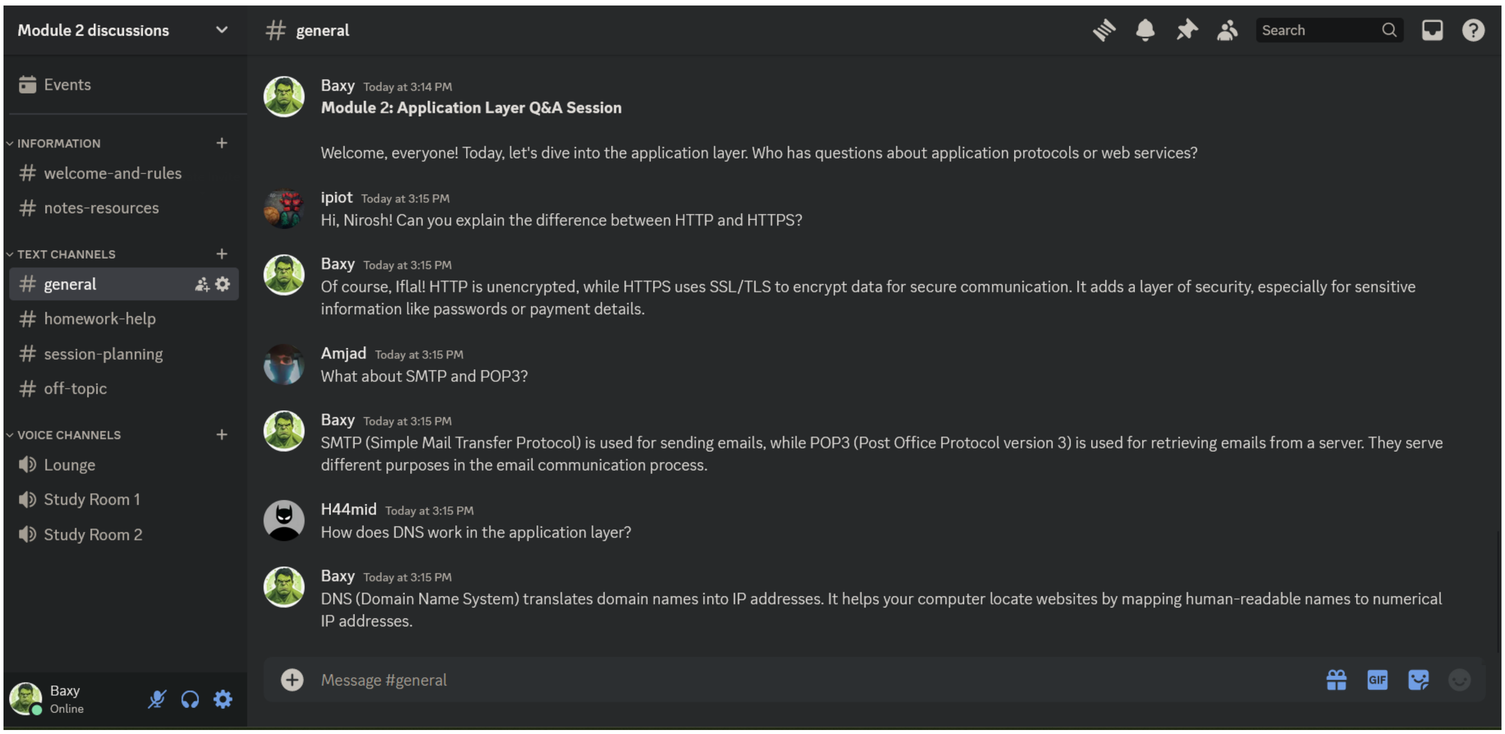
Reflection for the task 1.3HD

Introduction:

In the field of networking education, student success depends on creating a collaborative learning atmosphere. I actively contributed in helping my colleagues comprehend and apply the concepts covered in all seven modules from weeks 2 through 10 of the SIT202: Secure Networking unit. This reflective report explores the ways in which I supported peers and how these experiences correlated with the learning objectives of the unit.

Module 2: Application Layer:

During this module, which delved into the application layer of the OSI model, I engaged in lively discussions on our discord channel, addressing queries related to application protocols, web services, and email protocols. By providing detailed explanations and examples, I aimed to deepen my peers' understanding of application layer concepts.



Then for this Module 2, we did a group discussion on Understanding the Application Layer using HTTP. We altogether discussed what was HTTP all about and then we all an insightful knowledge about it afterwards. Through discussions with my peers, I have learned that application layer protocols serve as the interface for communication between software applications over a network. These protocols define the rules and conventions for data exchange, ensuring compatibility and interoperability.

Baxy Today at 2:15 PM
greetings earthlings...🌏 today we are all summoned here for the next quest....Active class 2: Bridging the gap between you and the technology: Understanding the Application Layer...... shall we begin our quest to finding the mystery of Application layer in the TCP/IP model......⚔️
so for today... our first question in Activity 1 will be..

1. why do we need application layer protocols?? Discuss some examples of application layer protocols.

ipiot Today at 2:20 PM
Application layer protocols are necessary for applications to communicate with each other because they provide a common set of rules and procedures for exchanging data. These protocols define the format of the data, the methods used to send and receive data, and the mechanisms used to ensure that data is delivered reliably and securely. Without application layer protocols, applications would have to use different methods to communicate with each other, which would make it difficult to develop and maintain applications.

Example:-
HTTP (Hypertext Transfer Protocol): HTTP is used to transfer web pages and other resources over the web.
HTTPS (Hypertext Transfer Protocol Secure): HTTPS is a secure version of HTTP that uses TLS/SSL to encrypt data.
DHCP (Dynamic Host Configuration Protocol): DHCP is used to assign IP addresses to devices on a network. (edited)
💯 1 👍 1

Baxy Today at 2:20 PM
Application layer protocols are necessary as they provide a standardized way for applications to communicate over a network. They define the rules and conventions for data exchange, ensuring compatibility and interoperability between different software applications.

Examples of application layer protocols include:
HTTP (Hypertext Transfer Protocol) for web browsing
SMTP (Simple Mail Transfer Protocol) for email communication
FTP (File Transfer Protocol) for file transfer
DNS (Domain Name System) for translating domain names to IP addresses.
👍 3

H44mid Today at 2:21 PM
Application layer protocols are necessary to enable communication and data exchange between different applications on different systems. They define the rules and formats for transmitting data and requests. Examples include HTTP for web browsing, SMTP for email, FTP for file transfer, and SSH for secure remote access. Without standardized application protocols, applications would not be able to communicate effectively across networks.
👍 2 💯 2

Amjad Today at 2:25 PM
Application layer protocols are vital in networking because they establish rules for effective communication between software applications. They ensure:

Standardization: Different applications can communicate seamlessly by adhering to common protocols.
Interoperability: Various systems can exchange data regardless of platform or device.
Abstraction: Applications can focus on functions without worrying about lower-level networking complexities.
Functionality: Protocols provide specific functions, like email exchange (SMTP) or file transfer (FTP).

Examples include HTTP for web browsing, SMTP for emails, FTP for file transfer, DNS for domain names, SSH for secure remote login, and SNMP for network device management. These protocols enable diverse communication and data exchange over networks. (edited)
👍 2 💯 1

mabrook Today at 2:33 PM

Application layer protocols are rules governing how software communicates over networks. HTTP fetches web pages, SMTP sends emails, while DNS translates domain names to IP addresses. These protocols ensure data travels securely and accurately between applications, enabling seamless internet functionality.

Some examples of application layer protocols are:

1. HTTP (Hypertext Transfer Protocol): Used for fetching web pages and resources from servers.
2. SMTP (Simple Mail Transfer Protocol): Facilitates the sending of emails between servers.
3. FTP (File Transfer Protocol): Enables the transfer of files between client and server.
4. DNS (Domain Name System): Converts domain names to IP addresses for internet communication.
5. DHCP (Dynamic Host Configuration Protocol): Assigns IP addresses to devices on a network automatically.

👍 2 2

Baxy Today at 2:46 PM

Let's consider one of the popular application layer protocols, HTTP to learn the principal operation of HTTP. To carry on this discussion, we can do a small role play.

But before that i have question:
a. HTTP is a client-server protocol. Can you identify the key features of HTTP?

👍 3

ipiot Today at 2:48 PM

Client-server architecture: HTTP involves a client (e.g., web browser) and a server (e.g., web server) communicating with each other.
Request-response model: The client sends requests to the server, and the server sends responses back to the client.
Stateless: HTTP does not store information about the client's previous requests.
Cacheable: HTTP supports caching, which can improve performance by reducing the number of requests sent to the server.
Extensible: HTTP can be extended with new features and functionality through the use of HTTP extensions.

Baxy Today at 2:50 PM

--> HTTP stands for Hypertext Transfer Protocol. It's a protocol used for communication between a client (usually a web browser) and a server (which hosts web resources like web pages, images, etc.). In this communication model, the client initiates requests for resources, and the server responds to those requests.

--> It operates over TCP/IP and typically uses port 80 for communication.

--> HTTP follows a request-response model where clients (web browsers) send requests to servers, and servers respond with the requested resources. (edited)

👍 3

Amjad Today at 2:50 PM

HTTP is a client-server protocol with key features including statelessness, request-response communication, text-based messages, support for various media types, connectionless nature, and extensibility. It facilitates communication between web browsers (clients) and web servers, allowing the retrieval and exchange of web resources over the internet.

