

## **WEBSITE NETWORK AUTOMATION DESIGN AND IMPLEMENTATION IN RT RW NET SENDEN DUSUN MAGELANG WITH DJANGO FRAMEWORK**

Yoel Chandra Eka Paksi<sup>\*1</sup>, Indrastanti R. Widiyanti<sup>2</sup>

<sup>1,2</sup>Teknik Informatika, Fakultas Teknologi Informasi, Universitas Kristen Satya Wacana, Indonesia  
Email: [672018296@student.uksw.edu](mailto:672018296@student.uksw.edu), [indrastanti@uksw.edu](mailto:indrastanti@uksw.edu)

(Naskah masuk: 29 Mei 2022, Revisi: 12 Juni 2022, diterbitkan: 20 Oktober 2022)

### **Abstract**

*RT RW NET is part of the Internet facilities owned by the hamlet of Senden Magelang, whose purpose is to provide cheap Internet facilities. However, behind the existing facilities several problems emerged, based on a statement from the hamlet of sending that the problem that arose was the configuration process for adding new users, which took a long time due to manual configuration in each RT (Rukun Tetangga) and RW (Rukun Warga) alley. Therefore, in this study, a website-based network automation application was created using the Django framework and developed using the Waterfall method, thus providing easy access and fast configuration when configuring new users. After designing the network automation website, the website will be operated by the network administrator, who hopes to simplify and shorten the time for new user configurations.*

**Keywords:** *network automation, framework Django, Waterfall, RT RW NET, internet, website*

## **PERANCANGAN DAN IMPLEMENTASI WEBSITE NETWORK AUTOMATION PADA RT RW NET DUSUN SENDEN MAGELANG DENGAN FRAMEWORK DJANGO**

### **Abstrak**

RT RW NET merupakan salah satu bagian dari fasilitas *internet* yang dimiliki oleh dusun Senden Magelang, yang memiliki tujuan untuk memberikan fasilitas *internet* yang murah. Namun dibalik fasilitas yang ada muncul beberapa permasalahan, berdasarkan pernyataan dari kepala dusun Senden bahwa permasalahan yang muncul adalah proses konfigurasi penambahan *user* baru yang memakan waktu yang cukup lama dikarenakan konfigurasi yang masih manual di setiap gang RT (Rukun Tetangga) dan RW (Rukun Warga). Oleh karena itu pada penelitian kali ini dibuat sebuah aplikasi *network automation* berbasis *web* yang menggunakan *framework Django* dan dikembangkan menggunakan metode *Waterfall*, sehingga memberikan kemudahan dalam mengakses serta konfigurasi yang cepat saat melakukan konfigurasi *user* baru. Setelah melakukan perancangan dari *website network automation* ini maka *website* akan dioperasikan oleh administrator jaringan, yang harapannya dapat mempermudah dan mempersingkat waktu konfigurasi *user* baru.

**Kata kunci:** *network automation, framework Django, Waterfall, RT RW NET, internet, website*

### **1. PENDAHULUAN**

Kebutuhan terhadap jaringan internet dewasa ini semakin meningkat, dibuktikan dengan banyaknya masyarakat yang menggunakan jaringan internet untuk melakukan komunikasi antar *device* [1]. Oleh karena itu kebutuhan akan internet WIFI (*wireless fidelity*) menjadi meningkat, Sehingga memunculkan inisiatif untuk merancang sebuah RT RW NET dusun Senden yang merupakan solusi yang baik karena harganya yang terjangkau dan dapat menjangkau ke masyarakat dari kelas bawah sampai dengan kelas atas [2]. Seperti di dusun Senden Magelang berhasil merancang sebuah RT RW NET sejak 2020 dengan tujuan untuk dapat

memudahkan masyarakat dari dusun Senden ini sendiri sehingga dapat mengakses internet dengan murah pada masa *pandemic corona virus*. RT RW NET dusun Senden Magelang sendiri menggunakan perangkat jaringan *Mikrotik* dan *Cisco* sebagai alat jaringan untuk melakukan *routing* dan *switching* dibuktikan dengan adanya perangkat *Mikrotik router RB 941* dan *Cisco Switch Catalyst 2960* pada setiap titik rumah, gang RT (Rukun Tetangga) dan gang RW (Rukun Warga).

Dibalik keberhasilan dusun Senden Magelang dalam menciptakan akses internet dengan murah ini terdapat beberapa permasalahan. Berdasarkan kepala dusun Senden Magelang permasalahan yang muncul di antaranya adalah pada saat proses konfigurasi

penambahan *user* baru yang dilakukan cukup lama dikarenakan konfigurasi yang masih manual di setiap gang RT (Rukun Tetangga), sehingga mengakibatkan instalasi *user* baru yang membutuhkan waktu yang cepat dalam menggunakan jaringan *internet* tidak dapat secara langsung menggunakan jaringan *internet* tersebut. Maka dari permasalahan itu diperlukan sebuah aplikasi yang dapat membantu persoalan yang ada yaitu sebuah *network automation* berbasis *website* [3][4]. Untuk proses perancangan dari *website network automation* menggunakan *framework Django* dengan *programming Python*, untuk penyimpanan dari *database* menggunakan *mysql server* sehingga data dari administrator dan data dari *network automation* dapat disimpan dengan baik dan benar. Lalu untuk menghubungkan antara *website* ke perangkat jaringan seperti perangkat *Mikrotik RB 941*, maka dapat menggunakan *library Paramiko, Netmiko* dari *Python* dan *API RouterOS* [5][6].

