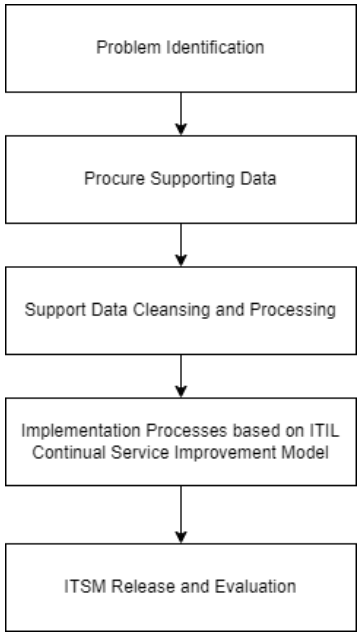


Fig. 3: Research Methodology

4. Result

4.1. What is the vision?

This step consists of defining the vision of the improvement process. The vision of this improvement is to automate the manual and repetitive multi-staging supports to the existing JIRA ITSM platform. Based on the manual and repetitive multi-staging supports shown in Fig. 1, ach request type's detailed explanation is in Table 1:

Table 1: Manual and Repetitive Multi-Staging Request Types Description

Request Type	Description
Start Request-Only Multi-Staging Server	Developers and Product Managers can request to turn on multi-staging server that listed as request-only server. Under normal circumstances, the request-only multi-staging server will not start unless requested.
Recreate Staging	Developers and Product Managers can request to recreate current multi-staging server with the latest template provided by Infrastructure Team.
Copy Database Data	Developers and Product Managers can request to replicate or copy database data between one multi-staging server to another multi-staging server.
Create New Domain	Developers or Product Managers can request to register new domain names for new services in a multi-staging server.

4.2. Where are we now?

To better understands current state of manual and repetitive multi-staging supports, the current support flow can be seen in the figure below:

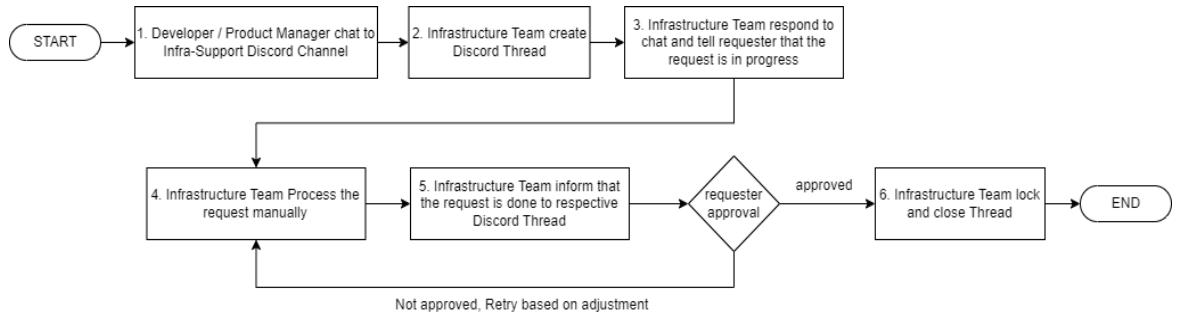


Fig. 4: Current Manual and Repetitive Multi-Staging Supports Flow

Based on the support flow shown in Fig. 4, one of the Infrastructure Team members will do the request manually based on the request types and multi-staging name. For example, in Start Request-Only Multi-Staging Server Request Type, an Infrastructure Team will log in to AWS Console and turn on the requested multi-staging instance manually.

After the understanding of the support flow has been established, all categories of manual and repetitive multi-staging request resolution time will be calculated as a baseline for the improved ITSM. The request resolution time will be calculated from the first developer or product manager chat request timestamp to the final response chat timestamp by the Infrastructure Team. Each chat timestamp is scrapped from the Infra-Support Discord Channel between December 1, 2022, to June 30, 2023. The average resolution time for each manual and repetitive request type can be found in Table 2.

Table 2: Average Resolution Time for Each Manual and Repetitive Request Type

Request Type	Average Resolution Time (hours:minutes:seconds)	Number of un-responded requests
Start Request-Only Multi-Staging Server	0:09:45	0
Recreate Staging	40:27:10	1
Copy Database Data	6:56:20	0
Create New Domain	7:03:20	3

Based on the Table 2, there are multiple instances where developers or product managers request is not responded or skipped by Infrastructure Team. This situation further supports the urgency to improve PT XYZ Multi-Staging ITSM platform.

