Implementasi Dan *Profiling* Integrasi *Opensource* ITSM iTop Dengan *Opensource* NMS Nagios

1st Azizan Hawari
Fakultas Rekayasa Industri
Universitas Telkom
Bandung, Indonesia
azizanhawari@student.telkomuniversity.ac.id

2nd Adityas Widjajarto
Fakultas Rekayasa Industri
Universitas Telkom
Bandung, Indonesia
adtwjrt@telkomuniversity.ac.id

3rd Muhammad Fathinuddin
Fakultas Rekayasa Industri
Universitas Telkom
Bandung, Indonesia
muhammadfathinuddin@telkomuniversity.ac.id

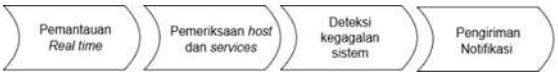
***Abstrak*— Dalam era kemajuan teknologi informasi yang pesat, ITSM menjadi sangat penting bagi organisasi untuk memastikan sumber daya TI digunakan secara optimal untuk mencapai tujuan bisnis. iTop, yang berbasis pada kerangka kerja ITIL v4, menyediakan berbagai modul untuk manajemen layanan TI, tetapi memiliki keterbatasan dalam pemantauan infrastruktur jaringan. Oleh karena itu, penelitian ini mengintegrasikan iTop dengan Nagios untuk mengatasi kelemahan tersebut.**

Penelitian ini menggunakan metodologi Network Development Life Cycle (NDLC) yang terdiri dari beberapa tahap, mulai dari analisis kebutuhan, rancangan sistem, hingga simulation prototyping. Pada penelitian ini dilakukan 3 pengujian dengan menjalankan 3 rancangan skenario percobaan yaitu pengujian pada sistem monitoring Nagios, sistem pembuatan tiketing pada iTop dan sistem integrasi layanan monitoring dan tiketing Nagios dengan iTop. Agar lebih rinci dalam memndapatkan hasil pengujian, dirancang juga 3 Data Flow Diagaram untuk mengetahui secara rinci alur informasi yang terjadi dari ketiga aspek pengujian. Pada 19 pengujian yang dilakukan dengan masing-masing capaian paramaeter keberhasilan, menunjukan status BERHASIL DIUJI dan Hasil penelitian menunjukkan bahwa integrasi antara iTop dan Nagios dapat dilakukan, meskipun pada pembuatan tiket kendala yang diperoleh dari hasil pemantauan jaringan Nagios tidak terjadi secara otomatis dan real time. Namun, seacara konektivitas iTop dan Nagios berhasil diintegrasikan dengan eksekusi script create-ticket-manual-incident.php yang membuat tiket secara manual pada layanan iTop.

***Kata kunci*—ITSM, iTop, Nagios, integrasi, pemantauan jaringan, Ticket**

I. PENDAHULUAN

Dalam era kemajuan teknologi informasi, pemanfaatan sumber daya TI seringkali menjadi tantangan dalam mencapai tujuan bisnis. *IT Service Management* (ITSM) berperan penting untuk memastikan organisasi memaksimalkan penggunaan TI secara efektif. *Framework IT Infrastructure Library (ITIL)* hadir sebagai pedoman terstruktur yang menawarkan praktik terbaik untuk meningkatkan manajemen layanan TI, sekaligus membantu dalam pengukuran dan peningkatan berkelanjutan kualitas layanan TI [1].



GAMBAR 1
Alur Pemantauan Jaringan

ITIL V4 adalah versi terbaru dari kerangka kerja ITSM, dirancang untuk memberikan panduan lengkap dan fleksibel dalam mengelola layanan TI di lingkungan yang dinamis, membantu organisasi dari demand hingga penciptaan nilai, serta mengembangkan kompetensi praktisi ITSM [2]. Dalam pengelolaan layanan, biaya juga menjadi pertimbangan penting dalam pengambilan keputusan. Meskipun perangkat ITSM komersial menawarkan pengelolaan layanan TI yang optimal, biayanya sering kali tinggi. Sebagai alternatif, ITSM *open source* menyediakan fitur yang hampir setara dengan biaya lebih rendah [3].

Information Technology Operational Portal (iTop) adalah perangkat ITSM open source berbasis *web* yang mengikuti praktik terbaik ITIL dan bersertifikasi ISO 20000, menjadikannya pilihan menarik bagi organisasi. iTop menyediakan berbagai modul, seperti CMDB, *Helpdesk*, dan *Incident Management, Problem management, Change Management Services Management, data administration* dan *Admin Tools* dengan keunggulan dalam pengelolaan layanan multi-klien serta biaya yang minimal [4] Namun, kelemahan iTop terletak pada keterbatasannya dalam pemantauan infrastruktur jaringan, sehingga deteksi masalah jaringan bisa lebih lambat dan visibilitas kinerja jaringan kurang mendetail [5]. Untuk mengatasi ini, perangkat pemantauan jaringan seperti Nagios diperlukan sebagai pendukung untuk mengoptimalkan kinerja iTop pada *level* infrastruktur jaringan.

Nagios adalah *open source* NMS yang memantau infrastruktur jaringan, termasuk *bandwidth, status up/down, memory*, dan CPU. Fitur utama meliputi pemantauan *network services* (SMTP, HTTP, PING), *host resources (processor, disk usage)*, serta kustomisasi melalui plugins dan notifikasi *log file* [6].

Integrasi Nagios dan iTop mengoptimalkan pengelolaan layanan TI dengan memantau performa jaringan secara realtime dan otomatis mengirim tiket saat terjadi masalah. Proses ini mengikuti pendekatan *Network Development Life Cycle* (NDLC) melalui enam tahap: analisis, desain, *prototyping,* implementasi, monitoring, dan manajemen, untuk meningkatkan efisiensi dan respons pemeliharaan.

II. KAJIAN TEORI

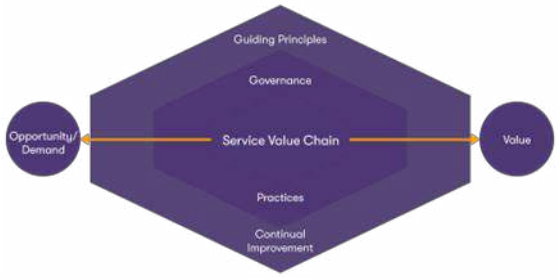
A. Sistem Pemantauan Jaringan

Pemantauan jaringan memastikan kinerja optimal dan mencegah masalah seperti gangguan koneksi atau kecepatan lambat. Tujuannya adalah mendeteksi dan memperbaiki masalah, meningkatkan keandalan, efisiensi, dan respons terhadap gangguan yang dapat memengaruhi operasional. [7].

NMS memantau infrastruktur jaringan secara *real-time*, termasuk status perangkat, layanan, dan kinerja. Ketika terjadi masalah pada *host* atau *services*, NMS mendeteksinya dan mengirimkan notifikasi kepada administrator atau kontak yang ditentukan.