*Website network automation* terdiri dari beberapa bagian diantaranya adalah *login*, pada bagian ini berfungsi untuk memberikan autentikasi pada akun. Selanjutnya adalah bagian *dashboard* di bagian ini memiliki fungsi untuk memberikan informasi kepada administrator tentang bagaimana menggunakan *website network automation*. Untuk bagian lainnya adalah *input device*, pada bagian ini memiliki fungsi untuk memasukkan data *device* yang nantinya akan digunakan untuk proses *network automation*. Untuk bagian selanjutnya adalah bagian cek *device* yang berfungsi untuk melakukan pengecekan data *device* yang sudah masuk. Kemudian, halaman konfigurasi *user* baru pemilihan IP dari *device* dan pemilihan konfigurasi untuk setiap *device*. Jika sudah, maka data pemilihan IP dan data pemilihan konfigurasi akan diproses dengan program *network automation*. Selanjutnya konfigurasi *command line interface* (CLI), dibagian ini berbeda dengan bagian konfigurasi *user* baru karena pada konfigurasi *command line interface* (CLI) bisa memasukkan *script* dari *Mikrotik* ke dalam *form* dan memilih IP *device* yang akan dieksekusi program. Di akhir bagian dari *website network automation* adalah *log* yang memiliki fungsi untuk mengecek bagian konfigurasi *user* baru.

Perancangan dari *website network automation* ini menggunakan metode pengembangan *Waterfall* dengan menggunakan pemrograman *Python* dan *framework Django*. Untuk pengujian sistem dilakukan dengan menggunakan metode *Black Box*.

## 2. METODE PENELITIAN

Metode pengembangan aplikasi adalah metode yang dipakai oleh seorang *developer* aplikasi untuk dapat mengembangkan sebuah aplikasi yang dibuat. Seperti metode pengembangan aplikasi *Waterfall*. Metode *Waterfall* merupakan sebuah metode yang digunakan untuk mengembangkan sebuah aplikasi, yang terdiri dari analisis, desain, pengkodean dan

pengujian [7]. Model metode *Waterfall* ditunjukkan oleh Gambar 1.



Gambar 1. Model metode *Waterfall* (Sumber : Purnia, 2019)

### 2.1. Analisis Kebutuhan

Analisis kebutuhan digunakan untuk mempersiapkan kebutuhan fitur dari perancangan sebuah aplikasi yang ada, sehingga dapat membantu dalam melakukan proses perancangan dari sebuah aplikasi. Pada penelitian ini dilakukan beberapa proses analisis kebutuhan diantaranya adalah melakukan observasi dengan menanyakan kepada administrator yang ada di RT RW NET tentang beberapa hal yang harus diperlukan dalam melakukan sebuah perancangan dari aplikasi ini. Setelah melakukan observasi dan melakukan beberapa pertanyaan kepada administrator maka selanjutnya adalah mendalami hasil dari observasi. Untuk hasil analisis dari observasi yang pertama terkait analisis kebutuhan perancangan dari aplikasi *network automation* adalah melakukan instalasi *Python* pada *operating system* ini bertujuan agar saat proses pengkodean *Python* sehingga dapat berjalan dengan baik [8]. Setelah itu yang kedua adalah instalasi dari *Django* yang berfungsi sebagai *framework* untuk proses pemrograman dari *Python* [9].

Tahap yang ketiga adalah melakukan instalasi *library* dari *Python* untuk dapat melakukan *remote connection* dari *Mikrotik* seperti *Netmiko, Paramiko* dan *API RouterOS*. Untuk penggunaannya *Netmiko* digunakan untuk beberapa perintah yang langsung dieksekusi jadi CLI (*Command line Interface*) sudah ada di dalam pemrograman *Python* [6]. Lalu yang keempat dalam penggunaan dari *Paramiko* sebaliknya dengan *Netmiko*, saat proses eksekusinya perlu memasukkan terlebih dahulu CLI (*Command line Interface*) dan baru melakukan eksekusi di dalam pemrograman *Python* [5]. Sedangkan untuk yang kelima *API RouterOS* di penelitian ini digunakan untuk melakukan olah data dari *Mikrotik* itu sendiri [10]. Jika sudah maka yang keenam adalah melakukan instalasi dari *mysql* ini berfungsi untuk tempat menyimpan dari *database* [11]. Tahap yang ketujuh adalah melakukan instalasi *web browser* yang berfungsi untuk tempat proses *testing* dari *website* [12].

Jika sudah maka tahap yang ke delapan adalah melakukan instalasi dari *Bootstrap*. Ini berfungsi untuk memberikan kemudahan dalam menyusun

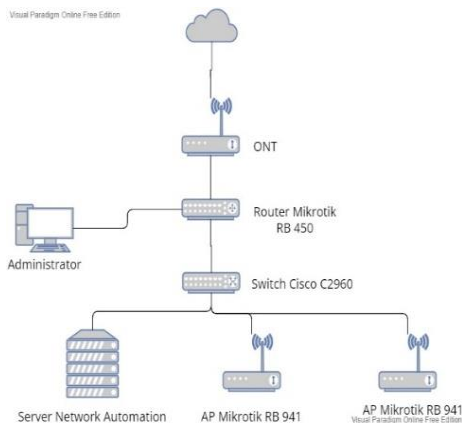
tampilan dari website dan agar tampilan yang dihasilkan lebih menarik dan *responsible* [13]. Selanjutnya jika proses observasi sudah berhasil dan dapat disimpulkan maka proses selanjutnya adalah melakukan perancangan dari *requirement specification*, ini berisi tentang beberapa spesifikasi yang diperlukan dalam melakukan perancangan dari sebuah aplikasi. Jika proses perancangan *requirement specification* sudah selesai maka itu akan menjadi sebuah acuan dalam melakukan proses perancangan dari sebuah aplikasi.

