ABSTRACT

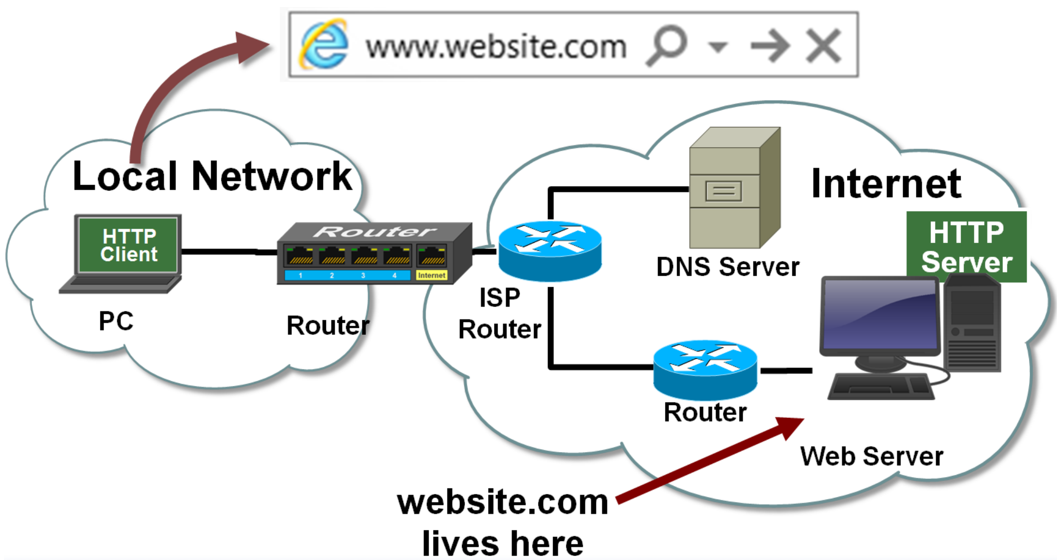
This project proposes the design and implementation of a smart lab system that facilitates the interconnection of multiple labs in a building or campus via a Local Area Network (LAN) infrastructure. Cisco Packet Tracer, a robust network simulation tool, will be utilized to simulate the networking devices and configurations needed for lab connectivity. The LAN network will be designed to support diverse traffic types, such as voice, video, and data, and will be scalable to accommodate future growth and expansion. The network will feature stringent security measures to prevent unauthorized access and protect confidential data.

Additionally, the smart lab system will incorporate a centralized monitoring and control system that can manage and oversee the network devices, applications, and services. This will enable lab administrators to proactively detect and resolve network problems, guarantee network availability, and optimize network performance. The project will entail a thorough analysis of the network requirements, development of the LAN network topology, selection and configuration of network devices, implementation of security measures, and creation of a monitoring and control system.

Overall, the smart lab system will provide an efficient, dependable, and secure networking infrastructure for connecting various labs within a building or campus, enabling seamless communication and collaboration among lab users.

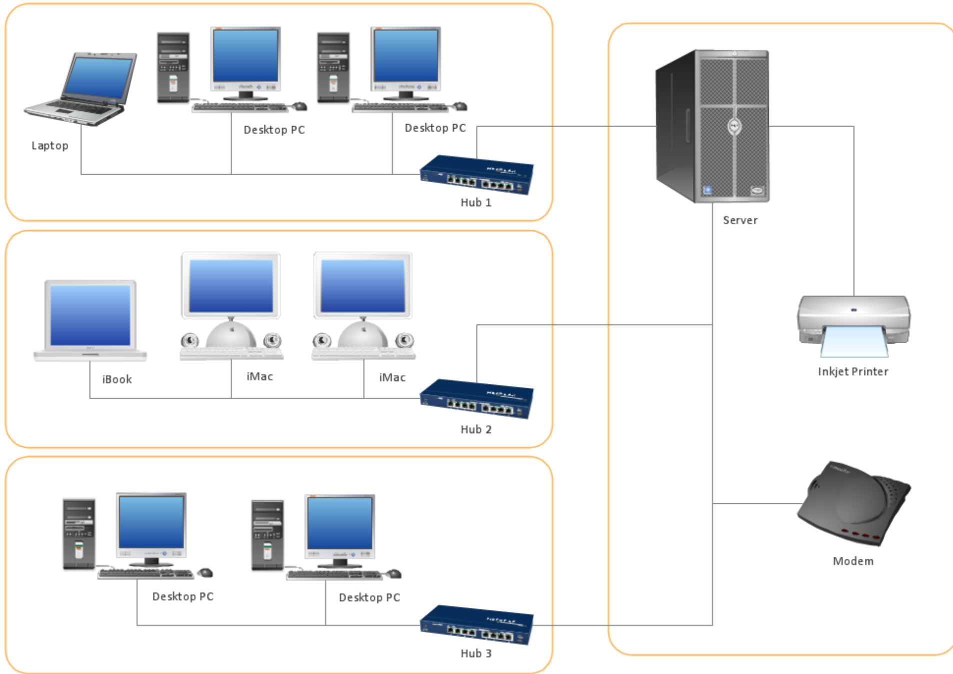
INTRODUCTION

This project focuses on designing and implementing a smart lab system that connects multiple labs in a building or campus using a Local Area Network (LAN) infrastructure. The LAN network topology will be carefully chosen to ensure optimal connectivity and scalability, and LAN networking protocols and technologies will be employed to facilitate efficient and secure communication between labs. The system will enhance productivity and collaboration among the labs. LAN network will be configured with appropriate LAN networking protocols and technologies such as Ethernet, TCP/IP, VLANs, and WLANs, among others, to facilitate efficient and reliable communication between the labs. The network will also be secured with robust security measures to prevent unauthorized access and safeguard sensitive data.



In Fig 1.1 Detailed TCP/IP communication example.

As part of the project to design and implement a smart lab system that connects multiple labs using a LAN network, implementing VLANs can be a critical step. VLANs will be used to isolate network traffic and ensure that each lab has its own secure and independent network. The first step will be to plan the VLANs required, taking into account the number of VLANs needed and the devices that will be in each VLAN. Next, VLANs will be configured on the switches and access ports will be assigned to their respective VLANs. VLAN trunks will also be created to allow multiple VLANs to travel over a single physical link. Once the VLANs are set up, thorough testing will be carried out to ensure that each VLAN can communicate with the devices in its respective VLAN and that the VLANs are isolated from one another. By implementing VLANs as part of the LAN network design, the project aims to provide a secure and efficient network infrastructure that enables seamless communication and collaboration among the labs.



In Fig 1.2 Network Topology – Local Area Network example.

BACKGROUND

HISTORY

The concept of smart labs has been around for several decades, with the first applications of laboratory automation appearing in the 1970s. However, it wasn't until the advent of the Internet of Things (IoT) and other smart technologies in the 21st century that the idea of a fully integrated smart lab system became a reality.