B. *Information Technology Services Management*

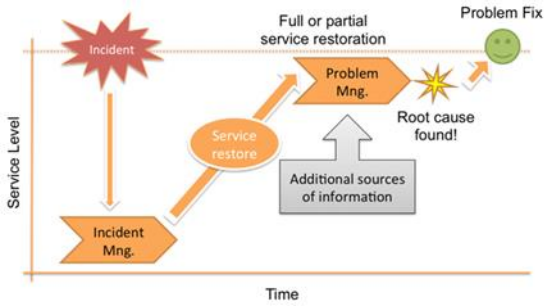
ITSM merupakan pendekatan yang digunakan organisasi dalam mengelola layanan TI di dalam sebuah organisasi. ITSM berfokus pada menyediakan dan mendukung layanan TI yang selaras dengan kebutuhan bisnis, dengan tujuan untuk meningkatkan kualitas layanan dan efisiensi operasional. Salah satu kerangka kerja tata kelola TI yang banyak dikenal dan digunakan dalam menerapkan ITSM adalah ITIL [8]. ITIL v4 adalah kerangka kerja terkini yang menyediakan panduan untuk mengelola layanan TI, termasuk manajemen insiden. Dirancang untuk membantu organisasi mengelola layanan TI secara efektif, fleksibel, dan terintegrasi, ITIL v4 memungkinkan organisasi menghadapi tantangan modern dalam dunia TI. [9].



GAMBAR 2
Services Value System ITIL V4

Gambar 2 menggambarkan Services Value System (SVS) ITIL V4, yang mencakup Services Value Chain (Plan, Engage, Design and Transition, Obtain/Build, Deliver and Support, Improve), praktik-praktik seperti Architecture Management dan Continual Improvement, prinsip panduan, dan Governance. SVS adalah kerangka kerja fleksibel yang mengelola layanan TI secara holistik, dari permintaan hingga penciptaan nilai, memastikan konsistensi dan kesesuaian dengan strategi organisasi. [10].

C. Nagios



GAMBAR 3
Alur kendala atau insiden layanan

Nagios adalah solusi monitoring open source yang dirancang untuk memantau jaringan komputer dan infrastruktur sistem. Dengan kemampuan mendeteksi dan merespons masalah pada server, perangkat, aplikasi, dan layanan, Nagios memantau *host* dan layanan serta mengirimkan notifikasi melalui email atau SMS saat terjadi masalah. Fitur lainnya mencakup pembuatan laporan, visualisasi data, dan integrasi dengan berbagai *plugin* untuk memperluas fungsionalitas. Secara arsitektur, Nagios menggunakan skrip untuk memeriksa status yang dimonitor dan dapat diintegrasikan dengan *web server* seperti Apache untuk menyediakan antarmuka *web* bagi pengguna [11].

itur-fitur Nagios, seperti monitoring, alert, dan reporting, menjadikannya pilihan unggul dalam Network Monitoring System. Kemampuan untuk memantau sumber daya seperti CPU load, memori, status up/down, uptime, dan bandwidth memungkinkan pemantauan yang fleksibel. Notifikasi yang diberikan oleh Nagios memungkinkan pengguna untuk segera mengetahui masalah jaringan pada host dan layanan dari mana saja [12].

D. *Information Technology Operational Portal*

iTop adalah aplikasi ITSM berbasis web yang bersifat *open source* dan berfungsi sebagai CMDB untuk membantu perusahaan dalam melaporkan kualitas layanan kepada klien serta menghubungkan operasional TI. Fitur-fitur iTop mencakup manajemen insiden dan permintaan pengguna, SLA transparan, manajemen masalah, CMDB yang dapat disesuaikan, dan chat *messenger*. Modul-modul dalam iTop meliputi CMDB untuk pengelolaan aset TI, *helpdesk* untuk pengaturan tiket layanan, *incident management* untuk deteksi masalah, *problem management* untuk menangani masalah berulang, *change management* untuk pengelolaan perubahan, *services management* untuk pengelolaan layanan pelanggan, *data administration* untuk impor dan ekspor data, serta *admin tools* untuk pengaturan sistem iTop [14].

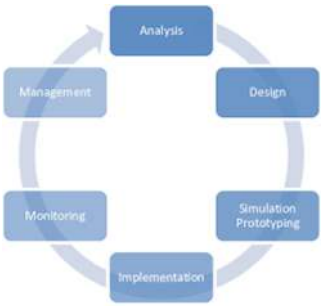
Di iTop, deteksi kendala dan tiket dikelola melalui dua komponen: *Incident Management* dan *Problem Management*. *Incident Management* bertujuan untuk mengelola dan memulihkan layanan yang mengalami gangguan, dengan fokus pada deteksi, identifikasi, dan pemulihan layanan untuk meminimalkan dampak pada bisnis. Tujuan utamanya adalah memulihkan layanan dengan cepat dan mengurangi waktu *down*.Sementara itu, *Problem Management* berfokus pada identifikasi dan pengelolaan masalah berulang atau kompleks yang mempengaruhi layanan. Proses ini melibatkan deteksi dan pemulihan masalah untuk meminimalkan dampak negatif

pada operasional layanan. Dengan demikian, *Problem Management* memastikan kualitas layanan yang stabil dan mendukung tujuan bisnis organisasi [15].

Gambar 3 menggambarkan alur Problem Management Reaktif, yang berfokus pada mengatasi insiden yang telah terjadi dengan mencari akar penyebab untuk mencegah terulangnya masalah. Ketika insiden muncul, pengelola TI melakukan perbaikan tanpa mempertimbangkan akar masalah yang lebih dalam. Namun, jika insiden berulang, pengelola akan melakukan analisis mendalam untuk menemukan penyebab dan mencegah terulangnya kendala. Fokus utama *Problem Management* adalah meningkatkan kepuasan dan perbaikan layanan jangka panjang. Dasar dari proses ini meliputi pengelolaan informasi dan hasil dari Permintaan Perubahan, pembaruan *Known Error Database* (KEDB) dan solusi, serta pencatatan masalah untuk menghasilkan informasi manajemen. [16].

E. *Network Development Life Cycle*

NDLC adalah siklus hidup pengembangan sistem jaringan komputer yang komprehensif, dengan integritas tinggi melalui tahapan yang spesifik untuk mencapai keluaran yang akurat dan produktif. Setiap tahapan memiliki karakteristik dan aktivitas tertentu, dan NDLC hanya dapat dilaksanakan setelah proses sebelumnya, seperti perencanaan strategis dan analisis data, selesai. Keberhasilan NDLC dalam mendistribusikan informasi secara tepat akan menentukan pencapaian tujuan strategi bisnis perusahaan. Model NDLC membantu perusahaan membangun arsitektur teknologi informasi yang efisien, meningkatkan produktivitas, mengukur spesifikasi informasi, menstandarkan dokumen, mengurangi keterlambatan, meminimalkan risiko, dan meningkatkan profit. [17]. Pada NDLC terdapat 6 tahap yang dilalui yaitu *Analysis, Design, Simulation Prototyping, Implementation, Monitoring* dan *Management* bisa dilihat pada gambar 4.