**2.2. Desain**

Merancang sebuah desain merupakan tahapan dalam perencanaan dan pengembangan berdasarkan informasi dari riset pengguna. Pada penelitian kali ini diagram yang memberikan gambaran terhadap perencanaan dan pengembangan aplikasi adalah pertama *activity diagram* yang berfungsi untuk mencari tau *use case* dengan cara menggambarkan aksi yang perlu dilakukan [14], kedua *use diagram* ini berfungsi untuk memberikan gambaran tentang hubungan *system* dan *actor* [15] dan topologi jaringan RT RW NET dusun senden ini memiliki fungsi untuk memberikan gambaran tentang topologi jaringan RT RW NET dusun senden Magelang.

**2.2.1. Topologi Jaringan**

Dalam merancang sebuah aplikasi *website network automation* dusun senden ini membutuhkan gambaran dari topologi jaringan, yang berfungsi untuk memberikan gambaran dalam melakukan proses pengetesan dan proses memasukkan *website* ke dalam *server* [5]. Untuk bentuk topologinya ditunjukkan pada gambar 4.



Gambar 2. Topologi jaringan

Pada gambar ini dapat digambarkan bahwa untuk topologi dari RT RW NET dusun senden terdiri dari beberapa tingkatan yang pertama adalah perangkat pusat, kedua distribusi, *access* dan *end device*, yang dimana untuk perangkat pada bagian pusat menggunakan perangkat *ONT*, lalu untuk distribusi menggunakan perangkat *Mikrotik RB450*, selanjutnya untuk perangkat *access* menggunakan *Cisco switch c2960* dan untuk dibagian paling

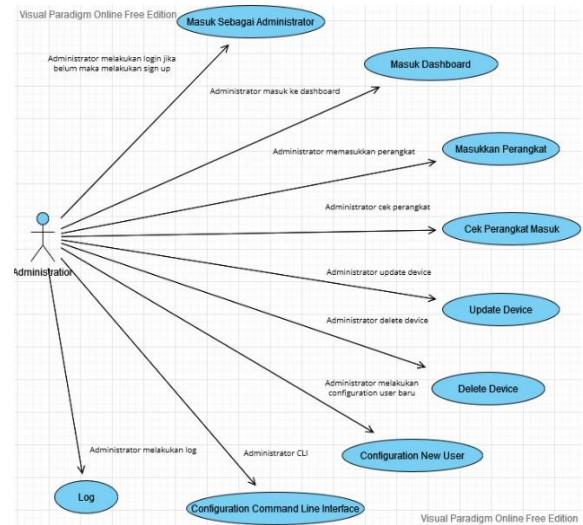
bawah pada *end deviceny* menggunakan *access point Mikrotik RB941*.

**2.2.2. UML (Unified Modelling Language)**

Perancangan dari sebuah system network automation berbasis website ini menggunakan UML (*Unified Modelling Language*), yang terdiri dari *activity diagram*, *class diagram* dan *use case diagram* [16][17].

**2.2.2.1. Use Case Diagram Administrator**

*Use case diagram* di penelitian ini terdiri dari beberapa tahapan yang dapat digambarkan pada gambar 3 *use case diagram* ini.



Gambar 3. Use case diagram

Pada gambar tersebut dapat diketahui bahwa untuk tahapan awal dari *use case diagram* ini adalah melakukan proses autentikasi seperti proses *login* dan melakukan *sign up*. Jika sudah maka admin dapat masuk ke dalam halaman *dashboard* di halaman ini admin dapat melakukan proses pembelajaran untuk menggunakan *website*. Bila di halaman *dashboard* sudah maka administrator dapat melakukan proses memasukkan data *router* di halaman *input* perangkat dan dilanjutkan masuk ke dalam halaman cek perangkat. Di halaman ini administrator dapat melakukan proses pengecekan pada perangkat yang sudah dimasukkan. Pada halaman cek perangkat maka akan diperlihatkan tampilan dari *update* dan *delete* dari perangkat tersebut bila ingin menghapus data admin dapat melakukan klik hapus data yang ada dan jika admin ingin melakukan update data maka admin dapat melakukan perubahan data dengan melakukan klik update perangkat yang ada. Lalu bila sudah maka admin dapat melakukan konfigurasi *user* baru dengan masuk ke dalam konfigurasi *user* baru dan memilih ip dan jenis konfigurasi yang akan digunakan. Selanjutnya adalah melakukan konfigurasi *CLI (Command Line Interface)* di mana konfigurasi ini juga memasukkan ip dan memasukkan konfigurasi *CLI (Command Line Interface)* kedalam form yang sudah diberikan,



bahasa pemrograman *Python* untuk menjalankan beberapa perintah pengkodean yang ada dan dibantu dengan *library* untuk menjalankan proses *network automation* yaitu *Netmiko*, *Paramiko* dan *RouterOS API Mikrotik* sehingga dapat melakukan pemrosesan konfigurasi *user* baru. Lalu dalam pemrosesannya dibantu dengan sebuah *framework* bernama *django*, dengan *framework* ini maka dapat memberikan sebuah kemudahan dalam melakukan pemrosesan *Python* dan *database*. Selanjutnya untuk pemrosesan data menggunakan *mysql* dengan *mysql* ini maka data yang disimpan tidak hanya dapat disimpan namun juga dapat diolah.

Mengenai macam-macam perangkat yang akan digunakan dipenelitian ini adalah sebagai berikut:

a. Hardware

Intel®Core™i5-7200 CPU @ 2.50GHz (4 CPUs), 2.7GHz, HDD 1 TB, RAM 4 GB, Model GPU intel®HD Graphics 620.

b. Software

Sistem operasi Windows 10, XAMPP, CMD untuk run *Djangoproject*, Sublime Text.