4.3. Where do we want to be?

In this step, the success criteria of each manual and repetitive support request type will be determined. The success criteria are based on average resolution time and request accuracy. For the request accuracy, the average number of "waiting further support" for all request tickets have to be below 10%, while the overall targeted resolution time for each request type is 10% faster than current average resolution time. The overall targeted resolution time will be used as Service Level Agreement in the improved ITSM Tickets with the details in Table 3.

Table 3: Targeted Resolution Time for Each Request Types

Request Type	Targeted Resolution Time (hours:minutes:seconds)
Start Request-Only Multi-Staging Server	0:08:45
Recreate Staging	40:00:36
Copy Database Data	06:14:42
Create New Domain	06:21:00

4.4. How do we get there?

The current Multi-Staging ITSM platform has to include manual and repetitive multi-staging request types to achieve this study's main objective. In this step, the author creates a Service Improvement Plan to plan improvement actions that need to be implemented with a brief target and improvement priority. The Service Improvement Plan can be seen in Table 4.

Table 4: Service Improvement Plan

Process or Service Concerned: **Create New Domain**	**Priority: HIGH**
Description: Developers or Product Managers can request to register new domain names for new services in a multi-staging server.	
Source of The Measure: Discord chat in Infra-Support channel	
Implementation Date: September 13 2023 – September 17 2023	
Expected Release Date: September 18 2023	
Business Case 1. Expected Outcome: New ITSM Menu "Create New Multistaging Domain" in "Staging Servers" Request Category 2. Desired Result: a. Maximum Ticket execution time is 6 hours and 21 minutes after ticket approval. b. Ticket with Incorrect Field will be Transitioned into Cancelled State. For example requested domain names contains other string than alphanumeric and dashes. c. New Domains stack is registered into multi-staging domain stack such as Amazon Certificate Manager, Amazon Cloudfront, Amazon Route 53, etc.	
Process or Service Concerned: **Start Request-Only Multi-Staging**	**Priority: MEDIUM**
Description: Developers and Product Managers can request to turn on multi-staging server that listed as request-only server.	
Source of The Measure: Discord chat in Infra-Support channel	
Implementation Date: September 14 2023 – September 18 2023	
Expected Release Date: September 19 2023	
Business Case 1. Expected Outcome: New ITSM Menu "Start Request Only Multistaging" in "Staging Servers" Request Category 2. Desired Result: a. Maximum Ticket execution time is 8 minutes and 45 seconds after ticket approval. b. Ticket with Incorrect Field will be Transitioned into Cancelled State. c. Requested Request-Only Multi-Staging can be accessed after the automation turn on the multi-staging instances.	
Process or Service Concerned: **Recreate Staging**	**Priority: LOW**
Description: Developers and Product Managers can request to recreate current multi-staging server with the latest template provided by Infrastructure Team.	
Source of The Measure: Discord chat in Infra-Support channel	
Implementation Date: September 16 2023 – September 26 2023	
Expected Release Date: September 27 2023	
Business Case	

1. Expected Outcome: New ITSM Menu “Recreate Multistaging Instance” in “Staging Servers” Request Category 2. Desired Result: a. Maximum Ticket execution time in 40 hours and 36 seconds after ticket approval. b. Ticket with Incorrect Field will be Transitioned into Cancelled State. For example, Re-created Multi-Staging Instance name cannot be the same as Current Multi-Staging Instance name. c. Old Multi-Staging Instance stack will be destroyed d. New Multi-Staging Instance stack can be used after ticket completion.

Process or Service Concerned: **Copy Database Data**	**Priority: VERY LOW**
Description: Developers and Product Managers can request to replicate or copy database data between one multi-staging server to another multi-staging server.	
Source of The Measure: Discord chat in Infra-Support channel	
Implementation Date: September 20 2023 – September 27 2023	
Expected Release Date: September 28 2023	
Business Case 1. Expected Outcome: New ITSM Menu “Copy Database Data” in “Staging Servers” Request Category. 2. Desired Result: a. Maximum Ticket execution time in 6 hours 14 minutes and 42 seconds after ticket approval. b. Ticket with Incorrect Field will be Transitioned into Cancelled State. For example: non-existent table / collection names. c. The Duplicated Database in target multi-staging have the same amount of data as the Source Database in source multi-staging.	

The created Service Improvement Plan can act as a guide for the improvement implementation process in PT XYZ. Improvement implementation will follow the schedules and plans in the Service Improvement Plan. Following the priority in the service improvement plan will ensure that the most critical and most requested request type is released first. To ensure that each request type works as intended, each request type also has a general overview of the expected outcome and desired results.