👍 3 2

mabrook Today at 2:54 PM

yah sure, here are some of the key features of http below.

1. Client-server architecture: Here the http operates on a client-server model, where clients send requests to servers for resources,(clients are known as web browsers) and servers respond with the requested data.
2. Request-Response Cycle: Here http follows a request-responce cycle , where clients initiate requests by sending HTTP requests to servers, and servers respond with HTTP responses containing the requested data.
3. Text-Based Protocol: HTTP messages are text-based, consisting of headers and an optional message body. Headers contain metadata about the request or response, while the message body contains the actual data being transferred, such as HTML content, images, or files.
4. Security: HTTPS secures HTTP communication by encrypting data between clients and servers, ensuring confidentiality and integrity.

These features use http protocols for communication between the clients and servers on the www. (edited)

👍 3

H44mid Today at 2:58 PM

The Hypertext Transfer Protocol (HTTP) is a client-server protocol that forms the foundation of data communication on the World Wide Web. It follows a client-server architecture, where the client, typically a web browser, initiates requests, and the server responds to those requests. HTTP is a connectionless and stateless protocol, meaning that each request-response transaction is independent, and the server does not maintain information about the client's previous requests.

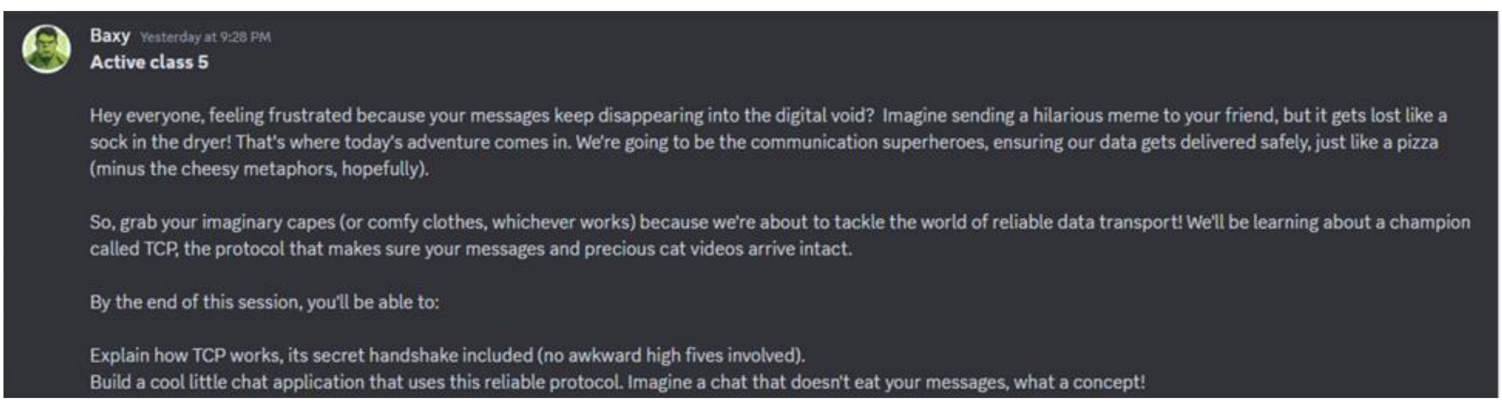
HTTP defines several request methods, such as GET, POST, PUT, and DELETE, which specify the action to be performed on the server. It also uses numerical status codes to indicate the result of a request, like 200 (OK), 404 (Not Found), or 500 (Internal Server Error). HTTP headers provide additional information about the request or response, such as content type, cache control, and authentication. One of HTTP's key features is its support for caching mechanisms, which improve performance by reusing previously fetched resources. The protocol's simplicity, scalability, and adherence to standards have contributed to its widespread adoption as the backbone of the World Wide Web.

👍 3

Module 3: Transport Layer:

In the transport layer module, I took the initiative to organize virtual study group sessions on Discord, focusing on topics such as TCP/IP protocols, UDP, and congestion control mechanisms. These sessions provided an interactive platform for peers to clarify doubts and discuss real-world applications of transport layer protocols.

I actively participated in the group discussions and collaborated with my friends to understand the importance of a reliable transport layer protocol. I can reference the group activity where we developed a transport layer protocol to ensure reliable image transmission.



ipiot Yesterday at 9:43 PM

1. Assume one of your group members sent you a message. However, you have never received it.
What went wrong? How do you make sure that the message is properly received?

What went wrong?
*The message may not have been received due to network issues such as packet loss, corruption, or congestion.
*The sender may have sent the message to the wrong address or port.
*The receiver's firewall or security software may have blocked the message.

How to make sure the message is properly received?
*Use error detection and correction techniques like checksums or CRCs to detect errors in the message.
*Implement acknowledgements (ACKs) where the receiver sends a message confirming receipt of the data.
*Set timeouts for each message and re-transmit if no ACK is received within the timeout period.

2. Assume you are trying to access a web page containing an image. Now, your job is to make sure
that you get the image properly. The image cannot be received in one packet, and it will be
broken down to 10 packets.
*Break down the image into smaller packets, each with a sequence number.
*Use acknowledgements (ACKs) for each packet received.
*Set timeouts for each packet and re-transmit missing packets.
*Implement error detection and correction techniques like checksums or CRCs.

ipiot Yesterday at 9:57 PM

3. Partner up with one of your group members and develop a transport layer protocol to
guarantee that the image is properly received by the client. One can act as the server where the
webpage/image is stored and the other can act as the client who wants to receive the image
(this means your group has 2 pairs of server-client networks).

I am acting as the **client** for this scenario

H44mid Yesterday at 9:58 PM

I'm acting as the **Server** for the Scenario

ipiot Yesterday at 9:58 PM

Client: "Sending SYN (Synchronize) to initiate connection."

H44mid Yesterday at 9:58 PM

Server: "Received SYN. Sending SYN-ACK (Synchronize-Acknowledgment) to acknowledge and synchronize." (edited)

ipiot Yesterday at 9:58 PM

Client: "Received SYN-ACK. Sending ACK (Acknowledgment) to finalize connection establishment."

H44mid Yesterday at 9:58 PM

Server: "Connection established. Ready to receive request." (edited)

ipiot Yesterday at 9:58 PM

Client: "Sending request for the image."

H44mid Yesterday at 9:58 PM

Server: "Received request. Preparing image for transmission." (edited)

ipiot Yesterday at 9:58 PM
Client: "Waiting for image transmission."

H44mid Yesterday at 9:59 PM
Server: "Dividing image into packets for sending." (edited)

ipiot Yesterday at 9:59 PM
Client: "Received packet 1. Sending ACK."

H44mid Yesterday at 9:59 PM
Server: "ACKnowledged. Transmitting packet 2." (edited)

ipiot Yesterday at 9:59 PM
Client: "Packet 2 received. Sending ACK."

H44mid Yesterday at 9:59 PM
Server: "ACKnowledged. Sending remaining packets."

ipiot Yesterday at 9:59 PM
Client: "All packets received. Sending ACK."

H44mid Yesterday at 9:59 PM
Server: "ACKnowledged. Image transmission complete."

ipiot Yesterday at 9:59 PM
Client: "Confirmed. Image received. Thank you."

H44mid Yesterday at 9:59 PM
Server: "You're welcome. Connection will be terminated."
4. Write down protocol that you would use, including the communication between the client and server and all the messages that you are passing.

The protocol involves the client requesting the image from the server, which divides it into 10 packets for transmission. The client acknowledges receipt of each packet, prompting re-transmission if necessary. Once all packets are received, the server confirms successful transmission, ensuring reliable image delivery. (edited)

ipiot Yesterday at 10:14 PM
5. How do you guarantee that your protocol does the job as you expected?

*By using **sequence numbers**and **timeouts**, the protocol can guarantee that the image is received properly, even if some packets are lost or delivered out of order.*
Double-Checking with Acknowledgements: Those confirmation messages you send back are like checking the pigeons made it to the right rooftop. The server knows each packet was delivered safely.**
*Rescuing Lost Packets**: If a packet goes missing, you can request a resend, just like calling for backup if a pigeon gets lost.
Keeping Things Orderly with Sequence Numbers: The numbered codes on the packets are like flight plans for the pigeons. They ensure everything arrives in the right order for reassembly. (edited)

6. Once you have finalized your protocol, check how other group members developed their protocols.

1. Is our protocol simpler or more complex than others? Does it involve fewer message exchanges?

Baxy Yesterday at 10:20 PM
Hey, our protocol seems pretty straightforward. It only uses four messages for the handshake and packet exchange. Maybe some groups have a more complex way of acknowledging packets? We could see if that affects the overall efficiency.

Amjad Yesterday at 10:20 PM
Our group's protocol uses a checksum for each packet to check for errors during transmission. That might be a bit more complex, but it could be helpful for catching corrupted data. (edited)

ipiot Yesterday at 10:21 PM
OK, thats great.

2. How do other protocols handle missing packets? Is there a more efficient way to request resends?

Amjad Yesterday at 10:21 PM
We included a timer in our protocol. If the client doesn't receive an acknowledgement within a certain time, it automatically requests a resend. Seems faster than waiting for a specific message. (edited)

Baxy Yesterday at 10:21 PM
For missing packets, we just ask for a resend directly. I wonder if there's a way to automatically detect missing packets without needing an extra message every time.

ipiot Yesterday at 10:22 PM
That's awesome to go on with discussions

3. Imagine this video was a giant movie with hundreds of packets. Can our protocol handle that much data? Would we need to add features to avoid duplicate packets (like having two pigeons arrive with the same message)?

