

OAJIS

Open Access
Journal of
Information
Systems

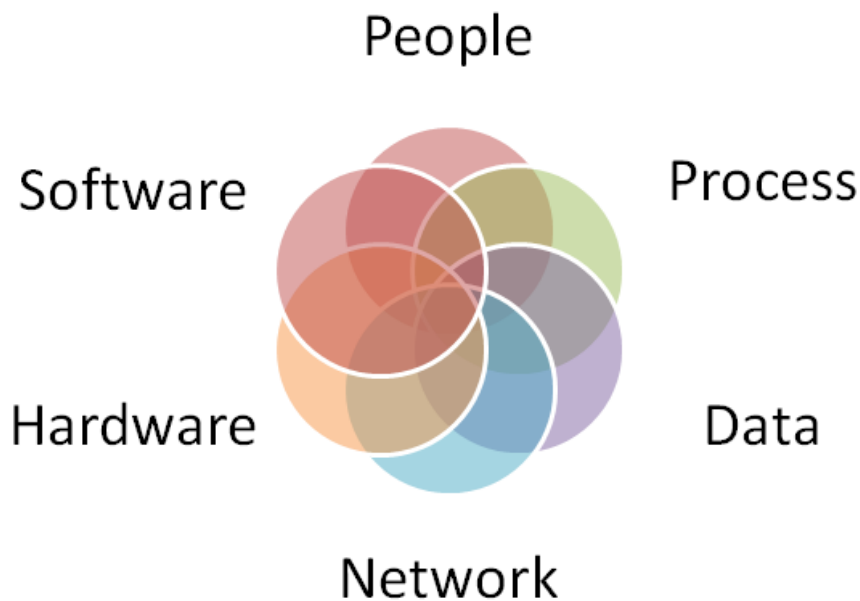
is.its.ac.id/pubs/oajis/

ISSN 1979-3979



jurnal sisfo

Inspirasi Profesional Sistem Informasi





Pimpinan Redaksi

Reny Nadlifatin

Dewan Redaksi

Faizal Mahananto

Mudjahidin

Penyunting

Rizal Risnanda Utama

Sekretariat

Jurusan Sistem Informasi – Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember (ITS) – Surabaya
Telp. 031-5999944 Fax. 031-5964965
Email: editor@jurnalsisfo.org
Website: <http://jurnalsisfo.org>

Jurnal SISFO juga dipublikasikan di *Open Access Journal of Information Systems* (OAJIS)

Website: <http://is.its.ac.id/pubs/oajis/index.php>



Mitra Bestari

Riyanto Jayadi, S. Kom, M.IM., Ph.D. (Universitas Bina Nusantara)

Yogantara Setya Dharmawan, S.Kom., MBusProcessMgt. (Institut Teknologi Sepuluh

Nopember)

Prof. Erma Suryani, ST., MT., Ph.D. (Institut Teknologi Sepuluh

Nopember)

Nisfu Asrul Sani, S.Kom., M.Sc. (Institut Teknologi Sepuluh

Nopember)

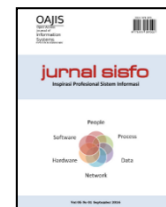
Arif Wibisono, S.Kom., M.Sc. (Institut Teknologi Sepuluh

Nopember)

Ilma Mufidah, S.T., M.T., Ph.D. (Telkom University)

Satria Fadil Persada, S.Kom., M.BA., Ph.D. (Universitas Bina Nusantara)

Izzat Aulia Akbar, S.Kom., M.Eng., Ph.D. (Institut Teknologi Sepuluh Nopember)



Daftar Isi

Evaluasi Gim Edukasi Berbasis Kerangka Berpikir MDA

Nisfu Asrul Sani, Ilham Cahya Suherman 1

Pembuatan Purwarupa Digital dengan Pendekatan Design Thinking Sebagai Strategi Bisnis di PT XYZ

Helena Hanindya Kartika Putri, Febby Candra Pratama 9

Digital Information Utilization to Evaluate Financial Performance in Mining Industry

Riza Rizqiyah 23

Transformasi Digital Startup: Pendekatan Berorientasi pada User

Riefky Prabowo, Febby Candra Pratama 41

Implementation of Website-Based Network Administration Automation within the Ministry of Social Affairs

Gumilar Santioko Mukti, Made Kamisutara 51

OAJIS

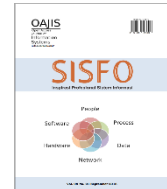
Open Access
Journal of
Information
Systems
is.its.ac.id/pubs/oajis/

jurnal sisfo

Jurnal Sisfo Vol. 10 No. 03 (2023) i-iv



Halaman ini sengaja dikosongkan



Implementation of Website-Based Network Administration Automation within the Ministry of Social Affairs

Gumilar Santioko Mukti¹, Made Kamisutara

Information Systems Department, Faculty of Computer Science, Universitas Narotama, Surabaya, Indonesia

Abstract

Today the use of computer networks has been implemented in almost every organization to communicate with parties who need them by granting access permissions. As a network management organization with a large number of devices, the challenge that currently has to be faced by the Data and Information Center of the Ministry of Social Affairs is that the administrative records of network devices and the installation of network devices are not well documented so it's difficult for network administrators every configure network device with different types and brands. Therefore, this research focuses on implementing a website-based network administration automation system so the data storage system is integrated and becomes a solution for carrying out complex network tasks that can be completed much more quickly and efficiently. The research method consists of identifying requirement, designing, implementing, and testing network administration automation systems. The results of this study indicate that a network administration automation system can facilitate device administration staff in managing device data; technicians in managing device installation data; coordinator in verifying device installation; and network administrators in configuring devices therefore minimizing the time needed by network administrators in configuring and reduce the possibility of network administrators making configuration errors.