4.5. Take Action

Based on the Service Improvement Plan stated in Table 4, several request types need to be included in the improved ITSM portal. One of the components of JIRA ITSM is workflow. The created JIRA ticket workflow for the manual and repetitive request types is Fig. 5:

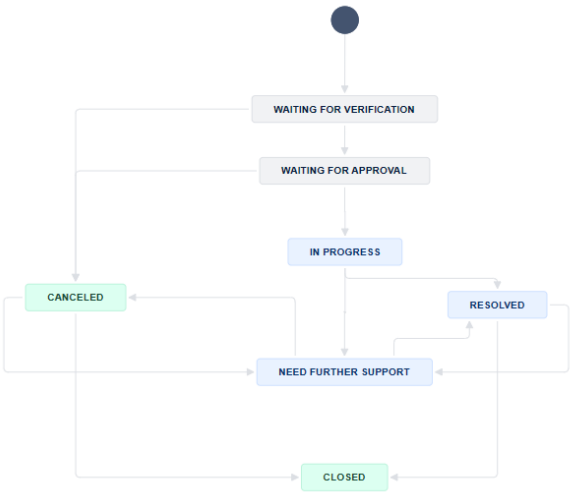


Fig. 5: JIRA Ticket Workflow for Manual and Repetitive Request Types

Based on Fig. 5, each new ticket field will be validated in the "Waiting for Verification" state. If the filled ticket fields are valid, the ticket will be transitioned into the "Waiting for Approval" state. Upon approval by the requester's direct supervisor, the ticket will be automatically set to "In Progress" State. All tickets in the "In Progress" state will send a webhook to the automation stack in PT XYZ Amazon Web Services to process the request. Upon request completion, the ticket will be set to "Waiting for Customer" state. If there are any problems with the ticket, the requester will comment on the ticket and the ticket status will be automatically set to "Need further Support". Ticket in the "Need further support" state will be retried manually by the Infrastructure Team. On the other hand, once the requester approves that all request has been satisfied, the ticket will be closed.

To implement automated support based on the created ticket, there will be a webhook upon the "In Progress" status change in each ticket. The webhook will automatically request the automation to do a task based on the created request into the affected multi-staging server. The architecture for the webhook and automation in Amazon Web Services can be found in Fig. 6.

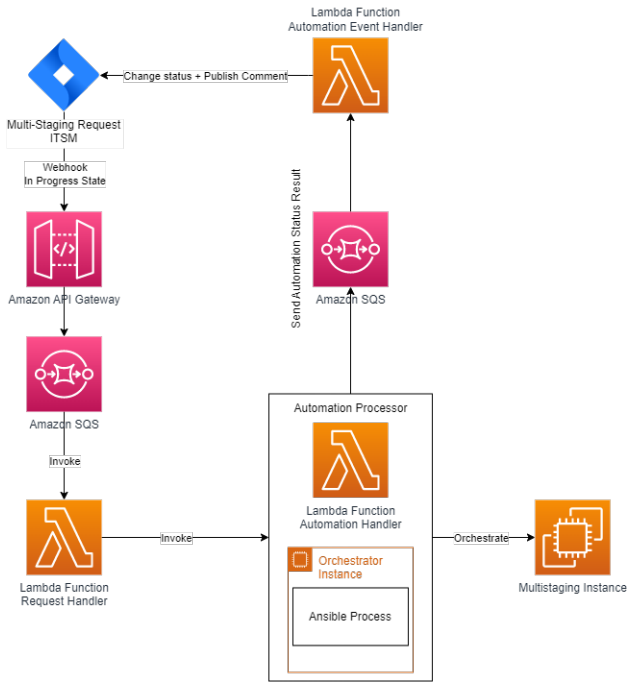


Fig. 6: ITSM Webhook Integration to Amazon Web Services

Each webhook sent from the ITSM ticket will be received by Amazon API Gateway. The Amazon

API Gateway will construct a payload to publish the request message into a respective Amazon Simple Queue Service. The Amazon Simple Queue Service will invoke a respective Amazon Lambda Function. The Amazon Lambda Function will send payload to automation processors such as Ansible, which resides in EC2 Instance or another Lambda Function. The Automation Processor will automatically process the request requested by each ticket into the respective multi-staging Instance. Upon automation completion, the automation processor will send the result to a response handler Lambda Function via another Amazon Simple Queue Service. The response handler lambda function will send a request to the respective ticket JIRA API based on the automation result (failed / success) to provide comments and change the ticket transition.

