#### A Project Report for Computer Networks Lab (22CS505PC)

On

Network Monitoring Tool

Submitted

to

CMR Technical Campus, Hyderabad

In Partial fulfillment for the requirement of the Award of the Degree of

#### BACHELOR OF TECHNOLOGY

##### in

#### COMPUTER SCIENCE & ENGINEERING

#### by

## AJAY

## (237R1A0503)

Under the esteemed guidance of

## Mrs. T. Vasavi

## Assistant Professor



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

CMR TECHNICAL CAMPUS

*An UGC Autonomous Institute*

Accredited by NBA & NAAC with A Grade

(Approved by AICTE, Affiliated to JNTU, Hyderabad)

Kandlakoya (V), Medchal (M), Hyderabad-501 401

(2025-2026)



CERTIFICATE

This to certify that, the Presentation entitled “**Network Monitoring Tool**” is submitted by **Ajay** bearing the Roll Number **237R1A0503** of B.Tech Computer Science and Engineering,In Partial fulfillment for the requirement of the Presentation and for the award of the Degree of Bachelor of Technology during the academic year 2025-26.

Subject Faculty

## Mrs. T. Vasavi

**CMR TECHNICAL CAMPUS**

**UGC AUTONOMOUS**

**Accredited by NBA & NAAC with ‘A’ Grade Approved by AICTE, New Delhi and JNTU Hyderabad**

Department of CSE

# Institute Vision:

To Impart quality education in serene atmosphere thus strive for excellence in Technology and Research.

# Institute Mission:

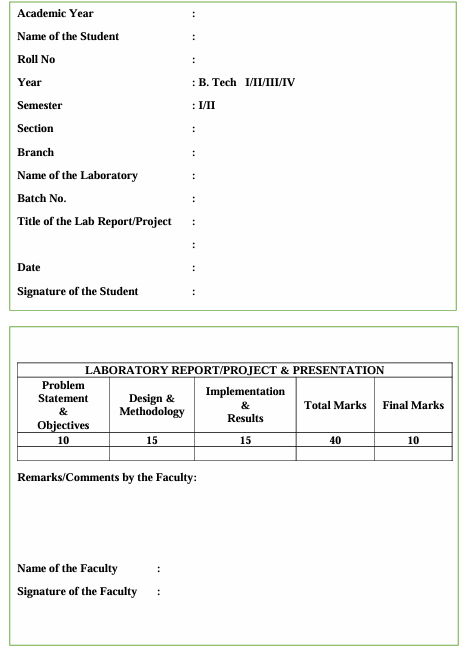
1. To Create state of art facilities for effective Teaching- Learning Process.
2. Pursue and Disseminate Knowledge based research to meet the needs of Industry &Society.
3. Infuse Professional, Ethical and Societal values among Learning Community.

# Department Vision:

To Provide quality education and a conducive learning environment in computer engineering that foster critical thinking, creativity, and practical problem-solving skills.

# Department Mission:

1. To educate the students in fundamental principles of computing and induce the skills needed to solve practical problems.
2. To provide State-of-the-art computing laboratory facilities to promote industry- institute interaction to enhance student’s practical knowledge.
3. To Inculcate self-learning abilities, team spirit, and professional ethics among the students to serve society.



**TABLE OF CONTENTS:**

|  |  |  |
| --- | --- | --- |
| 1 | ABSTRACT | 6 |
| 2 | INTRODUCTION | 7 |
| 3 | PURPOSE | 8 |
| 4 | OBJECTIVES | 9 |
| 5 | USER INSTRUCTIONS | 10-11 |
| 6 | SAMPLE CODE | 12-14 |
| 7 | OUTPUT | 15 |
| 8 | FUTURE IMPROVEMENTS | 16 |
| 9 | CONCLUSION | 17 |
| 10 | REFERENCES | 18 |

ABSTRACT

Network Monitoring Tool This work presents a Network Monitoring Tool designed to measure and analyze real-time network performance parameters such as download speed, upload speed, latency, and packet loss. The tool provides an interactive graphical interface that enables users to easily monitor their network’s behavior and performance. It utilizes Python libraries like speedtest-cli and ping3 to gather accurate network data, while Tkinter is used to create a simple and user-friendly GUI.

The system helps users understand the stability and quality of their internet connection by continuously tracking essential metrics. It serves as a practical implementation of computer networking concepts, offering insights into data transmission efficiency and network reliability. The tool can be further enhanced to include data visualization, alert systems, and historical performance tracking.

INTRODUCTION

In today’s digital world, reliable and high-speed internet connectivity has become essential for both personal and professional activities. Network performance directly affects online tasks such as video streaming, file transfer, cloud computing, and communication. Therefore, monitoring network parameters like speed, latency, and packet loss is crucial to ensure smooth and efficient data transmission.