Baxy Yesterday at 10:22 PM
A giant movie with hundreds of packets? Hmm, that's a good point. Our protocol might get bogged down with all those resends. Maybe we need a way to check for duplicates too, just in case some packets get sent twice.

Amjad Yesterday at 10:22 PM
For a massive file, we were thinking of adding a sequence number range instead of individual numbers for each packet. Like, instead of 1, 2, 3... we could say packets 1-1
received, 11-20 received, and so on. That way, we can acknowledge a whole bunch at once.

Baxy Yesterday at 10:23 PM
7. With your group, discuss whether your protocol valid when you need to access a large file that might be broken down into 1000 packets. If not, what are the changes you would like to make in your protocol?

mabrook Yesterday at 10:23 PM
So, our protocol seems to work well for the 10-packet image, but what if we're downloading a giant movie with, like, 1000 packets?

Baxy Yesterday at 10:23 PM
That's a good point! With so many packets, it might be easy to miss one and not even realize it. Our protocol might not catch it.

mabrook Yesterday at 10:24 PM
Maybe we could add a double-check system. After receiving a packet, the client could send a small message back to the server saying, "Hey, I got packet number 23!"

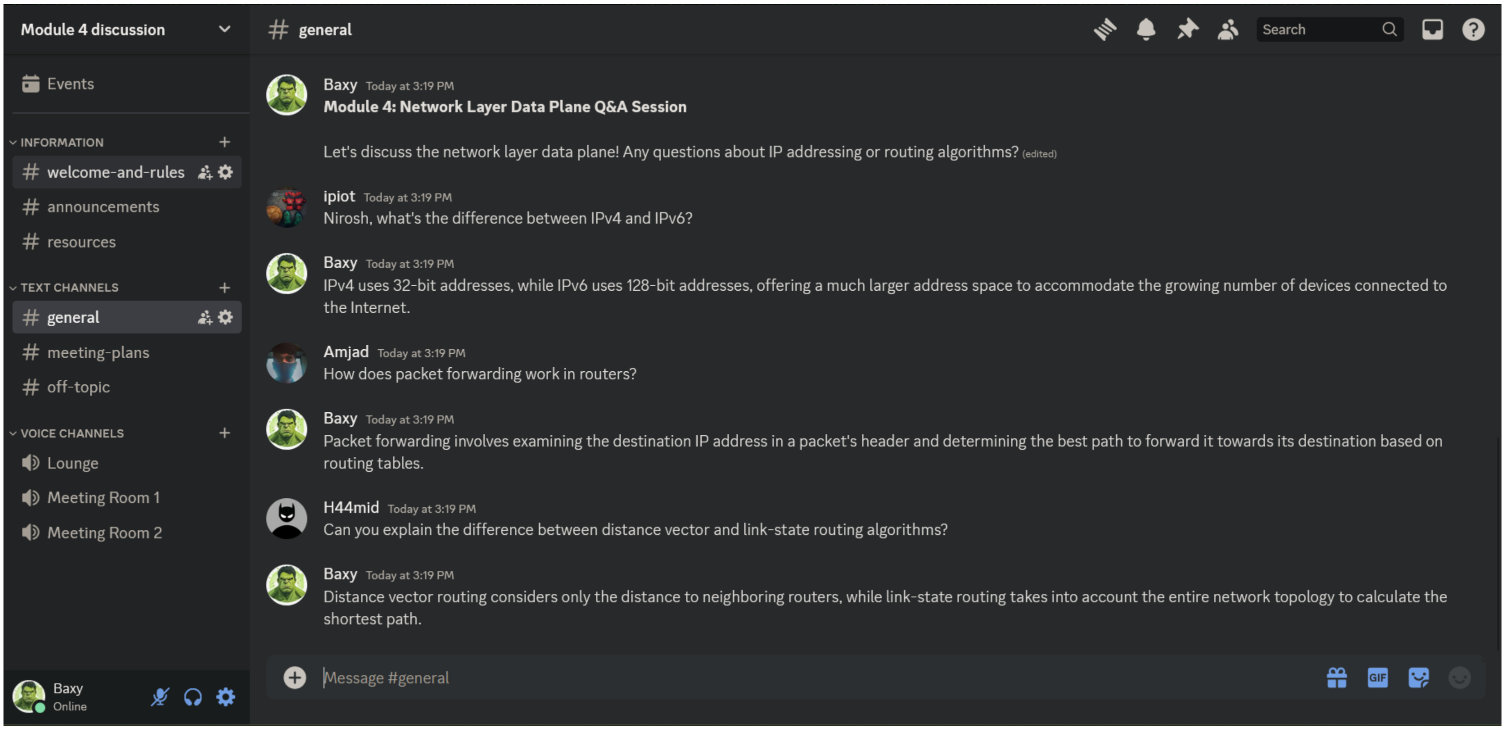
Baxy Yesterday at 10:24 PM
Ooh, that's clever! Then the server can keep track of which packets have been acknowledged. If it doesn't hear back about a specific one, it knows to resend it.

mabrook Yesterday at 10:24 PM
Exactly! We could call it a "packet confirmation" step. Also, with so many packets, what if we get duplicates? We don't want the movie to have two endings!

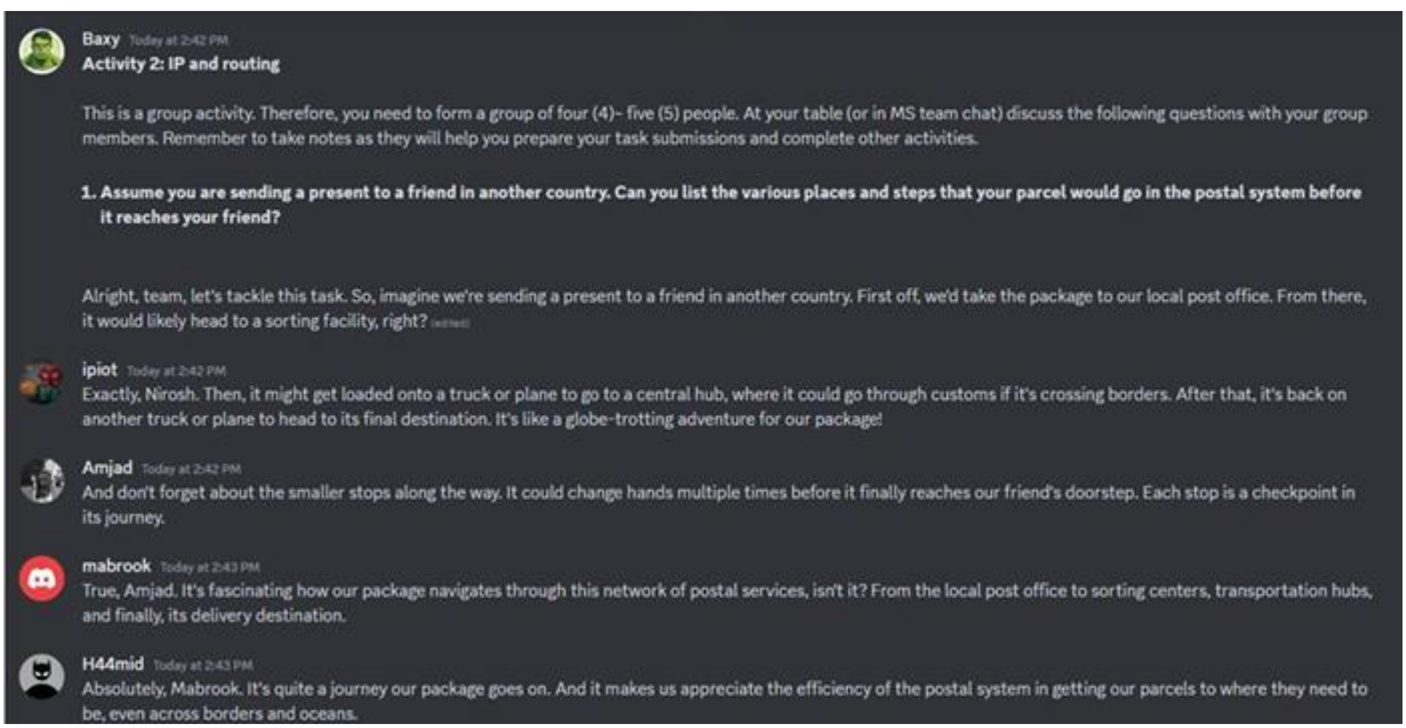
Baxy Yesterday at 10:24 PM
True! Maybe each packet could have a unique code besides the sequence number. That way, the client can recognize a duplicate and just ignore it.

Module 4: Network Layer Data Plane:

As the focus shifted to the network layer data plane, I explained concepts like IP addressing, routing algorithms, and packet forwarding in our discord discussion channel. These explanations were as valuable supplementary materials for peers seeking help in their learning.



This collaborative group activity fostered discussions surrounding IP and routing concepts. Within my group, we contemplated the analogy between sending a present through the postal system and transmitting a message over computer networks. We then tackled the task of building a network with two LANs and assigning roles to group members. As we configured devices, set IP addresses, and determined gateways, we strategized on enabling communication between LANs. Lastly, we meticulously outlined the steps a packet would traverse from PC1 to PC3.