The next action is implementing each request type automation into the existing ITSM Menu in the "Staging Servers" Request Category. The created Service Improvement Plan in Table 4 will be used as a guideline for request type automation priority. A request type with a higher priority will be created first before moving on to a lower-priority request type. Hence, the improvement process will start from Create New Domain, Start Request-Only Multi-staging, Recreate Staging, to Copy Database Data. Each implementation process can be seen in Fig. 7 below:

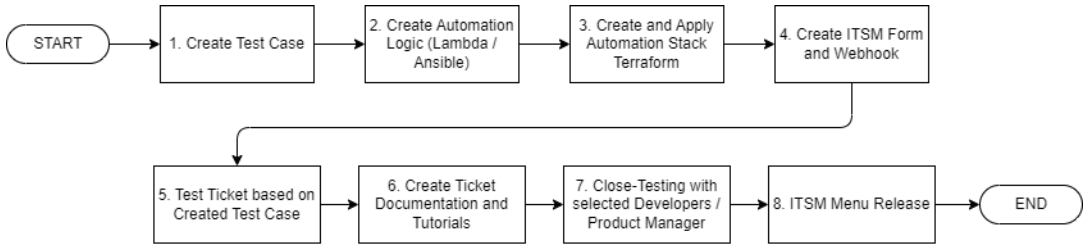


Fig. 7: Development Process for Each Request Types

Each Implementation will start with test case creation. After the test case creation, the automation logic such as Amazon Lambda Function or Ansible AWX Script will be coded. The next step is Terraform Code creation and applying all of the automation infrastructure stack listed in Fig. 6. The Infrastructure Team in PT XYZ mainly uses Terraform for Infrastructure as a Code to ensure that the created stack is standardized and prevent human errors on stack creation. After the infrastructure for automation stack creation, the next step is to create an ITSM form and webhook in JIRA ITSM. The next step is to test the ticket based on the test case, create documentation, close testing with selected Developers / Product Managers, and ITSM Menu Release.

4.6. Did We Get There

This step will evaluate the improved ITSM based on the created targets and test-cases in the created Service Improvement Plan.

Table 5: ITSM Automation Improvement Test Case Result

Request Type	Test Case	Status	Time To Resolution
Create New Domain	Test Automatic Cancel with Invalid Multi-Staging Name (RI-3373)	**PASSED**	3 seconds
	Test Automatic Cancel with Invalid Multi-Staging Name and Invalid Domain Pattern (RI-3374)	**PASSED**	27 minutes 18 seconds
	Negative Case Domain Creation Error, Transition to "Need Further Support" (RI-3387)	**PASSED**	46 minutes 37 seconds
	Positive Case Domain Creation with Valid Multi-Staging Name and Valid Domain Pattern (RI-3376)	**PASSED**	50 minutes 52 seconds

Start Request-Only Multi-staging	Negative Case Start Error, Transition to "Need Further Support" (RI-3390)	PASSED	10 minutes
	Positive Case Start Request-Only Multi-staging (RI-3396)	PASSED	47 seconds
Recreate Staging	Test Automatic Cancel with Invalid Current Multi-Staging Name (RI-3399)	PASSED	9 seconds
	Test Automatic Cancel with Invalid Re-Created Multi-Staging End Date (RI-3400)	PASSED	7 seconds
	Test Automatic Cancel with Invalid Re-Created Multi-Staging Name (RI-3402)	PASSED	7 seconds
	Positive Case Re-Create Multi-staging (RI-3477)	PASSED	10 hours 39 minutes
Copy Database Data	Test Automatic Cancel with Invalid Source Multi-Staging Name (RI-3436)	PASSED	3 seconds
	Test Automatic Cancel with Invalid Destination Multi-Staging Name (RI-3439)	PASSED	6 seconds
	Test Automatic Cancel with non-existent Database Name (RI-3443)	PASSED	7 seconds
	Positive Case Copy Database Data (RI-3458)	PASSED	1 hours 43 minutes

Below is the example of created testing tickets for the Improved ITSM Menu:

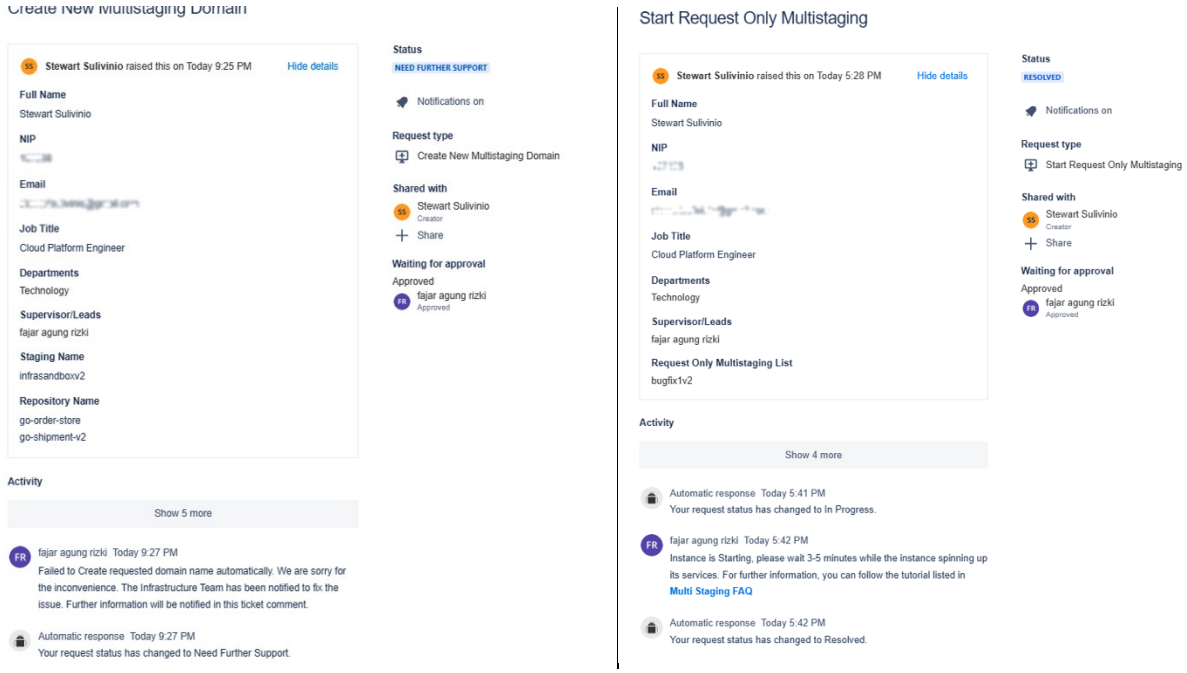


Fig. 8: Testing Ticket Example. Failed Ticket (Left), Success Ticket (Right)

The test ticket shown in Fig. 8, is based on the failed ticket for "Create New Domain" Request Type and "Start Request-Only Multi-staging" request type. Below is the example automation execution log in Amazon Lambda:

Fig. 9: Example Execution Logs in Amazon Lambda

Based on the test case result, the improved ITSM automation stack and request types passed all the test results. All test results also have a lower average time to resolution than the targeted average resolution time, which means the test result met the average time to resolution expectation. In the testing scenario, two tickets intentionally have a negative test case to test the "Need further support" ticket transition. All created tickets also met the ticket accuracy expectation due to the transition to the "Need further support" state only happening two times out of 14 test cases.

4.7. How do we Keep the Momentum Going

This step will evaluate ways to managed knowledge and lesson learned in this improvement to ensure that the improved ITSM can have a continual improvement and evaluation. To ensure that newly created request menu in "Staging Servers" request category can be understand in foreseeable future by developers or product managers, each request types have their own documentation. The documentation is embedded into the request type menu description as shown in the example below:

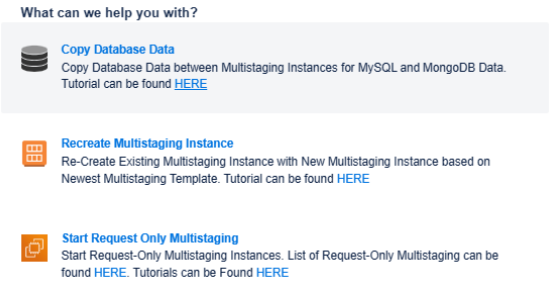


Fig. 10: Example of Ticket Tutorial Links

Besides ensuring the understanding of developers and product managers about how to fill the ITSM ticket, the author also created an internal JIRA documentation for future improvement of the created ITSM as shown below:

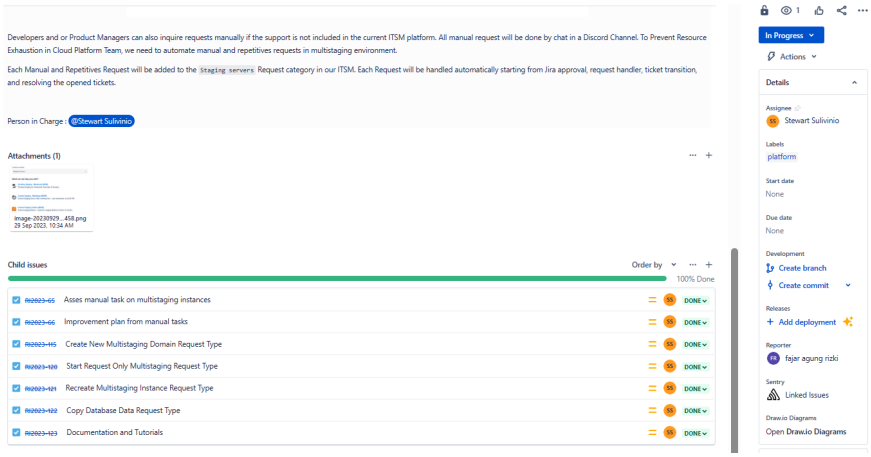


Fig. 11: Internal JIRA Documentation for ITSM

The internal documentation consists of details technical overview about how each component works, affected stacks in each request type, and how to re-deploy the created stack. The internal documentation can be used for future members of PT XYZ Infrastructure Team if an improvement should be made.

4.8. Implementation Evaluation

After the implementation has been completed, the next step is to release the improved to all developers and product managers in PT XYZ. Below is the schedule of each request type releases:

Table 6: Improved ITSM Menu Release Date

Request Type	Release Date
Create New Domain	September 13, 2023
Start Request-Only Multistaging	September 19, 2023
Recreate Staging	September 27, 2023
Copy Database Data	September 28, 2023

After the release, the author evaluates whether the improved Multi-Staging ITSM met the expectations. The first criteria are the 10% improvement of time to resolution for the improved Multi-Staging ITSM. To calculate the average resolution time for each ticket, JIRA provides a report for the Average Time to Resolution based on Request Type. The collected report is from the first implementation date, September 13, 2023, up to October 20, 2023. Below is the report from JIRA:

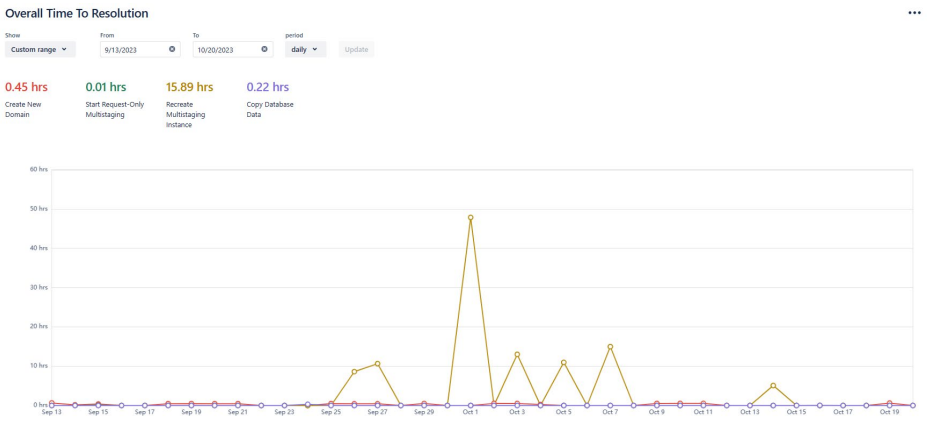


Fig. 12: Average Time to Resolution for Create New Domain Tickets

The final comparison for actual versus targeted average Time to Resolution for each request types can be seen in Table 6:

Table 7: Comparison between Average Time to Resolution for Each Request Type

Request Type	Targeted Resolution Time	Actual Average Resolution Time	Improvement Percentage
Start Request-Only Multi-Staging Server	8 minutes 45 seconds	36 seconds	1453.33%
Recreate Staging	40 hours 36 seconds	15 hours 53 minutes 24 seconds	257.79%
Copy Database Data	6 hours 14 minutes 42 seconds	13 minutes 32 seconds	2836.64%
Create New Domain	6 hours 21 minutes	27 minutes	84666.67%

Based on the comparison between the targeted and actual average time to resolution for the improved Multi-Staging ITSM in Table 6, the first criteria, which stated a 10% improvement of average time to resolution is met. The second criteria are the total occurrences of "Need Further Support" tickets transitioned is less than 10% of the total Tickets. The total number of ticket data is collected based on JIRA Reports from September 13, 2023, up to October 20, 2023, as shown below:

Table 8: Percentage of Occurrences for "Need Further Support " Ticket Transition

Total Number of Tickets	Need Further Support Transition Occurrences	Percentage of Occurrences
108	5	4.63%

Based on the calculation in Table 7, from the total of 99 tickets submitted between September 13 2023 to October 20, 2023, there are a total of 5 occurrences of the "Need further Support" transition which translates to 4.63% of all tickets. It shows that the improved Multi-Staging ITSM met all the expectations created in the "Where do we want to be" Steps.

