

Pengembangan Purwarupa Laboratorium Virtual Berbasis VMWare dengan Terraform

Michael Putra Kusuma¹, Nur Rohman Rosyid^{1,*}

¹Departemen Teknik Elektro dan Informatika, Sekolah Vokasi, Universitas Gadjah Mada;

michael.p.k@mail.ugm.ac.id

*Korespondensi: nrohmanr@ugm.ac.id;

Abstract – In recent years, Virtual Laboratory have increasingly been used by organizations around the world as learning media and a place to practice skills in the field of information technology by providing disposable virtual machines labs that can be created and terminated using buttons found on learning websites. With a Virtual Laboratory prototype that implements Infrastructure as Code, it will make things easier for infrastructure Administrators to manage virtual infrastructure. We need a website based interface that can create virtual machine lab, view the status of the created virtual machine lab and delete the virtual machine lab that have been created if they no longer want to use the virtual machine lab. The development of Virtual Laboratory prototype website uses PHP as server side scripting to run scripts for each of its features. Infrastructure as Code automation in this prototype functions as an automation of the management and provision of virtual machine lab on VMWare ESXi using Terraform. Before being able to access the website, users who are registered in the database are required to login first so that they can be directed to the folders that have been provided for each user. The hope with this website application is that it can make it easier for laboratory infrastructure Administrators to manage virtual machines labs, so there is no need to bother manually configuring each virtual machine for each user.

Keywords – Virtual Laboratory, VMWare ESXi, Terraform, Infrastructure as Code

Intisari – Pada tahun-tahun terakhir, Laboratorium Virtual telah semakin marak digunakan oleh organisasi di seluruh dunia sebagai media pembelajaran dan tempat melatih *skill* di bidang teknologi informasi dengan menyediakan *virtual machine lab* yang dapat dibuat dan dihentikan menggunakan tombol yang terdapat di *website* pembelajaran. Dengan adanya purwarupa Laboratorium Virtual yang menerapkan *Infrastructure as Code*, maka hal tersebut akan memudahkan *Administrator* infrastruktur dalam mengelola infrastruktur virtual. Diperlukannya sebuah antarmuka berbasis *website* yang dapat melakukan pembuatan *virtual machine lab*, melihat status *virtual machine lab* yang dibuat serta menghapus *virtual machine lab* yang telah dibuat jika sudah tidak ingin menggunakan *virtual machine lab* tersebut. Pengembangan purwarupa *website* Laboratorium Virtual ini menggunakan PHP sebagai *server side scripting* untuk menjalankan *script* dari tiap fiturnya. *Infrastruktur as Code* pada purwarupa ini berfungsi sebagai otomatisasi pengelolaan dan penyediaan *virtual machine lab* pada VMWare ESXi menggunakan Terraform. Sebelum dapat mengakses *website* tersebut, *user* yang terdaftar di *database* diharuskan untuk *login* terlebih dahulu sehingga dapat diarahkan ke *folder* yang telah disediakan untuk masing-masing *user*. Harapan dengan adanya aplikasi *website* ini adalah dapat memudahkan *Administrator* infrastruktur laboratorium dalam melakukan manajemen *virtual machine lab*, sehingga tidak perlu repot-repot lagi untuk melakukan konfigurasi tiap *virtual machine* secara manual bagi tiap *user*.

Kata kunci – Laboratorium Virtual, VMWare ESXi, Terraform, Infrastructure as Code

I. PENDAHULUAN

Pada era revolusi industri 4.0 ini, perkembangan teknologi yang digunakan oleh umat manusia terus mengalami kemajuan dan perkembangan yang sangat cepat dan pesat terutama di bidang infrastruktur dan virtualisasi. Akibat dari hal tersebut adalah semakin banyaknya penggunaan infrastruktur virtual pada organisasi di seluruh dunia. Salah satu contoh penerapan infrastruktur virtual adalah Laboratorium Virtual. Organisasi di dunia akhir-akhir ini banyak yang menggunakan Laboratorium Virtual pada *website* pembelajaran mereka sebagai salah satu media pembelajaran untuk melatih keterampilan penggunaannya, khususnya dalam di bidang teknologi informasi [1]. Dalam bidang virtualisasi, VMWare ESXi merupakan produk *Hypervisor* gratis milik VMWare yang dapat digunakan untuk men-deploy *Virtual Machines* (VM) dalam rangka menciptakan infrastruktur virtual untuk organisasi [2]. Sebelum Laboratorium Virtual terkenal, *Administrator*

infrastruktur laboratorium secara manual dan berulang mengelola infrastruktur virtual. *Administrator* membuat VM baru, mengkonfigurasi VM, menyalakan VM, memberi IP *Address* untuk VM, dan menghapus VM yang sudah tidak digunakan. Semakin bertambahnya VM yang terdapat pada *Hypervisor*, maka akan semakin bertambah pula waktu dan beban kerja yang dimiliki oleh *Administrator* infrastruktur, bahkan hal tersebut berpotensi menimbulkan terjadinya *human error*. Maka dari itu, diperlukan pengelolaan infrastruktur virtual berbasis kode atau yang biasa disebut *Infrastructure as Code* (IaC) [3]. IaC sangat cocok untuk Laboratorium Virtual yang melakukan hal-hal berulang dalam mengelola VM, IaC juga sangat penting karena dapat mengurangi waktu yang dihabiskan untuk melakukan hal-hal berulang tersebut, serta dapat memastikan konfigurasi VM yang akan dibuat, sehingga dapat mengurangi terjadinya *human error* [4].

Untuk mempermudah *Administrator* infrastruktur laboratorium dalam mengelola Laboratorium Virtual, pada VMWare ESXi, dibuatlah IaC dengan menggunakan Terraform. Selain IaC, diperlukan sebuah *website* sebagai antarmuka untuk pengguna laboratorium virtual sebagai aplikasi untuk membuat VM, menampilkan status berupa informasi dari VM yang dibuat pada antarmuka *website* serta menghapus VM tersebut jika sudah tidak digunakan. Sebelum dapat melakukan hal tersebut pada *website*, diperlukan fitur keamanan berupa halaman *login* yang akan mengarahkan pengguna ke halamannya masing-masing, sehingga pengguna tidak dapat mengakses dan menghapus VM milik pengguna lainnya. Mengelola infrastruktur virtual akan semakin sulit dilakukan seiring bertambahnya jumlah pengguna. Semakin banyak VM yang dibuat, maka akan memakan waktu yang banyak pula untuk mencari satu persatu VM yang digunakan oleh pengguna beserta IP *Address*-nya.

