

**National Institute Of Technology  
Srinagar  
INFORMATION TECHNOLOGY**



**Computer Networking**

---

**Project Report**

**Network Simulator**

---

Team Members	Enrolment No.
1. V Pavani	2021BITE084
2. Shubham	2021BITE011
3. Shubham Patel	2021BITE050

**Lecturer in charge:**

***Mrs Iqra Altaf Gillani***

# Introduction

The network communication simulation project is like a virtual playground where we get to explore how devices talk to each other in a computer network. It's kind of like playing with toy cars on a track, but instead of cars, we have data packets zooming around!

So, imagine you have this big setup with different play areas, each representing a part of how data moves through a network. There's a central area called the Hub, where all the devices connect and share their messages. It's like the hub of a wheel, making sure everything runs smoothly.

Then we have the CRC component, which is like our quality checker. It makes sure the messages don't get messed up during their journey, kind of like how you check your homework for mistakes before turning it in.

The Physical Layer is where things get really physical! It's all about how the actual data bits move through cables and wires. Think of it like the highway that the data travels on.

Next up, we have the Data Link Layer, which is like the traffic cop of our network. It makes sure that data packets are sent and received correctly between devices, kind of like how a post office makes sure your letters get to the right address without getting lost.

Now, if we want to get fancy, we can add in some extra components like the Network Layer, which deals with the big picture stuff like how data moves between different networks. It's like the GPS of our network, figuring out the best route for our data packets to take.

And then there's the Transport Layer, which is all about making sure data gets from one device to another without any hiccups. It's like a super reliable delivery service that guarantees your package arrives safely and on time.

Finally, we have the Application Layer, which is where all the fun stuff happens! This is where apps on different devices talk to each other, like when you send a message on your phone or share a file with a friend. It's like the playground where all the kids meet up to play games and share snacks.

So, in a nutshell, this project lets us dive deep into how networks work by playing around with all these different components. We get to see firsthand how data moves from one device to another, how errors are detected and fixed, and how applications communicate with each other. It's like peeking behind the curtain to see the magic of the internet at work!

# Objectives and Learning Outcomes

1. **Understanding How Devices Talk:** Get the basics of how computers chat with each other, focusing on how they connect physically (like plugging in cables) and how they package up their messages.
2. **Getting Hands-On:** Learn how to set up devices on a network, give them their own addresses (like giving each house on a street its own number), and make them chat using a special type of connection called TCP.
3. **Playing with Data:** Play around with sending messages between devices and watch how they move through different parts of the network, kind of like watching a letter travel through the postal system.
4. **Making Sure Data Stays Safe:** Figure out how to spot and fix mistakes that might happen to our messages as they travel, like a spell-checker for computer data.
5. **Understanding Different Jobs:** Learn about the different roles that parts of the network play, from the physical stuff like cables, all the way up to the software that makes everything work smoothly.
6. **Exploring What Each Part Does:** Dive into the details of how things like the central hub, the IP address giver (DHCP), and the connection maker (TCP) do their jobs to keep everything running smoothly.
7. **Dealing with Problems:** Look at what can go wrong in a network and come up with ways to fix it, kind of like being a detective trying to solve a puzzle.
8. **Learning to Fix Things:** Work on your skills for solving problems and thinking through tricky situations, like figuring out why your internet isn't working and how to fix it.
9. **Teamwork Time:** Practice working with others by setting up communication between lots of devices and apps, just like organizing a big group chat where everyone has to talk to each other.

# System Architecture

Let's break down the components of this network simulation project in simpler terms:

- **Physical Layer:** This is like the base level of our network, where the actual data moves around through cables and wires. Think of it as the stage where our actors (devices) perform. Each device, like Device A, Device B, etc., plays a role in sending and receiving data.

- **Data Link Layer:** This layer makes sure that data gets from one device to another without any mistakes. It's like the editor of our story, checking for errors and making sure everything is formatted correctly. The Data Link Layer module helps with this by handling tasks like breaking up data into manageable chunks and checking for errors using CRC.

- **Network Layer:** Here, we're dealing with how data finds its way through the network. It's like the GPS of our network, figuring out the best route for our data packets to travel. The Network Layer assigns addresses (like street addresses) to devices and keeps track of who's where using DHCP.

- **Transport Layer:** This layer makes sure that data gets to where it needs to go reliably and in the right order. It's like a traffic controller, managing the flow of data and making sure it doesn't get jumbled up. The TransportLayer class helps with this by keeping track of connections and making sure data is sent and received properly.