The result of the implementation shows that the existing multi-staging ITSM can be improved by utilizing the ITIL Continual Service Improvement Framework. In terms of practical implication, the improved ITSM can increase the efficiency of submitted support requests. The Infrastructure Team in PT XYZ can focus on other important supports or tasks instead of being overwhelmed by manual and repetitive multi-staging supports, which take a considerable amount of time for each request. Other organizations can also use a similar stack used in this study to integrate their ITSM platform and Cloud Computing Services.

In terms of theoretical implications, the author agrees that the ITIL Continual Service Improvement model can act as a framework to evaluate the existing multi-staging ITSM (Lamichhane, 2019; Reiter & Miklosik, 2020). The existing multi-staging ITSM needs to be improved over time to ensure that the ITSM platform always provides support types that are relevant to PT XYZ developer's and product managers' needs. The author also agrees that this improvement can be classified as Script-based Automated ITSM (Meng et al., 2018) due to its nature of migrating manual and repetitive support tasks with automation scripts.

5. Conclusions

In conclusion, this study demonstrates the implementation of the ITIL Continual Service Improvement model to successfully automate repetitive ITSM processes in the PT XYZ Cloud multi-staging environment, by significantly improving response time and accuracy. This study provides a useful framework for companies relying on cloud platforms to enhance service agility through ITSM automation.

This study might have some inaccuracies for the average resolution time especially for "Copy Database Data". The inaccuracies might occur due to the low number of requests and the size of data being copied across multi-staging instances. The size of the data being copied will affect the time needed for dumping the data in the source database and restoring the dumped data to the destination database.

This study also has limitations regarding ways to process scrapped chat data to find manual and repetitive multi-staging supports.

While this study only focuses on integration between JIRA ITSM and Services in Amazon Web Service due to PT XYZ usage for both providers, future research can implement ITSM which orchestrates resources across multiple cloud service providers. Future research can also employ emerging technologies such as Artificial Intelligence to advance ITSM data preparation and automation capabilities.

Acknowledgements

This research is funded by Research Grant from DIKTI (Directorate General of Higher Education) Republic of Indonesia entitle "Multistaging Server Orchestration with ITSM to Support Developers in PT XYZ", with grant number: "0536/E5/PG.02.00/2023" and contract number: "179/E5/PG.02.00.PL/2023; No 1402/LL3/AL.04/2023; 149/VR.RTT/VII/2023".
The author also acknowledged all the helps from Developers, QA engineer, and Product Managers in PT XYZ for their help in close-testing stage of the improved ITSM.

References

Alouffi, B., Hasnain, M., Alharbi, A., Alosaimi, W., Alyami, H., & Ayaz, M. (2021). A Systematic Literature Review on Cloud Computing Security: Threats and Mitigation Strategies. *IEEE Access*, *9*, 57792–57807. https://doi.org/10.1109/ACCESS.2021.3073203

Amazon Web Services. (2022). *What is cloud computing?* https://aws.amazon.com/what-is-cloud-computing/?nc1=h_ls

Atlassian. (2022). *What is IT Service Management (ITSM)?* https://www.atlassian.com/itsm

Axelos. (2019). *ITIL 4 Foundation Revision Guide*.

Bello, S. A., Oyedele, L. O., Akinade, O. O., Bilal, M., Davila Delgado, J. M., Akanbi, L. A., Ajayi, A. O., & Owolabi, H. A. (2021). Cloud computing in construction industry: Use cases, benefits and challenges. In *Automation in Construction* (Vol. 122). Elsevier B.V. https://doi.org/10.1016/j.autcon.2020.103441

Dhirani, L. L., & Newe, T. (2020). Hybrid Cloud SLAs for Industry 4.0: Bridging the Gap. *Annals of Emerging Technologies in Computing*, *4*(5), 41–60. https://doi.org/10.33166/AETiC.2020.05.003