The history of the smart lab system can be traced back to the development of laboratory information management systems (LIMS) in the 1980s. LIMS were initially developed to help laboratories manage data and streamline laboratory processes. Over time, LIMS evolved to include more advanced features, such as automated sample tracking, electronic signatures, and data analysis.

In the 1990s, laboratory automation technologies began to emerge, providing a way to automate repetitive tasks and increase laboratory throughput. These technologies included robotic sample handling systems, automated liquid handling systems, and high-throughput screening systems. These systems were expensive and complex, limiting their adoption to large pharmaceutical companies and research institutions.

In the early 2000s, the advent of the IoT and other smart technologies paved the way for a more connected and integrated smart lab system. With the proliferation of sensors, wireless communication, and cloud computing, it became possible to monitor and control laboratory processes in real-time, from anywhere in the world.

The smart lab system has continued to evolve over the past decade, with the development of more advanced sensors, data analytics tools, and machine learning algorithms. Today, smart lab systems are being used in a wide range of applications, from pharmaceutical research to environmental monitoring.

HOW IT IS INVENTED?

The smart lab system is a computer communication project aimed at enhancing laboratory operations by incorporating smart technologies into lab environments. The project utilizes a combination of software and hardware technologies to create a connected system that can monitor, control and optimize various aspects of laboratory operations.

The smart lab system is built on a robust network infrastructure that enables communication and data exchange between various components of the system. The network infrastructure comprises of wired and wireless technologies that are integrated with sensors, actuators, and control systems. The system is designed to be scalable, flexible and adaptable to different laboratory environments.

SUPERIORITY OF THE SYSTEM

1. Efficient Communication: The LAN network will provide a fast and reliable means of communication between the labs, enabling researchers and technicians to share data and collaborate in real-time.

2. Scalability: The network topology and VLAN implementation will be designed to allow for easy expansion as the labs grow and new devices are added to the network.

3. Enhanced Security: The network will be configured with robust security measures, including access control lists (ACLs), firewalls, and intrusion detection systems, to safeguard sensitive data and prevent unauthorized access.

4. Flexibility: The LAN network will be designed to accommodate a variety of devices, including desktops, laptops, servers, printers, and mobile devices.

5. Centralized Management: The network will be managed centrally, making it easier to monitor network performance, troubleshoot issues, and make changes to the network configuration as needed.

6. Cost-Effective: By using open-source software and standard networking equipment, the project can be implemented cost-effectively while still delivering a high-performance and secure network infrastructure.

MOTIVATION

The smart lab system is designed to address the challenges faced by lab administrators, researchers and technicians in managing laboratory processes. These challenges include monitoring and maintaining lab equipment, ensuring regulatory compliance, managing laboratory resources, and optimizing laboratory processes. The smart lab system aims to solve these challenges by automating various laboratory processes, providing real-time monitoring of lab equipment, and integrating laboratory data into a single platform for analysis and decision-making.

GOALS AND SPECIFICATIONS

The smart lab system consists of several modules that work together to create a comprehensive laboratory management system. The modules include:

1. Laboratory monitoring module - This module monitors the laboratory environment, including temperature, humidity, air quality, and other environmental factors that can affect laboratory processes. The data collected by this module is used to optimize laboratory processes and ensure regulatory compliance.

2. Equipment monitoring module - This module monitors laboratory equipment, including instruments, machines, and other devices. The module provides real-time data on the status of the equipment, including usage, maintenance requirements, and performance. This data is used to optimize equipment performance, reduce downtime and maintenance costs, and ensure equipment availability.

3. Resource management module - This module manages laboratory resources, including supplies, chemicals, and other materials. The module provides real-time data on inventory levels, usage, and ordering requirements. This data is used to optimize resource usage, reduce waste and costs, and ensure timely procurement.

4. Data management module - This module manages laboratory data, including experimental data, results, and reports. The module provides a centralized platform for data storage, analysis, and decision-making. This data is used to optimize laboratory processes, improve research outcomes, and ensure regulatory compliance.

The smart lab system is designed to be user-friendly, with a simple and intuitive interface that enables users to monitor and control various aspects of laboratory operations. The system is also designed to be secure, with robust authentication and encryption protocols that protect laboratory data from unauthorized access.

SUMMARY

Overall, the smart lab system is a transformative computer communication project that has the potential to revolutionize laboratory operations. By incorporating smart technologies into laboratory environments, the system can improve efficiency, reduce costs, and enhance research outcomes The continued development of smart lab technologies is expected to drive further innovation in laboratory operations and research outcomes in the years to come.

SYSTEM DESIGN

OVERVIEW OF THE RPOJECT

The Smart Lab System aims to revolutionize the way laboratories operate by leveraging advanced computer communication technologies. It involves the integration of sensors, actuators, networking infrastructure, and data processing capabilities to create an intelligent and efficient lab environment. The system enables real-time monitoring, control, and analysis of various parameters within the lab, enhancing productivity, safety, and research outcomes.

PROJECT RESEARCH

The project research designing a network topology in which a main server connects to various departmental labs using VLAN concepts. Core devices and a few end-to-end connectivity devices will also provide access to internet servers and external servers. Each department will operate on a separate IP network, with the switches configured with appropriate VLANs and security settings. To enable routing for the routers in the internet network, RIPv2 will be used, while static routing will be used for the external server. The building will obtain dynamic IP addresses from a DHCP server based on a router.

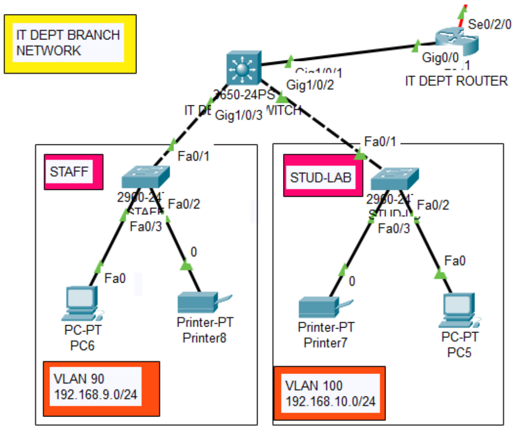
PROJECT LAYOUT

Designing a smart lab system for computer communication project involves creating a network infrastructure that enables efficient data transfer, communication, and control within the lab environment. Here's a high-level system design for a smart lab system:

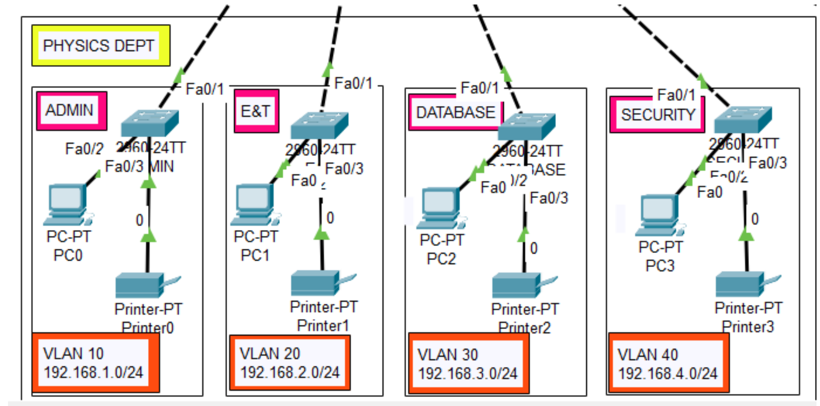
1.Network Infrastructure:

- ✓ Set up a robust wired and wireless network infrastructure to connect all devices in the lab, including computers, sensors, actuators, and other equipment.
- ✓ Use high-speed Ethernet switches to ensure reliable and fast communication between devices.