- **Application Layer:** This is where the fun stuff happens! It's like the party at the end of our journey, where devices get to chat and share information. The ApplicationLayer class helps devices talk to each other by sending messages back and forth.

- **Hub:** This is like the central meeting point in our network, where all the devices come together. It's like a town square where everyone gathers to share news. The Hub class helps by making sure messages from one device get passed on to all the others.

- **Switch:** Think of this as a private chat room within our network. It's like having separate rooms for different groups of people to talk in private. The Switch class helps by directing messages only to the devices that need to see them, based on their unique addresses.

So, in this project, we're basically setting up a whole little world where devices can talk to each other, with different layers of organization to make sure everything runs smoothly. It's like orchestrating a big party where everyone knows where to go and what to do!

# Implementation Details

**L**et's talk about how we actually put together this network simulation project:

- **Choosing the Tools:** We decided to use Python because it's pretty straightforward and has a ton of tools for networking and simulations. It made things easier for us.

- **Designing with Objects:** We organized our code like a big puzzle, with each piece fitting neatly into its own spot. Each layer of the network, like the Network Layer or Data Link Layer, got its own class to handle its specific job.

- **Creating a Family Tree:** We made a family tree of classes, with parent classes like NetworkLayer and child classes like Device A or Device B inheriting from them. It helped keep things organized and made it easy to add new devices later on.

- **Passing the Message:** Communication between layers was like passing a baton in a relay race. Each layer got data from the layer above, did its thing, and then passed it down to the next layer. This kept everything running smoothly and following the rules.

- **Keeping Things Wrapped Up:** We used a technique called data encapsulation, where we wrapped up data from higher layers in special packets as it moved down the protocol stack. It helped keep things tidy and made sure data arrived safely.

- **Giving Out Addresses:** Our DHCP module handled giving out IP addresses to devices, keeping track of who's who in the network. It was like the bouncer at a party, making sure everyone got their own spot.

- **Handling Devices:** We had two main types of network devices: hubs and switches. Hubs were like the central meeting spot, while switches were more like private chat rooms. Both made sure messages got where they needed to go.

- **Dealing with Mistakes:** We built in lots of checks to catch errors and make sure everything was running smoothly. If something didn't look right, we showed an error message to let the user know.

- **Talking to Users:** We made a simple text-based interface so users could interact with our simulation. They could make choices, see what devices were up to, and watch how data moved through the layers. It made things easy to understand and play around with.

So, by putting all these pieces together, we created a simulation that really felt like a mini network, helping people understand how all the parts work together in real life. It was a hands-on way to learn about networking without needing a

# Simulation Process

- **Getting Started:** The user kicked things off by picking which devices would be talking to each other. They also gave some data to send from the source device in a special binary format.

- **Dealing with the Basics:** The first stop was the physical layer, where the data from the source device arrived. We made sure it looked right and stored it in the device's memory. The user also got to choose how the devices would be connected: either in a star shape, like a hub, or with a direct connection.

- **Handling the Data:** Next, the data made its way to the data link layer. Here, we wrapped it up in a special package called a frame by adding some extra information like headers and a CRC checksum to check for errors. Then, we sent the frame on its way to the network layer for more action.

So, in a nutshell, the simulation started with the user setting things up, then moved through the layers of the network, making sure everything was packaged up nicely before it moved on to the next step. It's like making sure your package is securely wrapped before you send it off in the mail!

## Summarize

The network communication simulation project involved several key steps and interactions among the different layers of the network protocol stack. It all started with users inputting the source and destination devices, along with the source device's data in binary format. After that, the physical layer received and validated the data, storing it in the respective device object. Users also had the option to select the type of network connection, choosing between a star topology or a dedicated network connection. Following that, the data link layer encapsulated the source device data into a frame by adding headers and a CRC checksum. This framed data was then passed on to the network layer, where IP addresses were assigned to devices using the DHCP module.

# References

- [1] Stack Overflow.
- [2] Cisco Networking Academy.
- [3] Network World.
- [4] Computer Networking: A Top-Down Approach by James F. Kurose and Keith W. Ross.
- [5] ChatGPT (OpenAI).
- [6] IEEE Communications Society.
- [7] Networking for Dummies by Doug Lowe.
- [8] Network Warrior by Gary A. Donahue.
- [9] Computer Networking: Principles, Protocols and Practice by Olivier Bonaventure.
- [10] TCP/IP Illustrated, Volume 1: The Protocols by W. Richard Stevens, Gary R. Wright, and Addison-Wesley.