Heikkinen, S., & Jäntti, M. (2019). Studying Continual Service Improvement and Monitoring the Quality of ITSM. In M. Piattini, P. Rupino da Cunha, I. García Rodríguez de Guzmán, & R. Pérez-Castillo (Eds.), *Quality of Information and Communications Technology* (pp. 193–206). Springer International Publishing. https://doi.org/10.1007/978-3-030-29238-6_14

Heikkinen, S., Jäntti, M., & Saranto, K. (2020). Applying Continual Service Improvement Practices to Study Quality of Healthcare Information System Services: A Case Study. In M. Shepperd, F. Brito e Abreu, A. Rodrigues da Silva, & R. Pérez-Castillo (Eds.), *Quality of Information and Communications Technology* (pp. 155–168). Springer International Publishing. https://doi.org/10.1007/978-3-030-58793-2_13

Irfandhi, K., Indrawati, A., Alexandra, D., Wanandi, K., Harisky, Y., & Liawatimena, S. (2016). Impelementation of Information Technology Service Management at Data And Information System Center of XYZ University. *ComTech: Computer, Mathematics and Engineering Applications*, *7*(1), 41. https://doi.org/10.21512/comtech.v7i1.2220

Karkošková, S. (2018). Towards Cloud Computing Management model based on ITIL processes. *ACM International Conference Proceeding Series*, 1–5. https://doi.org/10.1145/3278252.3278265

Karkošková, S., & Feuerlicht, G. (2014). ITIL AS A FRAMEWORK FOR MANAGEMENT OF CLOUD SERVICES. In *IJRET: International Journal of Research in Engineering and Technology*. http://www.ijret.org

Lamichhane, N. (2019). *Implementing Continual Service Improvement In Business Enterprises: A Proposal To Improve Business Effectiveness Of Nepal*. https://www.researchgate.net/publication/334400797

Limanto, A., Khwarizma, A. F., Imelda, Rumagit, R. Y., Pietono, V. P., Halim, Y., & Liawatimena, S. (2017). A study of Information Technology Infrastructure Library (ITIL) framework implementation at the various business field in Indonesia. *2017 5th International Conference on Cyber and IT Service Management (CITSM)*, 1–4. https://doi.org/10.1109/CITSM.2017.8089244

Lubis, M., Cherthio Annisyah, R., & Lyvia Winiyanti, L. (2020). ITSM Analysis using ITIL V3 in Service Operation in PT.Inovasi Tjaraka Buana. *IOP Conference Series: Materials Science and Engineering*, *847*(1), 012077. https://doi.org/10.1088/1757-899X/847/1/012077

Meng, F. J., Xu, J., Zhang, X., Yang, L., Chen, P., Wang, Y., Liu, X., Ayachitula, N., Murthy, K., Shwartz, L., Galambos, G., Su, Z., & Zheng, J. (2018). Opportunities and Challenges Towards Cognitive IT Service Management in Real World. *Proceedings - 12th IEEE International Symposium on Service-Oriented System Engineering, SOSE 2018 and 9th International Workshop on Joint Cloud Computing, JCC 2018*, 164–173. https://doi.org/10.1109/SOSE.2018.00028

Pathak, R. C., & Khandelwal, P. (2018). A model for hybrid cloud integration: With a case study for it service management (ITSM). *Proceedings - 2017 IEEE International Conference on Cloud Computing in Emerging Markets, CCEM 2017*, *2018-January*, 113–118. https://doi.org/10.1109/CCEM.2017.26

PricewaterhouseCoopers. (2021). *The Impact of Cloud Computing on the Indonesian Economy*. https://www.pwc.com/id/en/publications/digital/the-impact-of-cloud-computing-on-the-indonesian-economy.pdf

Rashid, A., & Chaturvedi, A. (2019). Cloud Computing Characteristics and Services A Brief Review. *International Journal of Computer Sciences and Engineering*, *7*(2), 421–426. https://doi.org/10.26438/ijcse/v7i2.421426

Redhat. (2022, June 1). *What is a webhook ?* https://www.redhat.com/en/topics/automation/what-is-a-webhook

Reiter, M., & Miklosik, A. (2020). *DIGITAL TRANSFORMATION OF ORGANISATIONS IN THE CONTEXT OF ITIL® 4*. https://www.microsoft.com/sk-sk/microsoft-365/microsoft-teams/free

Serrano, J., Faustino, J., Adriano, D., Pereira, R., & da Silva, M. M. (2021). An it service management literature review: Challenges, benefits, opportunities and implementation practices. In *Information (Switzerland)* (Vol. 12, Issue 3). MDPI AG. https://doi.org/10.3390/info12030111