GAMBAR 4
Tahapan etode NDLC

F. *Personal Homepage : Hypertext Preprocessor*

Hypertext Preprocessor (PHP) adalah bahasa pemrograman *web* yang digunakan untuk membangun situs *web* dinamis. PHP memungkinkan pengembang membuat halaman yang berinteraksi dengan pengguna secara *real-time*, seperti menampilkan data yang berubah atau memproses informasi dari formulir. Keistimewaan PHP meliputi kecepatan eksekusi, sifat *open source*, kemudahan pembelajaran, dan dukungan teknis yang baik. Dengan lebih dari 25 juta situs web yang menggunakannya, PHP menjadi salah satu bahasa pemrograman *web* paling populer, sering dipadukan dengan teknologi lain seperti MySQL untuk membangun aplikasi web yang kompleks dan interaktif. [18].

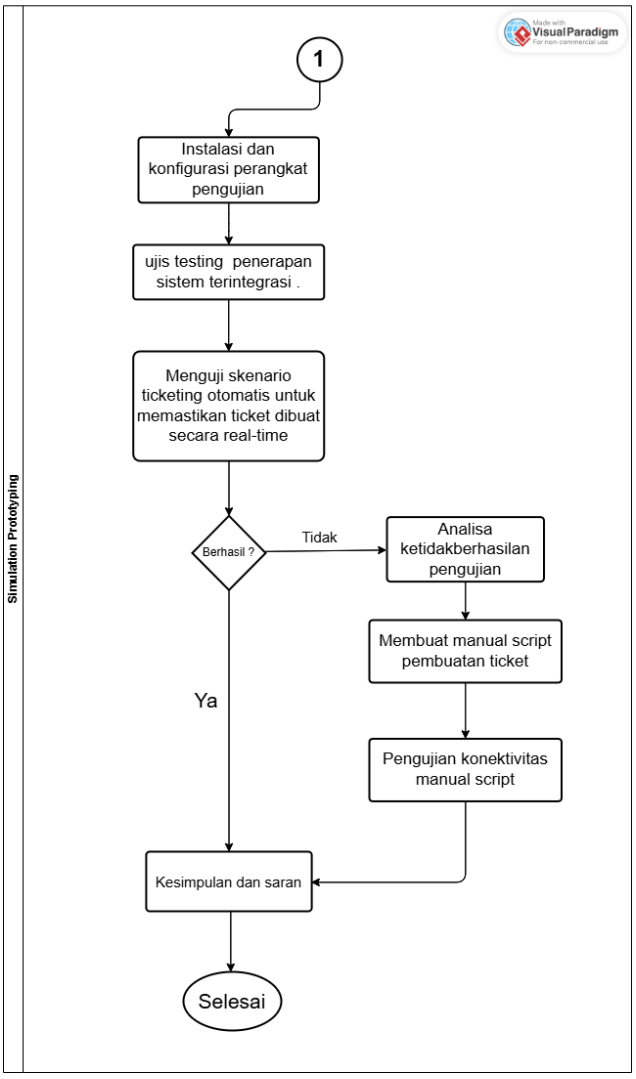
Menurut sejarah, PHP dahulunya merupakan singkatan dari Personal *Home Page Tools* saat pertama kali dikembangkan oleh Rasmus Lerdoft. Namun, setelah dikembangkan lebih lanjut oleh Zeev Suraski dan Andi Gutmans dengan penambahan fitur, singkatan PHP diubah menjadi *Hypertext Preprocessor*. PHP dikenal sebagai bahasa scripting yang sering digunakan di halaman *web*, di mana kodenya langsung dimasukkan ke dalam HTML.Salah satu keunggulan PHP adalah kemampuannya untuk berintegrasi dengan berbagai *database* dengan mudah. PHP mendukung berbagai *database* secara langsung tanpa memerlukan konektor tambahan seperti yang dibutuhkan oleh bahasa pemrograman lain, seperti Java. Fleksibilitas PHP dalam berhubungan dengan berbagai *database*, terutama MySQL, membuatnya sangat populer. Untuk menghubungkan PHP dengan *database*, cukup mengetahui nama *database*, lokasi, serta username dan password yang diperlukan untuk mengaksesnya [19]. Selain itu, terdapat keunggulan lain dalam penggunaan PHP ; Akses Cepat, PHP memiliki tingkat akses yang lebih cepat karena ditulis di tengah kode HTML, sehingga waktu respon programnya lebih cepat; Murah dan Gratis, PHP tidak memerlukan biaya untuk digunakan, sehingga sangat murah dan gratis; Mudah Dipakai, Fitur dan fungsinya lengkap, cocok digunakan untuk membuat halaman *web* dinamis; Kompatibilitas Sistem Operasi, PHP dapat dijalankan di berbagai sistem operasi, seperti Windows, Linux, macOS, dan varian lainnya; Integrasi dengan *Database*, PHP sangat fleksibel berhubungan dengan berbagai database. Salah satu database yang paling banyak digunakan bersama PHP adalah MySQL. Penghubungan dengan *database* hanya memerlukan mengetahui nama *database*, lokasinya, serta *username* dan *password* untuk menuju ke database tersebut; Keamanan, PHP memiliki tingkat keamanan yang tinggi, membuatnya populer digunakan dalam pengembangan *web*; Scripting yang Lazim Digunakan, Kode PHP langsung dimasukkan ke dalam kode HTML, sehingga dapat membuat halaman *web* yang dinamis dengan mudah.

G. Integrasi

Integrasi adalah proses pembauran sehingga menjadi kesatuan yang utuh. Dalam konteks teknologi informasi, integrasi merujuk pada adanya hubungan antar subsistem yang saling berkaitan sehingga data dari satu sistem secara rutin dapat melintas menuju, atau diambil oleh, satu atau lebih sistem yang lainnya. Ini berarti bahwa integrasi memungkinkan data dan proses untuk berinteraksi secara harmonis dan efektif antara berbagai sistem yang berbeda. Sistem integrasi adalah sebuah rangkaian proses untuk menghubungkan beberapa sistem-sistem komputerisasi dan software aplikasi baik secara fisik maupun secara fungsional. Sistem integrasi akan menggabungkan komponen sub sistem ke dalam satu sistem dan menjamin fungsi - fungsi dari sub sistem tersebut sebagai satu kesatuan sistem. Dengan demikian, sistem integrasi memungkinkan pengguna untuk mengakses dan mengelola data secara lebih efisien dan efektif melalui platform yang terintegrasi [20].

III. METODE

Dalam penelitian ini, berikut merupakan sistematika penyelesain masalah yang dilakukan:



A. Tahap Analisis

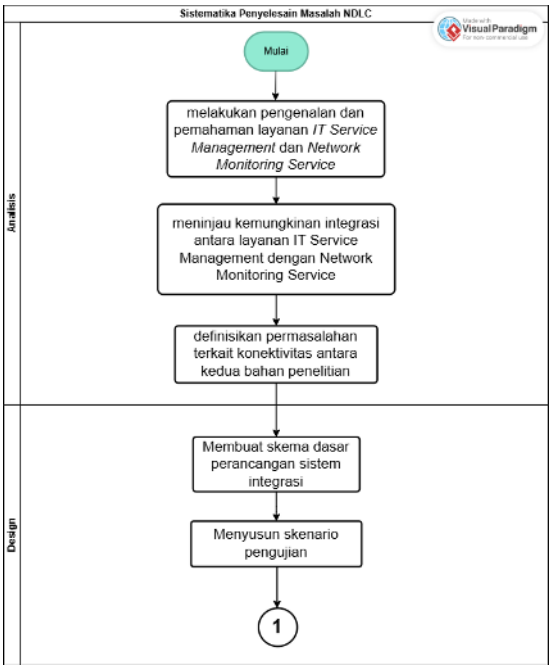
Penelitian ini dimulai dengan observasi sistematis terhadap layanan ITSM untuk memahami fitur-fiturnya dan kebutuhan integrasi sistem. Hasil observasi membantu mengidentifikasi permasalahan objek penelitian, dengan mempertimbangkan latar belakang masalah. Rumusan masalah dan batasan-batasan penelitian ditetapkan untuk memfokuskan tujuan penelitian. Tahap ini juga meliputi persiapan alat dan dokumentasi yang diperlukan untuk mendukung proses penelitian.