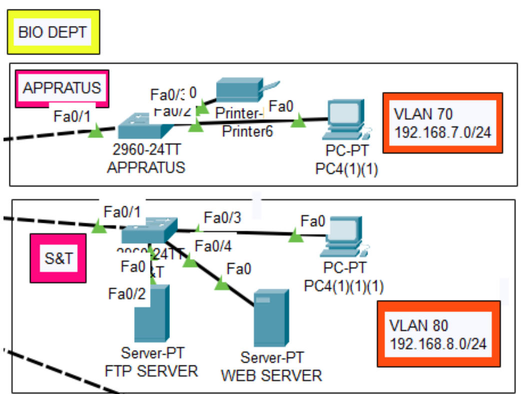
✓ Deploy wireless access points for seamless connectivity in the lab area.
✓ Implement network security measures like firewalls, intrusion detection systems, and access control mechanisms to protect sensitive lab data.



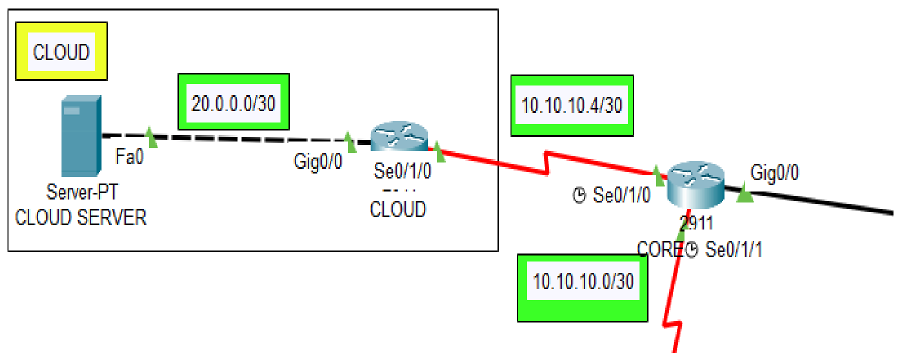
IT Dept. network giving access to staff and student



Physics dept. network topology giving access to admin, implementing database and security



Bio dept topology including all apparatus and implementation of VLAN



Cloud dept. and its connection of the server to the router and then the main switch

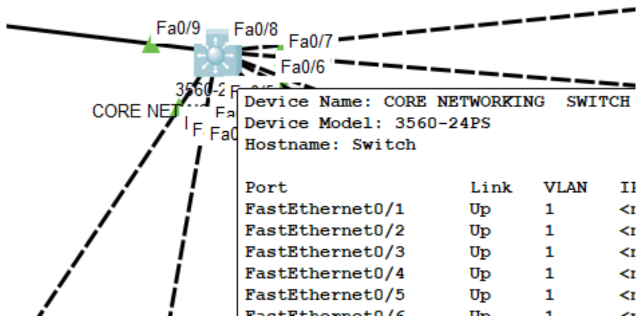
Devices in this network:

- PC's
- Web servers
- Printer
- Copper straight-through cables
- Switches
- Routers

2.Central Server:

✓ Have a central server that acts as the core of the smart lab system.
✓ The server should have sufficient processing power, memory, and storage capacity to handle data processing and storage requirements.
✓ Use a database system to store and manage lab data, experiment results, and other relevant information.
✓ Implement appropriate backup and recovery mechanisms to ensure data integrity.

Main switch configuration




```
Switch>en
Switch#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#int fa0/1
Switch(config-if)#swit
Switch(config-if)#switchport m
Switch(config-if)#switchport mode ac
Switch(config-if)#switchport mode access
Switch(config-if)#swit
Switch(config-if)#switchport acc
Switch(config-if)#switchport access v
Switch(config-if)#switchport access vlan 10
% Access VLAN does not exist. Creating vlan 10
Switch(config-if)#
Switch(config-if)#ex
Switch(config)#
```

Configuring in the main core switch to the other routers

3.Sensors and Actuators:

- ✓ Deploy a variety of sensors throughout the lab to monitor environmental conditions, such as temperature, humidity, light intensity, and air quality.
- ✓ Connect actuators to control lab equipment, such as robotic arms or automated testing apparatus.
- ✓ Use industry-standard communication protocols (e.g., MQTT, RESTful APIs) to enable communication between the sensors, actuators, and the central server.

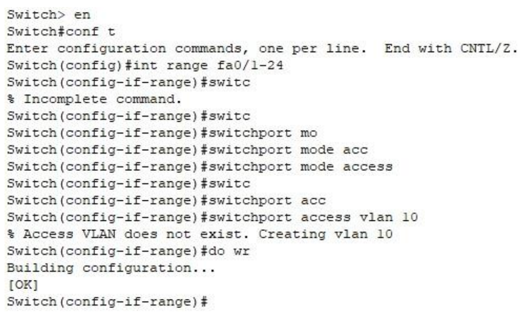
4.Data Processing and Analytics:

- ✓ Implement data processing capabilities on the central server to analyze sensor data and generate meaningful insights.
- ✓ Utilize data analytics techniques, such as machine learning algorithms, to extract patterns and trends from the collected data.
- ✓ Present the analyzed data and insights in a visual format, such as graphs or

dashboards, for easy interpretation.

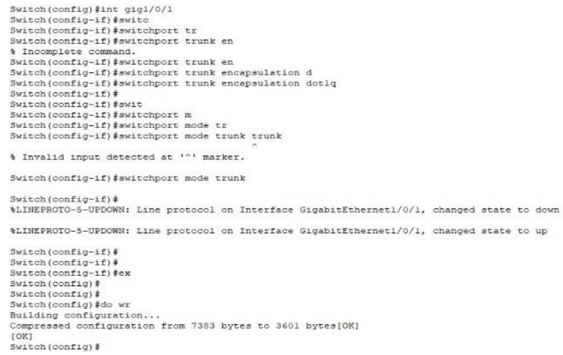
5.Integration and Configuration:

- ✓ Provide Configuration or integration points to allow other systems or external applications to interact with the smart lab system.
- ✓ Enable integration with laboratory management systems, inventory systems, or other relevant tools to streamline workflows and data sharing.