### 2.4. Pengujian

Pada tahapan pengujian merupakan tahapan setelah melakukan *development* dari aplikasi. Ini berfungsi untuk menguji apakah aplikasi yang dibuat sudah dapat bekerja dengan baik atau sebaliknya. Dipenelitian *website network automation* ini menggunakan metode *Black Box testing*. Untuk cara pengetesan yang digunakan *Black Box testing* adalah dengan menguji fungsional dari aplikasi, dan di akhir akan dibuat sebuah table hasil *testing* bila sudah selesai melakukan proses *testing*. Jika pengujian dapat berjalan dengan baik maka aplikasi dinyatakan dapat bekerja dengan baik jika sebaliknya maka tidak dapat berjalan dengan baik [5] [18].

## 3. HASIL DAN PEMBAHASAN

Dalam mengembangkan sebuah sistem *network automation* berbasis *website* ini menggunakan metode yang bernama *Waterfall*, dengan metode ini dapat menghasilkan sebuah *website* dengan performa yang cukup baik karena menggunakan pendekatan yang cukup sistematis dalam pengembangan sebuah sistem. Dalam oprasionalnya *website* ini digunakan oleh seorang administrator jaringan yang memiliki tanggungjawab untuk melakukan proses konfigurasi *user* baru.

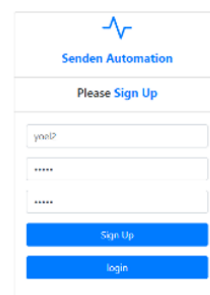
### 3.1. Hasil Tampilan Website

Hasil tampilan *website* merupakan hasil dari perancangan dari *webiste network automation* pada bagian ini akan dikelompokkan kedalam beberapa bagian yang pertama adalah tampilan *sign up*, *login*, setting awal, konfigurasi *user* otomatis dan *log*.

#### 3.1.1 Tampilan Halaman Sign Up

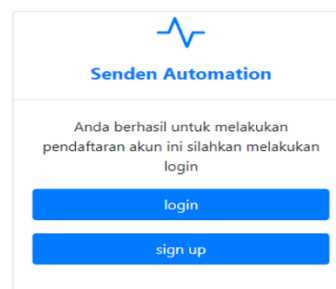
Pada halaman *sign up* terdiri dari beberapa bagian. Yang pertama adalah bagian *sign up* pada

bagian ini berfungsi untuk mendaftarkan akun dengan memasukkan *inputan* ke dalam *form*. Untuk tampilan login dapat ditunjukkan pada gambar 6.



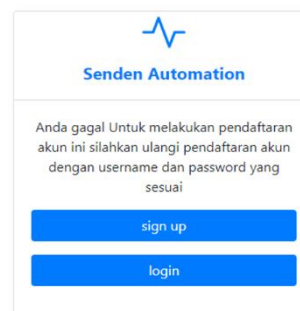
Gambar 6. Tampilan *sign up*

Terkait penggunaan dari tampilan *sign up* admin dapat memasukkan *username password* dan verifikasi *password* dengan benar sehingga dapat dilanjutkan ke tahapan validasi. Yang kedua adalah tampilan *sign up* pada tampilan *sign up* memiliki fungsi untuk untuk melakukan validasi apakah inputan sudah sesuai atau belum. Mengenai tampilan validasi dapat ditunjukkan pada gambar 7 dan gambar 8.



Gambar 7. Tampilan konfirmasi *sign up* berhasil

Di gambar 7 ini memberitahu kepada admin bahwa proses validasi sudah berhasil, ditunjukkan dengan muncul pesan bahwa proses validasi sudah berhasil dan admin dapat kembali melakukan proses *sign up* untuk akun baru atau masuk kedalam proses *login* dimana pada tampilan ini admin dapat melakukan proses validasi untuk masuk kedalam *dashboard*.



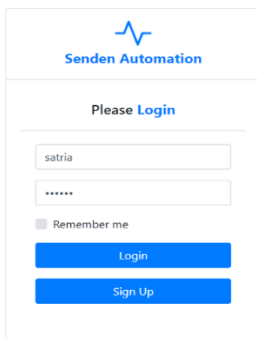
Gambar 8. Tampilan konfirmasi *sign up* gagal

Berbeda dengan gambar 7 pada gambar 8 admin diarahkan ke tampilan gagal melakukan validasi, ditunjukkan dengan pesan bahwa proses validasi

gagal. Maka dari pesan ini admin dapat melanjutkan ke proses *login* jika ada akun yang lain yang bisa digunakan atau membuat akun kembali dengan masuk ke tampilan *sign up*.

### 3.1.2. Tampilan Halaman *Login* admin

Tampilan halaman *login* memiliki fungsi sebagai pengaman atau sebagai autentikasi administrator pada sebuah *website network automation*, sehingga pengguna yang sedang menggunakan *website* memanglah seorang administrator dari *website* itu sendiri. Untuk tampilan *login* dapat ditunjukkan pada gambar 8.

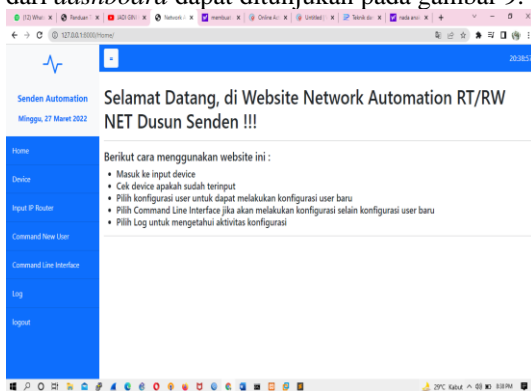


Gambar 9. Tampilan *login*

Untuk proses dalam menggunakan halaman *login* adalah dengan memasukkan *username* dan *password* sehingga jika berhasil maka akan dipindahkan ke halaman *dashboard* utama. Jika gagal dalam proses *login* maka administrator perlu melakukan koreksi terhadap akun yang administrator punya sehingga dapat di akses. Jika masih juga gagal maka administrator dapat mendaftarkan terlebih dahulu di halaman *sign up*.