Diharapkan dengan adanya purwarupa Laboratorium Virtual ini akan dapat membantu *Administrator* infrastruktur laboratorium agar tidak perlu lagi secara manual melakukan pembuatan VM, mengkonfigurasi IP *Address* ke VM, menginfokan IP *Address* ke setiap pengguna, mengkonfigurasi VM, menyalakan VM, dan menghapus VM yang sudah tidak digunakan, karena hal tersebut dapat dilakukan secara berulang menggunakan otomatisasi dari kode, sehingga lebih hemat waktu dan efisien dalam mengelola infrastruktur virtual.

II. DASAR TEORI

A. Laboratorium Virtual

Laboratorium Virtual merupakan sistem yang dapat digunakan untuk mendukung praktikum yang berjalan secara konvensional. Laboratorium Virtual dapat menjadi media pembelajaran yang digunakan sebagai solusi keterbatasan perangkat komputer [1].

Salah satu penerapan Laboratorium Virtual dilakukan oleh [5] mengenai “*The Ball and Beam System: A Case Study of Virtual and Remote Lab Enhancement With Moodle*”. Penelitian tersebut mendemonstrasikan manfaat dan pentingnya Laboratorium Virtual pada pendidikan jarak jauh, serta menunjukkan dampak positif dari *Virtual* dan/atau *Remote Lab* (VRL) dan penerapan *Moodle* terhadap pembelajaran siswa.

B. VMWare ESXi

Hypervisor merupakan lapisan virtualisasi yang berfungsi sebagai fondasi untuk lini produk lainnya. Di vSphere versi 5 dan yang lebih baru, *Hypervisor* dari produk vSphere hanya hadir dalam bentuk VMWare ESXi. VMWare ESXi adalah *Type 1 bare-metal Hypervisor* yang berjalan langsung pada perangkat keras sistem [2].

Salah satu penerapan VMWare ESXi dilakukan oleh [6] mengenai “*Virtualization and cyber security: arming future security practitioners*”. Penelitian tersebut menunjukkan bahwa VMWare ESXi keluar sebagai pemenang dari ketiga *platform* virtualisasi yang telah dipelajari (VMWare ESXi, Xenserver dan Hyper-V) setelah menjalankan lingkungan lab virtual untuk mengajar kursus keamanan di tingkat universitas. Dalam lingkungan lab virtual, fleksibilitas adalah kualitas penting karena perlu menyiapkan banyak skenario dengan parameter yang bervariasi. Konfigurasi dan opsi implementasi di VMWare adalah yang paling banyak intuitif, opsi manajemen pengguna lebih terperinci dan komprehensif, bahkan saat menjalankan ESXi sebagai *server* mandiri pada saat sistem lain membutuhkan konfigurasi yang jauh lebih kompleks untuk mencapai kontrol akses granular.

C. Infrastructure as Code

Infrastructure as Code (IaC) adalah pendekatan otomatisasi infrastruktur berdasarkan praktik dari pengembangan perangkat lunak. IaC menekankan rutinitas yang konsisten dan berulang untuk penyediaan (*provisioning*) dan perubahan sistem beserta konfigurasinya. *Administrator* membuat perubahan pada kode, lalu menggunakan otomatisasi untuk menguji dan menerapkan perubahan tersebut ke sistem yang dimiliki [3].

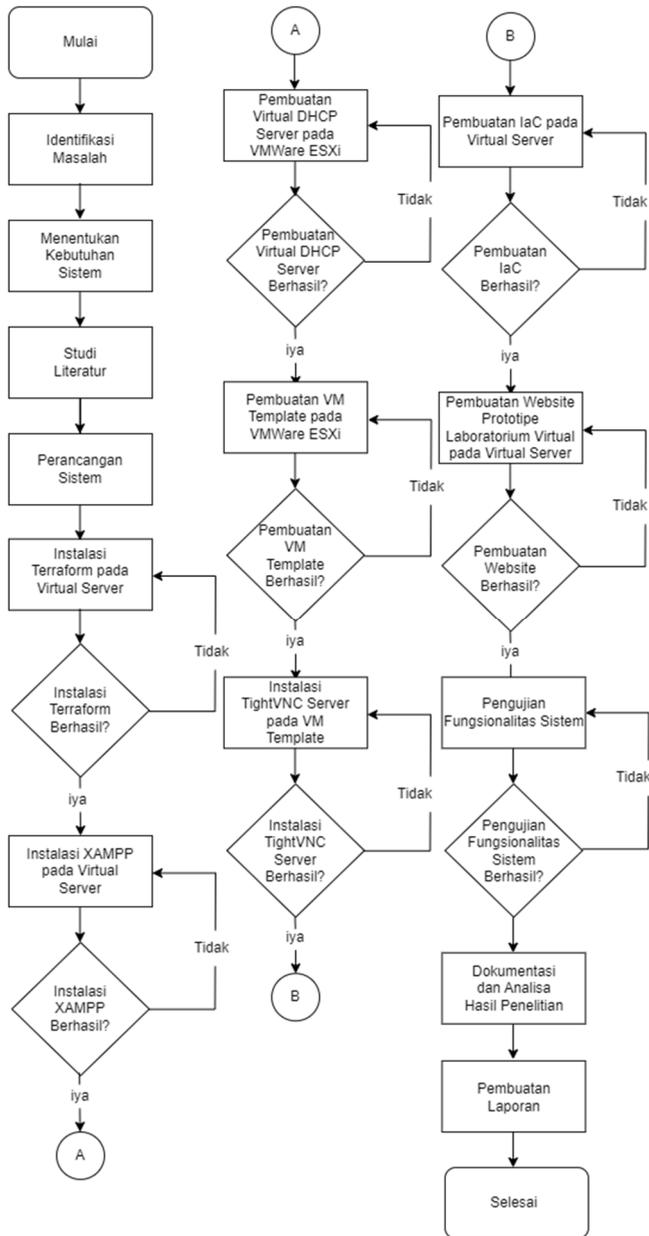
Terraform adalah *tool* IaC yang memungkinkan penggunaannya untuk menentukan sumber daya *cloud* dan lokal dalam *file* konfigurasi yang dapat dibaca manusia, dapat dibuat versi, digunakan kembali, dan dibagikan. Pengguna kemudian dapat menggunakan alur kerja yang konsisten untuk menyediakan dan mengelola semua infrastruktur pengguna sepanjang siklus hidupnya. Terraform dapat mengelola komponen tingkat rendah, seperti komputasi, penyimpanan, dan sumber daya jaringan [7].

Salah satu penerapan IaC dilakukan oleh [8] mengenai “*Infrastructure as Code for Business Continuity in Institutions of Higher Learning*”. Penelitian tersebut mendapatkan kesimpulan bahwa IaC memiliki manfaat yang sangat besar seperti penghematan waktu, konfigurasi awal dan pembaruan konfigurasi, investasi awal pada perangkat keras, infrastruktur yang andal dan terukur, serta *Recovery Time Objective* (RTO) yang lebih baik.