Keywords: Network Automation, Configuration, Python, Laravel

Abstrak

Sebagai organisasi yang mengelola jaringan dengan sejumlah besar perangkat, Pusat Data dan Informasi Kementerian Sosial dihadapkan pada tantangan signifikan. Tantangan tersebut terkait dengan kurangnya dokumentasi yang tepat terkait administrasi perangkat jaringan dan pemasangan perangkat jaringan yang tidak tercatat dengan baik. Hal ini mengakibatkan kesulitan bagi administrator jaringan saat harus mengkonfigurasi perangkat jaringan yang beragam jenis dan mereknya. Untuk mengatasi permasalahan tersebut, penelitian ini dilakukan dengan tujuan untuk menerapkan sistem otomasi administrasi jaringan berbasis website. Sistem ini bertujuan untuk mengintegrasikan penyimpanan data, memberikan solusi untuk menangani tugas-tugas jaringan yang kompleks dengan lebih efisien dan cepat. Penelitian ini mencakup berbagai tahapan, mulai dari identifikasi kebutuhan, perancangan, implementasi, hingga pengujian sistem otomasi administrasi jaringan. Hasil dari penelitian ini menunjukkan bahwa sistem otomasi administrasi jaringan mampu memberikan kemudahan kepada staf administrasi perangkat dalam mengelola data perangkat, teknisi dalam mengelola data pemasangan perangkat, koordinator dalam memverifikasi pemasangan perangkat, dan administrator jaringan dalam melakukan konfigurasi perangkat. Dengan demikian, waktu yang diperlukan oleh administrator jaringan untuk melakukan konfigurasi dapat diminimalkan, dan risiko kesalahan konfigurasi pun dapat dikurangi.

Kata kunci: Otomasi jaringan, konfigurasi, Python, Laravel

¹Corresponding Author

Email address: santiokomukti@gmail.com (Gumilar Santioko Mukti)
<https://doi.org/10.24089/j.sisfo.2020.11.005> (DOI)

1. Introduction

Today the use of information and communication technology (ICT) has covered all aspects of life. This can be seen from all aspects of life that cannot be separated from the use of the internet or computer networks. The use of computer networks has been implemented in almost every company or organization to communicate with parties who need them by granting access permissions [1]. This computer network makes it easy for everyone to access and share data and information using wired or wireless (wireless). Data access services via computer networks are essential in daily activities [2]. In line with the need for data access, the use of computer networks in today's organizations is increasingly complex and dynamic. The Data and Information Center of the Ministry of Social Affairs or commonly referred to as Pusdatin Kesos is a work unit under the Indonesian Minister of Social Affairs whose job is to carry out the management and dissemination of social welfare data and information, management and development of information systems and technology.

With this inherent task, Pusdatin Kesos is responsible for managing all computer networks within the Ministry of Social Affairs. As a network management organization with a large number of devices, some of the challenges currently faced by Pusdatin Kesos in network management are administration records of new devices, administration of device installation records, and configuration of network devices of different types, types and brands. Currently the work area within the Ministry of Social Affairs has geographical conditions that are separated into 2 large locations (areas) and consists of separate buildings. In each of these locations there are around 120 devices of different types, types and brands.

Administrative records of new network devices and the installation of network devices are currently not well documented, making it difficult or a significant obstacle for network administrators every time they make configurations. The absence of complete and valid data regarding the type and brand of the device, the serial number of the device, the device management IP, as well as the area and details of the device installation location causes the network administrator to manually search for these data every time he makes a configuration.

In the use of complex computer networks, it certainly requires many network components or devices used in building computer networks such as routers, access switches, core switches, farm switches and other supporting devices with different types, types and brands of devices. In addition, leadership level policies will also result in different configurations being implemented. The addition of devices and an increase in the number of users are some of the reasons why the network is dynamic or constantly changing. This complex and dynamic computer network makes it increasingly difficult for network administrators to configure because they have to always adjust network configurations in response to changing network conditions [3]. The number of scattered devices and the number of users is currently only managed by 2 network administrators by manually typing configuration commands using the command line interface (CLI) environment on devices of different types, types and brands every time there is a change in policy. Network administrators must log in every time they change devices, so the more devices, the more time it takes to configure. The manual or traditional method by remotely processing each of these devices will take a long time [4]. As a result of frequent configuration changes, network administrators must always require high concentration in every configuration. Such conditions increase the possibility that network administrators often make configuration errors on each device.

In line with previous research, network automation is a solution for doing complex network jobs and can be implemented on devices that support the SSH protocol so that work can be completed much faster and also

efficient in network maintenance with procedures that are easier to follow and implement on the network large scale [4]. Therefore, based on the conditions described above, it is necessary to have a website-based network administration automation system with an integrated graphical interface or GUI so that the data storage system is centralized. The network administration automation system in this study covers the recording of new network devices and new location areas by device admin staff; recording of network device installation by network technicians; verification of the installation of network devices by the network management coordinator; and configuration of network devices by network administrator.

2. Literature Review

2.1 Network Automation

Network Automation is an automation method for network administration. The automation method on the network is commonly known as Network Automation. Network automation is a way to configure network devices more quickly. The principles of network automation are the same as in other types of automation such as in applications, systems, storage and telephony. Network automation can reduce time when developing, changing configurations, and backup/restore network configurations [5]. Network automation is a solution for saving operational costs, not only reducing the time spent configuring network devices, but also network maintenance efficiency with procedures that are easier to follow and implement on a large scale [6]. Network automation has the main goal of simplifying the tasks associated with configuring, managing and operating network equipment, network topology, network services, and network connectivity [7]. In addition, the concept of network automation is also developing, namely where setting (configuration), managing, testing and operating both physical and virtual network devices is carried out using software or applications automatically by sending configuration commands to network devices via a secure shell (SSH) connection [2][8][9].

2.2 Python

Python is a multipurpose interpretive programming language or one that can execute a number of multipurpose instructions directly with an object oriented method (Object Oriented Programming) and uses dynamic semantics to provide a level of readability of code or syntax (easy to understand and learn). The Python language was designed by Guido van Rossum appeared in 1991 and is still being developed by the Python Software Foundation [10]. Python has several advantages over other programming languages, Python is an open source programming language that can be used on a variety of different platforms such as Windows, Mac, Linux, Raspberry Pi, and others. Python has a simple syntax like English and allows developers to write programs with fewer lines than some other programming languages. Python codes are easy to read, uncomplicated, and easy for users to learn [11][12].

Automation can be done using a programming language like python. Python is becoming the most popular programming language in automation because it is not as complex as java and C. When it comes to automation or controlling network devices, Python scripts are based on the Netmiko and Paramiko libraries. Both use SSH connection to control the device. SSH (Secure Shell) is a cryptographic network protocol for operating network services securely [6]. Paramiko is a Python implementation of the SSHv2 protocol that provides both client and server functionality. Netmiko is a multi-vendor library, developed based on Paramiko, simplifying connections across multiple vendors and network platforms [6]. The difference between Netmiko and Paramiko is that Netmiko uses an easier way to connect to the switch using ConnectHandler which also uses SSH on the backend. In addition, when using Netmiko, you must specify the type of device you want to control. Netmiko uses Paramiko but also creates new interfaces and methods needed to work on various network devices.[python] For scripts using Netmiko, the important functions used are ConnectHandler(), send_command(), and disconnect().