B. Tahap *Design*

Tahap *design* atau perancangan merupakan tahap yang membahas pembuatan desain perancangan sistem integrasi antara layanan Nagios dengan iTop, dengan memperhatikan hasil analisa kebutuhan untuk merancang desain sistem terintegrasi. Pada tahap ini dilakukan juga pembuatan rancangan skenario pengujian yang akan dilakukan dalam proses pengujian untuk mengetahui apakah pengujian yang dilakukan berhasil atau tidak.

C. Tahap *Simulation Prototyping*

Tahap *Simulation Prototyping* mencakup pemasangan dan pengujian Nagios dan iTop, termasuk integrasi kedua layanan untuk *ticketing* secara *real time*. Skenario pengujian



melibatkan host down dan layanan critical, dengan analisis dilakukan jika terjadi kegagalan. Tahap ini menghasilkan profil ticketing, evaluasi konektivitasa, kesimpulan penelitian dan saran perbaikan.

IV. HASIL DAN PEMBAHASAN

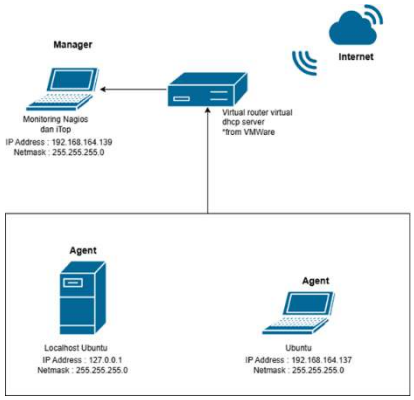
Bagian ini menjelaskan proses persiapan perangkat serta rancangan skenario terhadap sistem Nagios dan iTop serta skenario otomatisasi hingga hasil yang diperoleh dari penerapannya.

A. Topologi jarinngan

Topologi yang rancang adalah implementasi NMS Nagios dengan *output* yang dihasilkan berupa tiketing *incident* dan *problem* pada iTop. Dengan sistem monitoring

Gambar 6 Topologi Jaringan Pengujian

pada topologi ini, Nagios sebagai layanan yang memantau jaringan dapat memonitor objek di dalam jaringan secara *real time*, mampu mendeteksi status *bandwith* (*up* atau *down*) serta dapat mengetahui adanya *error* yang terjadi pada



jaringan tersebut. Nagios akan mengirimkan *alert* pada iTop apabila *host* maupun *services* yang berada pada sistem

Spesikasi VM 1	
Operating System	Ubuntu 20.04
Memory	2 GB
Processor	2
Harddisk	20 GB
Network Adapter	Network Address Transaltion (NAT)
Internet Protocol	192.168.164.137/24

pemantauan jaringan Nagios dalam keadaan *down/error* dan dapat di tangani secara otomatis oleh Nagios.

Gambar 6 menjelaskan bagaimana topologi kedua layanan yang diklasifikan menjadi dua yaitu sebagai agent dan manager. Agent atau host pada pengujian ini adalah Linux Ubuntu dengan alamat IP 192.168.164.137/24 dan Linux Ubuntu localhost dengan alamat IP 127.0.0.1/24 yang bertindak sebagai objek yang di monitoring host mapun servicesnya. Sedangkan yang bertindak sebagai manager adalah perangkat keras laptop yang telah terpasang NMS Nagios dan iTop yang telah terintegrasi

Dalam topologi jaringan diatas terdapat perangkat – perangkat yang saling terhubung satu sama lain untuk mendukung proses integrasi antara iTop dan Nagios. berikut

Gambar 5 Sistem Penyelesaian Masalah Penelitian

spesifikasinya.

TABEL 1
Perangkat Lunak Pengujian

Perangkat Lunak	Versi	fungsi
VMWare Workstation	16.2.5	Perangkat lunak virtualisasi yang memungkinkan pengguna untuk menjalankan beberapa sistem operasi secara bersamaan di satu komputer fisik. VMWare Workstation digunakan untuk menginstall sistem operasi Linux Ubuntu
Linux Ubuntu	20.04	Sebuah distribusi sistem operasi dari Linux yang memiliki ketersediaan berbagai perangkat lunak pendukung yang dapat diakses melalui repositori resmi. Linux Ubuntu digunakan untuk menginstall aplikasi iTop dan Nagios
iTop	3.2	Perangkat lunak manajemen layanan IT open-source yang digunakan untuk mengelola infrastruktur IT. iTop digunakan untuk keperluan integrasi Nagios dengan iTop.
Nagios	4.4.11	Platform monitoring sistem *open-source* yang digunakan untuk memantau infrastruktur IT, termasuk server, layanan, aplikasi, dan perangkat jaringan. Nagios digunakan untuk keperluan monitoring sebuah perangkat.

Tabel 1 merupakan spesifikasi perangkat lunak yang digunakan dalam pengujian. Dalam pengujian ini menggunakan 2 *virtual machine*.

TABEL 2
Spesfikasi Virtual Machine

Harddisk	20 GB
Network Adapter	Network Address Transaltion (NAT)
Internet Protocol	192.168.164.139/24

Spesifikasi VM 2	
Operating System	Ubuntu 20.04
Memory	2 GB
Processor	2

dari tabel diatas, pengujian menggunakan 2 *virtual machine* yang memeiliki perannya masing-masing. *Virtual machine* 1 berperan sebagai *host* yang akan di pantau untuk layanannya dengan alamat *internet protocol* 192.168.164.137/24, sedangkan *virtual machine* 2 bertugas sebagai *server* tempat dipasangnya iTop dan Nagios. Dapat disimpulkan *host* utama yang tidak masuk kedalam pemantauan *services* hanya tempat dijalankannya iTop dan Nagios. Pada kedua virtual machine digunakan NAT (*Network Address Translation*) sebagai *network adapter* agar dalam menjalankan *virtual machine* dengan akses internet tanpa perlu mengeksposnya ke jaringan eksternal, memungkinkan pengujian aplikasi dalam lingkungan yang aman. Selain itu, penggunaan NAT *adapter* dapat memudahkan komunikasi melalui jaringan internal yang dibuat oleh *virtual machine*. Ini memungkinkan dalam penelitian ini untuk bekerja sama dalam lingkungan yang aman dan terpisah tanpa memerlukan akses dari luar jaringan.