### 3.1.3 Tampilan Halaman *Dashboard*

Halaman *dashboard* merupakan tampilan setelah melakukan proses *login*. Fungsi dari tampilan ini adalah untuk memberikan pengarahan pada admin bagaimana cara untuk menggunakan *website network automation* ini. Mengenai tampilan dari *dashboard* dapat ditunjukkan pada gambar 9.



Gambar 10. Tampilan *dashboard*

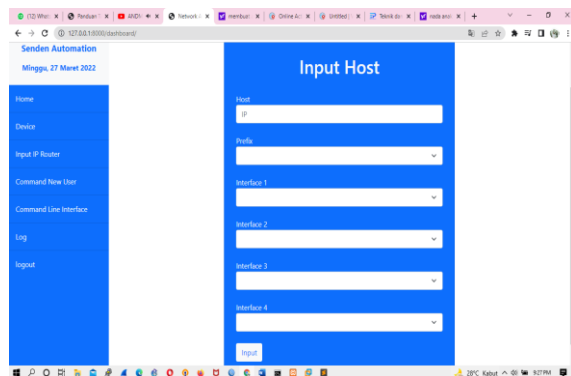
Untuk proses dari penggunaan dari tampilan ini bisa lewat akses *link dashboard* dan lewat *login*.

### 3.1.4. Tampilan *Setting Awal*

Proses *setting* awal berisi tentang persiapan perangkat sebelum proses konfigurasi otomatis *user* baru. Untuk tampilan dari halamannya terdiri dari beberapa bagian yaitu bagian *input device*, *update* perangkat dan yang terakhir adalah *delete* perangkat.

#### 3.1.4.1. Tampilan *Input Device*

Pada tampilan ini berisi tentang input dari *device*, ini berfungsi untuk mendukung proses konfigurasi *user* baru. Terkait gambar *input* dapat ditunjukkan pada gambar 10.

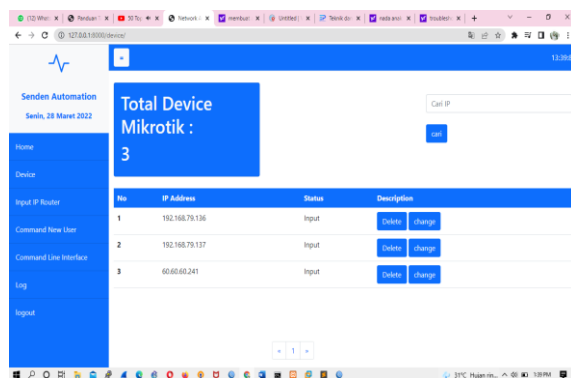


Gambar 11. Tampilan *input device*

Untuk penggunaannya admin dapat memasukkan beberapa kebutuhan *inputan* kedalam *form* yang sudah ada.

#### 3.1.4.2. Tampilan *Device*

Pada tampilan ini berisi tentang informasi yang ada pada *device* seperti IP status *update* dan *delete*. Terkait gambar dari tampilan *device* dapat ditunjukkan pada gambar 12.

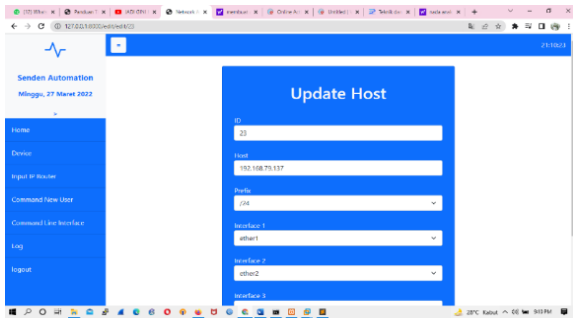


Gambar 12. Tampilan *device*

Untuk dapat masuk kedalam tampilan tersebut admin dapat masuk ke dalam tampilan *device*.

#### 3.1.4.3. Tampilan *Device Update*

Untuk tampilan *device update* berisi tentang *update* dari *device*. Untuk tampilan dari *update* ditunjukkan pada gambar 13.

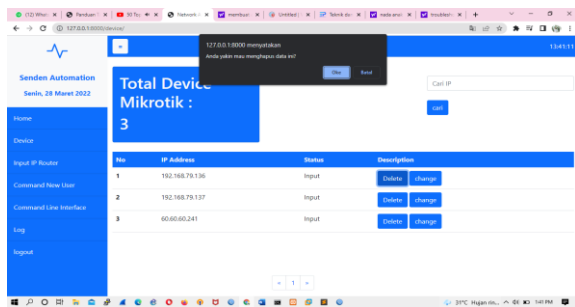


Gambar 13. Tampilan *update*

Mengenai proses dari *update* sendiri dimulai dengan melakukan klik pada *bottom* dari *device* dan melakukan *update* dengan merubah *inputan* yang sebelumnya dimasukkan pada proses *inputan*.

### 3.1.4.4. Tampilan *Device Delete*

Tampilan *device delete* berfungsi untuk menghapus *device*. Mengenai detail dari gambar dapat ditunjukkan pada gambar 14.



Gambar 14. Tampilan *delete*

Untuk proses penggunaannya agar *device* dapat dihapus maka dapat dengan klik *delete* dari *device* yang sudah dimasukkan.