This study uses Python because it is a high-level programming language that is easy to learn, becomes an integral part of large-scale networks, is a solution to streamline operations on networks, and network administrators do not need to type CLI commands to configure repetitive networks because with Python they can create interfaces automatically. graphics [13]. Python is a full-featured programming language with well-documented libraries. A network programmer will easily retrieve and request data from a web, and easily extract data into a common format via the web using only Python.

2.3 Laravel

Laravel is a PHP-based framework that is used in making websites and was created by Taylor Otwell and released under the MIT license with source code provided by Github or open source [14][15][16][17]. Laravel is an application framework with an elegant syntax and has extensive functions such as security, password storage, reminder and reset password, encryption, and validation [18]. Laravel summarizes program code because several lines of code used in php, are summarized in Laravel. Apart from that, Developers also don't need to write the same code to use, but just set it up in the controller so that it makes the code more concise. With Laravel which makes the code more concise and several modules (artisan, authentication and controller) provided to developers in website development, the time needed for making this website is faster. Laravel has complete documentation which can be accessed at id-laravel.com, learninglaravel.net, laracast.com and Laravel-news.com. Laravel can be used easily so it is said to be user friendly. Apart from that, Laravel guarantees the security of this framework. Solving problems related to using Laravel is quite a lot and easy to find because there is a Laravel community [15]. Just like other frameworks, Laravel is built on the MVC (Model-Controller-View) concept to make website development easier [16].

This study uses the Laravel framework for the user interface because there are several advantages that Laravel has, namely Laravel is an expressive PHP framework because the syntax in Laravel uses language that is easy to understand so even novice programmers will easily understand the use of a syntax; Laravel has quite complete documentation where each version has its own documentation, starting from how to install it to using its features; Laravel is used by many programmers so many libraries that support Laravel are created by programmers; Laravel is supported by Composer so libraries can be obtained easily from the internet using Composer; Laravel has its own template engine called blade which makes it easy to display data in HTML templates [19]. In addition, the Ministry of Social Affairs currently uses a lot of the Laravel framework for application development, so the selection of this Laravel framework can make it easier for the Ministry of Social programmers to develop this system in the future. By using Laravel and Python together, it shows the opportunity for a mix of technologies that is possible to bridge processes at layer 4 (tcp protocol ssh) to layer 7 (applications).

2.4 Previous Research

There are several studies related to network automation. Rheza conducted research related to the development of website-based network administration automation applications in terms of static, dynamic routing configurations, VLAN creation, maintenance in the form of backup and restore and can be accessed centrally so that it will be better managed [4]. This research uses the Rapid Application Development (RAD) method with Ubuntu as a server, Python 3.6. for application development, the Paramiko 2.4 library for automation, Django as a web framework, and GNS3 2.1 as a simulation tool. Prof. . The design of network automation using the Network Development Life Cycle (NDLC) research method uses Python scripts and the Paramiko library which runs on a Linux Ubuntu server. Donny Rahardika conducted research in the form of designing and building a command-line-based network automation system to increase the productivity of the network operator center so as to save time and resources in configuring new network devices at once with just a few easy steps [21]. This research uses the Network Development Life Cycle (NDLC) method and the Python programming language with the help of the Netmiko library. Abdul Aziz designed network automation related to configuration updates, operating system updates, username and password updates, as

well as performing backup restore configurations on customer routers simultaneously with commands typed in one web dashboard so that configurations can be carried out more effectively [22]. The system development method is structured, namely the waterfall model. Yoel Chandra created a website-based network automation application that provides convenience in accessing and configuring the addition of new users quickly without taking a long time due to manual configuration in every alley RT (Rukun Tetangga) and RW (Rukun Warga) [23]. The design of this network automation website uses the Waterfall development method with Python programming and the Django framework.

This research focuses on implementing website-based network administration automation which consists of recording new network devices and new location areas, recording network device installation, verifying network device installation, and network device configuration in the form of VLAN configurations and network port interface configurations. This study uses the Laravel framework, HTML, PHP, and JavaScript for web-based system development and Python programming and the Netmiko library to connect network devices with web-based systems. Even though several previous studies have developed network automation systems, most of the current research uses only Mikrotik devices, so research is still needed to find out how network automation is implemented on other devices such as Aruba and Cisco.

3. Methodology

The stages of the research conducted consisted of 4 main processes which are presented in Figure 1. The four processes consist of identifying needs, designing, implementing, and testing.

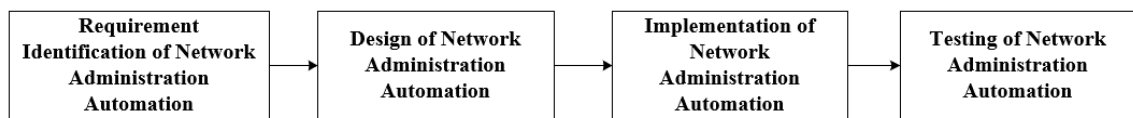


Figure 1. Research Methodology

3.1 Requirement Identification of Network Administration Automation

The requirement identification of network administration automation consists of several activities, namely conducting literature studies on previous studies related to network automation as a basis for conducting research; identify hardware and software requirements to be used; and identify functional requirements for network administration automation systems.

3.2 Design of Network Administration Automation

The design of network administration automation consists of several activities, namely designing UML diagram models in the form of use case diagrams and activity diagrams; designing the appearance of the GUI (Graphical user interface (GUI)); and designing the database.

3.3 Implementation of Network Administration Automation

Implementation of network administration automation consists of implementing or building a website-based network administration automation system using the Laravel web framework, HTML, PHP, JavaScript, and Python with the main Netmiko library to connect servers with network devices via the SSH protocol.

