



Pengembangan Aplikasi Administrasi Jaringan Berbasis Website

Ilham Aji Pangestu¹, Fani Apriyani², Ayattullah Damalin³, Aries Saifudin⁴

¹⁻⁴ Ilmu Komputer, Teknik Informatika, Universitas Pamulang, Tangerang Selatan, Indonesia
Email: [1wonogiriboy02@gmail.com](mailto:wonogiriboy02@gmail.com), [2faniapriyani26@gmail.com](mailto:faniapriyani26@gmail.com), [3ayatdamalin@gmail.com](mailto:ayatdamalin@gmail.com),
[4aries.saifudin@unpam.ac.id](mailto:aries.saifudin@unpam.ac.id)

Abstrak— Menyediakan perawatan dan konfigurasi jaringan untuk banyak perangkat yang terpasang merupakan tantangan bagi perusahaan. Administrator jaringan harus secara manual masuk ke setiap perangkat secara berurutan, yang memakan waktu dan tidak efisien. Untuk mengatasi hal ini, perusahaan telah mengembangkan sebuah sistem aplikasi berbasis web yang menggunakan otomatisasi jaringan. Metode pengembangan aplikasi cepat digunakan dalam pengembangan aplikasi ini untuk mengidentifikasi persyaratan aplikasi, merancang, membangun, dan mengimplementasikannya. Sistem aplikasi ini menggunakan perpustakaan utama Paramiko untuk menghubungkan dan mengotomatiskan server jaringan ke perangkat-perangkat jaringan melalui protokol SSHv2, serta menggunakan kerangka kerja Django untuk pengembangan web. Pengujian dilakukan menggunakan metode uji kotak hitam. Hasil dari proyek aplikasi ini adalah otomatisasi jaringan untuk konfigurasi perutean statis dan dinamis, pembuatan VLAN, serta pemeliharaan berupa pencadangan dan pemulihan terpusat untuk manajemennya yang lebih baik.

Kata Kunci: Otomatisasi jaringan, Python, Django, Paramiko, Manajemen jaringan, Pemeliharaan rutin, Administrator jaringan, Perangkat jaringan, Sistem aplikasi berbasis web

Abstract— *Providing maintenance and network configuration for many installed devices is a challenge for companies. Network administrators have to manually log in to each device sequentially, which is time consuming and inefficient. To overcome this, the company has developed a web-based application system that uses network automation. Rapid application development methods are used in this application development to identify application requirements, design, build and implement them. This application system uses the main Paramiko library to connect and automate network servers to network devices via the SSHv2 protocol, and uses the Django framework for web development. Testing was carried out using the black box test method. The output of this application project is network automation for static and dynamic routing configuration, VLAN creation, and maintenance in the form of centralized backup and restore for its better management.*

Keywords: *Network automation, Python, Django, Paramiko, Network management, Routine maintenance, Network administrator, Network device, Web based application system*

1. PENDAHULUAN

Perusahaan yang beroperasi di industri jaringan menghadapi tantangan dalam menghadapi jumlah perangkat jaringan yang terpasang yang sangat banyak. Perakitan dan perawatan peralatan menjadi tugas yang kompleks. Metode tradisional dalam mengelola setiap perangkat dari jarak jauh memakan waktu yang lama. Namun, otomatisasi jaringan dapat menjadi solusi untuk mengatasi tugas-tugas kompleks ini. Dengan menggunakan perangkat yang mendukung protokol SSH, otomatisasi jaringan memungkinkan tugas-tugas ini dilakukan dengan lebih cepat dan efisien. Selain itu, otomatisasi jaringan juga memudahkan dalam pemeliharaan jaringan dengan prosedur yang lebih sederhana yang dapat diikuti dan diimplementasikan dalam skala jaringan yang besar.

Otomatisasi jaringan menggunakan logika pemrograman untuk mengelola sumber daya dan layanan web, memungkinkan insinyur jaringan untuk dengan cepat mengkonfigurasi dan mengintegrasikan infrastruktur jaringan (Layer 1-3) dan layanan aplikasi (Layer 4-7). Salah satu library Python yang sering digunakan untuk otomatisasi ini adalah Paramiko. Paramiko adalah sebuah pustaka Python yang menggunakan protokol SSHv2 untuk interaksi dan komunikasi dengan perangkat lain yang mendukung SSHv2. Sebuah penelitian oleh Paul MIHÁILĂ, Titus BĂLAN, yang didasarkan pada penelitian oleh Radu CURPEN dan Florin SANDU berjudul "Network Automation and Abstraction using Python Programming Methods" menyoroti pentingnya otomatisasi dalam jaringan tradisional yang tidak mendukung protokol SDN (Software-Defined Networking) OpenFlow. Pada jaringan tradisional yang masih menggunakan perangkat dengan



model lama yang dibuat oleh berbagai produsen, sulit untuk melakukan konfigurasi atau pengiriman karena perbedaan dalam sintaks dasar dan ketidakmampuan mendukung protokol OpenFlow yang digunakan dalam jaringan SDN. Dalam penelitian ini, otomatisasi dilakukan dengan menggunakan skrip Python untuk membuat VLAN pada tiga switch Cisco yang dikontrol. Dalam implementasinya, dua perpustakaan, yaitu Paramiko dan Netmiko, digunakan. Netmiko digunakan untuk terhubung ke switch dengan menggunakan ConnectHandler, yang juga menggunakan SSH di belakang layar, dan dapat dikonfigurasi untuk jenis perangkat yang akan diatur.