Baxy Today at 2:44 PM
Agreed, Haamid. Well, that wraps up our discussion on the journey of a parcel through the postal system. Let's move on to the next part of our activity.
.
2. How this analogous to a situation where you want to send a message to a friend in another country over the computer networks?

Alright, team, let's dive into the next part of our activity. So, imagine instead of sending a physical package, we're sending a message to a friend in another country over computer networks. How do you think this is similar to sending a parcel through the postal system? (edited)

ipiot Today at 2:44 PM
Well, just like with the parcel, our message would go through multiple steps and locations in the digital realm. Instead of post offices and sorting facilities, we have routers and servers that help route our message to its destination.

Amjad Today at 2:44 PM
Exactly, Iflal. And just like our parcel, the message might go through various checkpoints or nodes along the way. Each router it passes through makes a decision on where to send it next, based on its destination address.

mabrook Today at 2:44 PM
And let's not forget about the concept of protocols. Just like how the postal system has rules and regulations, computer networks operate on protocols like TCP/IP. These protocols ensure that our message is properly formatted and transmitted across the network.

H44mid Today at 2:44 PM
And much like our parcel, our message might traverse across borders and oceans, going through different networks and possibly even encountering delays or congestion along the way. It's a journey through the digital landscape.

Baxy Today at 2:45 PM
Spot on, Haamid. So, in essence, sending a message over computer networks shares many similarities with sending a parcel through the postal system. It's all about navigating through a network of nodes and checkpoints to reach its intended destination. Great insights, team! Let's move on to the next part of our activity.

3. Assume we need to build the following network with two LANs (LAN1 and LAN2). Each group member has a role to play. One group member can be the router and four other group members could be PCs (PC1 and PC2 belong to LAN1 and PC3 and PC4 belong to LAN2). Each device needs to set their own network configuration. The Router needs to set its interfaces/port and PCs need to set its IP address and gateways to be able to make a communication between two LANs. Discuss the configurations of your own device with your group members.

. (edited)
Alright, team, let's get into the networking part of our activity. We need to build a network with two LANs, and each of us has a role to play. Who wants to be the router?

ipiot Today at 2:45 PM
I'll take the router role, Nirosh. I think it's a crucial part of the network, responsible for facilitating communication between the LANs.

Amjad Today at 2:45 PM
Great! So, Iflal will be our router. That leaves the rest of us as PCs. I'll take one of the PCs in LAN1, PC1 maybe.

mabrook Today at 2:46 PM
I'll join you, Amjad. I'll be the other PC in LAN1, PC2.

H44mid Today at 2:46 PM
I'll take one of the PCs in LAN2 then, leaving PC3 and PC4 for Nirosh and me.

Baxy Today at 2:46 PM
Perfect! Now, let's discuss the configurations. Iflal, as the router, you'll need to set up your interfaces or ports. How do you plan to do that?

ipiot Today at 2:46 PM
I'll configure two interfaces, one for each LAN. Each interface will have its own IP address within the respective LAN's subnet. For example, Interface 0 will be for LAN1 with an IP address of 192.168.1.1, and Interface 1 will be for LAN2 with an IP address of 10.1.1.1.

Amjad Today at 2:46 PM
Alright, Mabrook and I are in LAN1. So, for our PCs, we need to set up our IP addresses and gateways. We'll use the subnet 192.168.1.0/24. I'll set my IP address as 192.168.1.2, and Mabrook can be 192.168.1.3. Our gateway will be the IP address of the router's interface in LAN1, which is 192.168.1.1.

mabrook Today at 2:46 PM
Got it, Amjad. I'll configure my PC accordingly.

H44mid Today at 2:46 PM
Sounds good, Mabrook. So, Me and Nirosh are in LAN2. We'll use the subnet 10.1.1.0/24. My IP address will be 10.1.1.2, and Nirosh's will be 10.1.1.3. Our gateway will be the IP address of the router's interface in LAN2, which is 10.1.1.1. I'll set up my PC with those configurations. Looks like we're all set up to communicate between the two LANs now.

Baxy Today at 2:46 PM
Sounds good Haamid, I'll set My PC's configuration for LAN2 as well you mentioned above.

Excellent job, everyone! We've configured our devices properly, and now we're ready to make communication between LAN1 and LAN2 possible. Let's proceed to the next step of our activity.

4. Assume PC1 needs to send a packet to PC3, discuss the steps that the packet needs to go through to reach to PC3.

Amjad Today at 2:47 PM
(PC1) Alright, team, let's get into character for this role play scenario. I'll be PC1 in LAN1, and Mabrook, you're PC2 also in LAN1. Nirosh, you're PC4 in LAN2, and Haamid, you're PC3, also in LAN2. And of course, Iflal, you'll be the router. (edited)

ipiot Today at 2:47 PM
(Router): Got it, Amjad. I'll play my role as the router guiding traffic between LAN1 and LAN2. (edited)

Baxy Today at 2:47 PM
(PC4): And I'll handle both PC4 in LAN2 and the switches in both LANs. Let's make this network communication happen!

H44mid Today at 2:47 PM
(PC3): Ready to receive that packet, Nirosh!

Amjad Today at 2:48 PM
(PC1): Alright, so I have this packet that needs to go to Haamid in LAN2. First, I'll send it to the switch in LAN1.

Baxy Today at 2:48 PM
(PC4/switch): I'll be the switch in LAN1, receiving the packet from Amjad. Once I receive it, I'll check my MAC address table to see which port Haamid's PC is connected to in LAN2. When I receives the packet from PC1, I check its MAC address table to determine the MAC address associated with the destination IP address of the packet.

If the destination MAC address is not found in the my MAC address table, I'll flood the packet to all ports except the one from which I received, allowing the router(Iflal) to learn the MAC address associated with the destination IP address.

Once the I learn the correct port associated with the destination MAC address, I forward the packet only to that port, effectively directing it towards the **router(iflal)**.

mabrook Today at 2:49 PM
(PC2): And once the switch finds Haamid's PC, it forwards the packet directly to Haamid's PC without broadcasting it to every device in LAN1. That's efficiency for you!

Baxy Today at 2:49 PM
(PC4/switch): Exactly, Mabrook.

ipiot Today at 2:49 PM
(Router): Meanwhile, as the router, I've been observing the flow of traffic between LAN1 and LAN2. When the packet arrived at the router from LAN1, I ensured it was directed towards LAN2. Once it reached LAN2, I stepped back and let the switches handle the internal communication within the LAN.

When I receive the packet from LAN1, I examine its destination IP address. This IP address helps me determine if the packet needs to be forwarded to LAN2.

I consult my routing table, which contains information about the network topology and paths to different destinations. By comparing the destination IP address of the packet with the entries in my routing table, I determine that the packet is destined for LAN2.

Once I ascertain that the packet needs to be forwarded to LAN2, I encapsulate it with the appropriate data-link layer addressing information for the next hop, which in this case is the switch connected to LAN2.

I identify that the packet is destined for LAN2 by examining its destination IP address and comparing it against the entries in my routing table. If there's a match for LAN2's network address range in the routing table, I know the packet needs to be forwarded to LAN2.

Finally, I transmit the packet onto the interface connected to LAN2, ensuring that it reaches the switch in LAN2 for further distribution within that LAN. (edited)

Baxy Today at 2:50 PM
(PC4/switch): I received the packet from the router, I'll switch hats and act as the switch in LAN2 and forward it to the correct port associated with Haamid's PC (PC3) based on its MAC address table, enabling the packet to reach its final destination within LAN2.

H44mid Today at 2:50 PM
(PC3): Here I am, waiting for that packet!

Finally, I transmit the packet onto the interface connected to LAN2, ensuring that it reaches the switch in LAN2 for further distribution within that LAN. (edited)

Baxy Today at 2:50 PM
(PC4/switch): I received the packet from the router, I'll switch hats and act as the switch in LAN2 and forward it to the correct port associated with Haamid's PC (PC3) based on its MAC address table, enabling the packet to reach its final destination within LAN2.

H44mid Today at 2:50 PM
(PC3): Here I am, waiting for that packet!

Baxy Today at 2:50 PM
(PC4/switch): Once I locate Haamid's PC, I'll forward the packet directly to Haamid without any unnecessary delay or broadcasting.

Amjad Today at 2:50 PM
(PC1): And that's how the packet successfully traveled from PC1 in LAN1 to PC3 in LAN2, with the help of the switches managing communication within each LAN and the router facilitating communication between LANs.

mabrook Today at 2:50 PM
(PC2): It's amazing to see how all the components work together seamlessly to ensure efficient communication in our network.

Baxy Today at 2:50 PM
(PC4/switch): Absolutely, Mabrook. Each device and component has its role, and together, they make our network function smoothly.