3.4 Testing of Network Administration Automation

Testing network administration automation consists of two activities, namely testing using the black-box testing method to determine the functional success of the application system and testing using the

questionnaire method to provide an assessment of the feasibility quality of the application system whether it is in accordance with user requirements or not. Testing using black-box testing is done by creating test cases and providing the results of these tests, then seeing whether the results of functional testing are successful as expected or not. While testing using a questionnaire was carried out by giving statements to all users directly to assess them with a Likert scale. The Likert scale is designed to convince users to answer at several levels of the statements provided. The number of respondents to the questionnaire was 18 users, consisting of 2 device administration staff, 13 network infrastructure technicians, 1 network management coordinator, and 2 network administrators. In conducting the test using a questionnaire to the respondents, the researcher prepared 2 aspects of the statement used in the test, namely aspects of the application display and the application process. There are 11 statements for device administration staff, 9 statements for network infrastructure technicians, 9 statements for network management coordinators, and 11 statements for network administrators which respondents can then assess by checking a list of one of Strongly Agree (SA), Agree (A), Neutral (N), Disagree (D), Strongly Disagree (SD). Furthermore, the results of the Likert scale assessment that has been filled in by respondents, calculations are carried out using the following formula:

$$Y = \frac{x}{ideal\ score} \times 100\%$$

$$x = \sum (N \times R)$$

Description:

Y = The percentage value sought

x = the number of times the value of each answer with the respondent

N = value of each answer

R = number of respondents

4. Results and Discussion

4.1 Requirement Identification Result

Based on the ongoing business processes, there are several problems within the Ministry of Social Affairs, namely as follows:

- a. Data details for each network device contained in the warehouse such as brand, type, and serial number are not documented.
- b. Network device installation data is not well documented because it is not integrated or the data is fragmented based on the Network Technician team.
- c. Administrative recording of new network devices and installation of network devices that are not well documented in points 1 and 2 make it difficult or hinder the network administrator every time they make configurations. The absence of complete, valid and integrated data regarding the type and brand of the device, the serial number of the device, the IP of the device management, as well as the area and details of the device installation location causes the network administrator to manually search for these data so that it takes a long time every time to do so. configuration.
- d. Configuration that is carried out by network administrators manually by typing configuration commands using the command line interface (CLI) environment requires more time when more and more devices are configured, thus requiring high concentration and increasing the possibility of network administrators frequently making configuration errors on each device.

With the existence of business processes and problems in the Ministry of Social Affairs above, the proposed business processes initiated by researchers are shown by Figure 2 as follows:

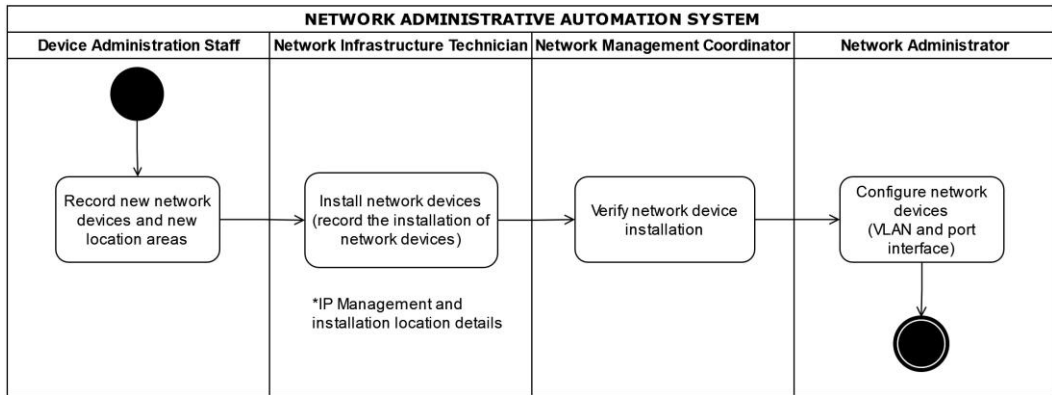


Figure 2. Proposed Business Process Network Administration Automation System

- Device Administration Staff records new network devices and new location areas using an automated network administration system.
- The Network Infrastructure Technician records the installation of network devices using an automated network administration system after installing the devices.
- The Network Management Coordinator verifies the installation of network devices that have been carried out by the Network Technician using an automated network administration system and forwards it to the Network Administrator for configuration.
- The Network Administrator performs VLAN configuration and port interface configuration on each verified network device using a network administration automation system.

4.2 Design Result

Based on the problems and proposed business processes, a use case diagram modeling with 19 menu features (9 main features) is produced. These features are interrelated and are used when recording network devices to configuring network devices. The following Figure 3 is a detailed use case diagram in a website-based network administration automation system.

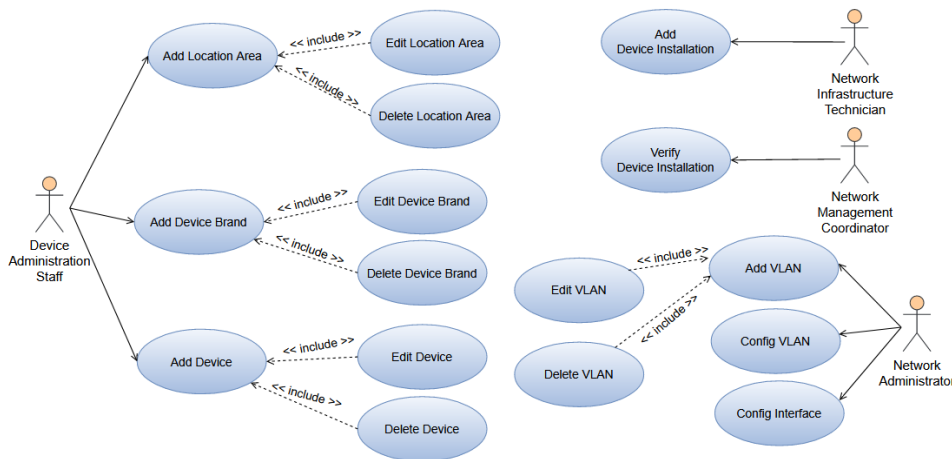


Figure 3. Network Administration Automation System Use Case Diagram

To support the implementation of a network administration automation system, database modeling is required in the form of a Physical Data Model (PDM). The PDM in the following Figure 4 is a physical description of the database to be created by considering DBMS that will be used.

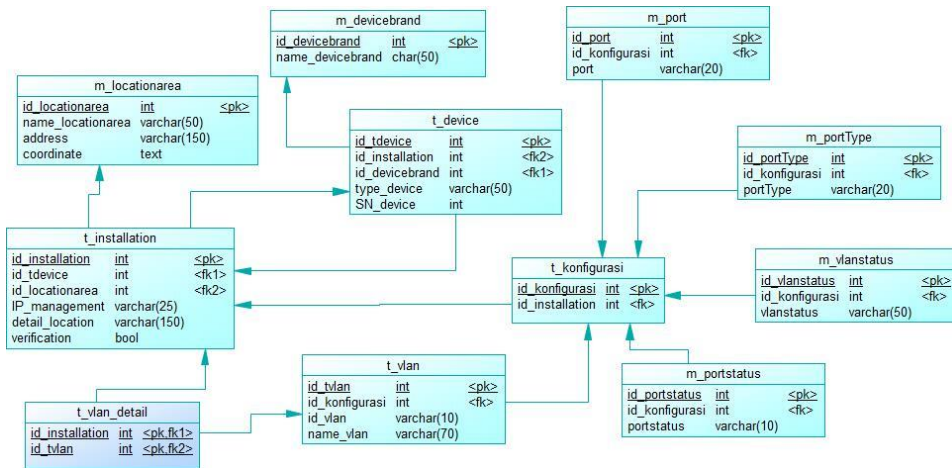


Figure 4. Physical Data Model of Network Administration Automation System

To simplify the process of implementing the system that has been modeled in the use case above, a workflow is created that contains the actions performed by the system based on the input provided by the user. In the following Figure 5 is a workflow that contains network and system administrator actions in configuring VLANs and port interfaces on selected network devices.

