

Otomatisasi Backup Konfigurasi Perangkat Jaringan Komputer Cisco

FIRMAN DWI JAYA PRASETYO, HADI SUTARJO, DANDI TRIANTORO,
DOLLY VIRGIAN SHAKA YUDHA SAKTI*

Fakultas Teknologi Informasi, Universitas Budi Luhur, Jakarta
Email: 2011510076@student.budiluhur.ac.id , *dolly.virgianshaka@budiluhur.ac.id

Received 30 Januari 2024 | *Revised* 15 Februari 2024 | *Accepted* 28 Mei 2024

ABSTRAK

Manajemen perangkat jaringan umumnya dilakukan oleh administrator sesuai dengan tingkat kompleksitas jaringan. Penelitian ini bertujuan untuk mempercepat proses pemeliharaan yang dilakukan pada beberapa perangkat router dan switch cisco. Setiap bulan, laporan bulanan diperlukan untuk merekam kegiatan pemeliharaan, yang secara inti membahas rincian perangkat mana yang masih beroperasi. Selain itu, kegiatan ini juga mencakup pembuatan cadangan konfigurasi pada setiap perangkat, dengan tujuan memberikan tim pemeliharaan akses cepat ke konfigurasi terakhir jika terjadi situasi darurat, sehingga mempercepat proses pemulihan. Hasil dari pelatihan model menggunakan dataset sintetis yang dibuat dengan menggunakan Python library menunjukkan bahwa model terbaik berhasil mencapai akurasi sebesar 99% terhadap 10 perangkat jaringan yang di uji coba untuk hal ini yang di uji ialah perangkat cisco saja, yang diperoleh dari percobaan para network engineer.

Kata kunci: *Back up konfigurasi, Otomatisasi, Log riwayat*

ABSTRACT

Network device management is generally carried out by administrators according to the level of network complexity. This research aims to speed up the maintenance process carried out on several Cisco router and switch devices. Each month, a monthly report is required to record maintenance activities, essentially discussing the details of which devices are still operational. In addition, this activity also includes creating a backup of the configuration on each device, with the aim of providing the maintenance team with quick access to the latest configuration in case of an emergency situation, thereby speeding up the recovery process. The results of model training using a synthetic dataset created using the Python library show that the best model managed to achieve an accuracy of 99% on 10 network devices that were tested. This was only a Cisco device that was tested, which was obtained from experiments by network engineers.

Keywords: *Back up configuration, Automation, Log history*

1. PENDAHULUAN

Dengan semakin berkembangnya Internet, teknologi informasi telah berkembang menjadi komponen penting dalam kehidupan manusia, terutama dalam proses transformasi data menjadi informasi. **(Akbar, dkk 2019)**. Saat ini, internet hampir menjadi kebutuhan vital bagi banyak orang, dan kebutuhan akan internet terus meningkat. **(Purba, dkk, 2021)**. Banyak penyedia peralatan jaringan muncul sebagai akibat dari kemajuan teknologi jaringan komputer. Topologi jaringan wide area network (WAN) membutuhkan banyak perangkat, khususnya router. Administrator jaringan yang mengonfigurasi beberapa router rentan terhadap kesalahan manusia. Selain itu, kebutuhan untuk beralih antar antarmuka juga meningkatkan waktu yang diperlukan untuk konfigurasi **(Nugroho, dkk, 2022)**. Dengan pengembangan otomatisasi jaringan diharap teknologi ini dapat mengatasi masalah. Otomatisasi jaringan merupakan metode pengelolaan atau administrasi beberapa perangkat jaringan secara bersamaan menggunakan bahasa pemrograman Python. Hal ini memungkinkan untuk mengelola atau mengatur puluhan bahkan ratusan perangkat jaringan dengan mudah dan cepat **(Mashudi, 2020)**. Proses otomatisasi jaringan ini menggunakan logika pemrograman untuk mengatur sumber daya dan layanan jaringan. Ini membuatnya lebih efisien daripada proses konfigurasi manual **(Neidinger, 2021)**. Sebagai contoh, tugas seorang network engineer adalah mengkonfigurasi sebuah perangkat dalam jaringan. Otomasi jaringan menggunakan konsep pemrograman dan menyediakan konsep manajemen jaringan yang terpusat. Oleh karena itu, ketika perangkat pertama kali dipasang, konfigurasi awal biasanya diberikan pada perangkat dengan jumlah perangkat yang relatif sama. Untuk mempermudah dan mempercepat proses konfigurasi, proses otomatisasi jaringan digunakan **(Lori macvittie, 2018)**. Dalam penerapannya sistem otomatisasi backup jaringan, dibutuhkan software atau program untuk mengirim perintah atau command pada perangkat jaringan. Salah satu di antara program yang paling sering digunakan untuk tujuan ini adalah dengan script Python. Seperti Paramiko dan Netmiko, Python memiliki beberapa library akses yang dirancang khusus untuk melakukan otomatisasi jaringan **(Nugroho, dkk, 2020)**.

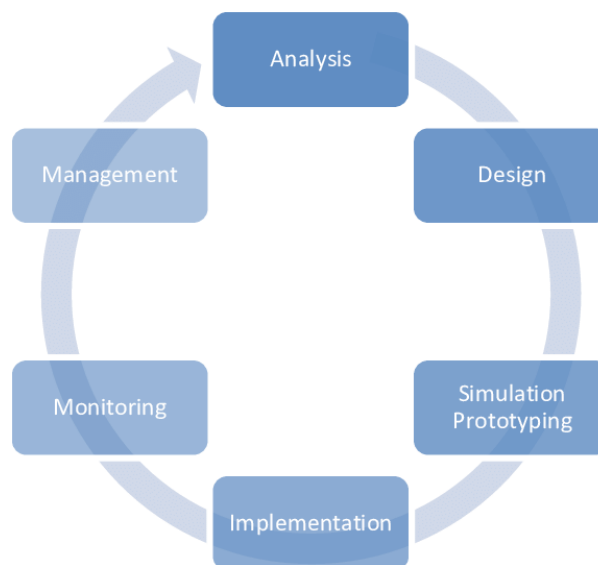
Paramiko merupakan library Python yang menggunakan koneksi SSH dan dapat berfungsi sebagai klien maupun garçon. Dengan memanfaatkan koneksi SSH, Paramiko memungkinkan otomatisasi konfigurasi pada perangkat jaringan seperti router atau switch Cisco yang terhubung. Untuk dapat masuk ke dalam perangkat jaringan, Paramiko memerlukan beberapa informasi, seperti alamat IP (Internet Protocol), username, dan word. Jika ketiga informasi tersebut valid, Paramiko dapat mengakses perangkat jaringan menggunakan SSH customer **(Mauboy dan Wellem, 2022)**. Sedangkan library Netmiko adalah yang dibangun di atas Paramiko dan digunakan secara khusus untuk otomatisasi perangkat jaringan. Ini menyediakan fungsi-fungsi yang lebih tinggi tingkat, seperti login ke perangkat, mengirimkan perintah, dan mengambil outputnya. Akan tetapi library Netmiko tergantung pada Paramiko, sehingga jika memerlukan kontrol yang sangat spesifik terhadap koneksi SSH, mungkin perlu menulis kode tambahan di atas Netmiko **(Kukuh , dkk, 2020)**. Beberapa jenis jaringan memiliki Otomatisasi, seperti *data center network*, *wide area network* (WAN), *local area network* (LAN), *wireless network*, dan *pall network*. Intinya, sumber daya jaringan apa pun yang dikontrol oleh API atau CLI dapat diotomatisasi. Penelitian ini akan menerapkan otomatisasi jaringan pada jenis jaringan *wide area network* (WAN) dengan menggunakan CLI sebagai perintah atau perintah untuk melakukan otomatisasi backup konfigurasi. Untuk menghubungkannya, penelitian ini akan menggunakan protokol SSH **(Netmonk, 2020)**. Otomatisasi dengan Python digunakan oleh director jaringan untuk meminimalkan waktu, tenaga, dan biaya yang dikeluarkan saat mengonfigurasi perangkat jaringan. Otomatisasi ini melibatkan penggunaan Internet Protocol (IP) pada perangkat jaringan dan protokol koneksi jarak jauh seperti Telnet atau SSH. director kemudian akan mengeluarkan perintah berdasarkan jenis perangkat menggunakan Python **(Fahmi, dkk, 2021)**. Dalam penelitian