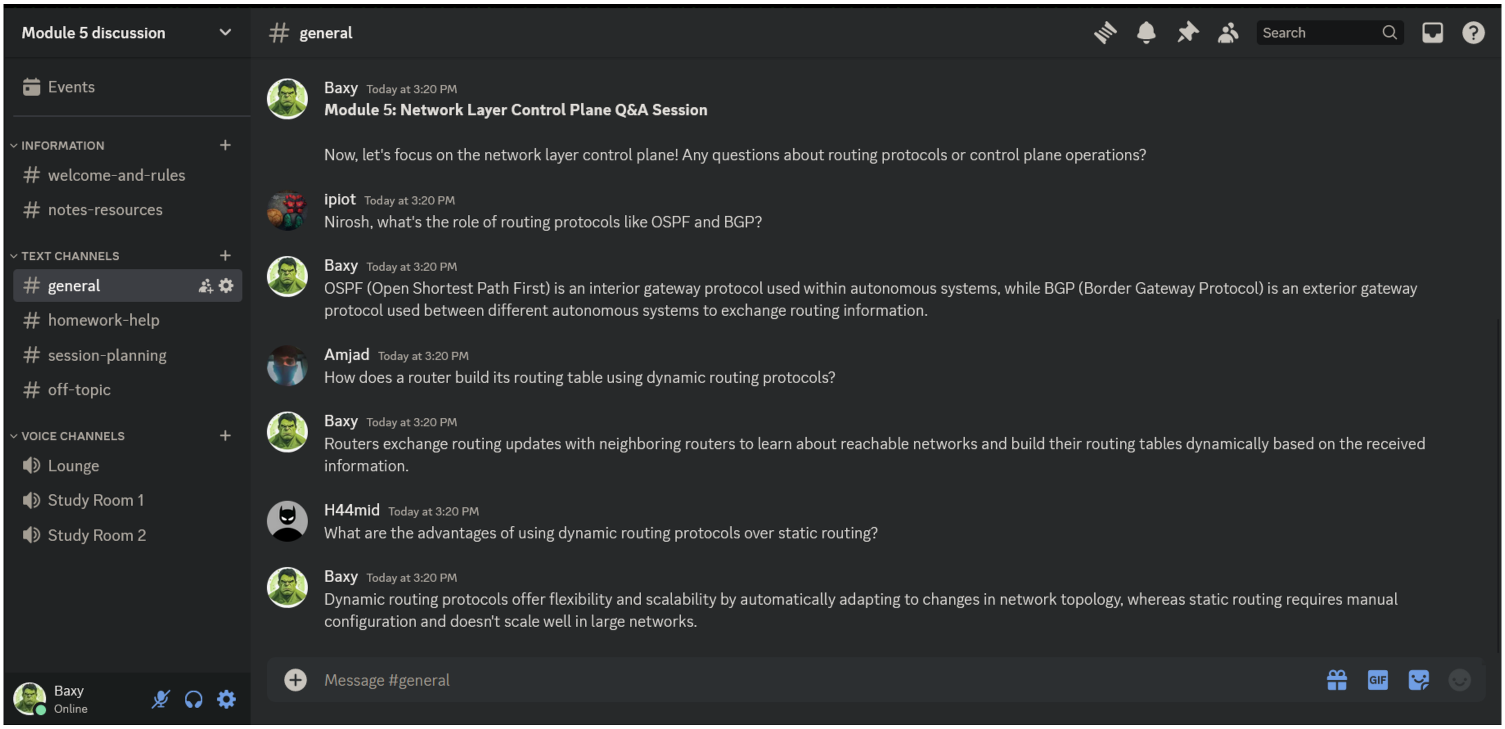
H44mid Today at 2:50 PM
(PC3): Thanks for the smooth delivery, everyone! Our network is running like a well-oiled machine.

NEW

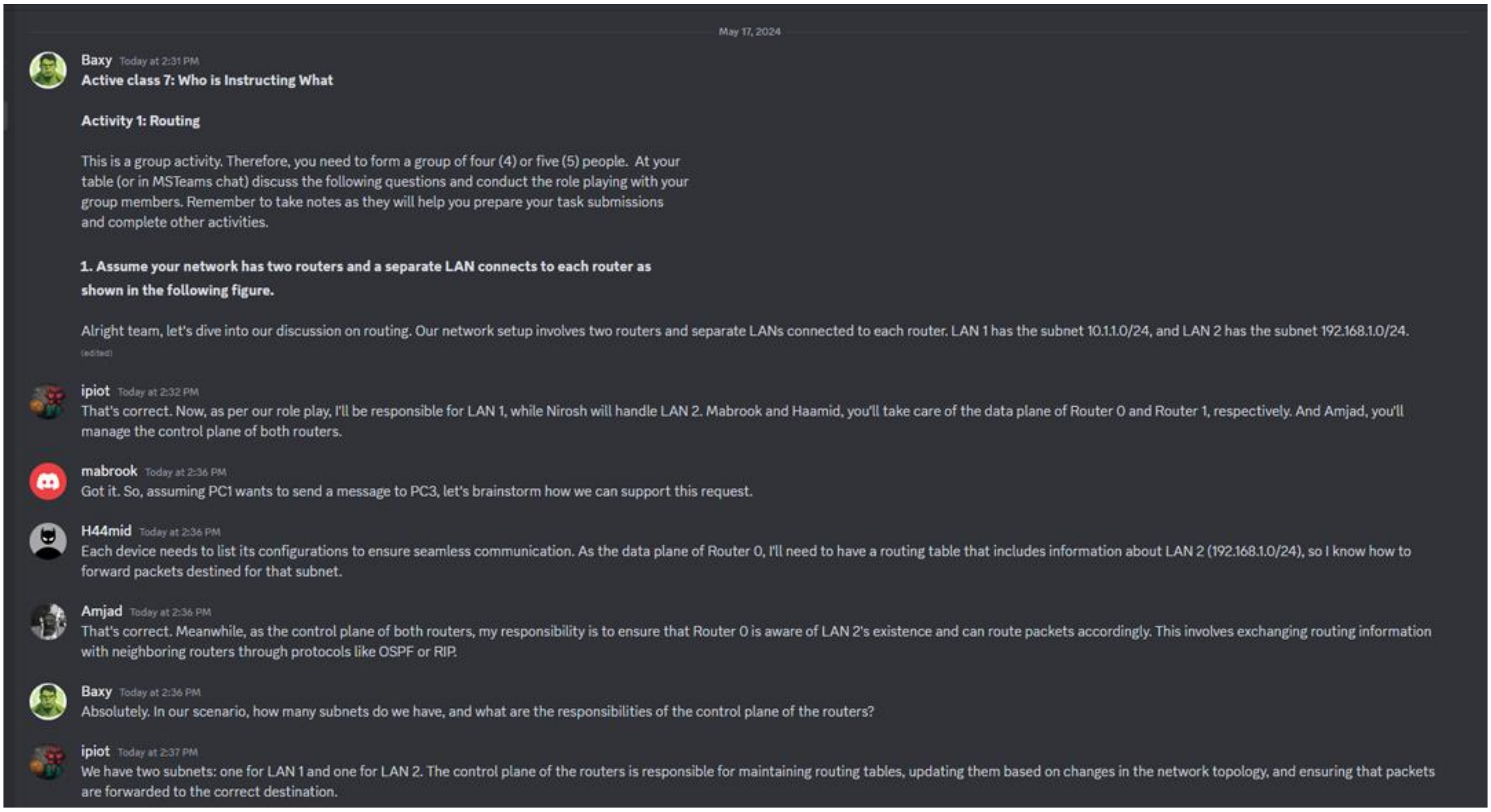
ipiot Today at 2:51 PM
(Router): Great job, team! We've successfully completed the activity, and I'm impressed by how well we coordinated to understand the steps a packet needs to go through to reach its destination in our network. Keep up the excellent work!

Module 5: Network Layer Control Plane:

During discussions on the control plane of the network layer, I facilitated peer review sessions for assignments and projects. By offering constructive feedback and suggestions for improvement, I aimed to enhance my peers' grasp of control plane protocols and routing protocols.



In Activity 1, we gather as a group to discuss routing concepts and simulate a network scenario. We form a team of four or five people and engage in discussions and role-playing exercises. We start by envisioning a network with two routers and separate LANs. Each LAN has its own IP address range: LAN1 with 10.1.1.0/24 and LAN2 with 192.168.1.0/24. To simulate the network, some of us act as LANs, responsible for PCs in each LAN, while others represent the data plane of Router 0 and Router 1. One member takes on the role of the control plane for both routers. We simulate a scenario where PC1 needs to send a message to PC3, requiring coordination and configuration among all devices involved. During our discussion, we analyze the network's subnet structure and discuss the responsibilities of the control plane in routers, including how Router 0 learns about LAN2 and Router 1.


May 17, 2024

Baxy Today at 2:31 PM
Active class 7: Who is Instructing What

Activity 1: Routing

This is a group activity. Therefore, you need to form a group of four (4) or five (5) people. At your table (or in MSTeams chat) discuss the following questions and conduct the role playing with your group members. Remember to take notes as they will help you prepare your task submissions and complete other activities.

1. Assume your network has two routers and a separate LAN connects to each router as shown in the following figure.

Alright team, let's dive into our discussion on routing. Our network setup involves two routers and separate LANs connected to each router. LAN 1 has the subnet 10.1.1.0/24, and LAN 2 has the subnet 192.168.1.0/24. (edited)

ipiot Today at 2:32 PM
That's correct. Now, as per our role play, I'll be responsible for LAN 1, while Nirosh will handle LAN 2. Mabrook and Haamid, you'll take care of the data plane of Router 0 and Router 1, respectively. And Amjad, you'll manage the control plane of both routers.