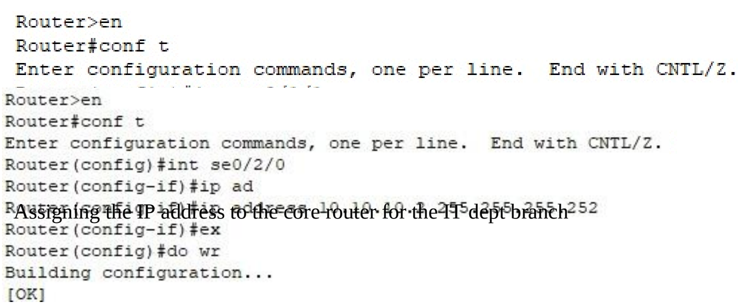
```
Switch> en
Switch#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#int range fa0/1-24
Switch(config-if-range)#switc
% Incomplete command.
Switch(config-if-range)#switc
Switch(config-if-range)#switchport mo
Switch(config-if-range)#switchport mode acc
Switch(config-if-range)#switchport mode access
Switch(config-if-range)#switc
Switch(config-if-range)#switchport acc
Switch(config-if-range)#switchport access vlan 10
% Access VLAN does not exist. Creating vlan 10
Switch(config-if-range)#do wr
Building configuration...
[OK]
Switch(config-if-range)#
```

CLI connection to routers within the range 0-24 in vlan 10



```
Switch(config)#int gig1/0/1
Switch(config-if)#switc
Switch(config-if)#switchport tr
Switch(config-if)#switchport trunk en
% Incomplete command.
Switch(config-if)#switchport trunk en
Switch(config-if)#switchport trunk encapsulation d
Switch(config-if)#switchport trunk encapsulation dotlq
Switch(config-if)#
Switch(config-if)#swit
Switch(config-if)#switchport m
Switch(config-if)#switchport mode tr
Switch(config-if)#switchport mode trunk trunk
                                        ^
% Invalid input detected at '^' marker.

Switch(config-if)#switchport mode trunk

Switch(config-if)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet1/0/1, changed state to down

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet1/0/1, changed state to up

Switch(config-if)#
Switch(config-if)#
Switch(config-if)#ex
Switch(config)#
Switch(config)#
Switch(config)#do wr
Building configuration...
Compressed configuration from 7383 bytes to 3601 bytes[OK]
[OK]
Switch(config)#
```

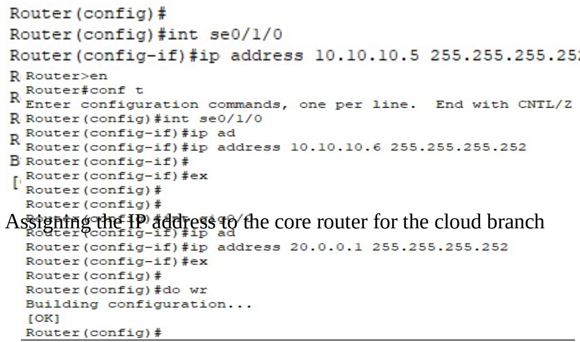
CLI connection of main switch to the connected router



```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#int se0/2/0
Router(config-if)#ip ad
Router(config-if)#ip address 10.10.10.1 255.255.255.252
Router(config-if)#ex
Router(config)#do wr
Building configuration...
[OK]
```

Assigning the IP address to the core router for the IT dept branch

CLI Connection of the IT dept. router to the core router



```
Router(config)#
Router(config)#int se0/1/0
Router(config-if)#ip address 10.10.10.5 255.255.255.252
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z
Router(config)#int se0/1/0
Router(config-if)#ip ad
Router(config-if)#ip address 10.10.10.6 255.255.255.252
Router(config-if)#
Router(config-if)#ex
Router(config)#
Router(config)#
Router(config-if)#ip ad
Router(config-if)#ip address 20.0.0.1 255.255.255.252
Router(config-if)#ex
Router(config)#
Router(config)#do wr
Building configuration...
[OK]
Router(config)#
```

Assigning the IP address to the core router for the cloud branch

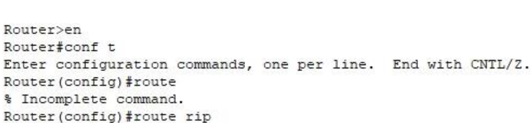
CLI Connection of the cloud router to the core router and connection of cloud router to the server

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#int gig0/0.90
Router(config-subif)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0.90, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0.90, changed state to up

Router(config-subif)#
Router(config-subif)#enc
Router(config-subif)#encapsulation d
Router(config-subif)#encapsulation dot1Q 90
Router(config-subif)#ip add
Router(config-subif)#ip address 192.168.9.1 255.255.255.0
Router(config-subif)#
Router(config-subif)#ex
Router(config)#
Router(config)#int gig0/0.100
Router(config-subif)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0.100, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0.100, changed state to up

Router(config-subif)#encapsulation dot1Q 100
Router(config-subif)#ip address 192.168.10.1 255.255.255.0
Router(config-subif)#
Router(config-subif)#ex
Router(config)#
Router(config)#do wr
Building configuration...
[OK]
```

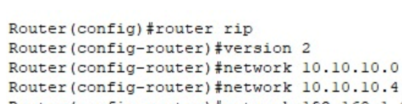
IT dept. branch router connecting to VLAN90 and VLAN100 via encapsulation