sebelumnya, pengujian otomatisasi jaringan berhasil. Dalam penelitiannya, diperlukan sejumlah hardware dan software untuk menjalankan sistem pengujian. Hardware yang akan di pakai menggunakan komputer atau laptop, garçon, router Cisco, dan switch unmanageable. Sementara itu, perangkat lunak yang diperlukan mencakup sistem operasi Windows yang berperan sebagai garçon, paket instalasi Python, dan library Paramiko (**Ginting, dkk, 2020**). Namun demikian, otomatisasi networking saat ini masih berupa program dalam CLI bukan mode GUI hanya berupa file script Python yang dijalankan. membuatnya sulit bagi engineer jaringan untuk melihat perangkat mana saja yang terhubung, melihat log tugas, dan memeriksa konfigurasi router Cisco yang pernah dilakukan. Oleh karena itu, pengembangan metode otomatisasi jaringan (robotisasi jaringan) diperlukan untuk memungkinkan konfigurasi berbagai perangkat secara bersamaan. Mengembangkan metode untuk otomatisasi jaringan yang menggunakan komponen programming (**Maharani, dkk, 2020**).

Penelitian ini bertujuan menghasilkan model yang dapat digunakan sebagai dasar metode otomatisasi konfigurasi backup jaringan pada perangkat Cisco dengan menggunakan script Python. Langkah inovatif ini diharapkan dapat membantu network director dalam mempercepat proses provisory dan mengelola sumber daya secara efektif, merespons serta menangani permasalahan yang muncul pada jaringan.

1. METODOLOGI PENELITIAN

Menurut Goldman (2004), pada "Gambar 1" dalam penelitian ini, NDLC (Network Development Life Cycle) akan digunakan dan sesuai dengan tahapan perancangan aplikasi, yang terdiri dari tahapan berikut:



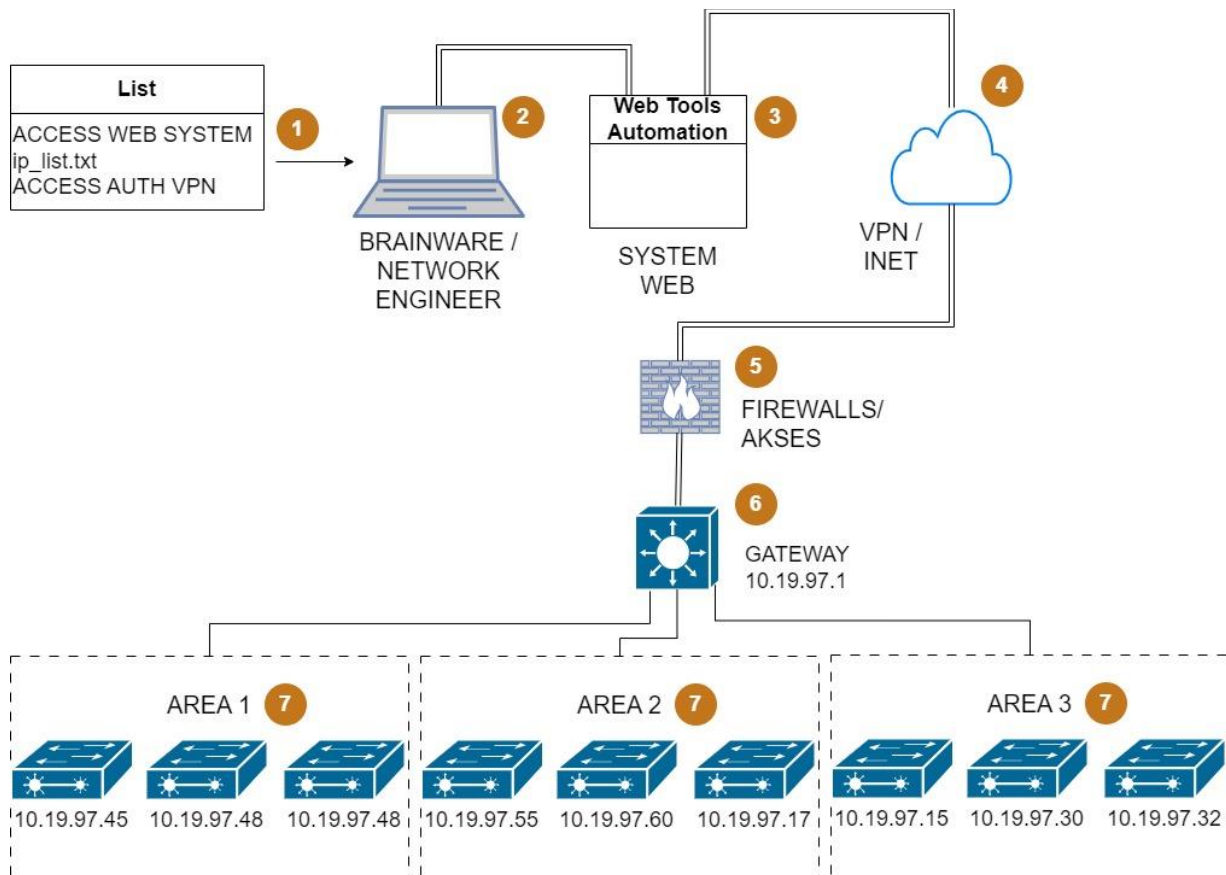
Gambar 1. Tahapan Metode Penelitian

- Analisis pada tahap awal melibatkan wawancara dengan teknisi jaringan untuk mendapatkan pemahaman mendalam tentang masalah yang dihadapi pada jaringan komputer lokal (LAN) dan jaringan komputer luas (WAN). Selain itu, dilakukan penelitian literatur tentang otomatisasi jaringan untuk menganalisis kebutuhan jaringan.
- Desain pada tahap ini, data yang telah didapat pada tahap analisis sebelumnya digunakan untuk merancang topologi jaringan interkoneksi, struktur navigasi, dan fitur aplikasi. Hal ini