The Network Monitoring Tool is developed to help users analyze and evaluate the quality of their internet connection in real time. It measures important performance metrics such as download and upload speeds, latency (ping), and packet loss. The tool provides a simple and interactive graphical user interface that displays the results clearly, making it easy for anyone to interpret the network status without technical expertise.

This application demonstrates key computer networking principles such as data transfer rate, delay, and network reliability. By integrating Python’s networking libraries with an intuitive GUI, it serves as an effective learning and analysis tool. Moreover, it can be extended to include advanced features such as graphical trends, alerts, and historical data visualization, enhancing its practical use and effectiveness.

PURPOSE

The main purpose of the Network Monitoring Tool is to provide users with an easy and effective way to assess the performance and stability of their internet connection. It helps identify key aspects of network quality such as speed, latency, and packet loss, which are essential indicators of how efficiently data is transmitted between devices and servers. By offering real-time results, the tool allows users to detect slowdowns or connectivity issues instantly.

Another important goal is to simplify the process of network analysis through an intuitive and user-friendly interface. Many existing monitoring tools are complex and designed for advanced users, but this tool focuses on accessibility and simplicity. With just one click, users can view their connection speed and latency, making it useful for both students and general users who want to understand their network better.

Furthermore, the tool aims to promote a practical understanding of computer networking concepts. It bridges the gap between theoretical learning and real-world application by allowing users to observe how network parameters behave under different conditions. This not only enhances technical knowledge but also encourages users to explore the factors that influence network performance and reliability.

OBJECTIVES

The objectives of this project are as follows:

1. Monitor Network Performance – Continuously track the health and speed of a network connection.
2. Measure Latency and Response Time – Display ping results to evaluate connection stability.
3. Detect Packet Loss – Identify and report data loss during transmission for reliability analysis.
4. Display Real-Time Results – Show network parameters dynamically through a user-friendly interface.
5. Visualize Network Health – Use color indicators or graphs to represent performance levels.
6. Simplify Network Troubleshooting – Help users quickly identify slow or unstable connections.
7. Support Multiple IP Checks – Allow users to test and compare multiple network addresses.
8. Provide Status Alerts – Notify users about connection errors, delays, or timeouts.
9. Enhance User Understanding – Offer a practical way to learn about basic networking concepts and performance evaluation.

USER INSTRUCTIONS

Below are step-by-step instructions on how to use the system effectively:

1. Launching the Application:
   1. Open the Network Monitoring Tool on your computer.
   2. The main window will display fields for IP address input and buttons for testing or monitoring.
2. Starting Network Monitoring:
   1. Enter the IP address of the device or network you wish to monitor.
   2. Click the Start Monitoring button.
   3. The tool will begin collecting data such as ping response time, packet loss, and network speed.
3. Viewing Real-Time Results:
   1. The interface displays key metrics like latency (ping), download/upload speed, and packet loss percentage.
   2. Graphs or indicators update continuously to show the current network status.
   3. Color codes represent performance — Green for stable, Orange for moderate, and Red for poor connection.
4. Stopping the Monitoring Process:
   1. Click the Stop Monitoring button to end the ongoing network test.
   2. The final performance summary will remain visible for review.
5. Refreshing or Changing IP:
   1. To monitor another device or connection, enter a new IP address.
   2. Click Start Monitoring again to view fresh results.
6. Error and Status Messages:
   1. If the IP is unreachable or connection times out, an alert message is displayed.
   2. Ensure your system is connected to the internet or LAN and try again.
7. Closing the Application:
   1. Once monitoring is complete, click Exit or close the window.
   2. The application safely stops all background processes before closing.

SAMPLE CODE:

*# network\_monitor.py*

*import tkinter as tk*

*from tkinter import messagebox*

*import threading*

*import speedtest*

*from ping3 import ping*

*import time*

*# ---------------- Functions ---------------- #*

*def check\_network():*

*try:*

*# Disable button during check*

*check\_btn.config(state="disabled")*

*status\_label.config(text="Checking network...", fg="blue")*

*root.update()*

*# Speed Test*

*st = speedtest.Speedtest()*

*download\_speed = st.download() / 1\_000\_000 # in Mbps*

*upload\_speed = st.upload() / 1\_000\_000 # in Mbps*

*download\_label.config(text=f"Download: {download\_speed:.2f} Mbps")*

*upload\_label.config(text=f"Upload: {upload\_speed:.2f} Mbps")*

*# Ping Test*

*host = "google.com"*

*latency = ping(host) \* 1000 # in ms*

*ping\_label.config(text=f"Ping to {host}: {latency:.2f} ms")*

*status\_label.config(text="Network check completed!", fg="green")*

*except Exception as e:*

*messagebox.showerror("Error", str(e))*

*status\_label.config(text="Error occurred!", fg="red")*

*finally:*

*check\_btn.config(state="normal")*

*def start\_thread():*

*t = threading.Thread(target=check\_network)*

*t.start()*

*# ---------------- GUI Setup ---------------- #*

*root = tk.Tk()*

*root.title("Network Monitoring Tool")*

*root.geometry("400x300")*

*root.resizable(False, False)*

*title\_label = tk.Label(root, text="Mini Project: Network Monitor", font=("Arial", 14, "bold"))*