Dalam buku Ahmad Rosid Komarudin "Mengotomasi manajemen jaringan dengan skrip Python" mengimplementasikan otomatisasi menggunakan Paramiko, Netmiko, Pyntc, Napalm, dan Tersedia dan digunakan pada perangkat Cisco. Semua alat yang digunakan bias diimplementasikan pada perangkat Cisco untuk otomatisasi jaringan. Dari pengujian yang dilakukan Alat Napalm dan Ansible adalah fitur yang paling kuat diimplementasikan pada perangkat Cisco. Ansible memiliki modulnya sendiri untuk jaringan dan dukungan banyak penyedia peralatan jaringan. Sedangkan dukungan dari pemasok napalm seperti Arista masih terbatas EOS, Cisco IOS, Cisco IOS-XR, Cisco NX-OS dan Juniper JunOS .

Pada masa lalu, komputasi dalam suatu organisasi dilakukan oleh sebuah model komputer tunggal yang menangani semua tugas. Namun, sekarang model tersebut telah digantikan oleh jaringan komputer, yang terdiri dari sekelompok node atau komputer terpisah yang saling terhubung untuk melaksanakan tugas-tugasnya. Jaringan komputer memungkinkan pertukaran informasi dan komunikasi yang lebih efisien. Seiring dengan perkembangan zaman, kebutuhan akan informasi dan komunikasi semakin meningkat, dan ini mendorong pembentukan jaringan yang semakin kompleks dan meluas, yang tersebar di seluruh dunia. Jaringan yang kompleks ini kemudian dikenal sebagai Interconnected Network, atau yang lebih dikenal dengan sebutan internet.

Implementasi otomatisasi jaringan di MikroTik RouterOS menggunakan Ansible, seperti yang dijelaskan dalam artikel "Mengotomasi Konfigurasi Router Mikrotik dengan Ansible" oleh I Made Bayu Swastika dan I Gede Oka Gartria di majalah Atitama, memberikan kemampuan untuk mengotomatisasi konfigurasi router MikroTik menggunakan fitur Queue Tree MikroTik. Untuk menghubungkan dan mengonfigurasi router melalui API MikroTik, modul API RouterOS di Ansible ditambahkan.

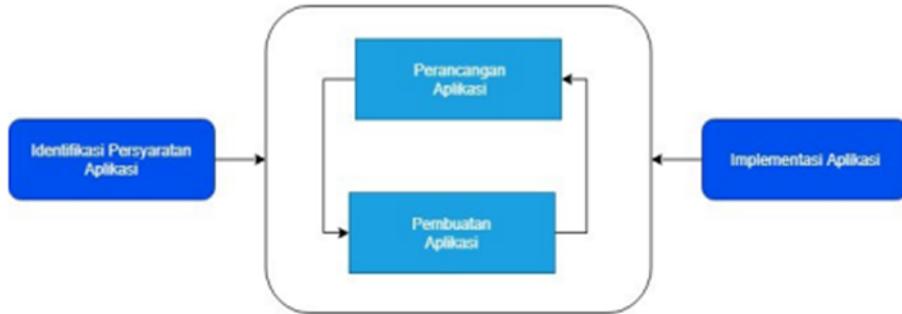
Pengujian dilakukan menggunakan metode black box dengan hasil bahwa perangkat lunak Ansible dapat menentukan beberapa pohon antrian secara berurutan dalam satu file yang dapat dieksekusi dalam waktu pemrosesan sekitar 1 menit untuk 19 konfigurasi pohon antrian. Ansible MikroTik juga dapat digunakan sebagai otomatisasi konfigurasi oleh vendor Cisco, selain MikroTik.

Meskipun demikian, sebagian besar jaringan saat ini masih menggunakan skrip untuk memfasilitasi otomatisasi. Oleh karena itu, perlu dikembangkan suatu aplikasi manajemen otomasi berbasis web yang memiliki keunggulan berupa layar atau antarmuka pengguna (GUI) dan dapat digunakan secara terpusat. Dengan adanya aplikasi ini, proses otomatisasi jaringan dapat lebih mudah diakses dan dikonfigurasi oleh administrator jaringan tanpa perlu terlalu banyak pengetahuan tentang skrip atau baris perintah.

2. METODE

Penelitian ini dimulai dengan melakukan studi literatur yang melibatkan beberapa jurnal dan buku sebagai sumber referensi untuk penelitian. Tujuan dari studi literatur ini adalah untuk memperoleh pemahaman yang mendalam tentang topik penelitian serta melihat perkembangan terkini dalam bidang tersebut.

Setelah studi literatur selesai, penelitian dilanjutkan dengan pengembangan sistem aplikasi menggunakan metode Rapid Application Development (RAD). Metode RAD terdiri dari empat fase utama yang harus dilalui dalam pengembangan aplikasi, yaitu:



Gambar 1. Fase Model RAD

Pada tahap identifikasi kebutuhan aplikasi, diperoleh kebutuhan perangkat keras dan perangkat lunak yang akan digunakan dalam pengembangan aplikasi. Berikut adalah rincian mengenai perangkat keras dan perangkat lunak yang akan digunakan:

a) Perangkat Keras:

- Komputer/laptop sebagai perangkat keras yang digunakan.
- Unit server untuk menjalankan aplikasi.
- Dua unit router Mikrotik RB951-2n.
- Dua unit router Cisco 2911.
- Dua router Cisco 2960.
- Satu sakelar yang tidak dikelola.

b) Perangkat Lunak:

- Ubuntu digunakan sebagai sistem operasi.
- Versi Python 3.6.7 digunakan sebagai server dalam pengembangan aplikasi.
- Paramiko library versi 2.4 digunakan untuk otomatisasi.
- Django digunakan sebagai kerangka kerja web.
- GNS3 versi 2.1 digunakan sebagai alat simulasi.