- dilakukan dengan menggunakan Diagram Penggunaan Aplikasi dan *activity diagram* aplikasi.
- Simulasi Prototipe , sistem prototipe akan dibangun menggunakan software GNS3 sebagai mereplikasi sistem jaringan komputer yang sedang beroperasi. Untuk membuat aplikasinya, bahasa pemrograman Python 3.x digunakan, dan untuk membuat antarmuka pengguna, HTML, CSS, dan JavaScript digunakan.
 - Implementasi pada bagian ini setelah simulasi prototipe, menerapkan semua yang telah direncanakan pada sebuah simulator.
 - Pada tahap monitoring ini, infrastruktur jaringan yang telah diimplementasikan diuji dengan memeriksa komputer jaringan klien yang terhubung ke jaringan dan melacak jalannya paket data.
 - Tahap terakhir, tahap manajemen adalah penetapan kebijakan untuk memastikan bahwa sistem aplikasi yang telah dibuat dan beroperasi dengan baik akan bertahan lama.

2.1. Topologi Jaringan Sistem

Pada rancangan aplikasi ini, topologi star digunakan oleh aplikasi *Automation Python Network Device* (APND) untuk memberikan akses wireless kepada pelanggan. Topologi star dipilih karena fleksibilitasnya, yang memungkinkan pemasangan perangkat untuk pelanggan baru tanpa mengganggu jaringan pelanggan lain. Adanya kontrol terpusat memungkinkan setiap pelanggan terhubung langsung ke port switch, memberikan keuntungan tambahan. Hal ini memudahkan pengawasan kinerja jaringan dan memungkinkan deteksi cepat kesalahan atau gangguan yang mungkin terjadi di seluruh jaringan (**Fahmi, dkk, 2021**). Berikut adalah rancangan topologi jaringan pada sistem aplikasi, dapat terlihat pada "Gambar 2".

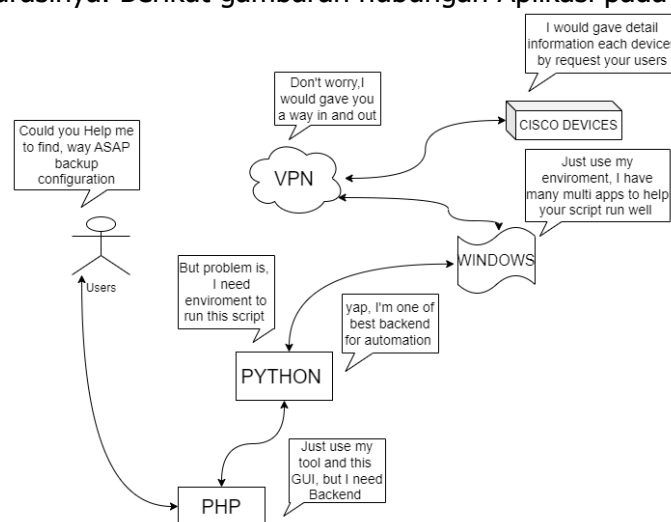


Gambar 2. Topologi Jaringan Sistem

1. Persiapan yang diperlukan untuk menjalankan program *Automation Python Network Device* (APND): a) Akses sistem web: Program ini berbasis web, sehingga memerlukan akses untuk membuka dan menjalankan program. b) *Ip_list.txt*: Daftar IP yang diunggah di program, berisi IP perangkat mana saja yang akan dilakukan backup konfigurasinya. c) Akses VPN: Router OSPF yang mengaktifkan protokol routing OSPF akan bertukar informasi rute. Jika mekanisme VPN (*Virtual Private Network*) ditambahkan, pertukaran informasi rute akan aman (**Musril, 2019**). Program ini menjalankan skrip Python di laptop masing-masing pengguna yang telah diberikan akses, sehingga memerlukan autentikasi VPN untuk mencegah peretasan pada sistem web.
2. Dalam uji coba aplikasi ini, yang melakukan adalah orang yang memiliki akses dan wewenang, yaitu *Network Engineer*.
3. Program yang berbasis web dapat dibuka dari perangkat mana pun yang tentunya browser mendukung HTML5.
4. VPN yang terautentikasi akan memberikan akses jalur masuk dan keluar dari perangkat lokal yang akan diakses.
5. *Firewall* dalam hal ini akan berfungsi untuk mencatat akses pengguna yang aktif dan juga mencatat log masuk dan keluar pengguna beserta aktivitas umum lainnya pada suatu jaringan (**Mauboy dan Wellem, 2022**).
6. *Gateway* di sini menggunakan sebuah *Core Switch* untuk menjadi gerbang utama masuk ke akses lokal. Dalam kasus ini, *gateway* tidak akan mengalami perubahan karena program tidak dirancang untuk merubah target.
7. *End point* yang diambil untuk backup adalah switch. Berikut adalah syarat dasar konfigurasi yang diperlukan agar program backup konfigurasi ini berjalan: a) Konfigurasi IP address management. b) Konfigurasi akses pengguna. c) Konfigurasi SSH; dalam kasus ini, port yang digunakan adalah default, yaitu port 22. d) Memastikan masing-masing switch dapat dijangkau (*reachable*) melalui gateway.

2.2. Gambaran Umum Hubungan Antar Aplikasi

Dalam penelitian ini akan menggunakan Python untuk menjalankan sebuah skrip yang bertujuan mengambil data backup konfigurasi. Secara umum, pengaturan jaringan dapat dilakukan melalui script, yang juga disebut sebagai pengaturan jaringan berbasis script, yang ditulis dalam bahasa pemrograman Python (**Alex, 2020**). Maka perlu menyiapkan file *ip_list.txt* yang berisi informasi alamat IP dari setiap perangkat Router dan Switch Cisco yang akan diambil konfigurasinya. Berikut gambaran hubungan Aplikasi pada "Gambar 3"