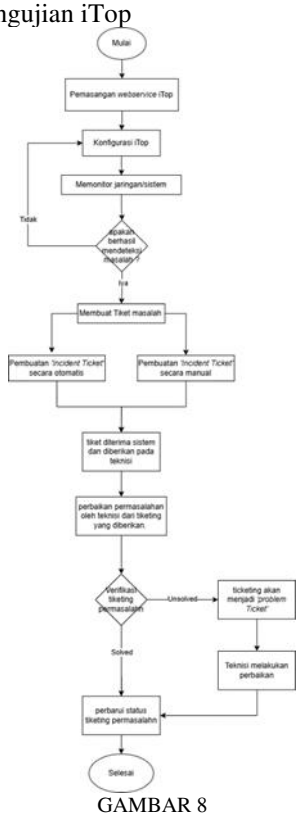
B. Skenario Percobaan

Skenario percobaan adalah serangkaian langkah terstruktur yang digunakan untuk menguji sistem, aplikasi, atau proses guna memastikan kinerja, keandalan, dan fungsionalitasnya sesuai dengan persyaratan yang ditetapkan. Skenario percobaan ini dirancang dalam bentuk flowchart, berikut daftar skenario yang dibuat:

1. Skenario Pengujian Nagios.

GAMBAR 7
Skenario pengujian sistem monitoring nagios

2. Skenario Pengujian iTop



GAMBAR 8
Skenario iTop dalam pembuatan ticket

3. Skenario Integrasi

GAMBAR 9
Skenario pengujian integrasi

C. Implementasi

Berikut tahapan implementasi yang dilakukan berdasarkan pada skenario percobaan sebelumnya:

1. Konfigurasi Nagios

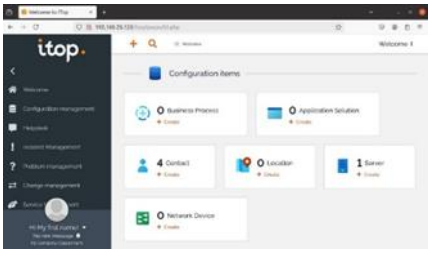
Dalam implementasi Nagios dilakukan serangkaian kegiatan yang berpedomana pada skenario pengujian nagios. Konfigurasi Nagios meliputi dari proses instalasi dan pengaturan perangkat lunak nagios agar bis digunakan sebagai alat NMS. Output yang diharapkan adalah nagios mampu mendefiniskan layanan pada host yang sudah di konfigurasi yang meliputi status, keadaan layanan dan deteksi masalah yang terjadi. Yang nantinya proses tersebut akan menjadi salah fungsionalitas acuang untuk menghubungkan sistem pemantauan yang di realisasikan pada sistem tiketing iTop. Berikut adalah tampilan halaman Nagios dan daftar layanannya :

GAMBAR 10
Halaman homepage nagios

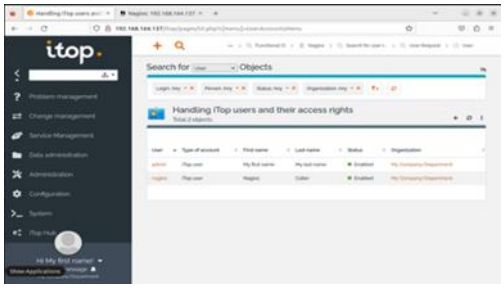
GAMBAR 11
Daftar Layanan yang berhasil di konfigurasi

2. Konfigurasi iTop

Dalam implementasi perangkat iTop, dilakukan serangkaian pengujian yang berpedoman pada skenario pengujian untuk mengetahui sistem pengelolaan layanan menggunakan iTop. Konfigurasi iTop dilakukan meliputi proses instalasi, pengaturan perangkat dan pengujian berdasarkan studi kasus pengelolaan layanan. Berikut tampilan laman iTop :



GAMBAR 12
Tampilan homepage iTop



GAMBAR 13
Handling user dari iTop

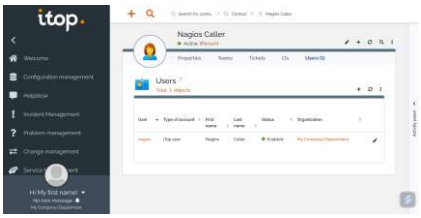
3. Konfigurasi Integrasi

Setelah melakukan pengujian, instalasi dan konfigurasi pada setiap fungsionalitas NMS nagios dalam sistem pemantauan jaringan dan sistem tiketing pengelolalaan layanan pada iTop, selanjutnya adalah untuk menggabungkan kedua layanan tersebut. Sistem integrasi dibuat untuk membuat efisisensi tim IT dalam sebuah organisasi dengan memanfaatkan kedua fungsi layanan. Sistem integrasi dirancang dengan memanfaatkan pemantauan jaringan pada perangkat nagios lalu, deteksi masalah akan direpresentasikan melalui pengelolaan layanan dengan sistem tiketing pada iTop. Pada implementasi sistem integrasi ini, menjalankan dan menghubungkan iTop dengan Nagios terdapat beberapa aspek yang perlu di konfigurasi yang dalam layanan iTop hal itu meliputi *user* dan *services*. Pada implementasi dibuatkan penerapannya menjadi dua yaitu pembuatan tiketing secara otomatis dan pembuatan tiket secara manual.

a. Pembuatan *Ticket* Otomatis

GAMBAR 14
REST User profile Nagios pada iTop

Dalam pembuatan tiket otomatis diperlukan skrip ‘createNagios-ticket.php’ sebagai dasar dalam atau *controler* pembuatan tiketnya. Selain itu, perlu di konfigurasi dari segi perangkat iTop dan Nagiosnya yang meliputi konfigurasi user pada itop, *profile nagios caller, REST, Services, Organisation, event handler pada server nagios dan NRPE pada server nagios* .



REST *Services user* dalam iTop berperan dalam kontrol akses pengguna dengan data program fungsionlitas melalui RESTful API. REST *Services* user juga berperan dalam operasi CRUD (*Create, Read, Update, delete*) pada data dalam iTop serta memfasiltasi integrasi iTop dengan aplikasi atau sistem ekternal iTop. mendefiniskan user Nagios dengan *person Nagios Caller* memiliki tanggung jawab atau peran sebagai administrator dan *REST services user*. Yang nantinya mendukung untuk pengambilan data hasil monitoring pada perangkat Nagios.

```
<?php
// Adjust these settings to fit your iTop
installation
// iTop Server location
define('ITOP_SERVER_URL',
'https://localhost/webservicess/iTop.wsdl.php');
// Valid credentials for connecting to iTop
define('ITOP_USERNAME', 'admin');
define('ITOP_PASSWORD', 'admin');
// Default settings for creating the incident ticket
define('DEFAULT_IMPACT', 2); // 1 = department, 2 =
services, 3 = person
define('DEFAULT_URGENCY', 2); // 1 = high, 2 =
medium, 3 = low
define('DEFAULT_CALLER_NAME', 'Caller Nagios');
define('DEFAULT_ORGANIZATION_NAME', 'My
Company/Department');
define('DEFAULT_SERVICES_NAME', 'Server');
define('DEFAULT_SERVICES_SUBCATEGORY', 'Ubuntu');
define('DEFAULT_WORKGROUP_NAME', 'Department
Customer');
// End of instance-specific parameters
```