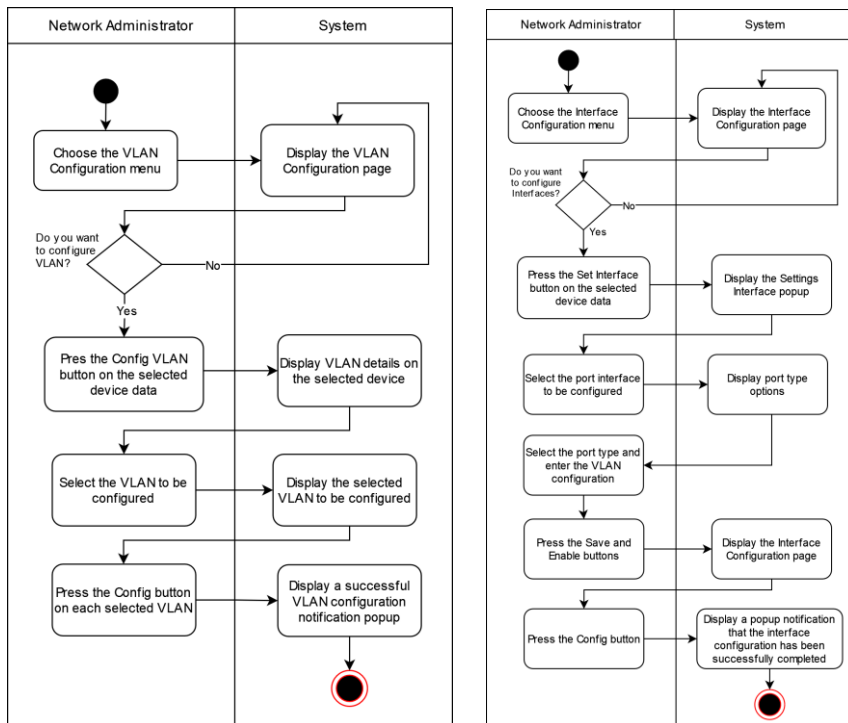


Figure 5. Network Administration Automation System Activity Diagram

4.3 Implementation Result

System implementation in this study uses the Laravel framework to build a website-based system and the Python library to connect network devices with the website-based system that has been built. The results of coding implementation using the Laravel framework and the Python library Netmiko are shown in Figures 6 and 7.

```

public function viewvlan($id, $idport)
{
    // vlan
    $data['pemasangan'] = T::pemasangan::where('t_pemasangan.id', $id)
    ->leftJoin('t_perangkat', 't_perangkat.id', 't_pemasangan.id_perangkat')
    ->leftJoin('m_merekperangkat', 'm_merekperangkat.id', 't_perangkat.m_merekperangkat_id')
    ->leftJoin('m_arealokasi', 'm_arealokasi.id', 't_pemasangan.id_arealokasi')
    ->leftJoin('t_vlan_detail', 't_pemasangan.id', 't_vlan_detail.pemasangan_id')
    ->select('t_vlan_detail.id as t_vlan_detail_id', 'm_arealokasi.nama_arealokasi as area', 'm_merekperangkat.nama_merekperangkat as merk', 't_perangkat.tipe_perangkat as t_perangkat_tipe_perangkat')
    ->first();
    $t_vlan = T::vlan::where('vlan_detail_id', $data['pemasangan']->t_vlan_detail_id->get());
    foreach ($t_vlan as $key => $vl) {
        $konfigurasi_vlan_radio = T::konfigurasi::where('id.t_vlan', $vl['id'])->where('id.port', $idport)->where('id.tipeport', '1')->where('id.pemasangan', $data['pemasangan']);
        $vlan_radio = ($konfigurasi_vlan_radio->id_statusvlan);
        if ($vlan_radio == '2') {
            $c = "checked";
        } elseif ($vlan_radio == '1') {
            $c = "";
        } else {
            $c = "";
        }
        $konfigurasi_vlan_check = T::konfigurasi::where('id.t_vlan', $vl['id'])->where('id.port', $idport)->where('id.tipeport', '2')->where('id.pemasangan', $data['pemasangan']);
        $vlan_check = ($konfigurasi_vlan_check->id_statusvlan);
        if ($vlan_check == '2') {
            $d = "checked";
        } else {
            $d = "";
        }
        $data['vlan_radio'][] = <div class="col-md-6"><div class="form-group" style="color: #888; padding-left: 10px; padding-right: 10px;"><label style="padding-left: 10px; padding-right: 10px;"><input type="checkbox" class="checkbox" /> $vlan_radio $vlan_radio $c </div></div>
        $data['vlan_check'][] = <div class="col-md-6"><div class="form-group" style="padding-left: 10px; padding-right: 10px;"><input type="checkbox" class="checkbox" /> $vlan_check $vlan_check $d </div></div>
    }
    return $data;
}

```

Figure 6. Coding Implementation Using Laravel Framework

```

global $device;
global $vlandetail;

def device_list():
    global $device;
    global $vlandetail;
    VDBCursor.execute("SELECT t_pemasangan.IP_manajemen, t_vlan.vlanid, t_vlan.nama_vlan")
    VLines = VDBCursor.fetchall()
    if VLines != "":
        for device_data in VLines:
            IP_perangkat = device_data[0]
            vlan = device_data[1]
            name_vlan = device_data[2]
            device = {
                "device_type": "cisco_ios",
                "host": device_data[0],
                "username": "username",
                "password": "password"
            }
            vlandetail = {
                "vlan": str(device_data[1]),
                "name": device_data[2]
            }
            # print(vlandetail)
            konfigurasi_vlan()
        else:
            print("error device")

def konfigurasi_vlan():
    # print(vlandetail)
    connection = ConnectHandler(**device)
    output = connection.send_config_set(vlandetail)
    print(output)
    output = connection.send_config_set('copy run start')
    print(output)
    connection.disconnect()

    device_data[0]
    else:
        device = {
            "device_type": "cisco_ios",
            "host": device_data[0],
            "username": "username",
            "password": "password"
        }
        interfacedetail = {
            "int": device_data[1]
        }
        modedetail = {
            "vlan": device_data[2] + " allowed " + device_data[3]
        }
        enabledetail = {
            device_data[0]
        }
        print(VLines)
        # konfigurasi_interface()
    else:
        print("error device list")

def konfigurasi_interface():
    # print(enabledetail)
    connection = ConnectHandler(**device)
    output = connection.send_config_set(interfacedetail, exit_config_mode=False)
    print(output)
    output = connection.send_config_set(modedetail, exit_config_mode=False)
    print(output)
    output = connection.send_config_set(enabledetail)
    print(output)
    output = connection.send_config_set('copy run start')
    print(output)
    connection.disconnect()