III. METODOLOGI

A. Tahap Penelitian

Tahapan penelitian dimulai dari identifikasi masalah, menentukan kebutuhan sistem, studi literatur, perencanaan sistem, hingga instalasi Terraform pada Virtual Server. Tahapan penelitian secara jelas disajikan pada Gambar 1 Diagram Alir Penelitian.



Gambar 1. Diagram Alir Penelitian

B. Alat dan Bahan

Terdapat beberapa bahan perangkat lunak dan perangkat keras yang diperlukan untuk menjalankan Proyek Akhir ini. Bahan-bahan yang digunakan untuk Proyek Akhir ini adalah sebagai berikut:

1) Perangkat Keras

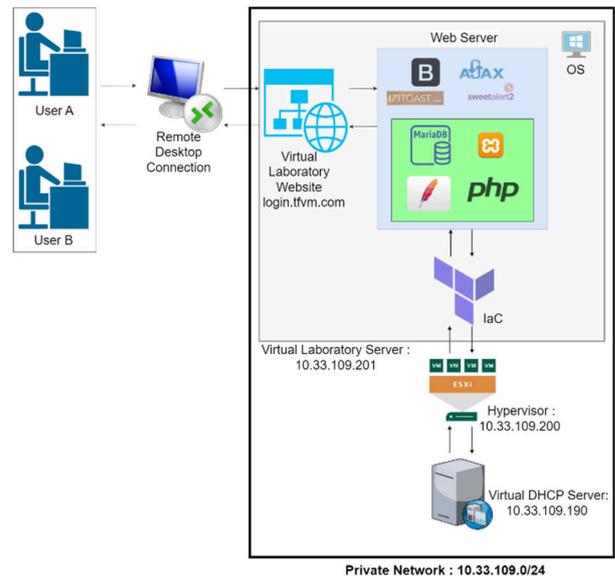
- Laptop
- VMWare ESXi
- Server purwarupa Laboratorium Virtual

2) Perangkat Lunak

- Sistem Operasi Windows 10 Home
- Sistem Operasi Kali Linux 2022.2
- Sistem Operasi CentOS 7
- Terraform 1.2.2
- VMWare Tools

- VMWare ESXi 7.0 U2
- XAMPP 7.2.34
- Terraform *provider esxi plugin* 1.10.2
- OVF Tool 4.4.3
- GO 1.18.3
- TightVNC (1.3.10 Kali Linux, 2.8.63 Windows)
- OpenVPN Connect 3.3.5
- Remote Desktop Connection
- jQuery 3.6.0
- Izitoast 1.4.0
- Sweetalert2 11.4.23
- Bootstrap 5.2
- OpenVPN Connect 3.3.5
- Dhcpd
- Wireshark
- Nmap

C. Perancangan Sistem

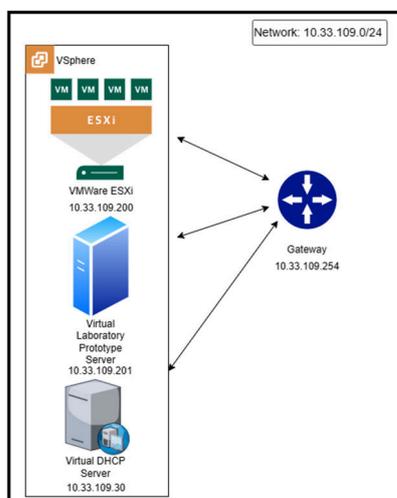


Gambar 2. Diagram Sistem Arsitektur Purwarupa

Alir kerja sistem purwarupa Laboratorium Virtual dapat dilihat pada Gambar 2. Proses dimulai dari *user* melakukan *remote access* ke *server* dari purwarupa Laboratorium Virtual menggunakan *Remote Desktop Connection* [9], aplikasi bawaan Windows untuk melakukan RDP. Pada *server* purwarupa Laboratorium Virtual inilah *web server* dari *website* purwarupa Laboratorium Virtual yang menggunakan XAMPP [10]. Hal yang dilakukan saat sudah dapat mengakses halaman *website* purwarupa Laboratorium Virtual adalah melakukan *login* dengan memasukkan *username* dan *password* agar *user* terverifikasi untuk dapat mengakses halaman utama dari *website* purwarupa Laboratorium Virtual milik setiap *user*. *Website* kemudian menggunakan Ajax [11] untuk melakukan verifikasi terhadap *username* dan *password* yang telah dimasukkan oleh *user* dengan data akun yang sudah ada pada *database* MariaDB [12]. Izitoast dan Sweetalert2 berfungsi sebagai notifikasi dan *alert* saat proses memverifikasi *user* [13], *website* akan menampilkan *alert* jika *user* salah memasukkan *username* atau *password*,

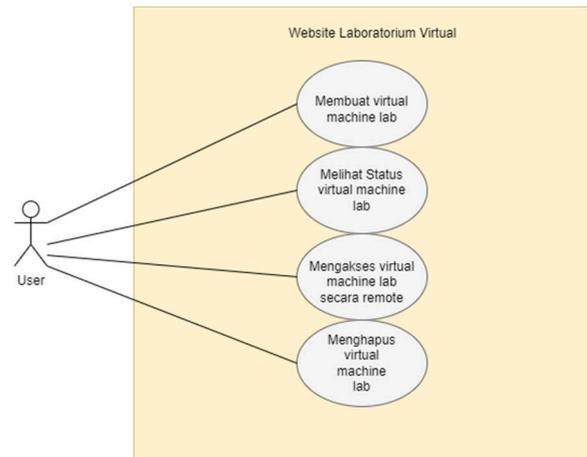
tidak memasukkan *username* atau *password* dan menampilkan notifikasi berhasil *login* jika *username* dan *password* yang dimasukkan benar. Setelah berhasil *login*, *user* dapat memilih salah satu dari empat modul purwarupa Laboratorium Virtual sebagai pilihan yang tersedia pada *submenu* dari menu *Deploy* yang terdapat di *navigation bar*. Keempat modul tersebut adalah “1 Linux”, “2 Linux”, “1 Windows”, dan “2 Windows”. Setiap modul tersebut berisi *virtual machine lab* sesuai dengan namanya. *Navigation bar* tersebut dibuat dengan menggunakan Bootstrap [14].