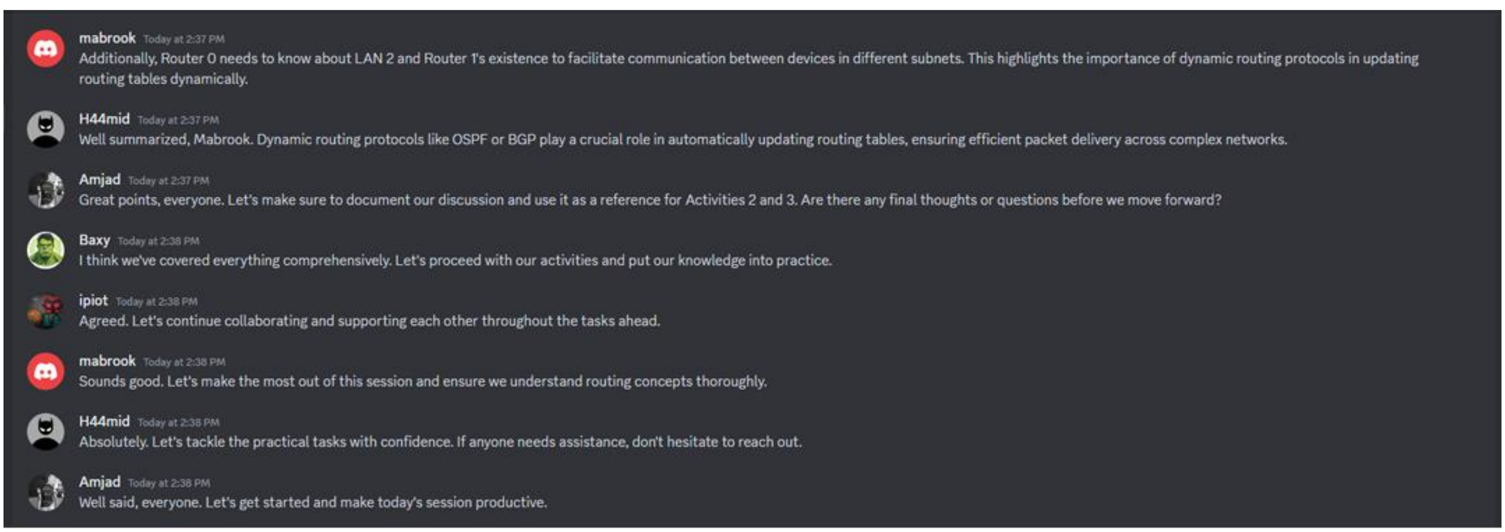
mabrook Today at 2:36 PM
Got it. So, assuming PC1 wants to send a message to PC3, let's brainstorm how we can support this request.

H44mid Today at 2:36 PM
Each device needs to list its configurations to ensure seamless communication. As the data plane of Router 0, I'll need to have a routing table that includes information about LAN 2 (192.168.1.0/24), so I know how to forward packets destined for that subnet.

Amjad Today at 2:36 PM
That's correct. Meanwhile, as the control plane of both routers, my responsibility is to ensure that Router 0 is aware of LAN 2's existence and can route packets accordingly. This involves exchanging routing information with neighboring routers through protocols like OSPF or RIP.

Baxy Today at 2:36 PM
Absolutely. In our scenario, how many subnets do we have, and what are the responsibilities of the control plane of the routers?

ipiot Today at 2:37 PM
We have two subnets: one for LAN 1 and one for LAN 2. The control plane of the routers is responsible for maintaining routing tables, updating them based on changes in the network topology, and ensuring that packets are forwarded to the correct destination.


mabrook Today at 2:37 PM
Additionally, Router 0 needs to know about LAN 2 and Router 1's existence to facilitate communication between devices in different subnets. This highlights the importance of dynamic routing protocols in updating routing tables dynamically.

H44mid Today at 2:37 PM
Well summarized, Mabrook. Dynamic routing protocols like OSPF or BGP play a crucial role in automatically updating routing tables, ensuring efficient packet delivery across complex networks.

Amjad Today at 2:37 PM
Great points, everyone. Let's make sure to document our discussion and use it as a reference for Activities 2 and 3. Are there any final thoughts or questions before we move forward?

Baxy Today at 2:38 PM
I think we've covered everything comprehensively. Let's proceed with our activities and put our knowledge into practice.

ipiot Today at 2:38 PM
Agreed. Let's continue collaborating and supporting each other throughout the tasks ahead.

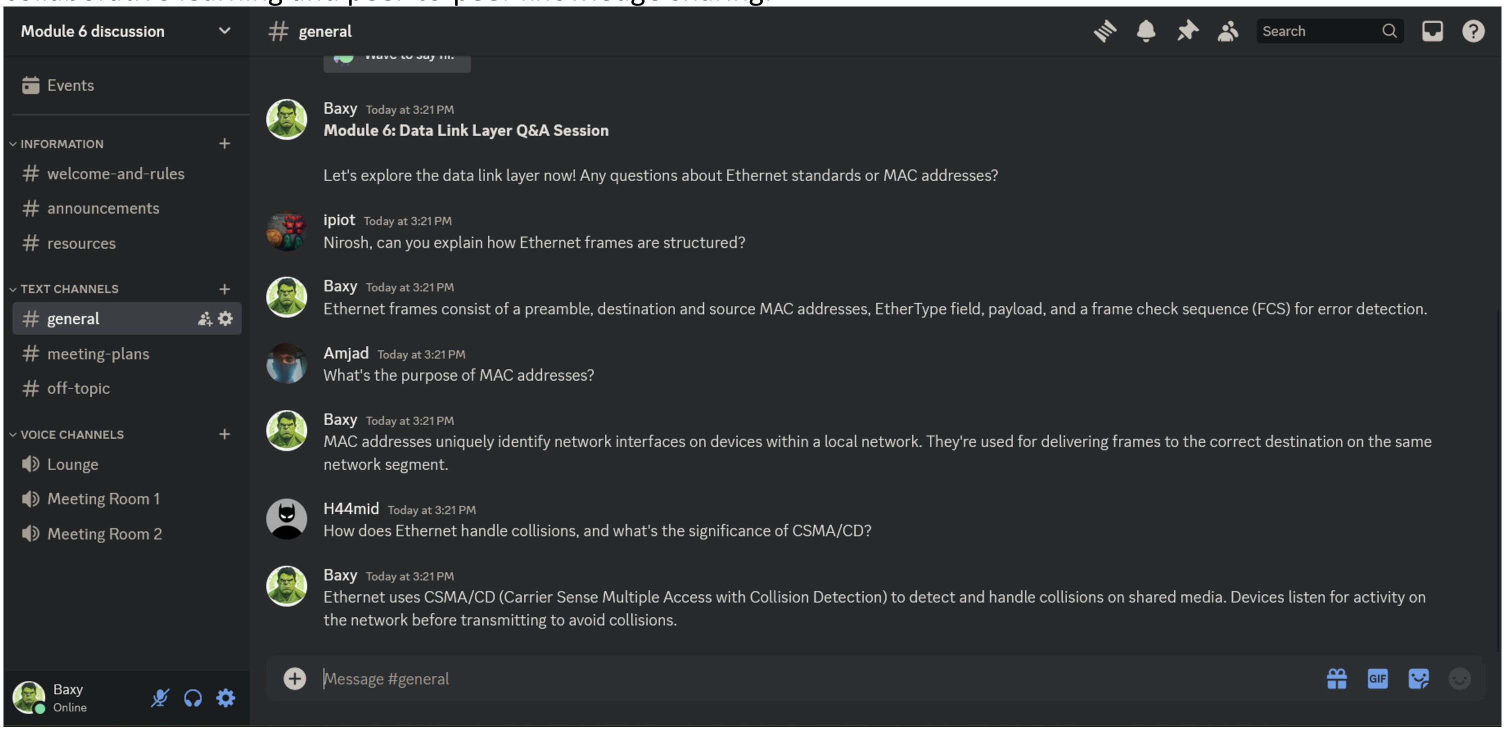
mabrook Today at 2:38 PM
Sounds good. Let's make the most out of this session and ensure we understand routing concepts thoroughly.

H44mid Today at 2:38 PM
Absolutely. Let's tackle the practical tasks with confidence. If anyone needs assistance, don't hesitate to reach out.