```

Figure 7. Coding Implementation Using Python Library-Netmiko

The implementation results which consist of a website-based network administration automation system display can be seen in Figure 8, 9, and 10. The Add Device page is used by the Device Administration Staff to add new devices to the warehouse. Furthermore, the Device Installation Page is used by the Network Infrastructure Technician to record device data after installing the device and the Verification Page is used by the Network Management Coordinator to verify the devices that have been installed by the Network Infrastructure Technician.

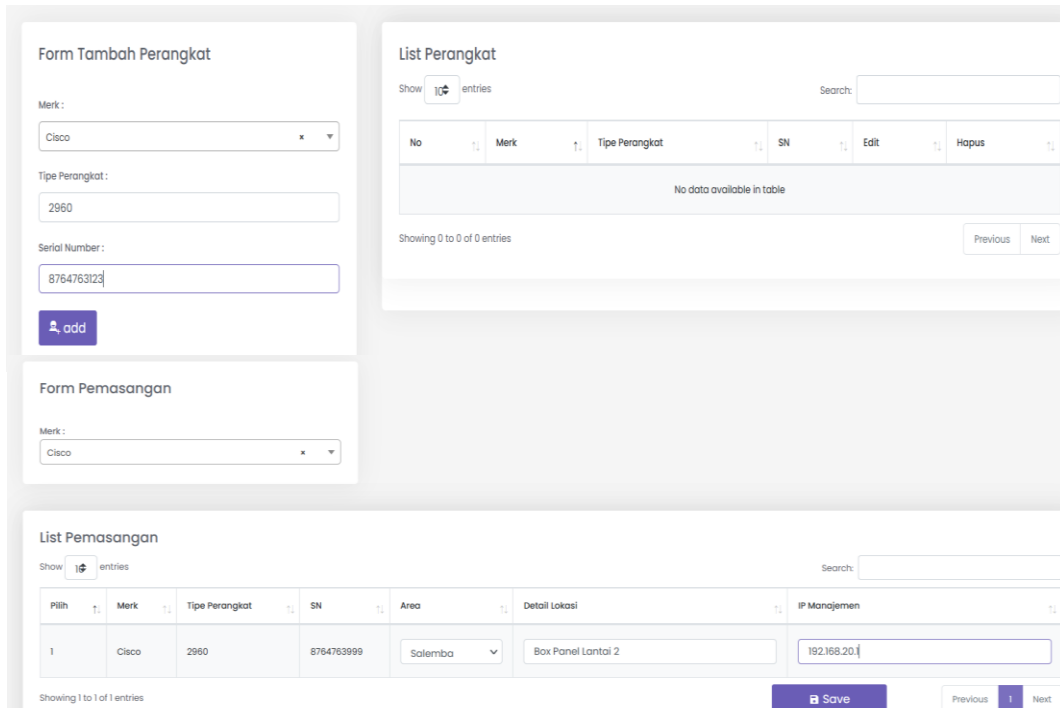


Figure 8. Add Device and Device Installation Page

The display of VLAN Configuration Page can be used by Network Administrator to add VLAN and Config VLAN. The Network Administrator can choose which VLAN to include on the network devices. In addition, on the Config VLAN menu display, the Network Administrator can press the Config button to run VLAN configuration on devices that have selected the VLAN.