```
Router(config)#service dhcp
Router(config)#
Router(config)#ip dhc
Router(config)#ip dhcp pool Staf-pool
Router(dhcp-config)#net
% Incomplete command.
Router(dhcp-config)#net
Router(dhcp-config)#network 192.168.9.0 255.255.255.0
Router(dhcp-config)#def
Router(dhcp-config)#default-router 192.168.9.1
Router(dhcp-config)#dn
Router(dhcp-config)#dns-server 192.168.9.1
Router(dhcp-config)#ex
Router(config)#
Router(config)#do wr
Building configuration...
[OK]
Router(config)#ip dhcp pool Studlb-pool
Router(dhcp-config)#network 192.168.10.0 255.255.255.0
Router(dhcp-config)#default-router 192.168.10.1
Router(dhcp-config)#dns-server 192.168.10.1
Router(dhcp-config)#
Router(dhcp-config)#ex
Router(config)#
Router(config)#do wr
Building configuration...
[OK]
Router(config)#
```

Assigning the Staf-pool and the Studlb-pool to the IT dept. router assigning DNS



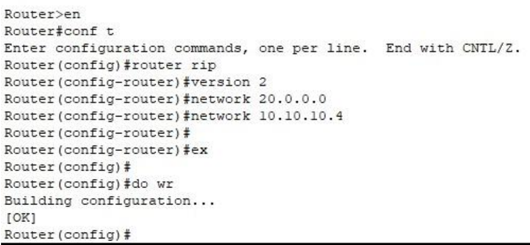
```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#route
% Incomplete command.
Router(config)#route rip
```



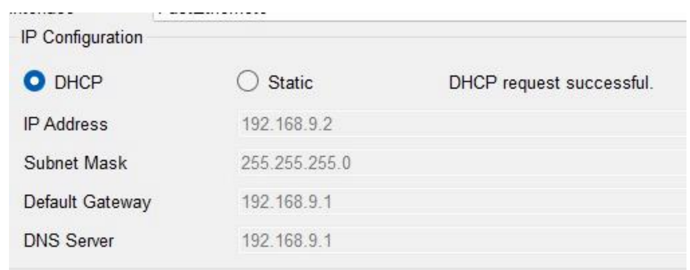
```
Router(config)#router rip
Router(config-router)#version 2
Router(config-router)#network 10.10.10.0
Router(config-router)#network 10.10.10.4
```

Assigning the RIP version 2 to IT dept. router

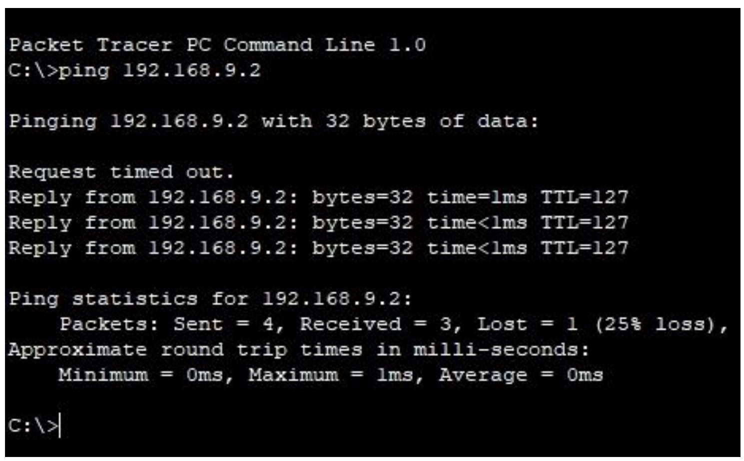
Assigning the RIP V2 in main router and then to networks



Final configuration of the cloud dept. to the core router using RIP V2



Use of DHCP(Dynamix Host Configuration Protocol) to automatically assign IP address



Successfully returning the bits back in VLAN100 using the Ping Command

6.Monitoring and Maintenance:

- ✓ Implement monitoring tools to track the health and performance of the smart lab system.
- ✓ Set up alerts and notifications for critical events or system failures.
- ✓ Regularly perform maintenance tasks, including software updates, security patches, and hardware checks, to ensure system reliability.

Remember, this is a high-level overview, and the actual implementation may require more detailed design decisions based on your specific requirements and constraints.

SOFTWARE HARDWARE REQUIREMENTS

To implement this project, we have to meet some software and hardware requirements.
For Software Requirement it is required to have (CISCO PACKET TRACER) installed on the System. Every implementation is done on this tool.
For Hardware Requirement it is required to have the followings -

- ✓ Intel Pentium 4, 2.53GHz or equivalent Processor
- ✓ 2GB Ram
- ✓ 1GB of free storage space
- ✓ Display of resolution 1024*768
- ✓ Language fonts supporting Unicode encoding
- ✓ Latest video card and OS updates

PROJECT IMPLEMENTATION

1.Project Overview:

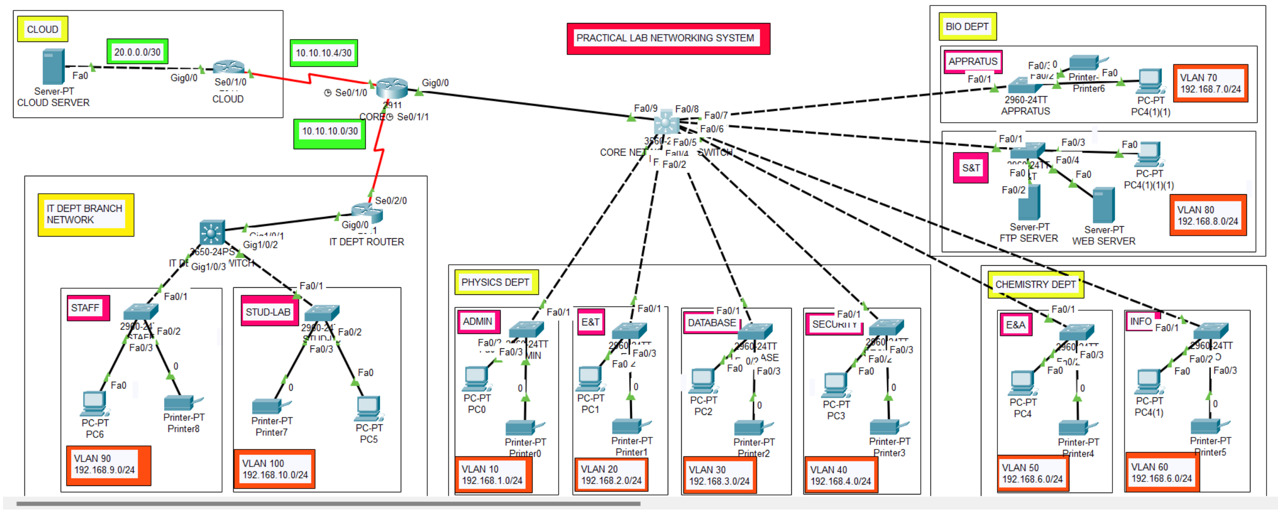
- ✓ The Smart Lab System project aims to create an advanced computer network infrastructure for a laboratory environment. The system will include various networked devices, such as computers, printers, servers, and sensors, to enable efficient lab management and monitoring. The implementation process will involve several steps, including planning, design, configuration, testing, and deployment.

2.Planning Phase:

- ✓ Define project objectives: Clearly outline the goals and objectives of the Smart Lab System project. Identify the specific requirements and functionalities desired in the lab environment.
- ✓ Assess existing infrastructure: Evaluate the current network infrastructure, including network topology, hardware, and software components. Identify any deficiencies or limitations that need to be addressed.
- ✓ Determine project scope: Define the scope of the project by identifying the number of labs, devices, and users that will be included in the system. Determine the budget and timeline for implementation.
- ✓ Assemble project team: Create a project team consisting of network engineers, system administrators, and other relevant stakeholders. Assign roles and responsibilities to team members.

3.Design Phase:

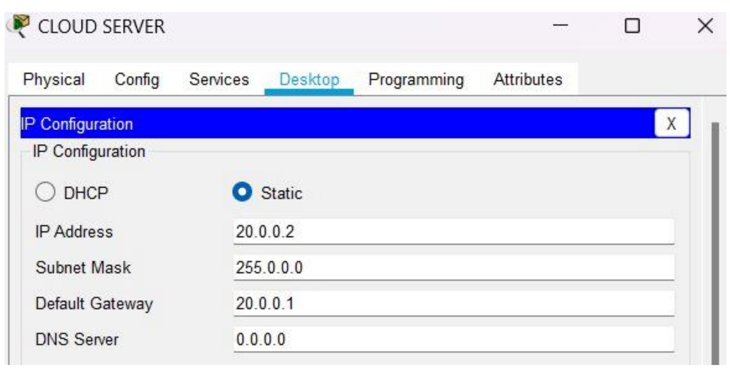
- ✓ Develop network architecture: Design a network architecture that meets the requirements of the Smart Lab System. Consider factors such as scalability, reliability, security, and performance. Determine the placement of network devices, including switches, routers, and servers.
- ✓ Create IP addressing scheme: Design an IP addressing scheme that provides efficient network addressing and subnetting. Allocate IP addresses to various devices and subnets within the lab environment.
- ✓ Plan network security: Identify and implement appropriate security measures, such as firewalls, intrusion detection systems, and access controls, to safeguard the lab network and data.



Final Design of the SMART LAB SYSTEMS

Define network services:

- ✓ Determine the network services required in the lab environment, such as DHCP, DNS, and file sharing. Design the necessary server infrastructure to support these services.



Defining the network service and assigning the IP address Subnet Mask and default gateway

4.Configuration Phase:

- ✓ Install and configure network devices: Physically install network devices, including switches, routers, and servers, in the lab environment. Configure their network settings, including IP addresses, VLANs, and routing protocols.
- ✓ Configure network services: Set up and configure the network services identified in the design phase, such as DHCP, DNS, and file sharing servers. Ensure proper functionality and integration with the lab network.
- ✓ Establish network connectivity: Establish connectivity between network devices by configuring appropriate routing protocols, VLANs, and access control policies. Test network connectivity and resolve any issues that arise.

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#int gig0/0.10
Router(config-subif)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0.10, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0.10, changed state to up

Router(config-subif)#en
Router(config-subif)#encapsulation d
Router(config-subif)#encapsulation dot1Q 10
Router(config-subif)#ip address 192.168.1.1 255.255.255.0
Router(config-subif)#ex
Router(config)#
```