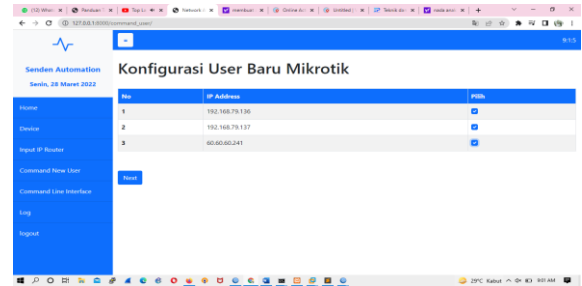
### 3.1.5. Tampilan Konfigurasi *User Automatis*

Konfigurasi *user* otomatis merupakan proses dari *network automation* pada *user* baru karena terdapat proses otomatis di dalam pemrosesan *command* ke *network*. Pada proses kali ini terdiri dari dua jenis proses konfigurasi *user* baru yang pertama adalah dengan memilih jenis konfigurasi untuk *user* baru sehingga beberapa CLI (*Command Line Interface*) pada *Mikrotik* sudah ada di dalam *coding* atau dibagian *backend*, sehingga *command* untuk bagian ini sudah masuk ke dalam halaman konfigurasi *user* baru dan yang kedua adalah dengan memasukkan CLI (*command line interface*) ke dalam *form* yang sudah ada sehingga konfigurasi dapat lebih fleksible. Untuk jenis konfigurasi ini masuk ke dalam halaman konfigurasi CLI (*Command Line Interface*).

#### 3.1.5.1. Tampilan *Configuration New User*

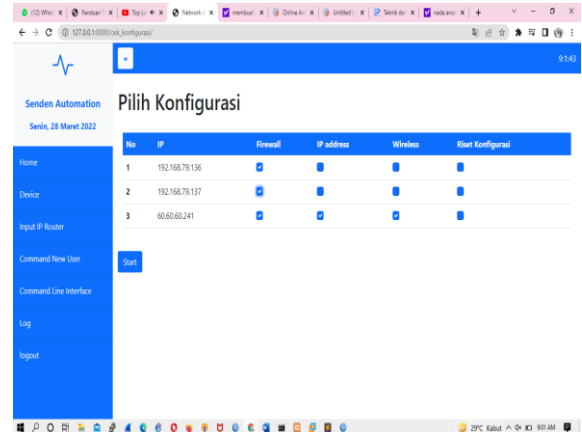
*Configuration new user* terdiri dari beberapa tahapan, yang pertama adalah melakukan pemilihan

IP. Untuk tampilan dari pemilihan IP dapat dilihat pada gambar 15.



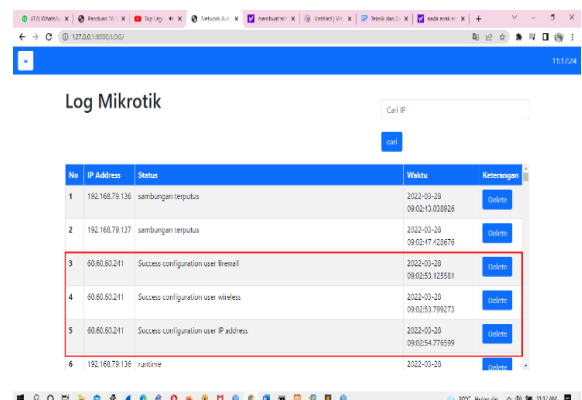
Gambar 15 Tampilan pilih IP di *configuration new user*

Di gambar 15 ini dapat diketahui untuk proses dari pemilihan IP (*Internet Protocol*) adalah dengan melakukan pemilihan dari cek *box* untuk dapat mengetahui *device* mana saja yang akan dilakukan konfigurasi. Yang kedua adalah melakukan pemilihan jenis konfigurasi. Terkait tampilan dari gambar pilih konfigurasi ditunjukkan pada gambar 16.



Gambar 16. Tampilan pilih konfigurasi

Pada tampilan gambar 16 menunjukkan bahwa dalam melakukan pemilihan konfigurasi dapat dengan klik cek *box* di bagian konfigurasi yang ingin dipilih. Lalu yang ketiga adalah melakukan pengecekan di halaman *log*. Untuk tampilan dari gambar *log* dapat ditunjukkan pada gambar 17.

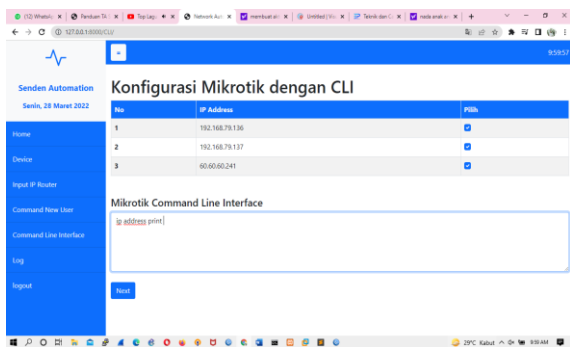


Gambar 17. Tampilan *log user* baru berhasil terkonfigurasi

Pada gambar 17 menunjukkan bahawa jika di halaman *log* sudah berhasil terlihat hasil konfigurasi *user* baru ke beberapa *device* secara bersamaan, maka proses konfigurasi pada *user* baru bisa dinyatakan sudah berhasil dan dapat berjalan dengan baik.

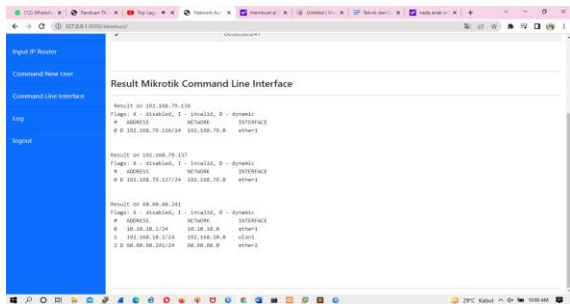
**3.1.5.2. Tampilan CLI (Command Line Interface)**

CLI (Command Line Interface) terdiri dari beberapa tahapan yang pertama adalah tahapan pilih IP (Internet Protocol) dan yang kedua adalah melakukan input CLI (Command Line Interface). Untuk tampilan dari konfigurasi IP dan CLI (command line Interface) dapat dilihat pada gambar 18.