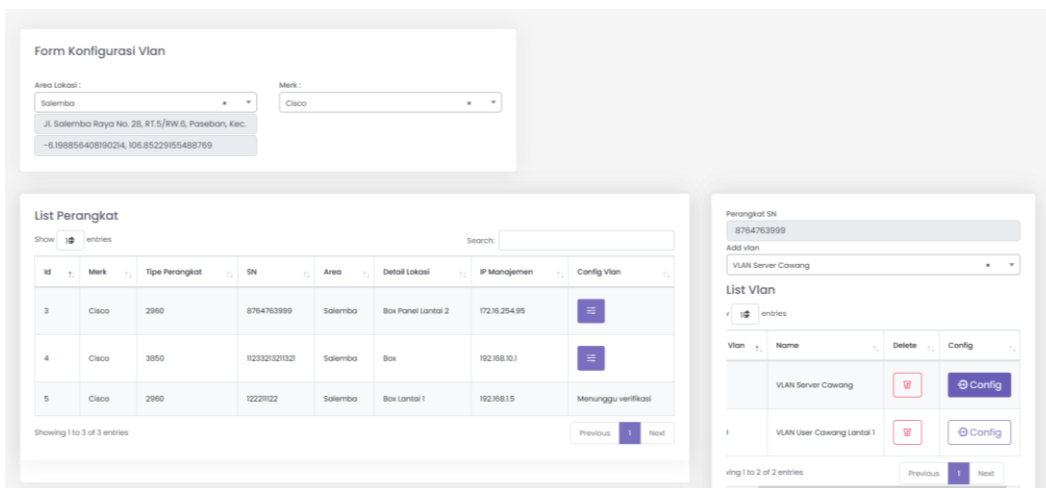


Figure 9. VLAN Configuration Page - Add VLAN and Config VLAN Menu

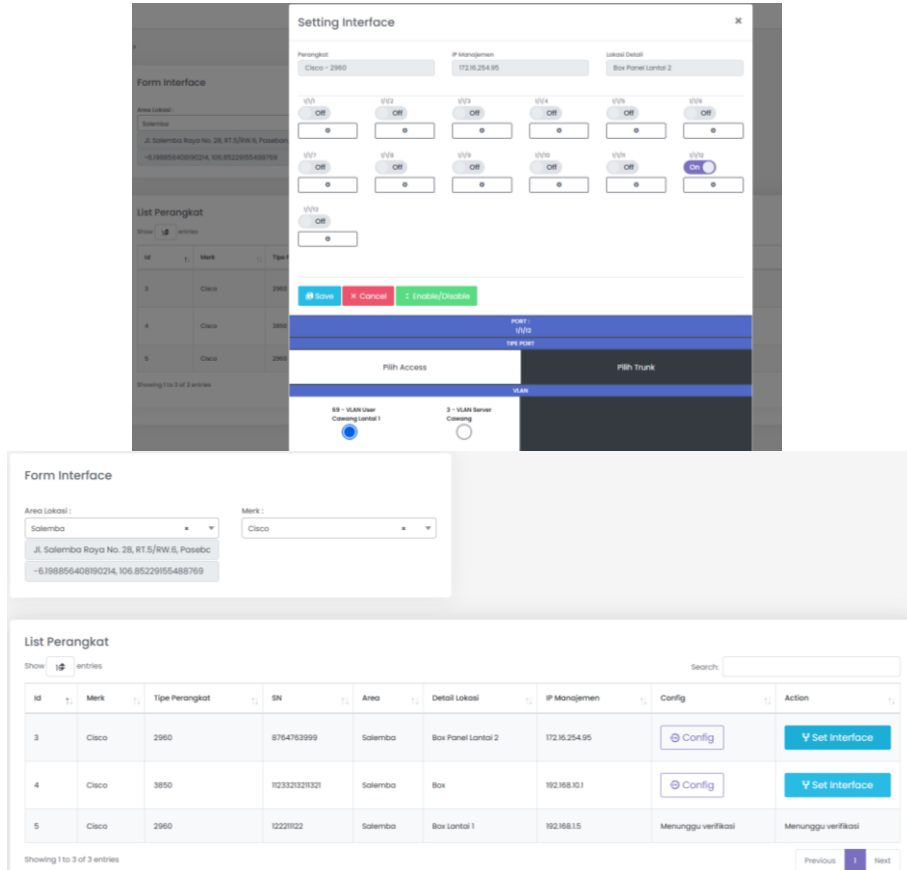


Figure 10. Interface Configuration Page - Set Interface and Config Interface Menu

The Interface Settings page is used by Network Administrators to configure interfaces on devices whose VLANs have been configured. The set interface menu display is used by the Network Administrator to select the VLAN type on each interface port. In addition, in the Config Interface menu display, the network Administrator can press the Config button to run the interface configuration on the interface ports that have selected the VLAN type.

4.4 Testing Result

The results of testing the network administration automation system using the blackbox testing method show that all features can function properly and correctly as expected. According to Table 1, it is known that one of the test result can add, change and delete VLAN data to configure VLANs as well as configure network interfaces. The next step is to measure the user perception of the technical performance of the system using questionnaire instruments. The survey was conducted to about 18 network administrators who are serving as the users. The questionnaire design is shown in Figure 11.

Table 1. System Test Results Using Black Box and Questionnaire

Testing Activity	Test Case	Expected Realization	Test Result
Access Rights: Network Administrator			
Add new VLAN ID and VLAN name data	ID VLAN : 2 Name : VLAN Server	The system saves the new device brand data and displays the results of adding new device brand data	✓ Correct (valid)
Modify VLAN name data (edit)	Name : VLAN Server Cawang	The system changes the device brand data correctly and displays the results of changing the device brand data	✓ Correct (valid)
Delete VLAN data	ID VLAN : 2 Name : VLAN Server Cawang	The system deletes the device brand data correctly and displays the results of deleting the device brand data	✓ Correct (valid)
Add new VLAN config on the selected network devices	Brand : Cisco Type : 2960 SN : 8764763999 Add VLAN : VLAN Server Cawang	The system saves the new VLAN configuration correctly and displays a notification that the new VLAN configuration on the selected device has been successful	✓ Correct (valid)
Add port interface config by specifying VLAN based on selected port type	Interface port : 1/1/12 Port Type : Access VLAN : VLAN User Cawang Lantai 1	The system saves the new port interface configuration correctly and displays a notification that the new port interface configuration on the selected device has been successful	✓ Correct (valid)

TESTING QUESTIONNAIRE
“WEBSITE-BASED NETWORK ADMINISTRATIVE AUTOMATION SYSTEM”

Please fill in by putting a mark “√” on the answer that fits with your opinion.

Information : SA : Strongly Agree D : Disagree
 A : Agree SD : Strongly Disagree
 N : Neutral

NETWORK ADMINISTRATOR

System View

	Statement	SA	A	N	D	SD
1	The system has an easy-to-understand appearance	√				
2	The system has an attractive appearance (matching design and color)		√			
3	The font used in the system is appropriate		√			
4	Available buttons are easy to understand		√			
5	The system is comfortable to use		√			

System Process

	Statement	SA	A	N	D	SD
1	The process on system doesn't take a long time • Add, Edit, and Delete Location Area (Device Administration Staff) • Add, Edit, and Delete Device Brand (Device Administration Staff) • Add, Edit, and Delete Device (Device Administration Staff) • Add Device Installation (Network Infrastructure Technician) • Device Installation <u>Verification</u> (Network Management Coordinator) • Add, Edit, and Delete VLANs (Network Administrator) • VLAN's Configuration (Network Administrator) • Interface Port Configuration (Network Administrator)	√				
2	All system features can work properly and correctly		√			
3	The system helps the business process become easier and well-documented • Network device logging process • The process of recording the installation of network devices • The process of checking installed devices • Network device configuration process	√				
4	The system has met the needs of network administrators		√			

Figure 11. System Test Results Using Questionnaire

From the results of direct assessment testing on 18 users using a questionnaire, then an analysis of the test results was carried out. For survey calculations from the system view aspect in statement number 1 regarding the easy-to-understand appearance that has been filled in by the user, the following results are obtained:

$$Y = \frac{77}{90} \times 100\% = 85\%$$

Table 2. The Results of The First Statement Questionnaire from The Aspect of The System View

Statement	Description	Scale (N)	Respondent (R)	N x R
1	Strongly Agree (SA)	5	5	25
	Agree (A)	4	13	52
	Neutral (N)	3	0	0
	Disagree (D)	2	0	0
	Strongly Disagree (SD)	1	0	0
Sum			18	77

Based on the percentage value of the first statement that shown by Table 2, it can be concluded that as many as 85% of respondents or users stated that they strongly agree that the website-based network administration automation system has an easy-to-understand appearance.