Pada setiap halaman dari keempat modul purwarupa Laboratorium Virtual terdapat tiga tombol yang tersedia untuk *user*. Saat tombol ditekan oleh *user*, PHP [15] akan mengeksekusi perintah Terraform [7] pada *Command Prompt* di *background* dari *server* purwarupa Laboratorium Virtual untuk menjalankan Terraform sesuai dengan fungsi Terraform yang ada di setiap tombolnya. Pertama, *user* menekan tombol “Start VM” untuk membuat *virtual machine lab*, saat tombol “Start VM” tombol ditekan VMWare ESXi akan membuat *virtual machine lab* menggunakan VM *template* yang telah tersedia. *Virtual machine lab* tersebut kemudian akan mendapatkan IP Address dari *Virtual DHCP Server* [16], setelah berhasil dibuat. Setelah *virtual machine lab* berhasil di-*deploy* Terraform akan mengeluarkan *output* pada kotak *output* di bawah tombol “Start VM” berisi durasi waktu untuk membuat *virtual machine lab* tersebut dan *output* berupa status dari *virtual machine lab*. Setelah itu, *user* menekan tombol “VM Resource” untuk melihat status dari *virtual machine lab* yang telah di-*deploy*, kemudian *user* dapat mengakses *virtual machine lab* tersebut secara *remote* lewat aplikasi TightVNC Viewer [17]. Terakhir, *user* menekan tombol “Stop VM” untuk menghapus *virtual machine lab* yang telah digunakan dari VMWare ESXi. *Virtual machine lab* yang telah di-*deploy* ini adalah hasil *clone* oleh Terraform dari VM *Template* yang telah dibuat pada VMWare ESXi. Terdapat dua VM *Template* pada penelitian ini, *Linux VM Template* dengan OS Kali Linux dan *Windows VM Template* dengan OS Windows 10 Home. *User* dapat keluar dari akun yang digunakan dengan menekan menu *Logout* pada *navigation bar*.



Gambar 3. Topologi Jaringan Sistem

Topologi jaringan sistem dapat dilihat pada Gambar 3. Pada rancangan sistem terdapat satu buah *server* untuk *website* purwarupa Laboratorium Virtual, satu buah VMWare ESXi, dan satu buah *Virtual DHCP Server*. *Server* purwarupa Laboratorium Virtual, *Virtual DHCP Server*, dan VMWare ESXi tersebut berada dalam jaringan yang sama yaitu “10.33.109.0/24” dengan *gateway* “10.33.109.254”.



Gambar 4. User Case Diagram

Dari pembahasan *website* purwarupa Laboratorium Virtual di atas, *user* memiliki akses untuk dapat membuat, melihat *status*, mengakses secara *remote* dan menghapus *virtual machine lab*. Seperti *user case diagram* pada Gambar 4, tidak ada *role user* secara khusus dan setiap *user* dapat menjalankan empat fitur utama yang ada pada *website* purwarupa Laboratorium Virtual.

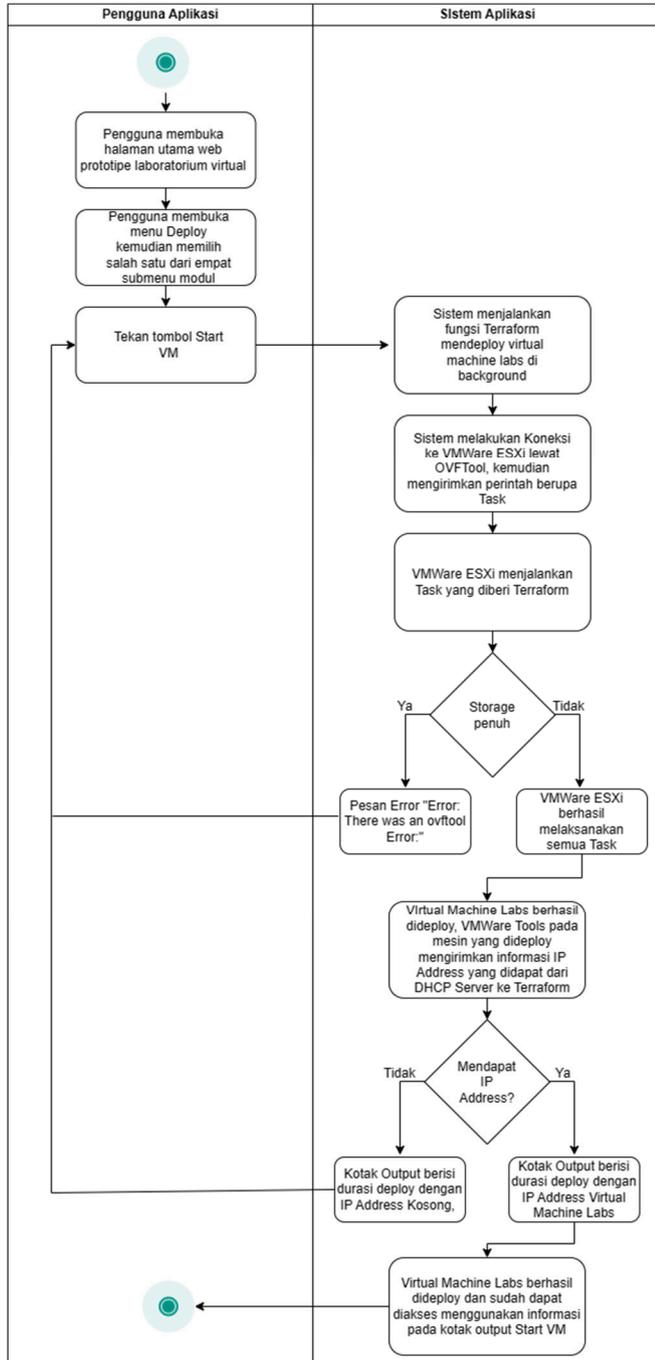
IaC pada penelitian ini digunakan untuk otomatisasi pengelolaan dan penyediaan *virtual machine lab*. *Tools* IaC yang digunakan pada penelitian ini adalah Terraform. Terraform memiliki tiga fungsi utama pada penelitian ini sebagai IaC [7]. Fungsi tersebut adalah *deploy virtual machine lab* menggunakan *command* “*terraform ini*” dan “*terraform apply -auto-approve*”, melihat *status virtual machine lab* menggunakan *command* “*terraform output*” dan menghapus *virtual machine lab* menggunakan *command* “*terraform destroy -auto-approve*”. *Provider* yang digunakan pada penelitian ini adalah terraform provider esxi [18].

D. Skenario Pengujian

Pengujian yang dilakukan pada penelitian ini menggunakan metode *Black Box Testing*. Pengujian ini dilakukan sebagai pengujian fungsionalitas setiap fitur pada sistem purwarupa Laboratorium Virtual yang telah dikembangkan. Proses pengujian sistem aplikasi dengan metode *Black Box Testing* tidak terlalu perlu mengetahui proses internal aplikasi dan kode program secara detail, tetapi cukup diketahui proses *testing* dari sisi bagian luar saja [19]. Metode tersebut akan digunakan untuk menguji fungsionalitas ketujuh fitur yang terdapat pada purwarupa Laboratorium Virtual.