Selain itu, terdapat persyaratan fungsional dan non-fungsional untuk aplikasi yang dihasilkan:

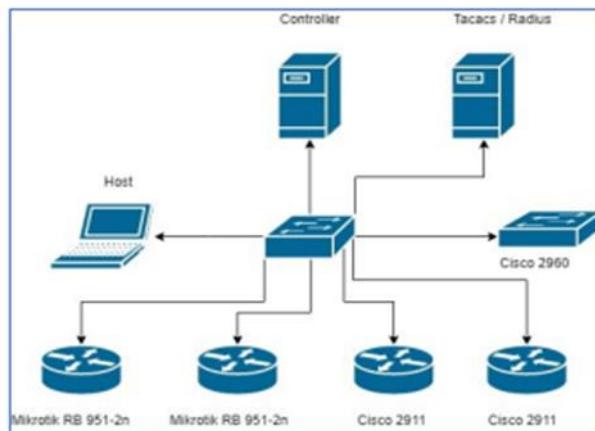
c) Persyaratan Fungsional:

- Sistem aplikasi harus dapat menambahkan konfigurasi perutean statis dan dinamis dalam bentuk OSPF, RIPv1, RIPv2, dan BGP.
- Pengaturan VLAN dapat ditambahkan melalui sistem aplikasi.
- Sistem aplikasi harus mendukung pengaturan pemulihan cadangan.

d) Persyaratan Non-Fungsional:

- Sistem aplikasi harus memiliki antarmuka yang ramah pengguna dan responsif.
- Aplikasi harus dapat digunakan di beberapa browser dan sistem operasi yang berbeda.

Langkah-langkah perancangan aplikasi meliputi desain topologi, model, dan tampilan. Ubuntu Server bertindak sebagai server pengembangan yang menjalankan sistem aplikasi dengan kerangka kerja Django, Paramiko, dan pustaka pendukung lainnya. TACACS+ digunakan untuk manajemen pengguna pada perangkat Cisco.

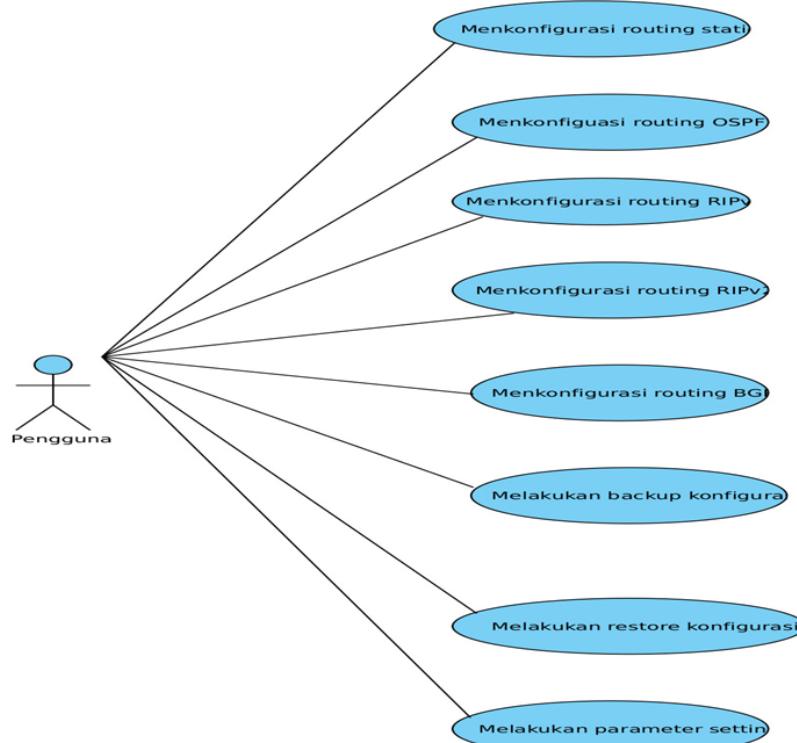


Gambar 2. Topologi Jaringan Aplikasi

Dalam tahap perancangan implementasi otomatisasi jaringan dengan Ansible menggunakan MikroTik RouterOS, diagram UML dapat digunakan untuk membantu memodelkan sistem secara visual. Dalam penelitian ini, digunakan dua jenis diagram UML yang umum digunakan, yaitu diagram use case dan diagram aktivitas.

Diagram use case digunakan untuk menggambarkan interaksi antara pengguna (aktor) dan sistem. Diagram ini mengidentifikasi fungsi-fungsi utama yang dapat dilakukan oleh pengguna dalam sistem. Dalam konteks implementasi ini, beberapa contoh use case yang mungkin termasuk:

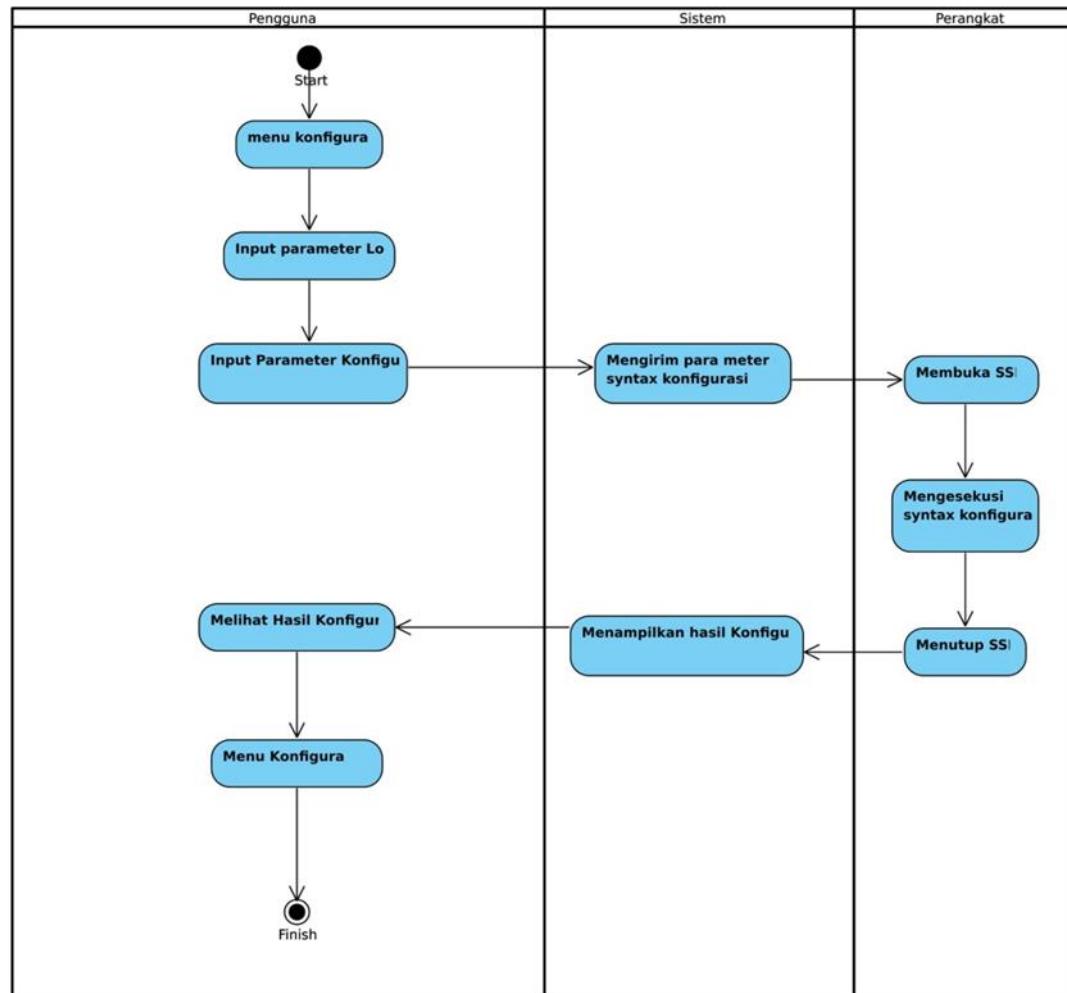
- Mengonfigurasi pohon antrian (queue tree) pada router MikroTik menggunakan Ansible.
- Menghubungkan ke router MikroTik melalui API menggunakan Ansible.
- Mengirimkan konfigurasi otomatis ke router MikroTik menggunakan Ansible.



Gambar 3. Use Case Diagram Aplikasi

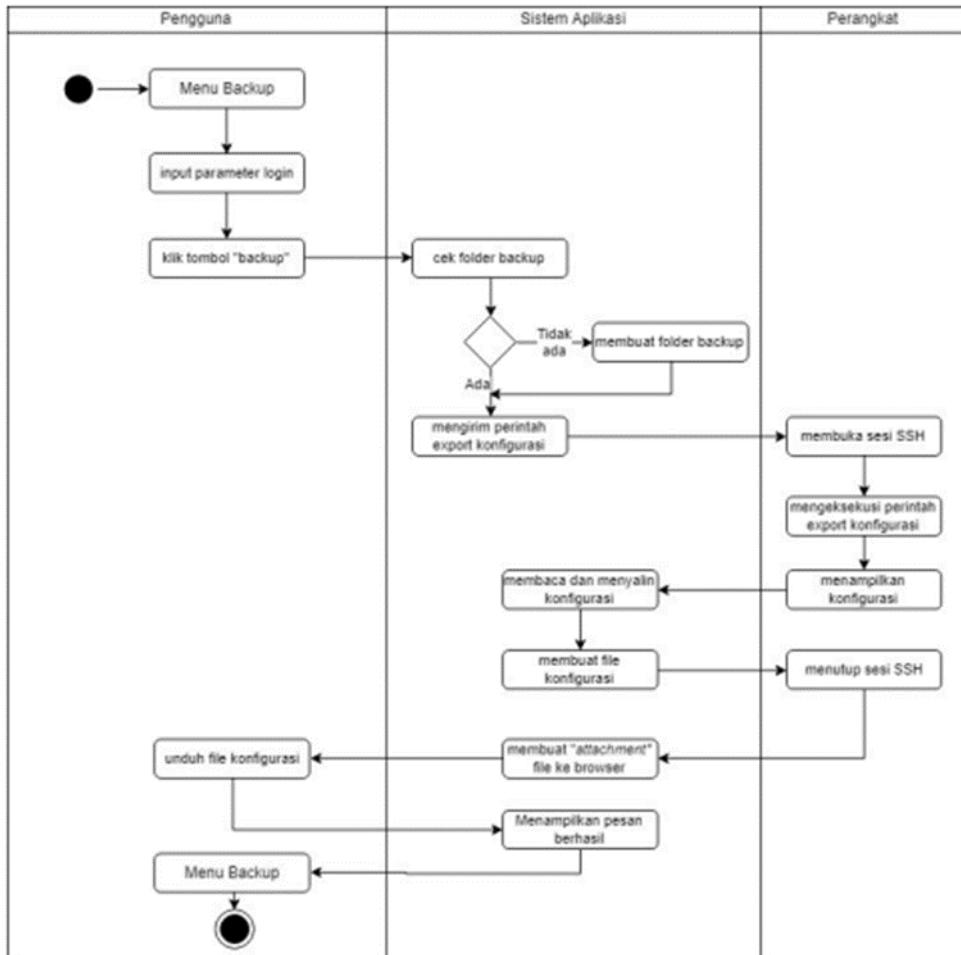
Diagram Aktivitas merupakan diagram UML yang digunakan untuk menggambarkan alur kerja dan fungsi-fungsi yang dapat dilakukan dalam suatu sistem. Saat menggambar diagram aktivitas, semua fungsi yang ada dalam sistem aplikasi ditampilkan sesuai dengan lima fitur utama yang terdapat dalam aplikasi tersebut.