5. Conclusion

5.1 Conclusion

The implementation of a network administration automation system using the Laravel framework and the Python library shows that based on the results of blackbox testing all features can function properly or have run as expected, such as the system being able to add, modify, and delete VLAN data; configure VLANs; and configure network interfaces. Besides that, based on the results of direct assessment of 18 users, it showed an average of 82% of users strongly agree that the system has an easy-to-understand appearance, attractive (design and color harmony), the fonts used are appropriate, the available buttons are easy to understand, comfortable to use. , the system process does not take a long time, all features function properly and have met user requirements. In addition, the results of the direct assessment to users stated that the system helped the process of recording network devices, recording network device installation, checking network devices that had been installed, and network device configuration easier and well-documented. With this, it can produce valid and integrated network management data so as to minimize the time needed by network administrators to configure and reduce the possibility that network administrators often make configuration errors on each device.

5.2 Suggestions

This research focuses on VLAN configuration and port interface configuration, so that for further research it is possible to develop device routing configuration features. In addition, this research focuses on website-based configuration automation, so that for further research it can be carried out to develop automation using social media by linking website-based network administration automation systems with social media so that configuration can be done using social media.

6. References

- [1] Rahmawati, "Konfigurasi Keamanan Jaringan Komputer Pada Router Dengan Metode Acl'S," *Journal of the American Chemical Society*, vol. I, no. 2, pp. 152–158, 2015.
- [2] L. G. Mauboy and T. Wellem, "Studi Perbandingan Library Untuk Implementasi Network Automation Menggunakan Paramiko Dan Netmiko Pada Router Mikrotik," *JURIKOM (Jurnal Riset Komputer)*, vol. 9, no. 4, pp. 790–799, 2022, doi: 10.30865/jurikom.v9i4.4420.
- [3] V. V. S. S. Balaram, C. Mukundha, and S. Bhutada, "Enhancement of Network Administration through Software Defined Networks," *IOSR Journal of Computer Engineering*, vol. 18, no. 1, pp. 30–36, 2016, doi: 10.9790/0661-18113036.
- [4] R. A. Wiryawan and N. R. Rosyid, "Pengembangan Aplikasi Otomatisasi Administrasi Jaringan Berbasis Website Menggunakan Bahasa Pemrograman Python," *Simetris*, vol. 10, no. 2, pp. 1–12, 2019.
- [5] G. S. Santyadiputra, I. M. E. Listartha, and G. A. J. Saskara, "The effectiveness of Automatic Network Administration (ANA) in network automation simulation at Universitas Pendidikan Ganesha," *Journal of Physics: Conference Series*, vol. 1810, no. 1, 2021, doi: 10.1088/1742-6596/1810/1/012028.
- [6] P. Mihailă, T. Bălan, R. Curpen, and F. Sandu, "Network Automation and Abstraction using Python Programming Methods,"

- MACRo - International Conference on Recent Achievements in Mechatronics, Automation, Computer Science and Robotics*, vol. 2, no. 1, pp. 95–103, 2017, doi: 10.1515/macro-2017-0011.
- [7] J. Edelman, S. Lowe, and M. Oswalt, *Network Programmability and Automation*. 2018.
- [8] I. Pinto, *Network Automation Made Easy*. Cisco Press, 2021.
- [9] K. Abuelenain, J. Doyle, A. Karneliuk, and V. Jain, *Network Programmability and Automation Fundamentals*. Cisco Press, 2021.
- [10] U. Admin, “Pemula Wajib Tahu, Python Dianggap sebagai High Level Programming,” 2022. <https://www.umn.ac.id/pemula-wajib-tahu-python-dianggap-sebagai-high-level-programming/>.
- [11] S. Hörning and B. Haese, “RMWSPy (v 1.1): A Python code for spatial simulation and inversion for environmental applications,” *Environmental Modelling and Software*, vol. 138, 2021, doi: 10.1016/j.envsoft.2021.104970.
- [12] B. Rhodes and J. Goerzen, *Foundations of Python Network Programming*. New York: Apress, 2010.
- [13] E. Chou, *Mastering Python Networking*. Birmingham: Packt Publishing, 2017.
- [14] Aminudin, *Cara Efektif Belajar Framework Laravel*. Yogyakarta: Lokomedia, 2015.
- [15] A. D. Manuputty, S. Hendrawan, and B. Haryanto, “Design of Information Systems for Research Permit Application with Agile Method and Website Based Laravel Framework,” *Journal of Information Systems and Informatics*, vol. 2, no. 1, pp. 60–78, 2020, doi: 10.33557/journalisi.v2i1.45.
- [16] D. Naista, *Codeigniter Vs Laravel - Kasus Membuat Website Pencari Kerja*. Yogyakarta: Lokomedia, 2017.
- [17] X. Chen, Z. Ji, Y. Fan, and Y. Zhan, “Restful API Architecture Based on Laravel Framework,” *Journal of Physics: Conference Series*, vol. 910, no. 1, 2017, doi: 10.1088/1742-6596/910/1/012016.
- [18] E. S. Soegoto, “Implementing Laravel Framework Website as Brand Image in Higher-Education Institution,” *IOP Conference Series: Materials Science and Engineering*, vol. 407, no. 1, 2018, doi: 10.1088/1757-899X/407/1/012066.
- [19] R. Abdulloh, *Membuat Aplikasi Point of Sale dengan Laravel dan AJAX*. Jakarta: PT ELEX MEDIA KOMPUTINDO, 2017.
- [20] M. Fahmi, M. Maisyaroh, I. Komarudin, S. Faizah, and I. Fadhilah, “Otomatisasi Jaringan Menggunakan Script Python Untuk Penyediaan Konfigurasi Internet Dan Manajemen Mikrotik,” *Bina Insani Ict Journal*, vol. 8, no. 1, p. 53, 2021, doi: 10.51211/biict.v8i1.1517.
- [21] D. Rahardika and N. Ratama, “Implementasi Network Automation Untuk Konfigurasi Jaringan Baru Dengan Netmiko,” vol. 2, no. 3, pp. 190–200, 2021.
- [22] P. Otomatisasi, J. Berbasis, A. Aziz, and H. Haerudin, “Perancangan Otomatisasi Jaringan Berbasis Web Dengan Django (Studi Kasus PT. Platinum Citra Indonesia),” *OKTAL : Jurnal Ilmu Komputer dan Sains*, vol. 1, no. 06, pp. 583–592, 2022.
- [23] Y. Chandra et al., “Website Network Automation Design And Implementation In RT RW Net Senden Dusun Magelang With Django Framework,” *Jurnal Teknik Informatika (JUTIF)*, vol. 3, no. 5, pp. 1313–1322, 2022.