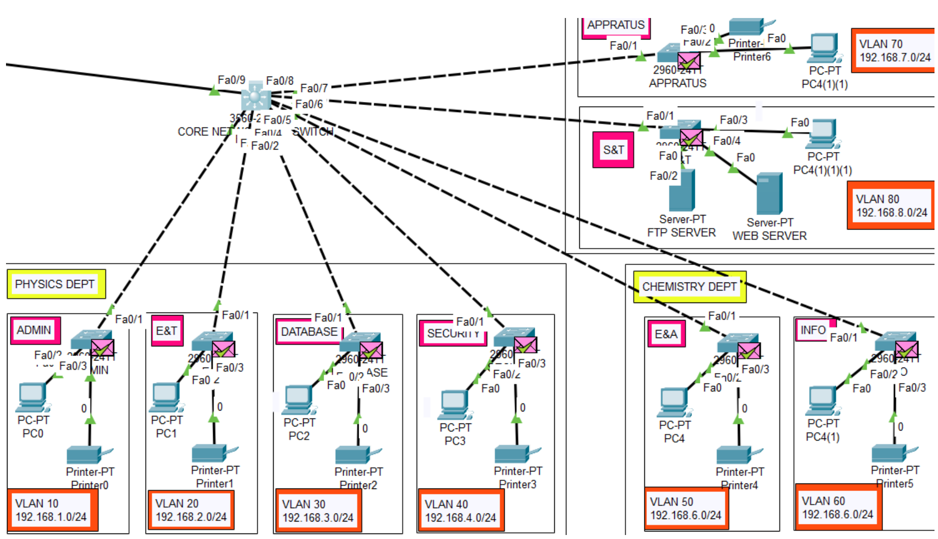
Targeting the core router in gigabit/0.10 followed by encapsulation and assigning the IP address till gigabit/0.80 followed by pool creation as follows

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#int gig0/0.10
Router(config-subif)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0.10, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0.10, changed state to up

Router(config-subif)#en
Router(config-subif)#encapsulation d
Router(config-subif)#encapsulation dot1Q 10
Router(config-subif)#ip address 192.168.1.1 255.255.255.0
Router(config-subif)#ex
Router(config)#
```

Assigning pool to every component

5.Testing of network :

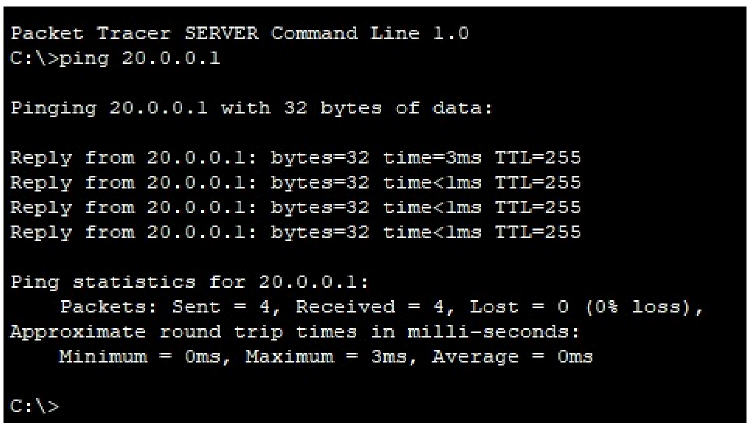
- ✓ Perform network testing: Conduct comprehensive testing of the lab network to ensure its performance, reliability, and security. Test various scenarios, such as heavy network traffic, device failures, and security breaches, to assess the system's resilience.
- ✓ Identify and resolve issues: Identify any issues or bottlenecks that arise during testing and resolve them promptly. Optimize network configurations and settings for better performance.
- ✓ Validate system functionalities: Verify that all the required functionalities, such as remote access, device monitoring, and resource sharing, are functioning as expected in the lab environment.
- ✓ Conduct user acceptance testing: Involve lab users and stakeholders in testing the system functionalities and collecting feedback. Make necessary adjustments based on the feedback received.



6.Deployment:

- ✓ Plan system rollout: Develop a deployment plan that outlines the steps for implementing the Smart Lab System across all labs. Consider factors such as downtime, user training, and data migration, if applicable.
- ✓ Execute deployment plan: Deploy the Smart Lab System according to the deployment plan. Coordinate with lab staff and users to minimize disruptions during the transition phase.
- ✓ Provide user training: Conduct training sessions for lab staff and users to familiarize them with the new system's features and functionalities. Address any questions or concerns they may have.
- ✓ Monitor and evaluate: Continuously monitor the system's performance after deployment. Collect feedback from lab.

Deployment after successfully getting the outcome of the ping command