Pada diagram aktivitas fitur Configuration, terdapat tiga objek yang memiliki aktivitas masing-masing. Objek pengguna memiliki fungsi untuk memilih pengaturan dan mengisi parameter Konfigurasi. Sementara itu, objek sistem aplikasi bertanggung jawab dalam mengirimkan konfigurasi dan menjalankan operasi pada Perangkat sesuai target yang ditentukan.



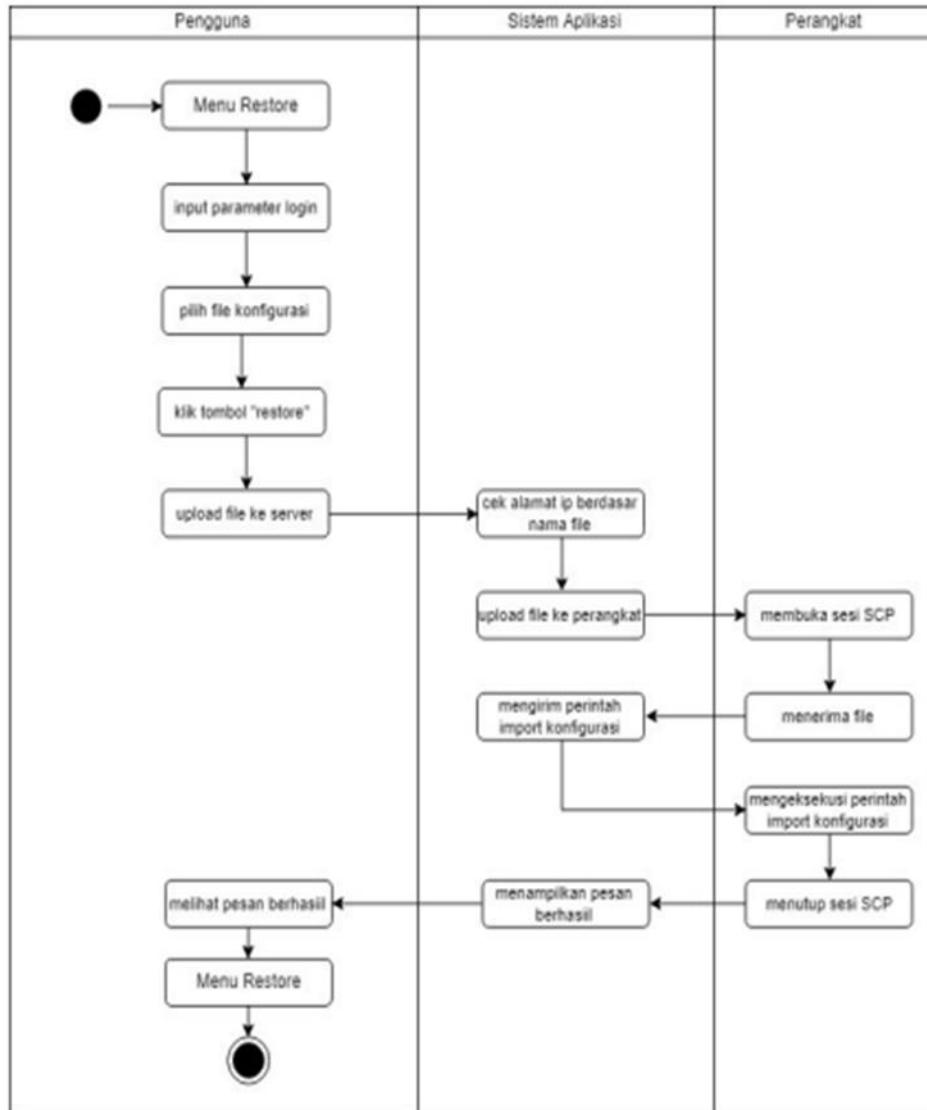
Gambar 4. Activity Diagram Fitur *Configuration*

Pada *activity* diagram fitur Backup, terdapat tiga objek yang memiliki aktivitas masing-masing. Objek Pengguna melakukan tindakan dengan mengisi parameter *login* untuk perangkat target. Objek sistem Aplikasi bertanggung jawab untuk membuat direktori cadangan yang berfungsi sebagai tempat penyimpanan sementara file cadangan, selain dari operasi pengiriman sintaks perakitan. Objek Perangkat digunakan sebagai tempat dilakukannya pencadangan konfigurasi.