1) Pengujian Fitur *Start VM*

Fitur *Start VM* berfungsi untuk membuat *virtual machine lab* dengan menjalankan fungsi *deploy* Terraform, PHP akan mengeksekusi *script* untuk menjalankan fungsi *deploy* Terraform di *background* saat tombol “Start VM” ditekan oleh *user*. Setelah *virtual machine lab* berhasil di-*deploy*, maka *output* fungsi Terraform yang berjalan di *background* akan ditampilkan oleh PHP di kotak *output* “Start VM”. Proses fitur *Start VM* dapat dilihat pada Gambar 5.

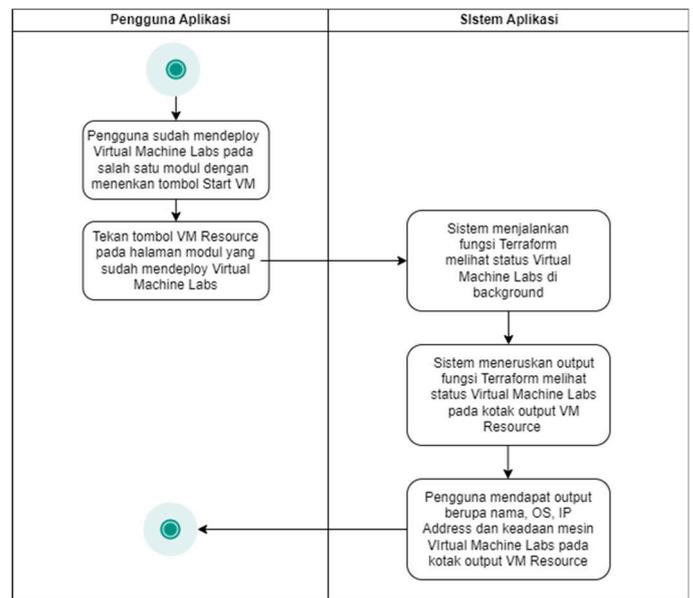


Gambar 5. Activity Diagram Fitur *Start VM*

Pengujian fitur ini bertujuan untuk memastikan sistem purwarupa Laboratorium Virtual dapat membuat *virtual machine lab* sesuai dengan modulnya dan *website* purwarupa Laboratorium Virtual dapat mengeluarkan *output* dari fungsi Terraform yang dijalankan pada kotak *output* di bawah tombol “Start VM”.

1) Pengujian Fitur *VM Resource*

Fitur *VM Resource* berfungsi untuk menampilkan *resource* berupa status dari *virtual machine lab* yang telah di-*deploy* pada halaman modul tersebut. PHP akan menjalankan fungsi Terraform melihat status *virtual machine lab* setelah tombol “VM Resource” ditekan oleh *user*. Setelah fungsi Terraform melihat status berhasil dieksekusi, maka *output* fungsi Terraform yang berjalan di *background* akan ditampilkan oleh PHP di kotak *output* “VM Resource”. Proses fitur *VM Resource* dapat dilihat pada Gambar 6.

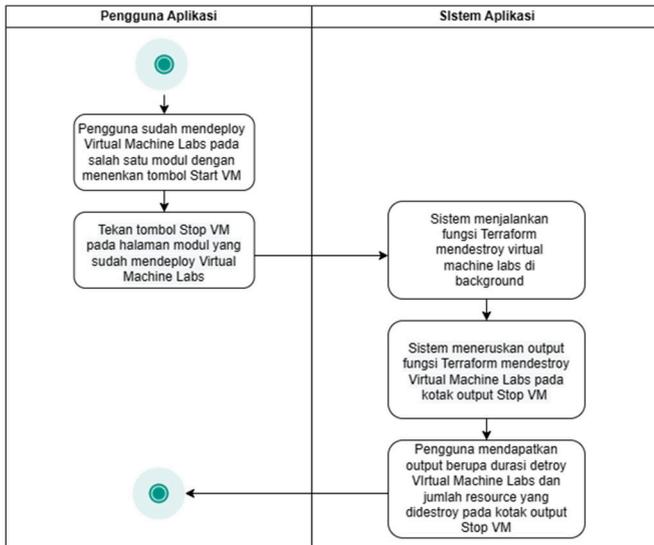


Gambar 6. Activity Diagram Fitur *VM Resource*

Pengujian fitur ini bertujuan untuk memastikan sistem purwarupa Laboratorium Virtual dapat menampilkan informasi status dari *virtual machine lab* yang sebelumnya telah di-*deploy* pada kotak *output* dibawah tombol “VM Resource”.

2) Pengujian Fitur *Stop VM*

Fitur *Stop VM* berfungsi untuk menghapus *virtual machine lab* yang telah di-*deploy* pada halaman modul. Fitur ini hanya berlaku untuk modul pada halaman modulnya sendiri, sehingga fitur ini tidak dapat men-*destroy resource* milik halaman modul lain, apalagi men-*destroy* milik *user* lain. PHP akan menjalankan fungsi Terraform menghapus *virtual machine lab* setelah tombol “Stop VM” ditekan oleh *user*. Setelah fungsi Terraform menghapus *virtual machine lab* berhasil dieksekusi, *output* fungsi Terraform yang berjalan di *background* akan ditampilkan oleh PHP di kotak *output* “Stop VM”. Proses fitur *Stop VM* dapat dilihat pada Gambar 7.

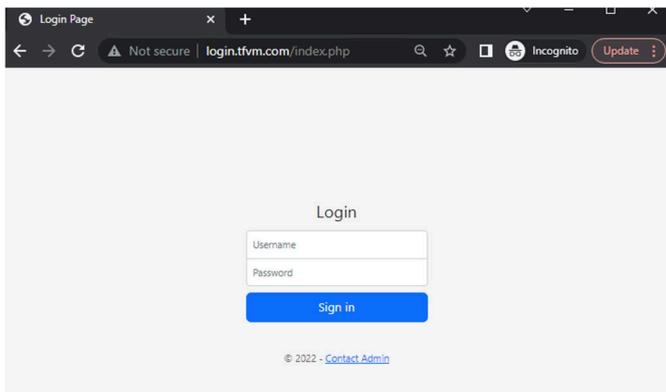


Gambar 7. Activity Diagram Fitur Stop VM

Pengujian fitur ini bertujuan untuk memastikan bahwa sistem purwarupa Laboratorium Virtual dapat menghapus *virtual machine lab* yang telah di-deploy dan *website* purwarupa Laboratorium Virtual dapat mengeluarkan *output* dari fungsi Terraform yang dijalankan pada kotak *output* di bawah tombol “Stop VM”.