Gambar 3. Hubungan Antar Aplikasi

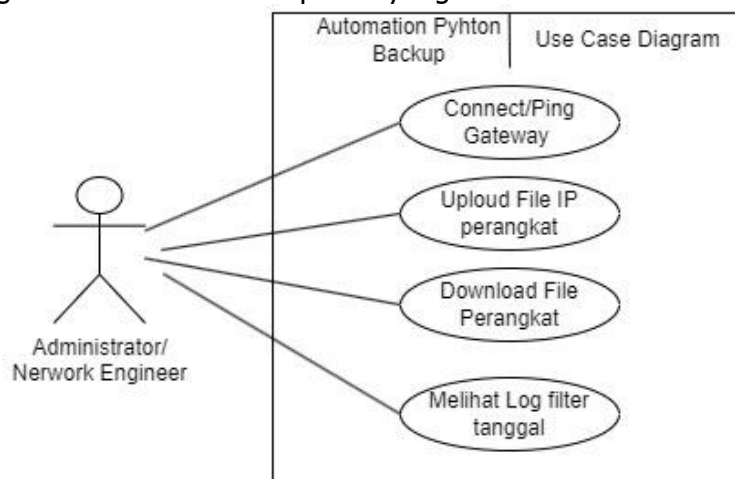
1. Users Dengan PHP : Users akan mengakses web aplikasi untuk backup konfigurasi otomatisasi pada perangkat cisco.
2. Php dengan python : php yang akan memiliki tugas untuk memberikan visual dari hasil yang didapatkan oleh python beserta melakukan interaksi ke users untuk dilanjutkan proses ke python sebagai *backend* dari program ini.
3. Windows dengan python : memberikan akses script python untuk terhubung dengan aplikasi jaringan vpn yang berada didalam os yaitu windows, serta memberikan akses app cmd untuk dijalankan dibackground.
4. VPN dengan windows : memberikan akses keluar masuk untuk script python yang memiliki tugas untuk mendapatkan informasi detail tiap perangkat cisco, sesuai list ip address yang telah diupload oleh users sebelumnya.
5. Perangkat Cisco dengan VPN : memberikan akses keluar masuk perangkat cisco untuk python yang memiliki fungsi memberikan informasi kumpulan ip perangkat yang mana saja untuk dilakukan backup otomatisasi, yang telah diupload sebelumnya oleh users dan menerima fungsi command yang diberikan pada python untuk hasil config dari backup konfigurasinya, memberikan hasil laporan ke python yang akan diteruskan ke php untuk di tampilkan hasil-nya berupa file teks.

Command yang diberikan, diantara sebagai berikut :

- a. show inv : Mendapatkan informasi detail hardware perangkat.
- b. show env all : Mendapatkan informasi status healthy tiap perangkat.
- c. show run : Mendapatkan Konfigurasi yang sedang berjalan/digunakan.

2.3. Diagram Penggunaan Aplikasi

Tahapan selanjutnya dalam penelitian ini, use case diagram dan activity diagram digunakan untuk tahap perancangan model. Use case diagram menunjukkan bagaimana user dan sistem berinteraksi dan fungsi apa yang dapat dilakukan oleh aplikasi (**Rezha, 2019**). "Gambar 4" menunjukkan diagram use case sistem aplikasi yang dibuat.



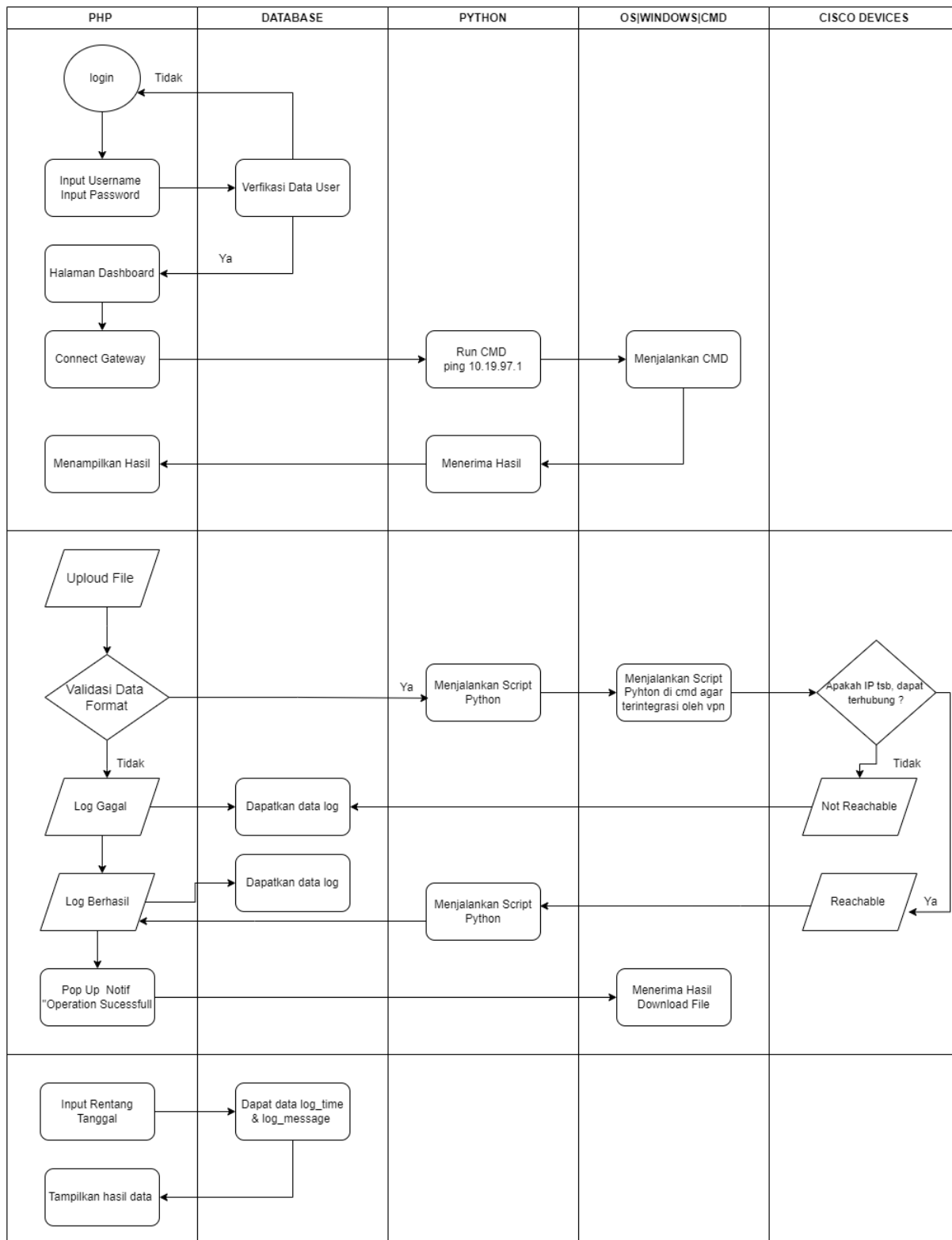
Gambar 4. Diagram Penggunaan Aplikasi

2.4. Aliran Kerja Aplikasi

Dalam perancangan aplikasi ini, Dirancang untuk jaringan skala besar, dalam pengujian ini kita melakukan dengan skala jaringan menengah dan perangkat jaringan menggunakan router switch Cisco. pengguna dapat mengimpor data dari perangkat untuk mengevaluasi proses input dan memproses data tersebut dalam aplikasi *Automation Python Network Device* (APND). Setelah itu, aplikasi akan melakukan proses impor data untuk membuat file dengan konten yang sesuai dengan konten yang telah diprogram sebelumnya. Pengguna kemudian

Otomatisasi Backup Konfigurasi Perangkat Jaringan Komputer Cisco

akan dapat melihat hasilnya dan secara otomatis akan mengunduhnya berdasarkan data perangkat yang telah dimasukkan.