```
Packet Tracer SERVER Command Line 1.0
C:\>ping 20.0.0.1

Pinging 20.0.0.1 with 32 bytes of data:

Reply from 20.0.0.1: bytes=32 time=3ms TTL=255
Reply from 20.0.0.1: bytes=32 time<1ms TTL=255
Reply from 20.0.0.1: bytes=32 time<1ms TTL=255
Reply from 20.0.0.1: bytes=32 time<1ms TTL=255

Ping statistics for 20.0.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 3ms, Average = 0ms

C:\>
```

COMMAND LINE INSTUCTIONS USED :

There are different commands used for implementing and enabling smart labs on that specific interface.

Commands	Description
enable	It is used to enable the networking device.
config t	It will enter the device into configuration mode.
show version	It is used to show version of router with some other details and security & data packages.
do reload	For reloading the router.
mkdir (directory_name)	Used for make a directory in router
ip ips config location (directory_name)	Assigning the location to store IPS signatures.
ip ips (name)	For creating a IPS rule

switchport access vlan 10	For switching port access to VLAN 10
int (interface_name)	To enter into an interface.
ip ips (rule_name) out	To apply the IPS signature inward at a interface.
logging on	Turning on the logging capability of the router.
status	To enter the status of this signature
enabled	To enable this category signature
trunk en	To carry the traffic of multiple VLANs over a single link and to enable it
encapsulation dot1q	To encapsulate dot1q vlan_id command in the subinterface
switch mode trunk	To use configure the trunking operational mode on a Layer 2 interface on a cisco IOS device.
int se0/1/1	Used for basic routing configuration
ip add	To add ip address
ex	To extend or to show ip route EXEC
do wr	To write in memory
int gig0/0	To target the router gig0/0
encapsulation d	To initiate encapsulation
ip dhcp	To automatically get the ip, default and subnet without manual assignment
ip dhcp pool Staf-pool	To creat pool with automation assignment of addresses
network ip add default gateway	To assign network with ip address, default and gateway address.
dns-server ip address	To resolve host names into the IP address
router rip	To configure the hostes as part of a RIP network
version 2	To shift to RIP version 2
ping	To ping and check for connection status
default-router ip address	To set the default router the ip address

Summary:

The Smart Lab System project involves the implementation of an advanced computer network infrastructure for a laboratory environment. The project follows a structured approach consisting of planning, design, configuration, testing, and deployment phases.During the planning phase, project objectives are defined, the existing infrastructure is assessed, and the project scope is determined. A project team is assembled, including network engineers and system administrators.
In the design phase, a network architecture is developed to meet the project requirements. An IP addressing scheme is created, network security measures are planned, and network services are defined. Consideration is given to scalability, reliability, security, and performance.The configuration phase involves the installation and configuration of network devices such as switches, routers, and servers. Network services are set up and connectivity between devices is established. Network settings are optimized, and any issues are resolved.
Testing and quality assurance are conducted to ensure the performance, reliability, and security of the lab network. Comprehensive testing is performed, issues are identified and resolved, and system functionalities are validated. User acceptance testing involves involving lab users and stakeholders to provide feedback.
In the deployment phase, a rollout plan is developed, and the system is deployed across all labs. User training is provided to familiarize lab staff and users with the new system. Monitoring and evaluation continue after deployment to ensure optimal performance.

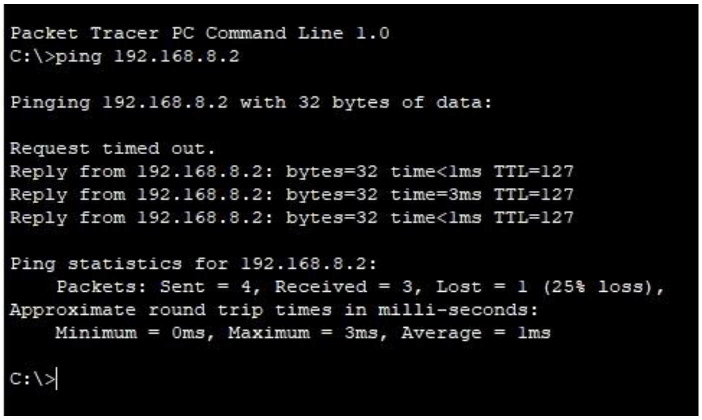
EXPERIMENTAL RESULTS AND DISCUSSION

Experimental Results and Discussions
Testing SMART SYSTEM LAB
The experimental results for a smart lab system can vary depending on the specific goals and specifications of the system, as well as the laboratory's unique requirements. However, some potential outcomes could include:

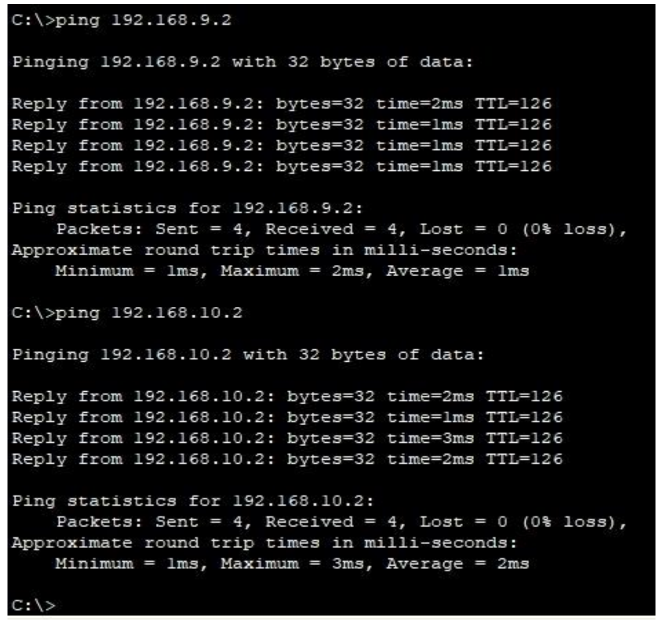
- ✓ Improved efficiency and productivity in laboratory operations, resulting in faster experimental turnaround times and increased output.
- ✓ Increased accuracy and reliability of experimental data, leading to better decision-making and more meaningful insights.
- ✓ Enhanced collaboration and communication among laboratory staff, resulting in improved teamwork and more innovative research outcomes.

- ✓ Reduction in errors and waste, resulting in cost savings and improved sustainability.
- ✓ Better management of laboratory resources, such as samples and inventory, resulting in more efficient use of laboratory space and equipment.

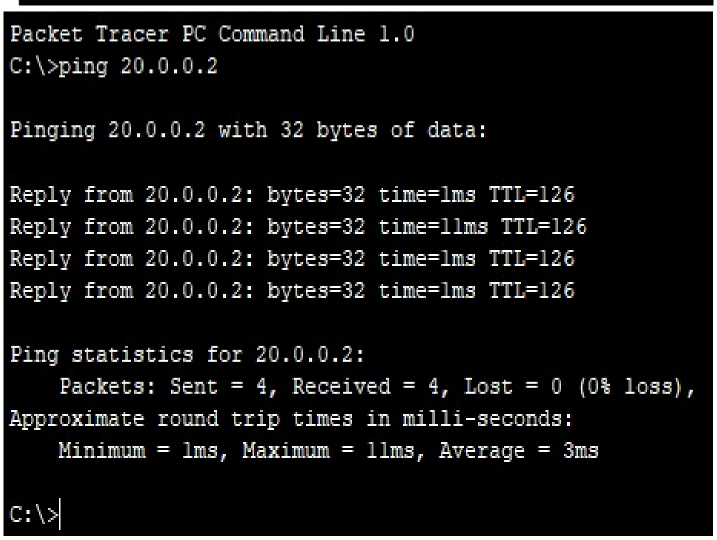
<u>SUCCESSFUL TESTING OUTCOME:</u>



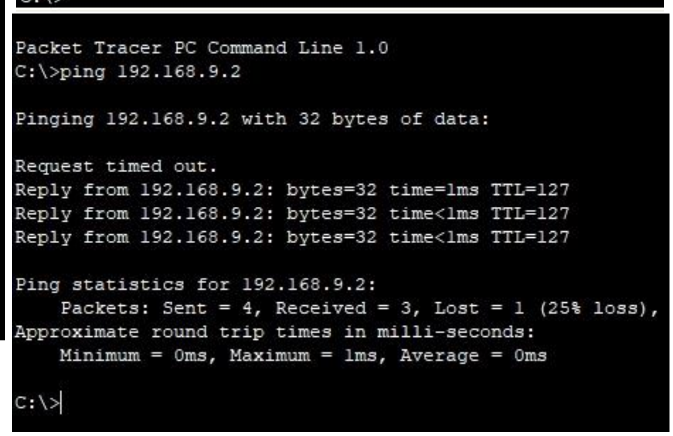
```
Packet Tracer PC Command Line 1.0
C:\>ping 192.168.8.2