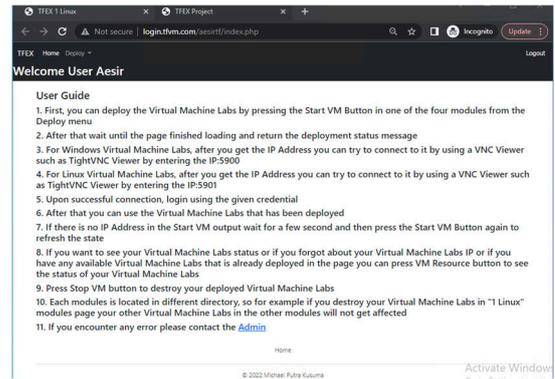
IV. HASIL DAN PEMBAHASAN

A. Tampilan Aplikasi



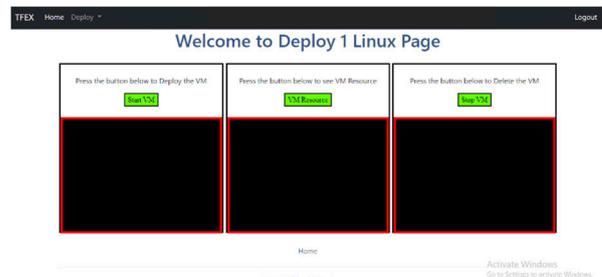
Gambar 8. Tampilan Halaman Login

Pengembangan antarmuka *website* purwarupa Laboratorium Virtual ini dibuat menggunakan Ajax, Bootstrap, SweetAlert2 dan iziToast, serta PHP dan MariaDB untuk *backend*. Pada Gambar 8, halaman utama *website* purwarupa Laboratorium Virtual terdiri dari *field username* dan *password* untuk memasukkan kredensial milik *user*.



Gambar 9. Tampilan Halaman Utama

Pada Gambar 9, halaman utama *website* purwarupa Laboratorium Virtual terdapat tiga menu yaitu “Home”, “Deploy” dan “Logout”, serta *user guide* dari purwarupa Laboratorium Virtual. Menu “Home” berfungsi untuk mengembalikan *user* ke halaman utama. Menu “Deploy” terdiri dari empat sub-menu berisi halaman modul purwarupa Laboratorium Virtual, keempat modul tersebut yaitu modul “1 Linux” untuk membuat satu *virtual machine lab* Linux, modul “2 Linux” untuk membuat 2 *virtual machine lab* Linux, modul “1 Windows” untuk membuat satu *virtual machine lab* Windows dan modul “2 Windows” untuk membuat 2 *virtual machine lab* Windows. Menu “Logout” berfungsi untuk menjalankan fitur *Logout*. *User guide* pada halaman utama merupakan panduan kepada *user* tentang purwarupa Laboratorium Virtual dan berisi cara *user* untuk dapat mengakses *virtual machine lab*.



Gambar 10. Tampilan Halaman Modul

Pada Gambar 10, halaman modul pada menu “Deploy” memiliki tampilan yang sama, kecuali bagian judul halaman di tengah. Pada halaman modul terdapat tiga fitur dengan tombol dan kotak *output* dibawahnya. Ketiga fitur tersebut adalah fitur *Start VM*, fitur *Stop VM* dan fitur *VM Resource*.

B. Hasil Black Box Testing

Seluruh proses pengujian dilakukan dengan membuat tabel berisikan skenario dan kasus pengujian yang akan digunakan sebagai acuan pengujian dari sistem purwarupa Laboratorium Virtual. Hasil *Black Box Testing* dapat dilihat pada Tabel 1.

Tabel 1. Hasil *Black Box Testing*

No	Skenario	Kasus Pengujian	Hasil		
1.	Fitur <i>Start VM</i>	User aesir modul 1 Linux	Berhasil		
		User aesir modul 1 Windows	Berhasil		
		User aesir modul 2 Linux	Berhasil		
		User aesir modul 2 Windows	Berhasil		
		User busir modul 1 Linux	Berhasil		
		User busir modul 1 Windows	Berhasil		
		User busir modul 2 Linux	Berhasil		
		User busir modul 2 Windows	Berhasil		
		User aesir modul 1 Linux dan User busir modul 1 Windows secara bersamaan	Berhasil		
		User aesir modul 1 Windows dan User busir modul 1 Linux secara bersamaan	Berhasil		
		User aesir modul 1 Windows dan User busir modul 1 Linux secara bersamaan	Berhasil		
		User aesir modul 1 Linux dan User busir modul 1 Windows secara bersamaan	Berhasil		
		User aesir modul 1 Windows dan User busir modul 2 Linux secara bersamaan	Berhasil		
		User aesir modul 2 Linux dan User busir modul 2 Windows	Berhasil		
		User busir modul 1 Linux dan User busir modul 1 Windows	Berhasil		
		User busir modul 2 Linux dan User busir modul 2 Windows	Berhasil		
		2.	Fitur <i>VM Resource</i>	User aesir modul 1 Linux dan User busir modul 1 Windows secara bersamaan	Berhasil
				User aesir modul 1 Windows dan User busir modul 1 Linux secara bersamaan	Berhasil
				User aesir modul 1 Windows dan User busir modul 1 Linux secara bersamaan	Berhasil
				User aesir modul 1 Linux dan User busir modul 1 Windows secara bersamaan	Berhasil
User aesir modul 1 Windows dan User busir modul 1 Linux secara bersamaan	Berhasil				
User aesir modul 1 Linux dan User busir modul 1 Windows secara bersamaan	Berhasil				
User aesir modul 1 Windows dan User busir modul 1 Linux secara bersamaan	Berhasil				
User aesir modul 1 Linux dan User busir modul 1 Windows secara bersamaan	Berhasil				
User aesir modul 1 Windows dan User busir modul 1 Linux secara bersamaan	Berhasil				
User aesir modul 1 Linux dan User busir modul 1 Windows secara bersamaan	Berhasil				
3.	Fitur <i>Stop VM</i>	User busir modul 2 Linux dan User busir modul 1 Windows secara bersamaan	Berhasil		
		User aesir modul 1 Linux dan User busir modul 1 Windows secara bersamaan	Berhasil		
		User aesir modul 1 Linux dan User busir modul 1 Windows secara bersamaan	Berhasil		
		User busir modul 1 Linux dan User busir modul 1 Windows	Berhasil		
		User busir modul 2 Linux dan User busir modul 2 Windows	Berhasil		
		User aesir modul 1 Linux dan User busir modul 1 Windows secara bersamaan	Berhasil		