Gambar 5. Activity Diagram Aplikasi

Kemudian aplikasi digunakan. Sebelum diimplementasikan secara nyata, simulasi dilakukan dengan GNS3. GNS3 adalah perangkat lunak yang memungkinkan pengguna untuk melakukan simulasi jaringan komputer (*network simulator*). Karena mendukung perangkat nyata dan virtual, pengguna dapat melakukan simulasi jaringan komputer pada GNS3 seperti pada perangkat jaringan yang sebenarnya. GNS3 dapat dijalankan pada sistem operasi Windows, Linux, dan Mac. Yang membedakan GNS3 dari aplikasi simulator lainnya, seperti Cisco Packet Tracer, adalah bahwa GNS3 memungkinkan simulasi untuk dijalankan pada berbagai perangkat dari berbagai vendor. Ini memberi pengguna kemampuan untuk mengeksplorasi berbagai perangkat dari berbagai vendor. Pengujian dilakukan setelah skenario dibuat metode Black-Box Testing digunakan untuk menguji fungsi pada sistem activity diagram aplikasi (**Islami, 2020**).

2. HASIL DAN PEMBAHASAN

Data *ip_list.txt* yang diperoleh dari implementasi uji coba ini perangkat jaringan router dan switch cisco dan untuk penempatan aplikasi backup tidak dapat diubah sesuai dengan topologi star "Gambar 2". Topologi Jaringan Sistem. Didukung oleh perangkat jaringan yang sudah dikonfigurasi SSH secara manual terlebih dahulu untuk menjadi syarat pertama perangkat bisa terhubung dengan perancangan aplikasi ini, dan administrator dapat dengan mudah mengupload file dari *Automation Python Network Device (APND)* untuk maintenance. Metode black box testing digunakan untuk melakukan pengujian dengan fokus pada spesifikasi fungsional aplikasi. Untuk memastikan bahwa aplikasi yang dibuat memenuhi harapan, pengujian ini dilakukan. Hasil dari pengujian halaman login adalah sebagai berikut:

Tabel 1. Pengujian Dashboard Login

No	Skenario Pengajuan	TestCase	Hasil yang ditentukan	Hasil uji coba	Hasil akhir
1	username dan password diisi dengan akun yang belum terdaftar	Username: (agus) Password : (test)	Pesan belum terdaftar akan ditampilkan dan pengguna akan ditolak oleh sistem.	Sesuai yang ditentukan	Valid
2	Username dan password diisi dengan akun yang telah terdaftar	Username: (nama) Password: (no telpon)	Halaman awal akan ditampilkan oleh sistem.	Sesuai yang ditentukan	Valid
3	Salah satu form dikosongkan	Username: (kosong) Password: (kosong)	Sistem akan menolak akses user, dan pesan form akan muncul yang harus diisi.	Sesuai yang ditentukan	Valid

Pengujian dilakukan sesuai dengan halaman pengujian aplikasi "Tabel 2". Hasilnya adalah bahwa semua fitur sistem aplikasi pada beranda dapat berjalan.

Tabel 2. Pengujian Activity Beranda

No	Skenario Pengajuan	Hasil yang ditentukan	Hasil uji coba	Hasil akhir
1	Tekan tombol login	Halaman beranda akan ditampilkan oleh aplikasi	Sesuai yang di tentukan	Valid
2	Tekan tombol logout	Aplikasi akan menampilkan halaman login.	Sesuai yang di tentukan	Valid
3	Tekan tombol Connect	Aplikasi akan menunjukkan akses ke gateway	sesuai yang di tentukan	Valid

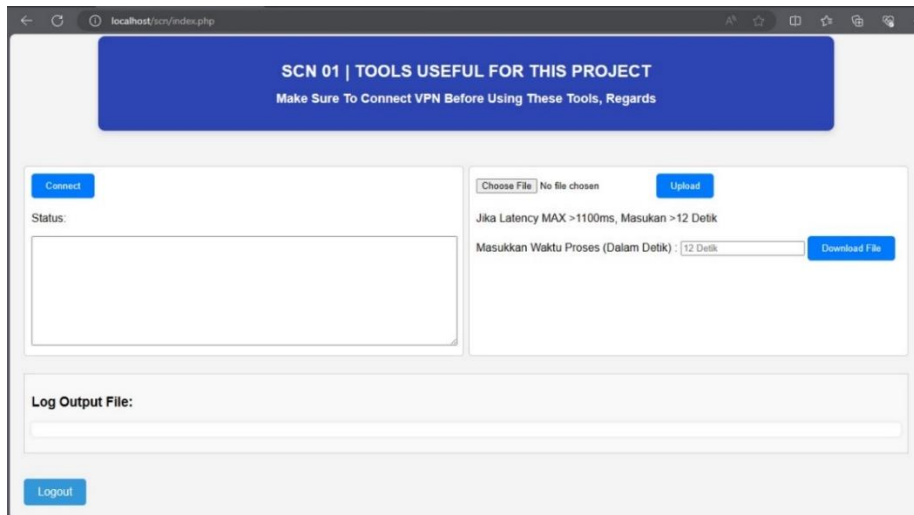
Pada skenario pengujian *exertion* halaman beranda ini menunjukkan hasil bahwa fungsi yang dibuat pada sistem aplikasi berhasil berjalan dan berfungsi dengan baik. Hasil uji coba *Activity* Halaman Beranda *Black-Box Testing* dapat dilihat pada " Tabel 3 ".

Tabel 3. Pengujian Activity Halaman Beranda

No	Skenario Pengajuan	Hasil yang ditentukan	Hasil uji coba	Hasil akhir
1	Tekan tombol choose file	Menampilkan uploaded	Sesuai yang di tentukan	Valid
2	Tekan tombol choose file bila kosong	Your file was not uploaded	Sesuai yang di tentukan	Valid
3	Tekan tombol download show run	Muncul pop up done	Sesuai yang di tentukan	Valid
4	Tekan tombol download show run jika format salah	format salah	Sesuai yang di tentukan	Valid