Pinging 192.168.8.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.8.2: bytes=32 time<1ms TTL=127
Reply from 192.168.8.2: bytes=32 time=3ms TTL=127
Reply from 192.168.8.2: bytes=32 time<1ms TTL=127

Ping statistics for 192.168.8.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 3ms, Average = 1ms

C:\>
```



```
C:\>ping 192.168.9.2

Pinging 192.168.9.2 with 32 bytes of data:

Reply from 192.168.9.2: bytes=32 time=2ms TTL=126
Reply from 192.168.9.2: bytes=32 time=1ms TTL=126
Reply from 192.168.9.2: bytes=32 time=1ms TTL=126
Reply from 192.168.9.2: bytes=32 time=1ms TTL=126

Ping statistics for 192.168.9.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 2ms, Average = 1ms

C:\>ping 192.168.10.2

Pinging 192.168.10.2 with 32 bytes of data:

Reply from 192.168.10.2: bytes=32 time=2ms TTL=126
Reply from 192.168.10.2: bytes=32 time=1ms TTL=126
Reply from 192.168.10.2: bytes=32 time=3ms TTL=126
Reply from 192.168.10.2: bytes=32 time=2ms TTL=126

Ping statistics for 192.168.10.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 3ms, Average = 2ms

C:\>
```



```
Packet Tracer PC Command Line 1.0
C:\>ping 20.0.0.2

Pinging 20.0.0.2 with 32 bytes of data:

Reply from 20.0.0.2: bytes=32 time=1ms TTL=126
Reply from 20.0.0.2: bytes=32 time=11ms TTL=126
Reply from 20.0.0.2: bytes=32 time=1ms TTL=126
Reply from 20.0.0.2: bytes=32 time=1ms TTL=126

Ping statistics for 20.0.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 11ms, Average = 3ms

C:\>
```



```
Packet Tracer PC Command Line 1.0
C:\>ping 192.168.9.2

Pinging 192.168.9.2 with 32 bytes of data:

Request timed out.
Reply from 192.168.9.2: bytes=32 time=1ms TTL=127
Reply from 192.168.9.2: bytes=32 time<1ms TTL=127
Reply from 192.168.9.2: bytes=32 time<1ms TTL=127

Ping statistics for 192.168.9.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms

C:\>
```

SUCCESSFUL INTERNAL ROUTING VERIFIED FROM PING COMMAND

CONCLUSION:

In conclusion, a smart lab system is a comprehensive solution for laboratory automation that integrates advanced technologies and software tools to improve laboratory efficiency, productivity, and collaboration. The development of smart lab systems was motivated by the need to manage increasing amounts of data generated in modern laboratories and to improve collaboration between researchers. Smart lab systems automate routine laboratory tasks, reduce errors, optimize resources, and provide a platform for sharing data and results, accelerating scientific research and development. They offer several advantages, including increased efficiency, improved accuracy, enhanced collaboration, better data management and analysis, resource optimization, and scalability.

The goals and specifications of a smart lab system typically include laboratory automation, integration with other lab equipment, data management and analysis, sample and inventory management, security and compliance, and user interface. The design of a smart lab system involves a thorough analysis of laboratory operations and workflow, followed by the identification of specific areas for improvement. The experimental results for a smart lab system are expected to be positive, with improvements in laboratory operations and outcomes.

Overall, smart lab systems are an ideal solution for academic, industrial, and government laboratories seeking to streamline laboratory operations and improve research outcomes. They provide researchers with an end-to-end solution for laboratory operations that helps them to focus on more complex tasks and achieve better results. As laboratory operations continue to become more complex, the demand for smart lab systems is likely to grow, making them a valuable tool for modern scientific research and development.

Difficulties

- ✓ Integration with existing laboratory equipment and systems can be complex and time-consuming.

- ✓ The cost of implementing a smart lab system can be high, and budget constraints may limit the ability to adopt the latest technology.

- ✓ The design of a smart lab system must take into account the specific needs of the laboratory, which can be highly varied and complex.

- ✓ The implementation of a smart lab system requires significant changes to laboratory operations and workflow, which can be difficult to manage.

Future Work

Future work in smart lab systems involves the integration of AI and machine learning algorithms, continued optimization of workflows, and expansion of compatibility with new laboratory equipment and systems.