Gambar 5. Activity Diagram Fitur Backup

Pada *Activity diagram* fitur *Restore* memiliki tiga objek yang memiliki aktivitas masing-masing. Objek pengguna mempunyai aktivitas dalam mengisi parameter *login* ke perangkat yang dituju dan memilih *file* konfigurasi yang akan digunakan sebagai *file restore*, objek sistem aplikasi memiliki tanggung jawab dalam aktivitas mengirim *file* konfigurasi dan mengirim *syntax* konfigurasi. Perangkat sebagai objek yang akan dilakukan *restore* konfigurasi.



Gambar 6. Activity Diagram Fitur *Restore*

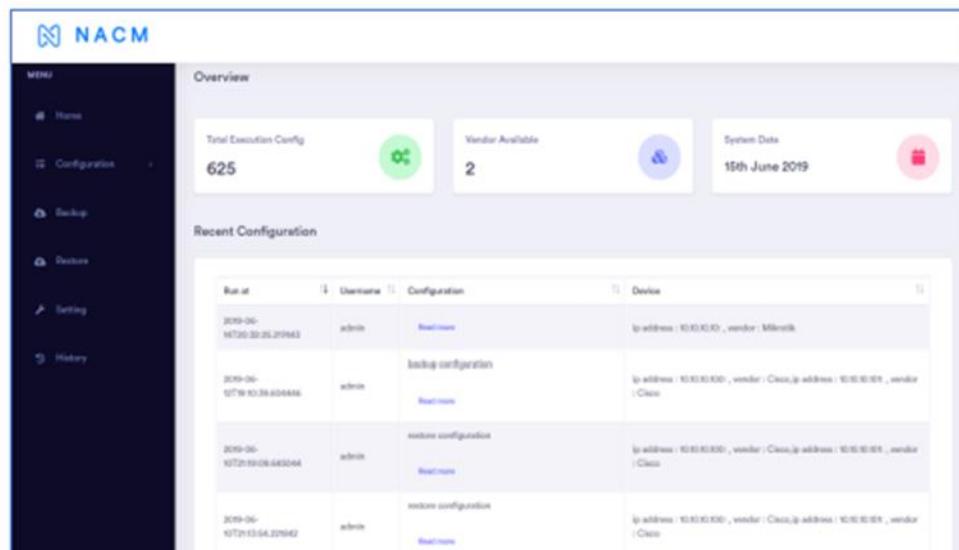
Tahap berikutnya adalah pembuatan aplikasi. Dalam pengembangan aplikasi otomatisasi administrasi jaringan, digunakan library utama Paramiko untuk otomatisasi. Library Paramiko digunakan untuk menghubungkan server dengan perangkat jaringan melalui protokol SSH. Aplikasi ini dikembangkan berbasis website dengan menggunakan web framework Django. Bagian backend sistem aplikasi menggunakan bahasa pemrograman Python, sedangkan bagian frontend menggunakan HTML, *Cascading Style Sheet* (CSS), dan JavaScript.

Setelah itu, dilakukan tahap implementasi aplikasi. Sebelum diimplementasikan secara real, dilakukan simulasi terlebih dahulu menggunakan GNS3. Implementasi pada jaringan yang sebenarnya melibatkan penggunaan router MikroTik RB951, Cisco 2911, dan switch Cisco 2960. Pengujian dilakukan sesuai dengan skenario yang telah dibuat. Metode pengujian yang digunakan adalah Black-Box Testing untuk memastikan bahwa fungsi-fungsi sistem aplikasi berjalan dengan baik atau tidak.

3. ANALISA DAN PEMBAHASAN

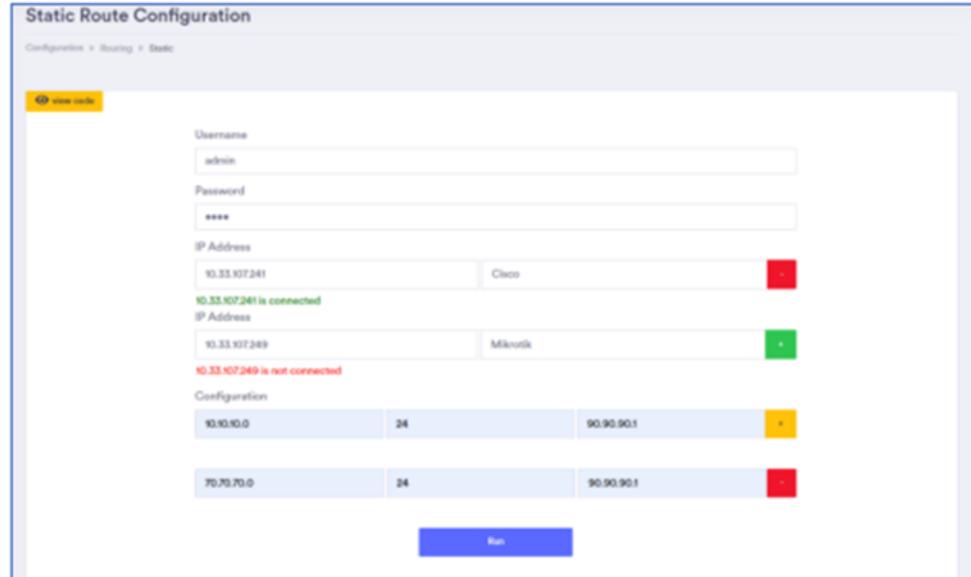
Sistem aplikasi yang sedang dikembangkan diberi nama *Network Automation Configuration Management* (NACM). Antarmuka sistem aplikasi didesain berdasarkan rancangan tampilan yang telah dibuat sebelumnya. Antarmuka tersebut terdiri dari daftar menu, halaman utama sistem aplikasi, serta submenu seperti Routing Static, Routing Dynamic, Routing BGP, Vlan, Codebased, Backup, Restore, dan History. Dalam pengembangan antarmuka aplikasi, digunakan framework CSS Bootstrap untuk mempermudah pembuatan tampilan. Antarmuka sistem aplikasi yang dihasilkan responsif, artinya tampilannya akan menyesuaikan dengan ukuran layar pengguna.