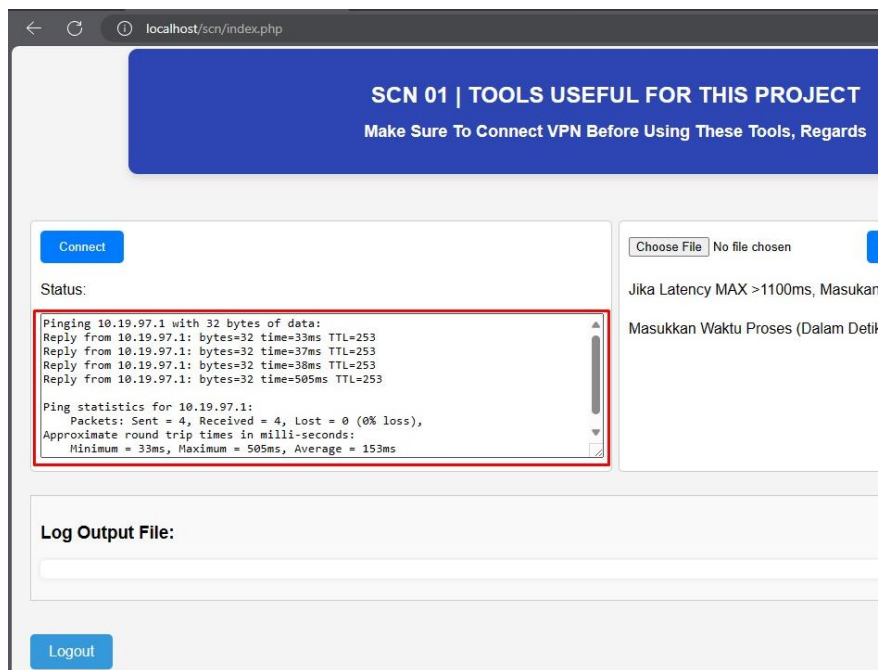
Pada bagian ini, rancangan yang telah dibuat akan diuraikan terkait tampilan layar dari aplikasi *Automation Python Network Device* (APND) berbasis web, mulai dari saat pertama kali dijalankan hingga selesai. Saat pertama kali dimulai, halaman dashboard login akan muncul. Di sana, pengguna akan memasukkan kata sandi yang telah mereka daftarkan untuk di gunakan mengakses aplikasi. Jika kata sandi yang di masukan salah, otomatis akan kembali ke halaman login, tetapi jika kata sandi benar, otomatis akan langsung masuk ke halaman home aplikasi, atau Dashboard (**Olva, dkk, 2021**). Pada tampilan menu utama ini terdapat kumpulan kumpulan submenu yang bisa diakses untuk penggunaan aplikasi *Automation Python Network Device* (APND). Tampilan Layar Menu Utama seperti pada "Gambar 6" dibawah ini:

- a. Melakukan cek koneksi gateway.
- b. Melakukan upload file yang berisi ip adres perangkat.
- c. Melakukan input proses pada masing-masing perangkat.
- d. Melakukan download file backup konfigurasi perangkat.
- e. Melakukan filter log output perangkat.



Gambar 6. Layar Dashbord Menu Utama

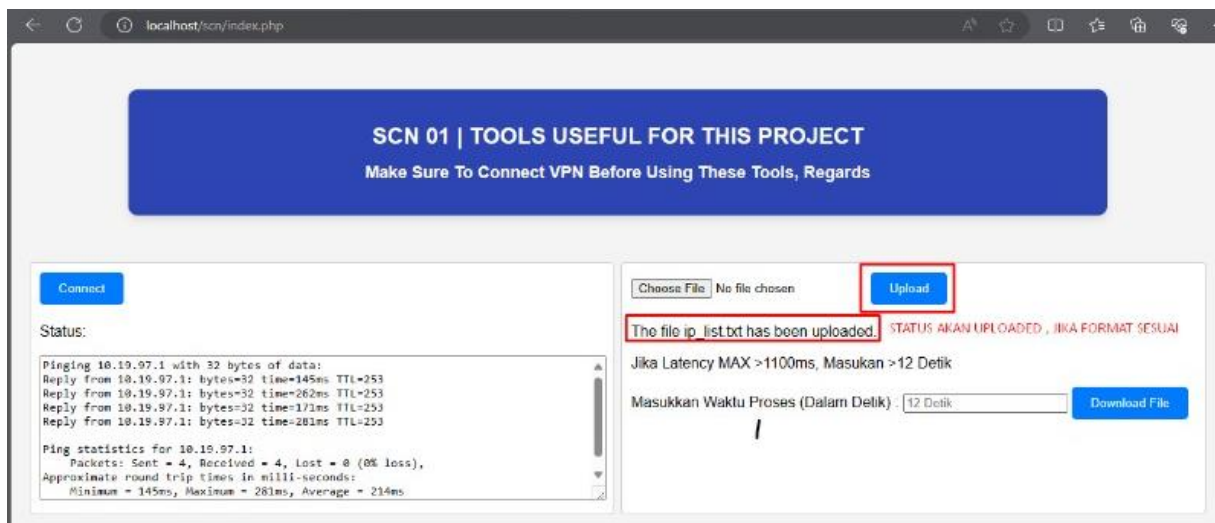
Pada Layar Dashbord Menu Utama aplikasi *Automation Python Network Device* (APND) tahap awal di harus kan melakukan cek koneksi gateway agar mengetahui bahwa tes ping diharuskan di bawah 1100ms agar dapat hasil yang maksimal, yang menandakan untuk dilanjutkan dengan mengupload file Ip Address seperti "Gambar 7"



Gambar 7. Tampilan Tes Koneksi Gateway

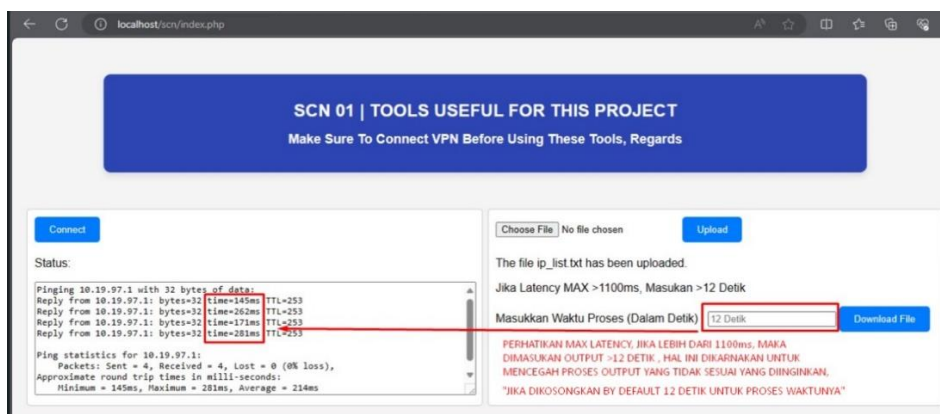
Pada layar ada terdapat button "Upload" yang fungsinya untuk mengupload dan memproses data seperti pada "Gambar 8" dibawah:

Otomatisasi Backup Konfigurasi Perangkat Jaringan Komputer Cisco



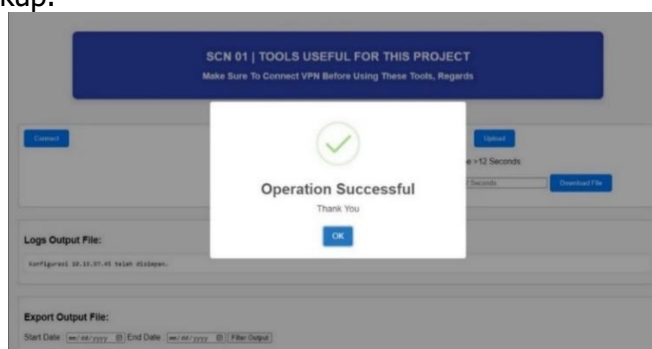
Gambar 8. Tampilan Layar Pemrosesan

Perlu diingat bahwa waktu pemrosesan hanya 12 detik jika koneksi gateway melebihi 1100 ms, ada kegagalan atau hasil yang tidak diharapkan. Oleh karena itu, untuk mendapatkan hasil terbaik, membutuhkan jaringan internet yang cepat dan stabil.