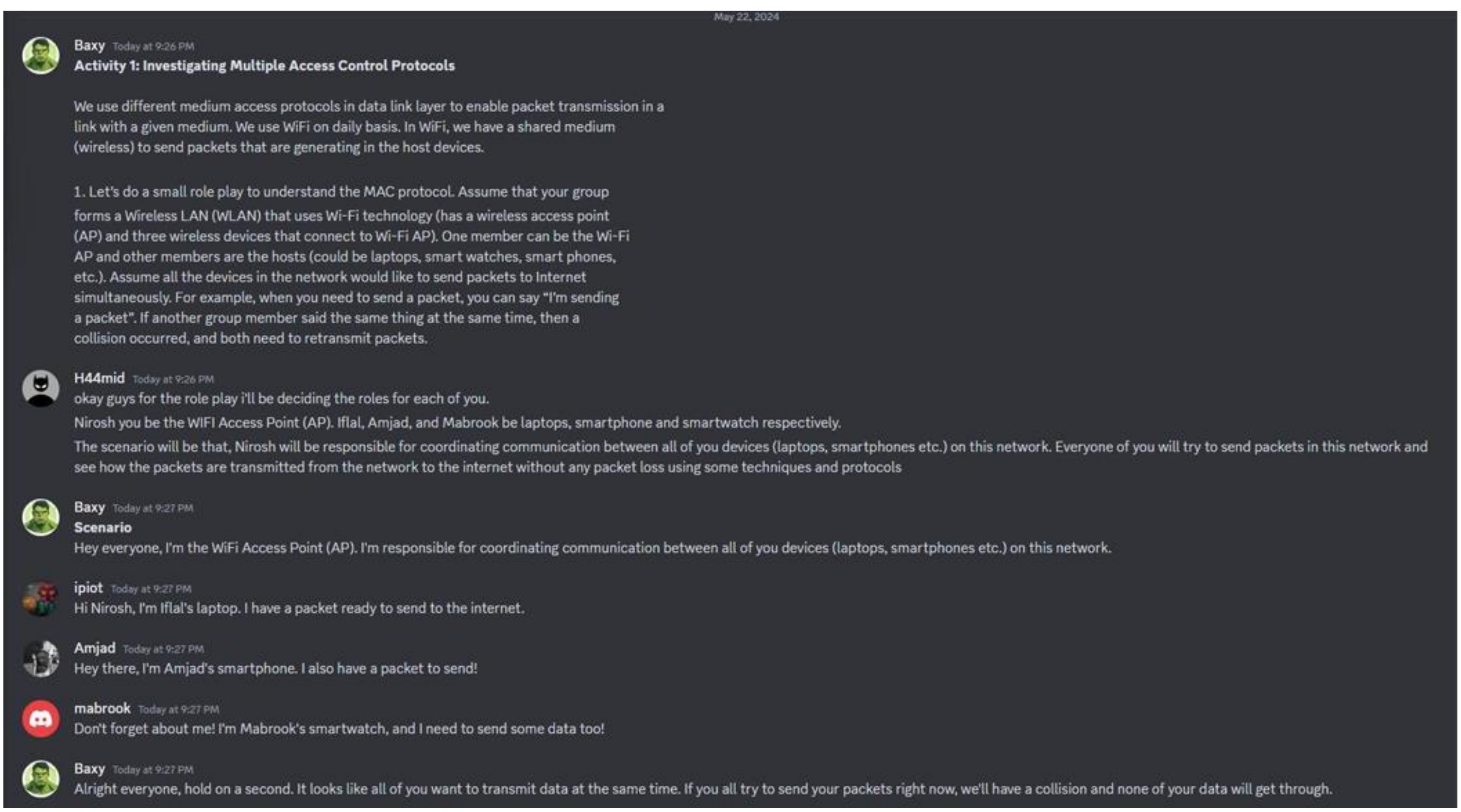
Amjad Today at 2:38 PM
Well said, everyone. Let's get started and make today's session productive.

Module 6: Data Link Layer:

In the data link layer module, I actively participated in Discord study group sessions, leading discussions on topics such as Ethernet standards, MAC addresses, and error detection techniques. These sessions encouraged collaborative learning and peer-to-peer knowledge sharing.



In this activity, we'll be acting out a WiFi network! I'll be the access point (AP), and others will be devices like laptops or phones. Imagine we all want to send data at the same time. Since WiFi is like a shared conversation, we need a plan to avoid garbled messages. We'll design a system where we check if the "airwaves" are free before talking. If two devices try to talk at once, we'll use a random waiting time to avoid interrupting each other again. We'll also draw a timeline to show how this waiting time works. Finally, we'll learn about the real system WiFi uses to manage communication.



H44mid Today at 9:27 PM
Introducing the Protocol:

To avoid collisions, we're going to use a Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) protocol. Here's how it works:
Carrier Sense: Before transmitting a packet, each host will first listen to the wireless channel to see if it's being used by another device.

Short Wait Time: If the channel is idle, the host will wait for a short random amount of time before transmitting. This helps to further reduce the chance of collisions if multiple devices happen to detect an idle channel at the same time.

Transmit Data: After waiting, the host will transmit its packet.

Collision Detection: If a collision occurs while transmitting, all devices will be able to detect it.

Backoff and Retry: In the case of a collision, each device involved will wait for a random amount of time (usually increasing the wait time with each retransmission attempt) before trying to resend their packets.

Baxy Today at 9:28 PM
Running the simulation:

Okay, let's try this out. Iflal, Amjad, and Mabrook, all of you listen to the channel and see if it's idle.

ipiot Today at 9:28 PM
Round 1 (Collision):

The channel is idle! I'm going to transmit my packet now. (edited)

Amjad Today at 9:28 PM
Hold on, I also see an idle channel and I'm transmitting!

mabrook Today at 9:29 PM
Me too!

Baxy Today at 9:29 PM
Uh oh, looks like we have a collision! Everyone stops transmitting and wait for a random amount of time before retrying.
Round 2 (Successful Transmission):

Alright, everyone try again. Check the channel and wait a random amount of time before transmitting.

ipiot Today at 9:29 PM
The channel is clear! I'm transmitting my packet.
(Mabrook and Amjad wait silently) (edited)

Baxy Today at 9:30 PM
Great job Iflal, your packet looks good! I'm sending it to the internet now.

Amjad Today at 9:30 PM
Round 3 (Another Transmission):

The channel is idle now. I'm going to transmit my packet.

Baxy Today at 9:30 PM
Perfect Amjad, your packet came through clearly. I'm sending it on its way.

mabrook Today at 9:31 PM
(Mabrook transmits)
Okay, my turn. The channel is clear and I've waited a bit. Here comes my packet!

Baxy Today at 9:31 PM
Excellent Mabrook, your packet looks good and I'm sending it off now.

There you go! By using CSMA/CA, we were able to avoid collisions and transmit everyone's data packets successfully.
2. What is the medium access control (MAC) protocol that can be used in WiFi?

This is a simplified example of CSMA/CA, and there are more complex variations used in Wi-Fi like CSMA/CA with Collision Detection (CSMA/CA with CD) that can further improve efficiency.

ipiot Today at 9:32 PM
That makes sense! Is CSMA/CA the only protocol used in Wi-Fi?

Baxy Today at 9:32 PM
That's a great question! There are actually other protocols used in Wi-Fi, but CSMA/CA is a common foundation for many of them. Think of them as teammates that help us manage the network traffic smoothly.

Amjad Today at 9:32 PM
Cool! Can you give us some examples?

Baxy Today at 9:32 PM
Sure! One is a set of standards called IEEE 802.11. Imagine these as different versions of Wi-Fi, like having different playbooks for different games. Each version, like 802.11ac or 802.11ax, specifies how we handle things like data speed, how many devices can communicate at once, and which frequency bands we use (think of them as different courts on which we play). All these variations work within the framework of CSMA/CA, but they add extra plays to our game.

mabrook Today at 9:32 PM
Interesting! Are there any other teammates on the field?

Baxy Today at 9:32 PM
Absolutely! Sometimes, even with CSMA/CA, collisions can happen. To minimize that risk, we can use a handshake called Request to Send/Clear to Send (RTS/CTS). It's like raising your hand before speaking in class. The device wanting to transmit asks the access point for permission (sends an RTS message). If the channel is clear, the access point gives the green light (sends a CTS message). This way, everyone knows who's talking and avoids interrupting each other.

ipiot Today at 9:32 PM
That makes sense! Are there any other tricks up our sleeves?

Baxy Today at 9:32 PM
You bet! Large data packets can be a bit cumbersome to transmit, so we can break them down into smaller pieces (fragmentation) before sending them. This reduces the risk of collisions and keeps things moving smoothly. On the other hand, if we have many small packets, we can combine them (aggregation) for better efficiency. It's like grouping smaller tasks together to save time.

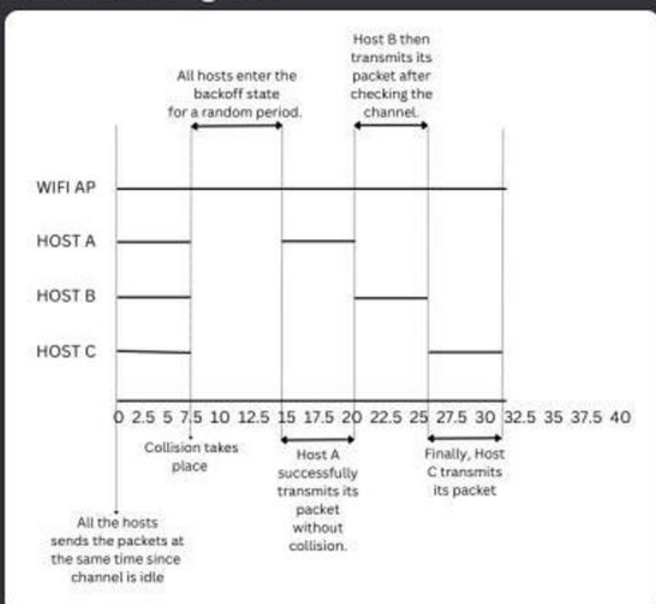
Amjad Today at 9:33 PM
Whoa, that's some complex teamwork!

Baxy Today at 9:33 PM
It can be, but it all works together to make Wi-Fi a reliable and efficient way for us to connect and share data. There are even more protocols used depending on the situation, but these are some of the key players.

mabrook Today at 9:33 PM
This is fascinating! Thanks for explaining Nirosh, I feel like I understand Wi-Fi a lot better now.