Halaman utama atau home page sistem aplikasi memberikan gambaran umum dengan menampilkan informasi tentang jumlah pengaturan yang telah dilakukan, jumlah penyedia yang telah melakukan pengaturan, dan tempat penyimpanan data yang diambil dari database SQLite. Bagian "Recent Setting" menampilkan informasi tentang 5 konfigurasi terakhir yang selesai dilakukan. Informasi tersebut mencakup tahun, bulan, tanggal, dan waktu pelaksanaan konfigurasi, serta username yang melakukan konfigurasi. Informasi yang ditampilkan mencakup pengaturan perutean, VLAN, berbasis kode, pencadangan, dan pemulihan. Terdapat juga informasi tentang perangkat konfigurasi, seperti alamat IP dan jenis perangkat pemasok peralatan. Halaman utama sistem aplikasi berfungsi sebagai log sederhana yang memungkinkan pelacakan pengguna yang melakukan konfigurasi pada perangkat jaringan menggunakan sistem aplikasi.



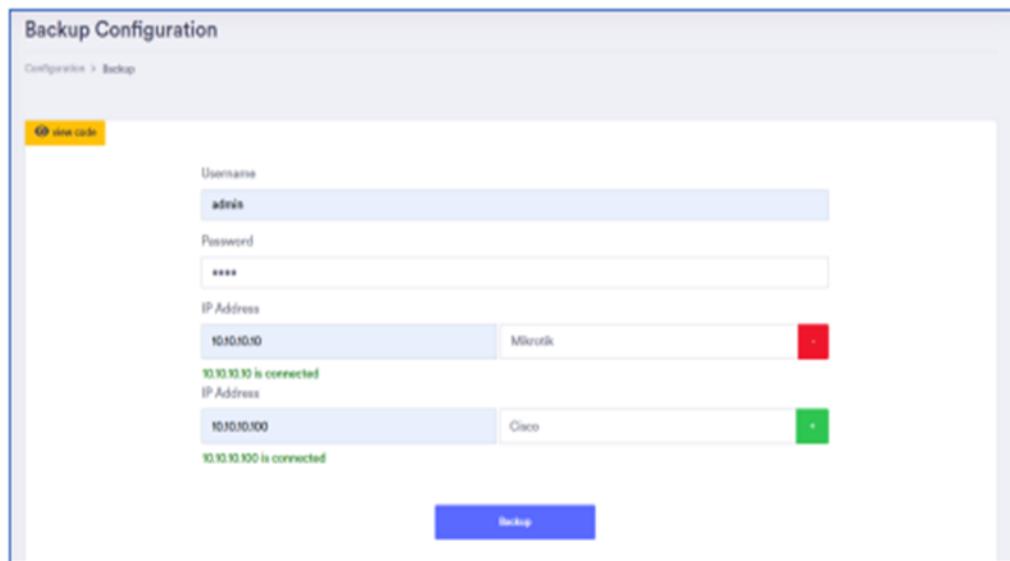
Gambar 7. Tampilan Halaman Utama Aplikasi

Fungsi konfigurasi terdiri dari beberapa submenu, seperti Static, Dynamic, BGP, VLAN, dan Routing berbasis kode. Setiap submenu memiliki kolom yang memungkinkan pengguna untuk memasukkan nama pengguna dan kata sandi SSH, serta kolom untuk mengisi alamat IP dan produsen perangkat target. Setiap submenu juga memiliki kolom-kolom yang sesuai dengan parameter konfigurasi yang digunakan. Misalnya, submenu Routing Statis memiliki kolom untuk parameter routing statis, seperti tujuan, prefiks, dan gateway.



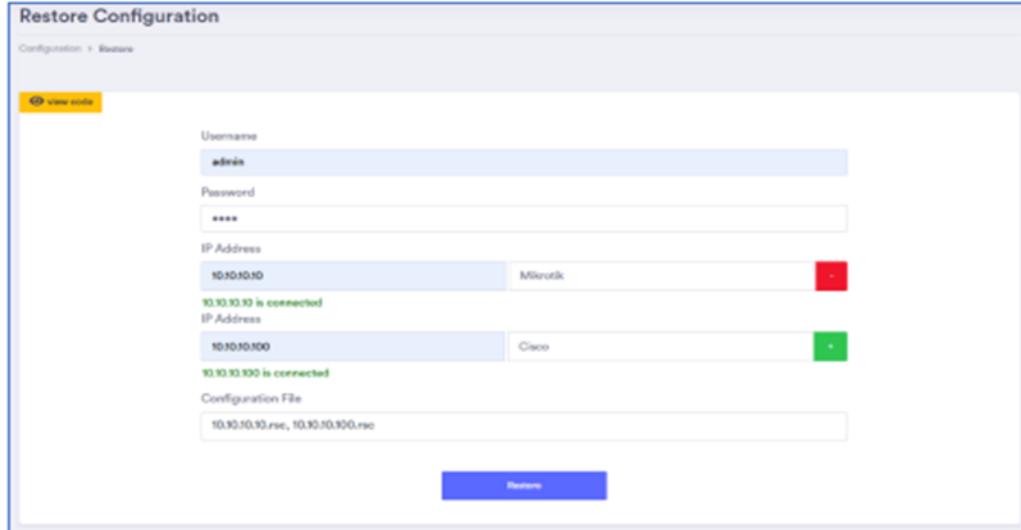
Gambar 8. Tampilan Halaman Fitur *Configuration Routing Static*

Fitur backup memiliki parameter input berupa kolom untuk memasukkan username, password, alamat IP, dan vendor. Setelah pengguna menekan tombol "Backup", akan muncul sebuah jendela pop-up yang menampilkan informasi tentang proses pengunduhan file konfigurasi. Pada jendela pop-up tersebut terdapat tombol "Finish" yang digunakan untuk memuat ulang halaman setelah proses pengunduhan selesai.