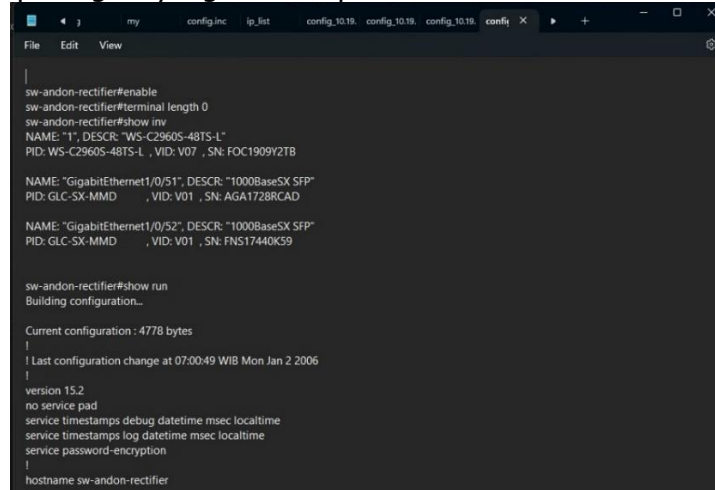
Gambar 9. Tampilan Layar Pemrosesan

Tampilan "Operation Successful" jika format yang di masukan sudah benar dalam waktu pemrosesan yang telah di jalankan dalam waktu 12 detik dan di lanjutkan untuk mendownload file yang telah di backup.



Gambar 10. Tampilan Finish

Hasil dari backup konfigurasi yang telah berhasil dilakukan dapat diunduh di bawah ini. File tersebut berisi data terkini mengenai konfigurasi router yang telah di-backup. Dengan demikian, pengujian dinyatakan berhasil, di mana fitur backup konfigurasi berhasil diujicobakan. Konfigurasi perangkat berhasil di-backup dan tersimpan dengan baik ke perangkat komputer yang dapat mengakses sistem aplikasi. Isi dari backup konfigurasi sesuai dengan konfigurasi perangkat yang di-backup.



```

sw-andon-rectifier#enable
sw-andon-rectifier#terminal length 0
sw-andon-rectifier#show inv
NAME: "1", DESCR: "WS-C2960S-48TS-L"
PID: WS-C2960S-48TS-L , VID: V07 , SN: FOC1909Y2TB

NAME: "GigabitEthernet1/0/51", DESCR: "1000BaseSX SFP"
PID: GLC-SX-MMD , VID: V01 , SN: AGA1728CAD

NAME: "GigabitEthernet1/0/52", DESCR: "1000BaseSX SFP"
PID: GLC-SX-MMD , VID: V01 , SN: FNS17440K59

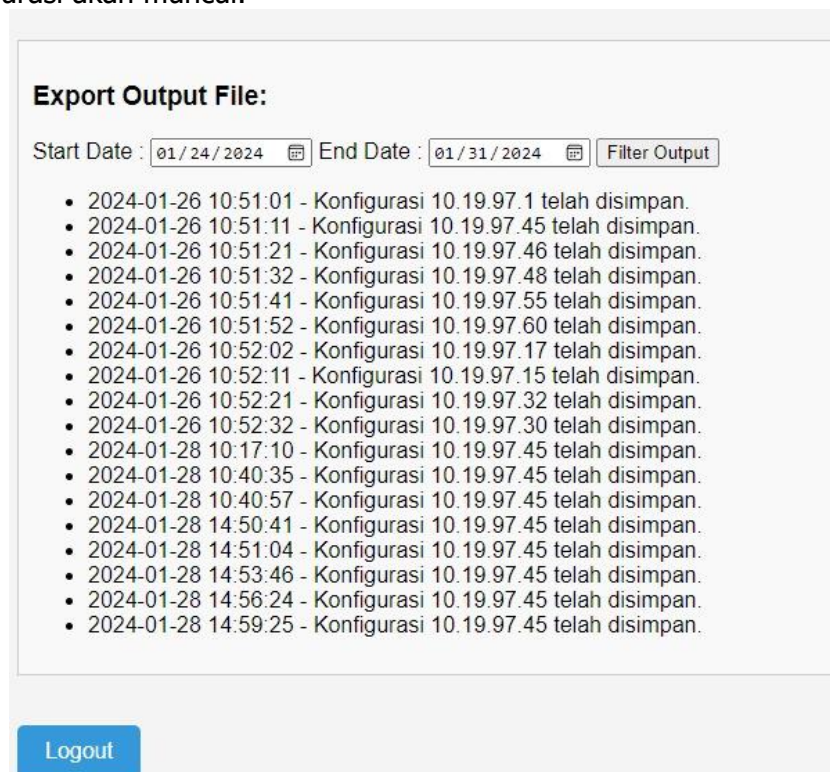
sw-andon-rectifier#show run
Building configuration...

Current configuration : 4778 bytes
!
! Last configuration change at 07:00:49 WIB Mon Jan 2 2006
!
version 15.2
no service pad
service timestamps debug datetime msec localtime
service timestamps log datetime msec localtime
service password-encryption
!
hostname sw-andon-rectifier

```

Gambar 11. Tampilan Hasil Backup Konfigurasi

Pada langkah terakhir, "Gambar 12" menawarkan opsi "Export Output File" di sub menu aplikasi. Dengan mengetik waktu yang telah ditentukan, riwayat log terakhir melakukan backup konfigurasi akan muncul.



Export Output File:

Start Date : End Date :

- 2024-01-26 10:51:01 - Konfigurasi 10.19.97.1 telah disimpan.
- 2024-01-26 10:51:11 - Konfigurasi 10.19.97.45 telah disimpan.
- 2024-01-26 10:51:21 - Konfigurasi 10.19.97.46 telah disimpan.
- 2024-01-26 10:51:32 - Konfigurasi 10.19.97.48 telah disimpan.
- 2024-01-26 10:51:41 - Konfigurasi 10.19.97.55 telah disimpan.
- 2024-01-26 10:51:52 - Konfigurasi 10.19.97.60 telah disimpan.
- 2024-01-26 10:52:02 - Konfigurasi 10.19.97.17 telah disimpan.
- 2024-01-26 10:52:11 - Konfigurasi 10.19.97.15 telah disimpan.
- 2024-01-26 10:52:21 - Konfigurasi 10.19.97.32 telah disimpan.
- 2024-01-26 10:52:32 - Konfigurasi 10.19.97.30 telah disimpan.
- 2024-01-28 10:17:10 - Konfigurasi 10.19.97.45 telah disimpan.
- 2024-01-28 10:40:35 - Konfigurasi 10.19.97.45 telah disimpan.
- 2024-01-28 10:40:57 - Konfigurasi 10.19.97.45 telah disimpan.
- 2024-01-28 14:50:41 - Konfigurasi 10.19.97.45 telah disimpan.
- 2024-01-28 14:51:04 - Konfigurasi 10.19.97.45 telah disimpan.
- 2024-01-28 14:53:46 - Konfigurasi 10.19.97.45 telah disimpan.
- 2024-01-28 14:56:24 - Konfigurasi 10.19.97.45 telah disimpan.
- 2024-01-28 14:59:25 - Konfigurasi 10.19.97.45 telah disimpan.