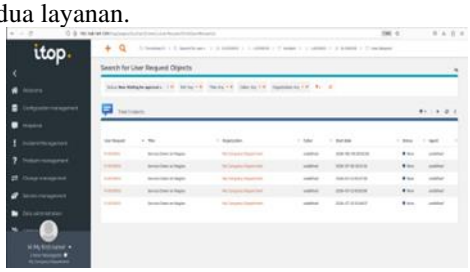
GAMBAR 15
Profile Nagios Caller pada iTop

command diatas merupakan skrip pembuatan tiket secara otamatis.

b. Pembuatan *Ticket* Manual

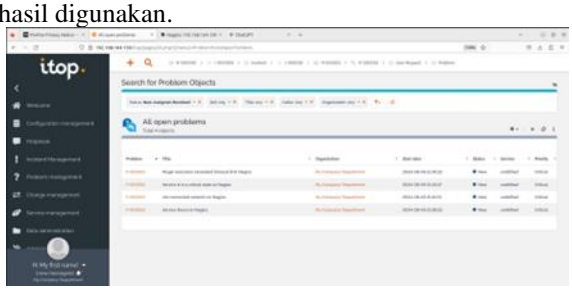
Pada pengujian yang dilakukan, jika *ticketing* tidak berhasil dibuat secara otomatis dan tidak melampirkan

hasil *down host* atau *ciritical* status *services* yang terjadi pada layanan Nagios. untuk itu penulis membuat solusi untuk membuat *script ticket* secara manual. Pada implementasi pembuatan tiket manual, menggunakan skrip *create-ticket-manual-incident.php* yang dibuat untuk menguji pembuatan dan konektivitas integrasi kedua layanan.



GAMBAR 16
Ticketing Incident Management

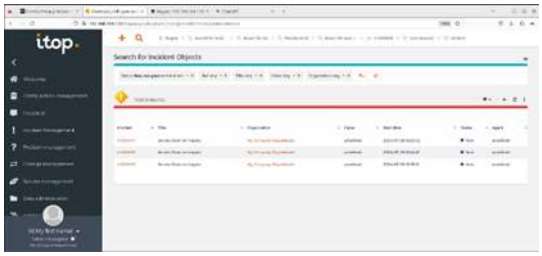
Gambar 16 merupakan *output ticketing* berhasil di buat dan ditampilkan pada modul atau *tab incident management* ini membuktikan bahwa pembuatan *ticke*t secara manual berhasil digunakan.



GAMBAR 17
Ticketing Helpdesk

Gambar 17 merupakann output dari pembuatan tiket manual yang di *assign* pada modul *Helpdesk* iTop.

Tiket manual bisa dibuat dan di assign pada modul problem management yang menampilkan kendala yang terjadi pada Nagios secara manual dan memberikan informasi rincian tiket berupa ID tiket judul, organisasi, *start date, services dan priority.*



GAMBAR 18
Ticketing Problem Management

D. Hasil Pengujian Nagios

Berikut ini adalah tabel yang menggambarkan langkah-langkah pengujian dan kriteria keberhasilan dari proses *monitoring* pada Nagios

TABEL 3
Hasil Pengujian Nagios

Kode	pengujian	Hasil pengujian		Status
		Berhasil	gagal	
N1a	1		⊗	Berhasil Diuji
	2	✓		
	3	✓		
N1b	1	✓		Berhasil Diuji
	2		⊗	
	3	✓		
N1c	1	✓		Berhasil diuji
	2	✓		
	3	✓		
N1d	1	✓		Berhasil diuji
	2		⊗	
	3	✓		
N2a	1	✓		Berhasil diuji
	2	✓		
	3	✓		
N2b	1		⊗	Berhasil diuji
	2	✓		
	3	✓		
N2c	1	✓		Berhasil diuji
	2		⊗	
	3	✓		

Keterangan :
N1a : Mengakses web services Nagios dengan *username* dan *password* yang telah ditentukan.
N1b : Menampilkan *homepage web services* Nagios dengan status *daemon* ada tanda centang hijau
N1c : Menampilkan informasi *host*
N1d : Menampilkan informasi *services* dari *host*
N2a : Menampilkan informasi status layanan pada *host*.
N2b : Menampilkan rincian informasi layanan pada *host.*
N2c : Menampilkan status *host (up and down).*

Tabel 3 adalah rangkuman hasil dari data pengujian Nagios yang telah dikumpulkan pada bab 4. Pengujian meliputi fungsionalitas antarmuka Nagios dengan 4 indikator penilaian dan fungsionalitas sistem monitoring layanan dengan 3 indikator penilaian. Dari pengujian dilakukan semua indikator berhasil diuji dan mencapai parameter keberhasilan.

E. Hasil Pengujian iTop

Berikut ini adalah tabel yang menggambarkan langkah-langkah pengujian dan kriteria keberhasilan dari sistem *ticketing* pada iTop

TABEL 4
Tabel hasil pengujian iTop

Kode	pengujian	Hasil pengujian		Status
		Berhasil	gagal	
iT1a	1		⊗	Berhasil Diuji
	2	✓		
	3	✓		
iT1b	1		⊗	Berhasil Diuji
	2	✓		
	3	✓		
iT1c	1	✓		Berhasil diuji
	2	✓		
	3	✓		
iT1d	1	✓		Berhasil diuji
	2		⊗	
	3	✓		
iT1e	1	✓		Berhasil diuji
	2	✓		
	3	✓		
iT2a	1	✓		

	2	✓		Berhasil diuji
	3	✓		
iT2b	1	✓		Berhasil diuji
	2	✓		
	3	✓		
iT2c	1	✓		Berhasil diuji
	2	✓		
	3	✓		

Keterangan :
iT1a : Masuk sebagai *administrator* menggunakan *username* dan *password* yang telah di konfigurasi sebelumnya.
iT1b : Menampilkan *web services* iTop beserta modul penunjang iTop.
iT1c : Mendapatkan hak akses dan menampilkan semua modul penunjang pada iTop.
iT1d : Menampilkan dan memodifikasi *user profile* serta *user account* pada modul *administration.*
iT1e : Menambahkan dan menampilkan *services* dan akun pengguna.
iT2a : Membuat dan memodifikasi *ticket user request.*
iT2b : Membuat dan memodifikasi *ticket incident*.
iT2c : Membuat dan memodifikasi *ticket problem.*

Pada tabel 4 merupakan hasil rekapitulasi dari pengujian iTop yang diambil dari implentasi dan data pengujian setiap fungsionalitasnya. Terdapat 2 pengujian yaitu pengujian fungsionalitas antarmuka iTop dengan 5 indikator keberhasilan dan pengujian sistem *ticketing* dengan 3 indikator keberhasilan.

F. Hasil Pengujian integrasi

Berikut ini adalah tabel hasil dari pengujian yang dilakukan pada sistem integrasi kedua layanan. Pada pengujian impelementasi integrasi antara perangkat Nagios dengan iTop yang diuji adalah bagaimana konektivitas dan pembuatan tiket dengan penerapan sistem integrasi.