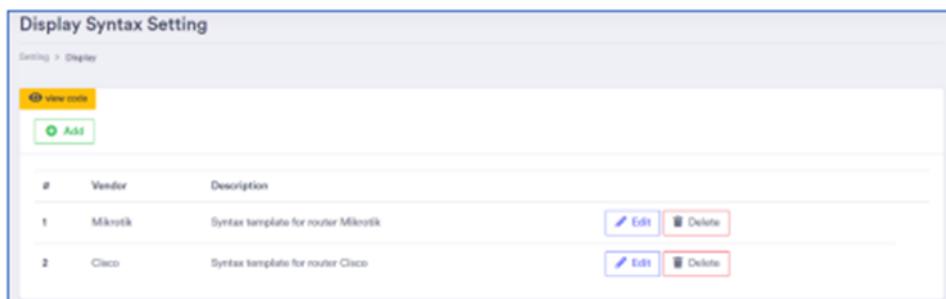
Gambar 9. Tampilan Halaman Fitur Backup

Fitur *restore* dilengkapi dengan kolom "Configuration File" yang memungkinkan pengguna untuk memilih file konfigurasi yang akan digunakan sebagai file *restore* pada perangkat tujuan. Ketika pengguna menekan kolom tersebut, akan muncul sebuah jendela file yang memungkinkan pengguna untuk memilih file konfigurasi yang diinginkan.



Gambar 10. Tampilan Halaman Fitur *Restore*

Fitur Setting menampilkan informasi tentang vendor yang telah diatur untuk parameter sintaks konfigurasi perangkat. Informasi yang ditampilkan meliputi nama vendor, deskripsi, serta tombol add untuk menambahkan setting vendor baru, tombol edit untuk mengubah parameter sintaks vendor, dan tombol delete untuk menghapus setting vendor.



Gambar 11. Tampilan Halaman Fitur *Setting*

4. KESIMPULAN

Berdasarkan data yang diperoleh dan analisis yang telah dilakukan dalam penelitian ini tentang Pengembangan Aplikasi Otomatisasi Administrasi Jaringan Berbasis Website Menggunakan Bahasa Pemrograman Python, dapat disimpulkan hal-hal berikut:

- Telah berhasil dikembangkan aplikasi otomatisasi administrasi jaringan berbasis website menggunakan library utama Paramiko dan web framework Django.
- Pengembangan aplikasi menghasilkan lima fitur utama, yaitu konfigurasi Routing, Vlan, Backup, Restore, dan Setting. Melalui fitur-fitur ini, aplikasi dapat melakukan fungsi utama dalam melakukan konfigurasi administrasi jaringan, seperti routing statis, dynamic OSPF, RIPv1, RIPv2, BGP, serta backup dan restore konfigurasi.
- Pengujian aplikasi menggunakan metode Black-box Testing menunjukkan bahwa semua fungsi dalam aplikasi yang dikembangkan pada penelitian ini berjalan dengan baik dan berhasil diimplementasikan pada vendor yang berbeda, seperti Cisco dan Mikrotik.



REFERENCES

- Chou, E. 2017. "Mastering Python Networking." 2nd ed. packt. Birmingham.
- Jurnal, J. T. I. K. (2021). 2021. "Black-Box Testing: Analisis Kualitas Aplikasi Source Code Bank Programming." *Jurnal JTIK (Jurnal Teknologi Informasi dan Komunikasi)* 5: 1.
- Komarudin, A. 2018. "Otomatisasi Administrasi Jaringan Dengan Script Python." *Jasakom. Jakarta.*
- Kose, M. A., & Ozkaya, M. "Owards Extending Uml's Activity Diagram for the Architectural Modeling, Analysis, and Implementation." In *2020 15th conference on computer science and information systems (fedcsis)* pp. 639-64.
- Mihailă, P., Bălan, T., Curpen, R. dan Sandu, F. 2017. "Network Automation and Abstraction Using Python Programming Methods." s. *Electronics and Computer Department. Transilvania University.*
- Mirdita, M., Steinegger, M., & Söding, J. 2019. "MMseqs2 Desktop and Local Web Server App for Fast, Interactive Sequence Searches." *Bioinformatics*, 35.16: 2856-2858.
- Swastika, I., Atitama, G. 2017. "Otomatisasi Konfigurasi Mikrotik Router Menggunakan Software Ansible." *Fakultas Matematika dan Ilmu Pengetahuan Alam. Universitas Udayana.*
- Wijaya, J. 2018. "Network Automation Using Ansible for Cisco Router Basic Configuration." *Teknik Elektro dan Informatika. Institut Teknologi Bandung*
- Anriani, I. A. (2022). Aplikasi Pelayanan Administrasi Kependudukan dan Penetapan APBDES Berbasis Web pada Desa Kampung Baru. *Jurnal Ilmiah Informatika.*
- Handriyanto, D. F. (2019). Kajian Penggunaan Mikrotik Router OS™ sebagai Router pada Jaringan Komputer.