Gambar 12. Tampilan Daftar Riwayat Log

3. KESIMPULAN

Pada penelitian ini, metode NDLC (*Network Development Life Cycle*) telah diimplementasikan sebagai landasan perancangan aplikasi. Hasil penelitian menunjukkan bahwa model terbaik berhasil mencapai akurasi sebesar 99% terhadap sepuluh perangkat jaringan network cisco. Dalam pengujian ini, aplikasi hanya digunakan untuk kegiatan internal belum dilakukan dengan pengujian lain yang memiliki kebutuhan yang sama, dikarenakan setiap kebutuhan dan lingkungan jaringan berbeda-beda oleh karena itu dibutuhkan penyesuaian kembali disisi aplikasi APND. namun pada pengujian perangkat yang lebih banyak, 50 perangkat jaringan bermerek cisco maka angka keberhasilan menurun menjadi 90% dikarenakan faktor utama yaitu , koneksi internet yang dimiliki harus stabil dibawah 1100ms dengan rekomendasi dibawah 300ms, jika ping yang dimiliki tinggi maka proses mengambil data tiap perangkat akan menjadi lama namun untuk mengatasi masalah tersebut telah tersedia sebuah input proses time sebelum melakukan Download file, jika dalam pengetesan ping ke gateway mendapatkan lebih dari 1100ms diperlukan untuk melakukan input time (detik) yang lebih lama, karna secara default waktu untuk proses download file tiap perangkat yaitu 12 detik. dan dalam pengujian selanjutnya jika memiliki koneksi internet yang stabil maka tidak ada faktor yang dapat menjadi penghambat keberhasilan proses backup otomatisasi. Sebelumnya, semua konfigurasi perangkat disimpan secara manual sebelum diterapkan otomatisasi jaringan. Perbaikan konfigurasi, baik pada saat provisioning maupun pembaruan konfigurasi perangkat secara menyeluruh, dapat dilakukan dengan efisien melalui otomatisasi backup konfigurasi jaringan menggunakan script Python. Terapkan otomatisasi backup konfigurasi jaringan pada perangkat router dan switch cisco yang sudah ada akan meningkatkan efisiensi manajemen jaringan, terutama dalam hal perbaikan Single Point of Failure (SPOF) pada infrastruktur jaringan.

DAFTAR RUJUKAN

- Fahmi, M., Maisyaroh, Komarudin, I., Faizah, S., Fadhilah, I. Otomatisasi Jaringan Menggunakan Script Python Untuk Penyediaan Konfigurasi Internet Dan Manajemen Mikrotik. Vol 8 No 1 (2021): Bina Insani ICT Journal (Juni) 2021.
- Ginting, E. S., and Hadi, I. Pengujian Konfigurasi Otomatis Penambahan Gateway Pada Virtual Router Menggunakan Aplikasi Otomatisasi Jaringan Berbasis Web. Vol. 4, pp. 1126–1131, 2020.
- H. A. Musril, "Desain Virtual Private Network (VPN) Berbasis Open Shortest Path First (OSPF)," *InfoTekJar (Jurnal Nas. Inform. dan Teknol. Jaringan)*, vol. 3, no. 2, pp. 83–88, 2019.
- J. Alex, *Network Automation using Python 3: An Administrator's Handbook*, 1st ed. Independent, 2020
- Kukuh Nugroho, Anggi Dzikri Abrariansyah, Syariful Ikhwan. Perbandingan Kinerja Library Paramiko dan Netmiko Dalam Proses Otomasi Jaringan. *INFOTEKJAR: JURNAL NASIONAL INFORMATIKA DAN TEKNOLOGI JARINGAN- VOL. 5 NO.1 (2020)*

- Luis Geraldo Mauboy, Theophilus Wellem. Studi Perbandingan Library Untuk Implementasi Network Automation Menggunakan Paramiko Dan Netmiko Pada Router Mikrotik, JURIKOM (Jurnal Riset Komputer), Vol. 9 No. 4, Agustus 2022.
- LORI MACVITTIE, F5 Networks and Red Hat, NetOps Meets DevOps: The State of Network Automation. 2018.
- Maharani, Anggita. (2020). Computational thinking dalam pembelajaran matematika menghadapi Era Society 5.0. *Euclid*, 7(2), 86–96.
- Mashudi, M., NETWORK AUTOMATION MENGGUNAKAN BAHASA PEMROGRAMAN PYTHON, (2022), *Jurnal Teknik Industri, Sistem Informasi dan Teknik Informatika*, vol. 9, no. x, pp. xx-xx, 2021. doi:
- M. F. Islami, P. Musa, and M. Lamsani, "Implementation of Network Automation using Ansible to Configure Routing Protocol in Cisco and Mikrotik Router with Raspberry PI," *J. Ilm. Komputasi*, vol. 19, no. 2, pp. 127–134, 2020, doi: 10.32409/jikstik.19.2.80
- M. Neidinger, *Python network programming techniques: 50 real-world recipes to automate infrastructure networks*, 1st ed. Packt Publishing, 2021.
- Netmonk. (2020, Januari 2020). Apa Itu Network Automation. Retrieved from netmonk.id: <https://netmonk.id/apa-itu-network-automation>.
- Nugroho, K., Abrariansyah, A. D., and Ikhwan, S. Perbandingan Kinerja Library Paramiko dan Netmiko Dalam Proses Otomasi Jaringan. Vol. 1, 2020.
- Olva, Maria, Permatasari, Ririt Dwiputri, Majid, Sanusi, Syair, Pratiwi, & Suganda, Afdal. (2021). Pemanfaatan Dasbor pada Pemantauan Data Transaksi Penjualan. *Journal of Engineering, Technology, and Applied Science*, 3(1), 1–15.
- Purba, Nabillah, Yahya, Mhd, & Nurbaiti, Nurbaiti. (2021). Revolusi Industri 4.0: Peran Teknologi dalam Eksistensi Penguasaan Bisnis dan Implementasinya. *Jurnal Perilaku Dan Strategi Bisnis*, 9(2), 91–98.
- Setiya Nugroho, Bambang Pujiarto. Network Automation Pada Beberapa Perangkat Router Menggunakan Pemrograman Python. *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIK)*. Vol. 9, No. 1, Februari 2022.
- Wiryawan, R.A ,Rosyid, N.R. Pengembangan Aplikasi Otomatisasi Administrasi Jaringan Berbasis Website Menggunakan Bahasa Pemrograman Python. *Jurnal SIMETRIS*, Vol. 10 No. 2 November 2019.