Gambar 18. Tampilan pilih IP dan input CLI (Command Line Interface)

Pada gambar 18 ini dapat diketahui bahwa proses untuk melakukan konfigurasi dengan CLI (Command Line Interface) dapat dengan memilih ip dan memasukkan perintah dari Mikrotik. Lalu yang kedua adalah proses hasil dari konfigurasi dari CLI (Command Line Interface). Terkait tampilan dari hasil konfigurasi dari CLI (Command Line Interface) dapat dilihat pada gambar 19.



Gambar 19. Tampilan hasil konfigurasi CLI (Command Line Interface)

Di gambar 19 ini dapat diketahui bahwa proses selanjutnya jika sudah berhasil melakukan konfigurasi dari CLI (Command Line Interface) akan muncul hasil hasil di halaman *result*

**3.2 Pengujian Website dengan Black Box**

Pengujian *website* dengan *Black Box* merupakan bentuk pengujian yang lebih mengandalkan fungsional yang ada mulai dari *input* dan *output*.

Untuk proses pengujian sendiri menggunakan sebuah perangkat PC dengan spesifikasi Intel®Core™i5-7200 CPU @ 2.50GHz (4 CPUs), 2.7GHz, HDD 1 TB, RAM 4 GB, Model GPU intel®HD Graphics 620.

Tabel 1. Uji Website dengan Metode Black Box

No	Pengujian	Test Case	Hasil	Kesimpulan
1	Uji login	Masukkan <i>username</i> validasi	Berhasil melakukan validasi	Valid
2	Uji sign up	Masukkan <i>nameuser</i> validasi	Berhasil melakukan validasi	Valid
3	Masuk <i>dashboard</i>	Masuk link <i>dashboard</i>	Berhasil masuk ke <i>dashboard</i>	Valid
4	Masuk <i>device</i>	Masuk ke link <i>device</i>	Berhasil masuk	Valid
5	Masuk <i>device</i> dan <i>update</i>	Masuk ke <i>update</i> lewat halaman <i>device</i>	Berhasil masuk dan melakukan <i>update</i>	Valid
6	<i>Device</i> dan <i>delete</i>	Masuk ke halaman <i>device</i> dan melakukan <i>delete</i>	Berhasil menghapus <i>device</i>	Valid
7	Masuk <i>Configurati on User Baru</i>	Masuk ke halaman <i>configurati on user baru</i>	Berhasil Masuk	Valid
8	Uji pilih IP pada halaman konfigurasi <i>user</i> baru	Pilih cek <i>box</i> dan lakukan <i>submit</i>	Berhasil dilakukan	Valid
9	Uji pilih konfigurasi pada halaman <i>configurati on user</i> baru	Pilih cek <i>box</i> dan lakukan <i>submit</i>	Berhasil dilakukan	Valid
10	Uji coba <i>log</i> dari hasil konfigurasi <i>user</i> baru	Pilih halaman <i>log</i>	Berhasil terkonfigurasi terlihat di <i>log</i>	Valid
11	Uji halaman konfigurasi <i>user</i> baru	Masuk link konfigurasi CLI	Berhasil masuk	Valid
12	Uji pilih IP dan masukkan <i>command line interface</i> pada halaman <i>command line interface Mikrotik</i>	Pilih cek <i>box</i> IP dan masukkan CLI ( <i>command line interface</i> ) Mikrotik	Berhasil berjalan	Valid
13	Uji hasil pemrosesan dari halaman <i>command line interface</i>	Masuk halaman setelah halaman <i>command line interface</i>	Berhasil terlihat	Valid
14	Uji log	Masuk ke halaman <i>log</i>	Berhasil <i>terupdate</i>	Valid



#### 4. KESIMPULAN

*Website network automation* untuk RT RW NET dusun Senden memberikan kemudahan dalam melakukan konfigurasi *user* baru. Dibuktikan dengan konfigurasi yang langsung berjalan di beberapa *device* dengan beberapa IP yang sudah terdaftar ke dalam *website network automation*. Pengembangan dari *website* ini menggunakan metode *Waterfall*, metode ini memberikan kemudahan dalam melakukan pengembangan dari aplikasi karena bersifat sistematis dan mudah digunakan. Untuk pengujian dari *website* ini menggunakan metode *Black Box* pada metode ini menerapkan mode pengetesan fungsional, di *website* ini berhasil untuk menyelesaikan beberapa tahapan pengetesan *Black Box* diibuktikan dengan fitur yang ada pada *website* dapat berjalan dengan baik dan benar. Mungkin untuk kedepannya *website* ini dapat dikembangkan lagi dengan lebih banyak fitur seperti *management hotspot* agar lebih mudah lagi dalam melakukan *management hostpot*.