No	Skenario	Kasus Pengujian	Hasil
		Windows dan User busir modul 1 Windows secara bersamaan	Berhasil
		User aesir modul 1 Linux dan User busir modul 1 Linux secara bersamaan	Berhasil
		User aesir modul 1 Windows dan User busir modul 1 Linux secara bersamaan	Berhasil

Dari hasil yang didapat pada Tabel 1, dapat disimpulkan bahwa pengujian seluruh fitur *website* purwarupa tersebut berhasil, karena seluruh fitur berfungsi dengan baik dan mendapatkan hasil sesuai dengan yang diharapkan untuk pengujian setiap kasusnya. Pada skenario fitur *Login*, *user* berhasil mendapatkan *pop-up* berupa notifikasi atau *alert* sesuai dengan rancangan pada setiap kasusnya. Pada skenario fitur *Redirect Unverified User*, *user* berhasil dialihkan sesuai dengan rancangan pada setiap kasusnya. Pada skenario fitur *VM Resource*, *user* berhasil melihat informasi status *virtual machine lab* sesuai dengan rancangan pada setiap kasusnya. Pada skenario fitur *Remote Access*, *user* berhasil mengakses *virtual machine lab* yang telah *di-deploy* dan dapat mengerjakan modul tugas yang ada pada *virtual machine lab*. *User* dapat menjalankan *command Nmap* [20] pada *virtual machine lab* Linux dan menjalankan *Wireshark* untuk menangkap *traffic ethernet* [21] pada *virtual machine lab* Windows. Pada skenario fitur *Logout*, *user* berhasil mengakhiri sesi pada *website* purwarupa Laboratorium Virtual sesuai dengan rancangan pada setiap kasusnya.

Tabel 2. Durasi *Deploy* Pengujian Fitur *Start VM*

Kasus Pengujian	Durasi Deploy
<i>User aesir deploy</i> modul "1 Linux"	17 menit 16 detik
<i>User aesir deploy</i> modul "1 Windows"	17 menit 12 detik
<i>User aesir deploy</i> modul "2 Linux"	19 menit 32 detik
<i>User aesir deploy</i> modul "2 Windows"	17 menit 45 detik
<i>User busir deploy</i> modul "1 Linux"	15 menit 38 detik
<i>User busir deploy</i> modul "1 Windows"	15 menit 34 detik
<i>User busir deploy</i> modul "2 Linux"	15 menit 45 detik
<i>User busir deploy</i> modul "2 Windows"	17 menit 37 detik
<i>User aesir deploy</i> modul "1 Linux" bersamaan dengan <i>user busir deploy</i> modul "1 Windows"	17 menit 20 detik untuk <i>user aesir</i> dan 18 menit 26 detik untuk <i>user busir</i>
<i>User aesir deploy</i> modul "1 Windows" bersamaan dengan <i>user busir deploy</i> modul "1 Windows"	17 menit 21 detik untuk <i>user aesir</i> dan 18 menit 29 detik untuk <i>user busir</i>
<i>User aesir deploy</i> modul "1 Linux" bersamaan dengan <i>user busir deploy</i> modul "1 Linux"	17 menit 57 detik untuk <i>user aesir</i> dan 19 menit 48 detik untuk <i>user busir</i>
<i>User aesir deploy</i> modul "1 Windows" bersamaan dengan <i>user busir deploy</i> modul "1 Linux"	18 menit 3 detik untuk <i>user aesir</i> dan 18 menit 5 detik untuk <i>user busir</i>

Dari hasil yang didapat pada Tabel 1 dan data durasi *Deploy* pada Tabel 2, dapat disimpulkan bahwa durasi dari *deployment virtual machine lab* pada fitur *Start VM* stabil dan tidak ada perbedaan durasi *deployment* yang signifikan antara membuat satu *virtual machine lab* dengan membuat dua *virtual machine lab*.

Data durasi yang didapat dari dua *user* melakukan *deploy* secara bersamaan membuktikan bahwa VMWare ESXi dapat *men-deploy virtual machine lab* dari dua *user* sekaligus tanpa adanya peningkatan durasi *deployment* yang signifikan.

Dari pengujian ini didapat rata-rata durasi *deployment* dari delapan *virtual machine lab* Linux sebesar 17 menit 40 detik dan rata-rata durasi *deployment* dari delapan *virtual machine lab* Windows sebesar 17 menit 33 detik. Dari data rata-rata *deployment* tersebut dapat disimpulkan bahwa waktu durasi *deploy virtual machine lab* Windows lebih cepat dari pada waktu durasi *deploy virtual machine lab* Linux. Hal ini terjadi karena proses *Task "Import VApp"* pada VMWare ESXi yang berfungsi untuk *deploy VM* dengan OS Linux memakan waktu lebih banyak.

Tabel 3. Durasi *Destroy* Pengujian Fitur *Stop VM*

Kasus Pengujian	Durasi Destroy
User aesir <i>deploy</i> modul "1 Linux"	23 detik
User aesir <i>deploy</i> modul "1 Windows"	39 detik
User aesir <i>deploy</i> modul "2 Linux"	23 detik
User aesir <i>deploy</i> modul "2 Windows"	49 detik
User busir <i>deploy</i> modul "1 Linux"	23 detik
User busir <i>deploy</i> modul "1 Windows"	43 detik
User busir <i>deploy</i> modul "2 Linux"	22 detik
User busir <i>deploy</i> modul "2 Windows"	48 detik
User aesir <i>deploy</i> modul "1 Linux" bersamaan dengan user busir <i>deploy</i> modul "1 Windows"	23 detik untuk user aesir dan 40 detik untuk user busir
User aesir <i>deploy</i> modul "1 Windows" bersamaan dengan user busir <i>deploy</i> modul "1 Windows"	40 detik untuk user aesir dan 41 detik untuk user busir
User aesir <i>deploy</i> modul "1 Linux" bersamaan dengan user busir <i>deploy</i> modul "1 Linux"	22 detik untuk user aesir dan 22 detik untuk user busir
User aesir <i>deploy</i> modul "1 Windows" bersamaan dengan user busir <i>deploy</i> modul "1 Linux"	39 detik untuk user aesir dan 21 detik untuk user busir

Dari hasil yang didapat pada Tabel 1 dan data durasi *Destroy* pada Tabel 3, dapat disimpulkan bahwa durasi dari *destroy virtual machine lab* pada fitur *Stop VM* stabil dan tidak ada perbedaan durasi *destroy* yang signifikan antara menghapus satu *virtual machine lab* dengan menghapus dua *virtual machine lab*. Data durasi yang didapat dari dua *user* melakukan *destroy* secara bersamaan membuktikan bahwa VMWare ESXi dapat menghapus *virtual machine lab* dari dua *user* sekaligus tanpa adanya peningkatan durasi *destroy* yang signifikan. Dari pengujian ini didapat rata-rata durasi *destroy* dari delapan *virtual machine lab* Linux sebesar 22 detik dan rata-rata durasi *destroy* dari delapan *virtual machine lab* Windows sebesar 42 detik. Dari data rata-rata *destroy* tersebut dapat disimpulkan bahwa waktu durasi *destroy virtual machine lab* Linux lebih cepat dari pada waktu durasi *destroy virtual machine lab* Windows. Hal ini terjadi karena proses *Task "Shutdown Guest"* pada VMWare ESXi yang berfungsi untuk melakukan *shutdown* terhadap VM dengan OS Windows memakan waktu lebih banyak.