Baxy Today at 9:35 PM

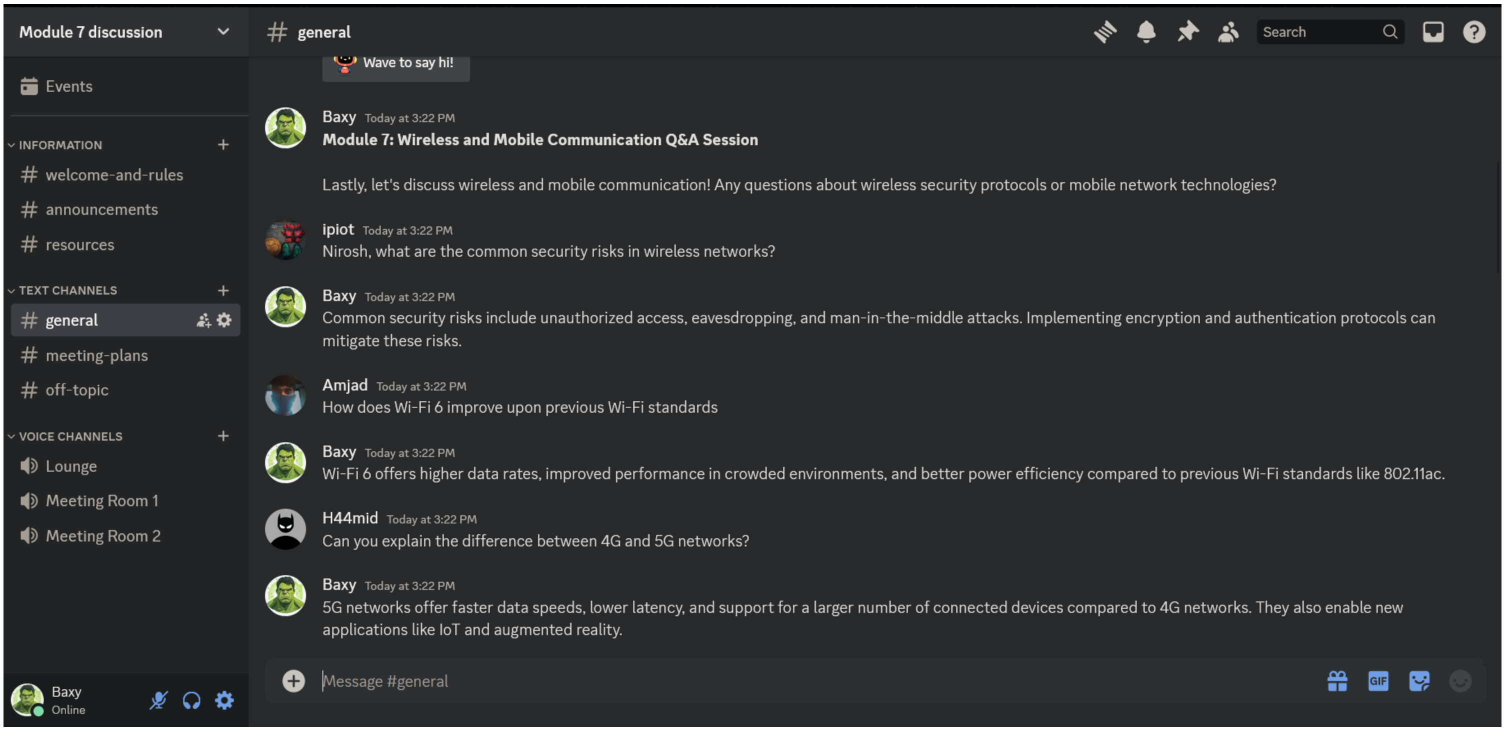
The timeline diagram



👍 2

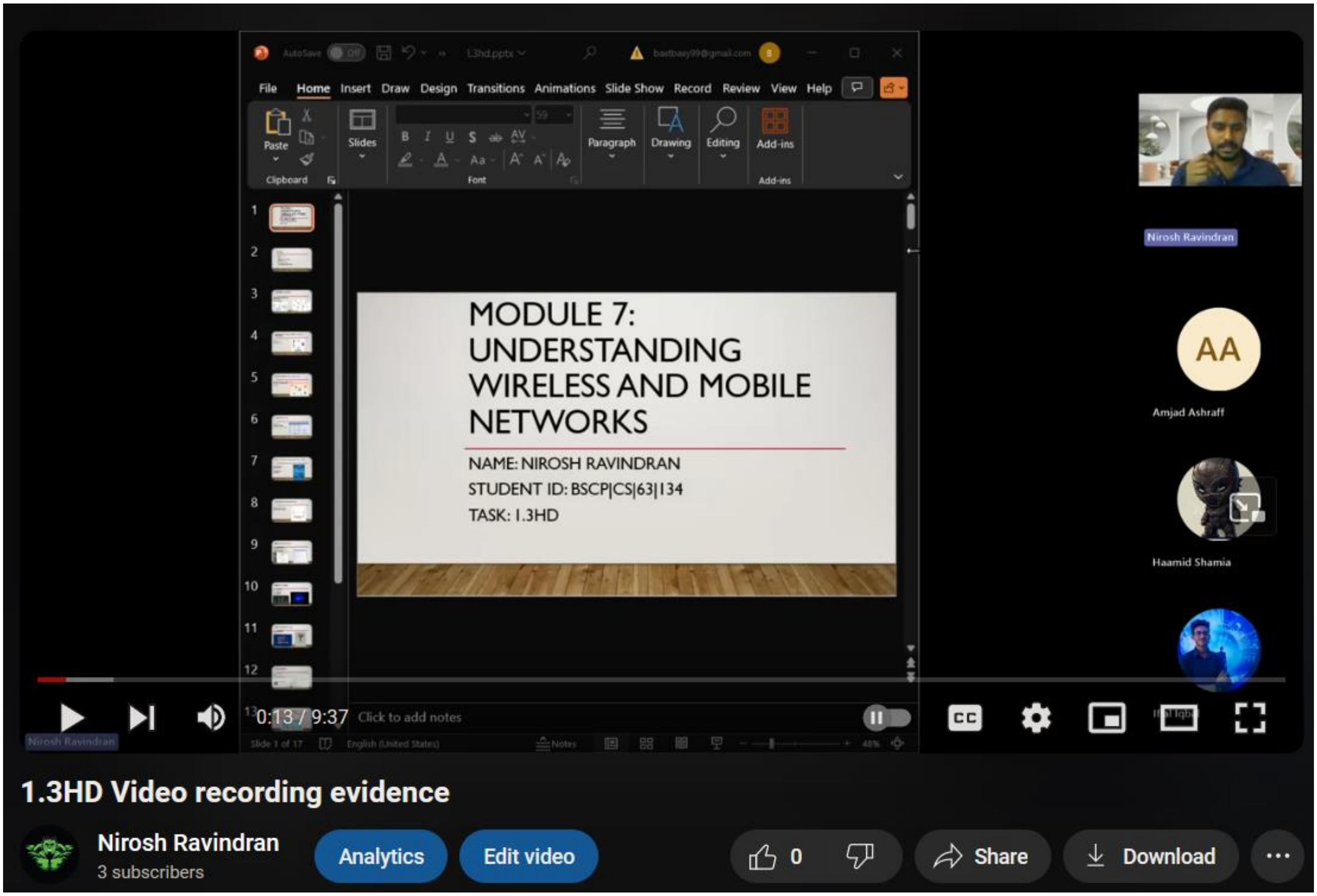
Module 7: Wireless and Mobile Communication:

Finally, during the wireless and mobile communication module, I continued to engage in discussion board conversations, and video tutorial explanation, focusing on wireless security protocols, Wi-Fi standards, and mobile network technologies. By sharing practical examples and case studies, I aimed to bridge the gap between theoretical concepts and real-world applications.



The video tutorial and explanation I did with my peers

https://www.youtube.com/watch?v=sIymBZGNZ98



I have attached the slides I created for the discussion after the conclusion.

Unit Learning Outcomes:

Discipline-specific knowledge and capabilities (GLO1): By actively participating in peer support activities, I deepened my understanding of networking concepts and protocols, strengthening my discipline-specific knowledge and capabilities in the field of information technology.

Communication (GLO2): Engaging in discussions, organizing study group sessions, and creating tutorial videos honed my communication skills, enabling me to convey complex ideas clearly and effectively to my peers.

Digital literacy (GLO3): Through the creation of tutorial videos and utilization of digital collaboration tools like Discord and Google Drive, I enhanced my digital literacy skills, becoming proficient in leveraging technology for educational purposes.

Critical thinking (GLO4): Addressing peers' questions and providing feedback on assignments required critical thinking and analytical skills, as I evaluated different perspectives and approaches to networking challenges.

Problem solving (GLO5): Assisting peers in solving coursework problems improved my problem-solving abilities, as I learned to approach issues methodically and identify effective solutions through collaborative efforts.

Self-management (GLO6): Balancing peer support activities with personal studies taught me valuable time management and self-management skills, enabling me to prioritize tasks effectively and meet deadlines efficiently.

Teamwork (GLO7): Collaborating within study groups and discussion forums enhanced my teamwork skills, as I experienced the benefits of collective problem-solving and active participation within a team environment.

Global citizenship (GLO8): Interacting with peers from diverse backgrounds and perspectives enriched my understanding of global citizenship, fostering a greater appreciation for diversity and the importance of inclusive communication in a globalized world.

Conclusion:

In summary, my participation in peer support activities during the networking modules helped in both my professional development as an information technology specialist and the group's overall learning experience. My grasp of networking principles has grown as a result of my efforts being in line with the unit's learning objectives. I have also developed vital abilities that will help me in my future academic and professional pursuits. This introspective journey emphasizes the value of peer support and the transformational potential of group learning in promoting academic achievement and professional competency in the information technology area.