TABEL 5
Tabel hasil pengujian integrasi

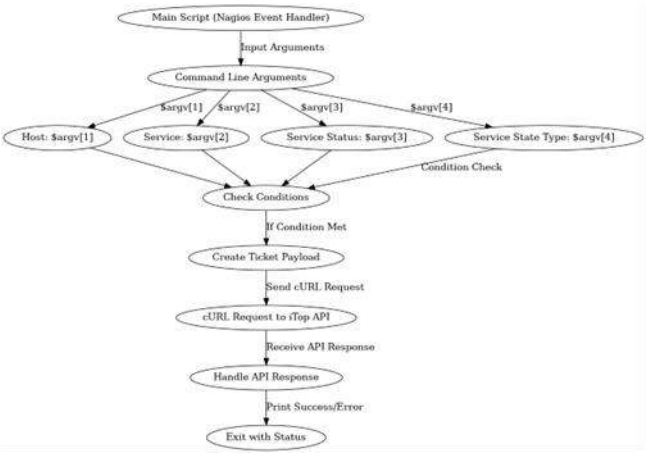
No.	Pengujian	Hasil yang diharapkan	Hasil pengujian	status
1.	Inisialisasi konfigurasi integrasi iTop dan Nagios	Nagios dan iTop berhasil dikonfigurasi sistem integrasinya tanpa error	Kedua sistem berjalan	Berhasil
2.	Konektivitas integrasi script createTicket-Nagios.php	Nagios dapat mendeteksi atau menangkap event handler skrip tiket otmatis	Event handler skrip	Berhasil
3.	Pembuatan ticket hasil monitoring Nagios secara otomatis.	Hasil monitoring *host* secara otomatis akan membuat ticket kendala dengan mengeksekusi skrip createTicket-Nagios.php	iTop tidak mendeteksi kendala dari monitoring. Ticket tidak dibuat	Tidak berhasil
4.	Konektivitas dan Pembuatan ticket hasil monitoring Nagios secara manual.	Dapat membuat ticket dari server Nagios dengan mengeksekusi skrip create-ticket-manual-incident.php	ticket terbuat sesuai isi skrip.	berhasil

Dalam pengujian integrasi terdapat indikator untuk mengindikasi integrasi antara Nagios dengan iTop berjalan sesuai prosedurnya. Indikator **instalasi dan konfigurasi integrasi Nagios dengan iTop** yang meliputi proses konfigurasi plugins NRPE dalam memantau kondisi kinerja jaringan secara efektif dan konfigurasi user, *services* serta organisasi pada iTop untuk membentuk jalan penghubung antara kedua layanan. pada indikator indikator ini memperoleh hasil pengujian berhasil karna setiap komponen mencapai hasil yang diharapkan. **pengujian konektivitas skrip createTicket-Nagios.php** merupakan tahap untuk menunjukkan apakah secara konektivitas skrip terhubung satu sama lain yang dapat dilihat pada munculnya *event handler* skrip pada perangkat Nagios. Pada indikator ini memproleh hasil pengujian yaitu Nagios mampu mendeteksi *event handler* dari skrip sehingga secara konektivitas kedua layanan telah terhubung. Indikator **Pembuatan *ticket* hasil monitoring Nagios secara otomatis** memperoleh status **TIDAK BERHASIL** pada pengujiannya walau komponen pendukung pada pengujian ini berhasil di uji untuk itu pada indikator **pembuatan ticket manual** dengan mengeksekusi skrip create-ticket-manual-incident.php dilakukan untuk memeriksa kembali konektivitas dan pembuatan ticket secara manual dari server Nagios pada perangkat iTop.

Skrip create-ticket-manual-incident.php adalah salah satu bentuk alternatif dalam pembuatan *ticket* yang terhubung dengan *server* Nagios dan sebagai salah satu analisa konektivitas integrasi kedua layanan.

TABEL 6
Kapabilitas script create-ticket-manual-incicdent.php

Kapabilitas	Code	Definisi
Konfigurasi konektivitas dan autentifikasi ke iTop	`$ITOP_URL, $ITOP_USER $ITOP_PWD`	mendefinisikan URL endpoint, username, dan password yang digunakan untuk mengakses API iTop.
Argument handling	`$argc dan $argv`	mengambil parameter yang diberikan dari *command line*. Jika jumlah parameter tidak tepat, maka program akan memberikan pesan tentang cara penggunaan yang benar.
Pemeriksaan status layanan	`$services_status, $services_state_type`	Kondisi berikut digunakan untuk memeriksa dan mendefiniskan kondisi layanan.
Pembuatan Payload untuk iTop	`Operation, class, fields, org_id dan fungsional_list`	data yang akan dikirim ke iTop melalui API.
Pengiriman permintaan ke iTop menggunakan cURL	`CURLOPT_POSTFIELDS, CURLOPT_SSL_VERIFYPEER dan CURLOPT_SSL_VERIFYHOST`	Kode menggunakan cURL untuk mengirim *request* HTTP POST ke API iTop



GAMBAR 19
UML script create-ticket-manual-incident.php

Dalam tabel diatas merupakan kapabilitas atau fungsi yang dapat dijalankan dari skrip pembuatan tiket manual dengan mengeksekusi skrip create-ticket-manual-incident.php. dalam Tabel 6 menjelaskan 5 kapabilitas yang dapat dilakukan oleh skrip create-ticket-manual-incident.php.

Gambar V.4 merupakan UML diagram dalam bentuk statechart diagram yang menjabarkan alur kerja dalam mengeksekusi perintah pembuatan tiket manual.

1. Skrip ini dimulai dengan eksekusi oleh Nagios sebagai bagian dari *event handler*, yaitu sebuah mekanisme yang dijalankan ketika terjadi suatu *event* di Nagios.
2. Setelah eksekusi dimulai, skrip menerima input berupa argumen dari *command line*. Argumen-argumen ini memuat informasi penting seperti nama *host* ($argv[1]), nama layanan atau *services* ($argv[2]), status layanan ($argv[3]), dan tipe status layanan ($argv[4]). Informasi ini kemudian digunakan untuk menentukan apakah kondisi tertentu terpenuhi.
3. Skrip kemudian melakukan pengecekan kondisi berdasarkan status layanan dan tipe status layanan. Jika status layanan tidak berada dalam keadaan *"OK"* atau *"UP"* dan tipe statusnya adalah *"HARD",* maka kondisi dianggap terpenuhi. Apabila kondisi ini terpenuhi, skrip melanjutkan proses untuk membuat payload tiket. *Payload* ini berisi informasi seperti *host* dan *services* yang bermasalah serta detail masalah yang dideskripsikan.
4. Setelah *payload* tiket berhasil dibuat, skrip mengirimkan *payload* ini ke API iTop menggunakan cURL, sebuah alat untuk melakukan permintaan HTTP. Setelah permintaan dikirimkan, skrip menerima *respons* dari API iTop. Respons ini bisa berupa informasi bahwa tiket berhasil dibuat atau sebaliknya, sebuah pesan *error* jika terjadi masalah dalam proses pembuatan tiket.
5. Skrip kemudian menangani *respons* yang diterima dengan menampilkan pesan hasil dari permintaan tersebut, baik itu pesan keberhasilan atau pesan *error*. Akhirnya, skrip mengakhiri proses dengan keluar sesuai dengan status yang diperoleh, menandakan bahwa seluruh proses telah selesai dijalankan.