*title\_label.pack(pady=10)*

*check\_btn = tk.Button(root, text="Check Network", font=("Arial", 12), command=start\_thread)*

*check\_btn.pack(pady=10)*

*status\_label = tk.Label(root, text="Click 'Check Network' to start", fg="blue")*

*status\_label.pack(pady=5)*

*download\_label = tk.Label(root, text="Download: -", font=("Arial", 12))*

*download\_label.pack(pady=5)*

*upload\_label = tk.Label(root, text="Upload: -", font=("Arial", 12))*

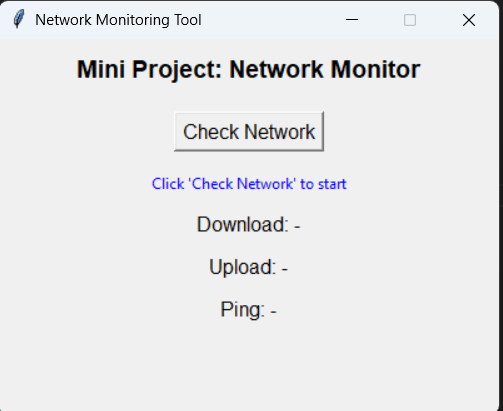
*upload\_label.pack(pady=5)*

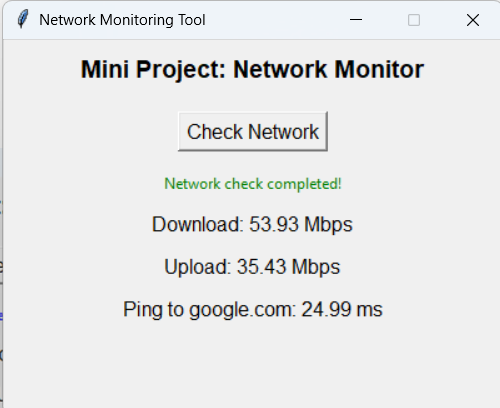
*ping\_label = tk.Label(root, text="Ping: -", font=("Arial", 12))*

*ping\_label.pack(pady=5)*

*root.mainloop()*

OUTPUT





FUTURE IMPROVEMENTS

1. **Graphical Charts:**
   * Display network speed, latency, and packet loss trends over time for better visualization.
2. **Multi-Server Monitoring:**
   * Allow checking multiple servers or IP addresses simultaneously for comprehensive analysis.
3. **Alerts & Notifications:**
   * Notify users when network performance drops below a threshold or packet loss occurs.
4. **Data Logging:**
   * Save historical network data locally for later review and comparison.
5. **Export Reports:**
   * Allow exporting network performance data as CSV or PDF for documentation.
6. **Custom Monitoring Intervals:**
   * Enable users to set custom intervals for periodic network testing.
7. **Advanced Metrics:**
   * Include additional parameters like jitter, throughput, and connection stability score.
8. **Cross-Platform Support:**
   * Make the tool compatible with Windows, Linux, and macOS for wider usability.
9. **Dark Mode:**
   * Add a dark theme to improve readability and reduce eye strain during long monitoring sessions.

CONCLUSION

The Network Monitoring Tool provides a simple and effective way to track and evaluate network performance in real time. By measuring key parameters such as download and upload speeds, latency, and packet loss, it allows users to quickly identify slow or unstable connections. The intuitive graphical interface ensures that even users with minimal technical knowledge can understand and interpret the results easily.

This tool also serves as a practical demonstration of fundamental networking concepts, bridging the gap between theory and real-world application. With potential future enhancements like data visualization, alerts, and cross-platform support, it can be further developed into a more comprehensive solution for network monitoring and analysis.

REFERENCES

 Python Official Documentation – https://docs.python.org/3/

 Tkinter GUI Documentation – https://docs.python.org/3/library/tkinter.html

 Speedtest CLI Library – <https://pypi.org/project/speedtest-cli/>

Ping3 Library Documentation – <https://pypi.org/project/ping3/>

 Stack Overflow – For troubleshooting Python and networking issues – <https://stackoverflow.com>

 GeeksforGeeks – Networking tutorials and Python examples – https://www.geeksforgeeks.org/

 YouTube Tutorials – Python GUI & Networking demonstrations (various creators)