#### DAFTAR PUSTAKA

- [1] K. Kristio, W. Widiyaningsih, and M. Lailasari, "DESIGNING CLEANERS WEBSITE FOR HELMET CLEANSING AND HELMET," *J. Tek. Inform.*, vol. 3, no. 2, pp. 193–200, 2022.
- [2] Y. K. Ningsih, Y. S. Rochman, and N. Kurniawati, "Implementasi RT/RW-Net Menggunakan Metode User dan Bandwidth Management," *J. Tek. Media Pengemb. Ilmu dan Apl. Tek.*, vol. 19, no. 2, pp. 120–129, 2020, doi: 10.26874/jt.vol19no02.305.
- [3] M. Fahmi, M. Maisyaroh, I. Komarudin, S. Faizah, and I. Fadhilah, "Otomatisasi Jaringan Menggunakan Script *Python* untuk Penyediaan Konfigurasi Internet dan Manajemen *Mikrotik*," *Bina Insa. Ict J.*, vol. 8, no. 1, p. 53, 2021.
- [4] E. S. Ginting and I. Hadi, "Testing Automatic Configuration of Adding Gateways to a Virtual Router Using Web-Based Network Automation Applications Elin," *J. Media Inform. Budidarma*, vol. 4, no. 4, pp. 1126–1131, 2020, doi: 10.30865/mib.v4i4.2485.
- [5] R. A. Wiryawan and N. R. Rosyid, "Pengembangan Aplikasi Otomatisasi Administrasi Jaringan Berbasis Website Menggunakan Bahasa Pemrograman *Python*," *Simetris J. Tek. Mesin, Elektro dan Ilmu Komput.*, vol. 10, no. 2, pp. 741–752, 2019.
- [6] D. Rahardika and N. Ratama, "Implementasi Network Automation Untuk Konfigurasi Jaringan Baru Dengan *Netmiko*," *J. Artif. Intell. Innov. Appl. Vol. 2, No. 3, Agustus 2021 ISSN*, vol. 2, no. 3, pp. 190–200, 2021.
- [7] D. S. Purnia, A. Rifai, and S. Rahmatullah, "Penerapan Metode *Waterfall* dalam Perancangan Sistem Informasi Aplikasi Bantuan Sosial Berbasis Android," *Semin. Nas. Sains dan Teknol.* 2019, pp. 1–7, 2019.
- [8] M. Fuadi and U. Darusalam, "Face Recognition Menggunakan *Opencv* Dengan Bahasa Pemrograman *Python* Oop Untuk Sistem," *J. Artif. Intell. Innov. Appl.*, vol. 2, no. 3, pp. 180–184, 2021.
- [9] D. Saputra, "Analisis Perbandingan Performa Web Service Rest Menggunakan Framework *Laravel*, *Django* Dan *Ruby On Rails* Untuk Akses Data Dengan," *J. Bangkit Indones.*, vol. 7, no. 2, p. 17, 2018, doi: 10.52771/bangkitindonesia.v7i2.90.
- [10] A. H. Albana and F. F. Coastera, "SISTEM INFORMASI MONITORING USER MANAGER DAN VOUCHER PAYMENT *MIKROTIK* ROUTER BERBASIS WEB MOBILE ( Studi Kasus : Ruko Queenkostel Kota Bengkulu )," *J. Rekursif*, vol. 8, no. 2, pp. 163–175, 2020.
- [11] R. R. Fadila, W. Aprison, and H. A. Musril, "Perancangan Perizinan Santri Menggunakan Bahasa Pemrograman *PHP/MySQL* Di SMP Nurul Ikhlas," *CSRID (Computer Sci. Res. Its Dev. Journal)*, vol. 11, no. 2, p. 84, 2021, doi: 10.22303/csrid.11.2.2019.84-95.
- [12] A. Rasool and Z. Jalil, "A Review of Web Browser Forensic Analysis Tools and Techniques," *Res. J. Comput.*, vol. 1, no. 1, pp. 15–21, 2020, [Online]. Available: <https://www.researchgate.net/publication/342338294>.
- [13] M. F. Santoso, "Teknik Responsive Web Design *Bootstrap* 4 Serta Penerapannya Dalam Rancang Bangun Layout Web," *J. Pilar Nusa Mandiri*, vol. 15, no. 1, pp. 61–68, 2019, doi: 10.33480/pilar.v15i1.101.
- [14] M. (2018) Siregar, H. F., Siregar, Y. H., & Melani, "Perancangan Aplikasi Komik Hadist Berbasis Multimedia. *JurTI (Jurnal Teknologi Informasi)*, 2(2), 113-121.," *JurTI (Jurnal Teknol. Informasi)*, vol. 2, no. 2, pp. 113–121, 2018, [Online]. Available: <http://www.jurnal.una.ac.id/index.php/jurti/article/view/425>.
- [15] S. Handayani, "Perancangan Sistem Informasi Penjualan Berbasis E-Commerce Studi KaHandayani, S. (2018). Perancangan Sistem Informasi Penjualan Berbasis E-Commerce Studi Kasus Toko Kun Jakarta. *ILKOM Jurnal Ilmiah*, 10(2), 182–189. <https://doi.org/10.33096/ilkom.v10i2.310>," *Ilk. J. Ilm.*, vol. 10, no. 2, pp. 182–189, 2018.

- [16] Suendri, “Implementasi Diagram UML (Unified Modelling Language) Pada Perancangan Sistem Informasi Remunerasi Dosen Dengan Database Oracle (Studi Kasus: UIN Sumatera Utara Medan),” *J. Ilmu Komput. dan Inform.*, vol. 3, no. 1, pp. 1–9, 2018, [Online]. Available: <http://jurnal.uinsu.ac.id/index.php/algoritma/article/download/3148/1871>.
- [17] Rahayuning Putri Mahardikawati and Nurgiyatna, “Sistem Informasi Industri Kecil Menengah Pemerintahan Kabupaten Boyolali Berbasis Website,” *J. Tek. Inform.*, vol. 1, no. 2, pp. 53–60, 2020, doi: 10.20884/1.jutif.2020.1.2.13.
- [18] D. Gunawan, I. A. Ar Raniri, R. N. Setyawan, and Y. D. Prasetya, “Web-Based Library Information System in Madrasah Ibtidaiyah Negeri Surakarta,” *J. Tek. Inform.*, vol. 2, no. 1, pp. 33–41, 2021, doi: 10.20884/1.jutif.2021.2.1.44.