Secara keseluruhan, diagram ini memberikan gambaran yang sistematis mengenai langkah-langkah yang diambil oleh skrip PHP untuk memantau status layanan di Nagios dan otomatis membuat tiket di iTop apabila ditemukan masalah yang sesuai dengan kriteria yang ditentukan.

V. KESIMPULAN

Berdasarkan penelitian yang telah dilakukan dalam "Implementasi dan Profiling Software Opensource iTop Dengan Software Opensource Nagios Berdasarkan Aspek Integrasi", didapat kesimpulan bahwa :

A. Nagios efektif memantau layanan pada host melalui antarmuka dan sistem monitoring-nya. Perangkat ini mampu mendeteksi kendala spesifik layanan dan menghasilkan *alert*, yang kemudian menjadi dasar pembuatan *ticket* di iTop.

B. iTop berfungsi efektif dalam pembuatan dan pengelolaan ticketing melalui antarmuka yang baik. Perangkat ini mendukung pembuatan tiket spesifik melalui modul *Helpdesk*, *Incident Management*, dan *Problem Management*. iTop juga memfasilitasi pengelolaan ticket layanan, termasuk penambahan *user, service,* organisasi, dan informasi spesifik dalam proses pembuatan *ticket*.

C. Analisis dari penerapan sistem integrasi dalam mengubungkan pemantauan layanan pada kelas infrastruktur jaringan menggunakan Nagios dengan pengelolaan layanan IT menggunakan iTop menunjukkan bahwa, terdapat indikasi kemungkinan untuk pembuatan *ticket* otomatis secara *real time*.hal ini dibktikan dari hasil yang di peroleh yaitu :

1. *Ticket* terbuat karena dengan menghubungkan kinerja pemantauan layanan pada skala infratruktur jaringan, yang menggunakan Nagios dalam memperoleh informasi pemantauannya kemudian, diberikan pada perangkat iTop dalam bentuk *ticket* kendala. Hal ini dibuktikan dalam pengujian konektivitas yang telah dilakukan, *server* Nagios berhasil mendeteksi *event handler* skrip createTicket-Nagios.php dalam pembuatan *ticket* otomatis namun dalam eksekusi skrip tersebut, tidak terbentuk sebuah *ticket* pada layanan iTop.
2. dirancang sebuah skrip dalam menguji konektivitas dan pembuatan *ticket* dari *server* Nagios ke perangkat iTop yaitu dengan menjalankan skrip create-ticket-manual-incident.php. penerapan skrip tersebut berhasil dalam segi konektivitas dan membentuk ticket layanan pada modul perangkat iTop yang meliputi Modul *helpdesk, Incident management dan Problem management.* Dari pengujian yang dilakukan, skrip berhasil dieksekusi diengan output *ticket* pada masing – masing modul yang di konfigrasikan secara manual.

REFERENSI

[1] Cartlidge, A., Ashley, X.-S., Hp, H., Rudd, C., Macfarlane, I., Windebank, J., Stuart, S., & Hp, R. (2007). The IT Infrastructure Library An Introductory Overview of ITIL® V3.

[2] Foundation ITIL ® ITIL 4 Edition ITIL ® OFFICIAL PUBLISHER. (2019).

[3] Marrone, M., & Kolbe, L. M. (2011). Impact of IT Service Management Frameworks on the IT Organization. Business & Information Systems Engineering, 3(1), 5–18. https://doi.org/10.1007/s12599-010-0141-5

[4] Firdaus, M. (2024). iTop Aplikasi Sistem Pencatatan Yang Gratis Untuk Manajemen Layanan IT Dalam Industri Software. https://doi.org/10.5281/zenodo.11216722

[5] Ahmad, N., Amer, N. T., Qutaifan, F., & Alhilali, A. (2013). Technology adoption model and a road map to successful implementation of ITIL. Journal of Enterprise Information Management, 26(5), 553–576. https://doi.org/10.1108/JEIM-07-2013-0041

[6] Oktivasari, P., & Habibullah, T. (2017). Kajian Network Monitoring System menggunakan Nagios dengan Whatsapp sebagai Notifikasi Allert KAJIAN NETWORK MONITORING SYSTEM MENGGUNAKAN NAGIOS DENGAN WHATSAPP SEBAGAI NOTIFIKASI ALERT.

[7] Chaidir, A., Anwar, S., Permana, S., Herdiana, R., & Narasati, R. (2022). Jurnal Ilmiah Manajemen Informatika Dan Komputer, 06(02), 9–16.

[8] A. N. Salim and T. Sutabari, *Jurnal Nuansa Informatika,* pp. volume 17 nomor 1 144 -153, 2023.

[9] Ilyasa, M. K., & Bisma, R. (n.d.). Analisis Manajemen Insiden dan Masalah Layanan IT pada Balitbang Jatim. JEISBI, 03, 2022.

[10] Foundation ITIL ® ITIL 4 Edition ITIL ® OFFICIAL PUBLISHER. (2019).

[11] Burgess, C. (2005). The Nagios Book - Chris Burgess.

[12] Fahreza and M. Rifqi, "Nagios Core Optimization By Utilizing Telegram as Notification of Disturbance," Journal of Applied Science, Engineering, Technology, and Education, p. 122, 2020.

[13] Rahmawati, A., Wijaya, A. F., Fakultas, A. R., & Informasi, T. (n.d.). ANALISIS RISIKO TEKNOLOGI INFORMASI MENGGUNAKAN ISO 31000 PADA APLIKASI ITOP Penulis Korespondensi. http://www.jurnal.umk.ac.id/sitech.

[14] Firdaus, M. (2024). iTop Aplikasi Sistem Pencatatan Yang Gratis Untuk Manajemen Layanan IT Dalam Industri Software. https://doi.org/10.5281/zenodo.11216722

[15] Cartlidge, A., Ashley, X.-S., Hp, H., Rudd, C., Macfarlane, I., Windebank, J., Stuart, S., & Hp, R. (2007). The IT Infrastructure Library An Introductory Overview of ITIL® V3.

[16] Zitek, "ITIL Reactive and Proactive Problem Management: Two sides of the same coin," ISO 20000 Processes and Functions, 2017.

[17] Kosasi, S., Studi, P., Informasi, S., Tinggi, S., Informatika, M., Pontianak, K., Merdeka, J., 372 Pontianak, N., & Barat, K. (2011). PENERAPAN NETWORK DEVELOPMENT LIFE CYCLE UNTUK PENGEMBANGAN TEKNOLOGI THIN CLIENT PADA PENDIDIKAN KSM PONTIANAK. Jurnal Ilmiah Komputasi Dan Elektron, 125–141.

[18] Solichin, A., & Kom, S. (n.d.). Pemrograman Web dengan PHP dan MySQL. http://achmatim.net

[19] Madcoms, Aplikasi Web Database dengan Dreamweaver dan PHP - MySQL, Indonesia: C.V Andi Offset, 2011.

[20] Anggriawan, F. and S. S. P. N. and N. S. (2016). UNIKOM_Firman Anggriawan_KP_BAB 2.