V. SIMPULAN

Berdasarkan data yang diperoleh serta analisis yang dilakukan dari Proyek Akhir Pengembangan Purwarupa Laboratorium Virtual Berbasis VMWare dengan Terraform, dapat ditarik kesimpulan sebagai berikut:

- Telah dikembangkan sistem purwarupa Laboratorium Virtual yang dapat membuat *virtual machine lab*, melihat status *virtual machine lab* dan menghapus *virtual machine lab* lewat *website* purwarupa Laboratorium Virtual berbasis VMWare ESXi dengan menggunakan Terraform sebagai *Infrastructure as Code*.
- Pengembangan sistem menghasilkan fitur-fitur lainnya yaitu fitur *Login* yang dapat memverifikasi kemudian mengarahkan *user* ke halaman utama *website* purwarupa Laboratorium Virtual milik masing-masing *user*, fitur *Redirect Unverified User* yang mengalihkan *user* ke halaman *login* jika belum melakukan *login*, fitur *Remote Access* yang membuat *user* dapat mengakses *virtual machine lab* secara *remote* dan fitur *Logout* yang dapat mengakhiri sesi *user* kemudian mengalihkan *user* ke halaman *login*.
- *Infrastructure as Code* pada purwarupa Laboratorium Virtual digunakan sebagai otomatisasi pengelolaan dan penyediaan *virtual machine lab* dapat membantu meringankan beban kerja *Administrator* infrastruktur laboratorium yang dilakukan secara berulang-ulang.
- Proses pembuatan *virtual machine lab* memerlukan waktu rata-rata 17 menit 40 untuk Linux dan 17 menit 33 detik untuk Windows. Sedangkan proses penghapusan *virtual machine lab* memerlukan waktu rata-rata 22 detik untuk Linux dan 42 detik untuk Windows. Sehingga proses pembuatan dan penghapusan *virtual machine lab* membutuhkan waktu sebanyak 18 menit 2 detik untuk Linux dan 18 menit 15 detik untuk Windows.
- Metode *Black Box Testing* sebagai uji fungsionalitas sistem menunjukkan bahwa seluruh fungsi yang terdapat pada sistem dapat bekerja dengan baik.

REFERENSI

- [1] W. Susanti et al., *Model Virtual Lab dan Remote Lab melalui Inkuiri dan Kolaborasi Berbantuan Android*. Penerbit Lakeisha, 2022.
- [2] N. Marshall, M. Brown, G. B. Fritz, and R. Johnson, *Mastering VMware vSphere 6.7*. John Wiley & Sons, 2018.
- [3] K. Morris, *Infrastructure as code*. O'Reilly Media, 2020.
- [4] V. HROMÁDKA, "Infrastructure As Code in Agile Software Development," 2022.
- [5] L. De La Torre, M. Guinaldo, R. Heradio, and S. Dormido, "The ball and beam system: A case study of virtual and remote lab enhancement with moodle," *IEEE Trans. Ind. Inform.*, vol. 11, no. 4, pp. 934–945, 2015.
- [6] M. B. Tharayanil, G. Whitney, M. Aiash, and C. Benzaid, "Virtualization and cyber security: arming future security practitioners," in *2015 IEEE Trustcom/BigDataSE/ISPA*, IEEE, 2015, pp. 1398–1402.
- [7] T. Terraform, "Documentation | Terraform | HashiCorp Developer," *Documentation | Terraform | HashiCorp Developer*, 2022. <https://developer.hashicorp.com/terraform/docs> (accessed May 29, 2023).
- [8] S. Muthoni, G. Okeyo, and G. Chemwa, "Infrastructure as Code for Business Continuity in Institutions of Higher Learning," in *2021 International Conference on Electrical, Computer and Energy Technologies (ICECET)*, IEEE, 2021, pp. 1–6.

-
- [9] QuinnRadich, "Remote Desktop Protocol - Win32 apps," Aug. 19, 2020. <https://learn.microsoft.com/en-us/windows/win32/termserv/remote-desktop-protocol> (accessed May 29, 2023).
- [10] A. Apache, "About the XAMPP project," 2022. <https://www.apachefriends.org/about.html> (accessed May 29, 2023).
- [11] L. Babin, *Beginning Ajax with PHP: From novice to professional*. Apress, 2007.
- [12] M. MariaDB, "About MariaDB Server," *MariaDB.org*, 2022. <https://mariadb.org/about/> (accessed May 29, 2023).
- [13] I. udiawan Sitorus, "Cara Membuat Login Multi User Ajax Menggunakan PHP, Sweetalert Dan IziToast," *BelajarwithIB*, Apr. 09, 2021. <https://www.belajarwithib.my.id/2021/04/cara-membuat-login-multi-user-ajax.html> (accessed May 29, 2023).
- [14] J. Spurlock, *Bootstrap: responsive web development*. O'Reilly Media, Inc., 2013.
- [15] M. McGrath, *PHP & MySQL in easy steps*. In Easy Steps, 2012.
- [16] Hewlett-Packard Development Company, "DHCPv4 server," 2022. https://techhub.hp.com/eginfolib/networking/docs/switches/WB/15-18/5998-8162_wb_2920_mcg/content/ch06s04.html (accessed May 29, 2023).
- [17] TightVNC, "TightVNC for Windows: Installation and Getting Started." 2012.
- [18] J. Senkerik, "Terraform Provider." May 29, 2023. Accessed: May 29, 2023. [Online]. Available: <https://github.com/josenk/terraform-provider-esxi>
- [19] A. Nordeen, *Learn Software Testing in 24 Hours: Definitive Guide to Learn Software Testing for Beginners*. Guru99, 2020.
- [20] Nmap, "Nmap: the Network Mapper - Free Security Scanner," 2022. <https://nmap.org/> (accessed May 29, 2023).
- [21] R. Sharpe, E. Warnicke, and U. Lamping, "Wireshark User's Guide: Version 4.1.0."
-