Project Name : Network Analyzer Dashboard

College Name : JECRC Foundation

Project Team :

1. Arpit Khandelwal
2. Avi Maheshwari
3. Divesh Jain
4. Lakshya Gupta
5. Madhur Kedia

Mentors :

Mr. Isaac Theogaraj
Ms. Vishakha Hegde
Ms. Ruchika Solanki

GitHub Link : https://github.com/Avi-Maheshwari121/Network-Analysis-Dashboard-HPE

Problem Statement

01

Metric Complexity

Current tools present an overwhelming set of granular metrics, lacking visual hierarchy & graphical context

02

Interpretability Barrier

Novices find raw technical formats difficult to interpret, leading to data underutilization.

03

Manual Data Bottleneck

Current tools are restricted to manual, offline file analysis from live network interfaces offering no streamlined solution.

04

No Automated Insights

Tools lack an automated system to instantly transform raw statistics into simplified, understandable observations.

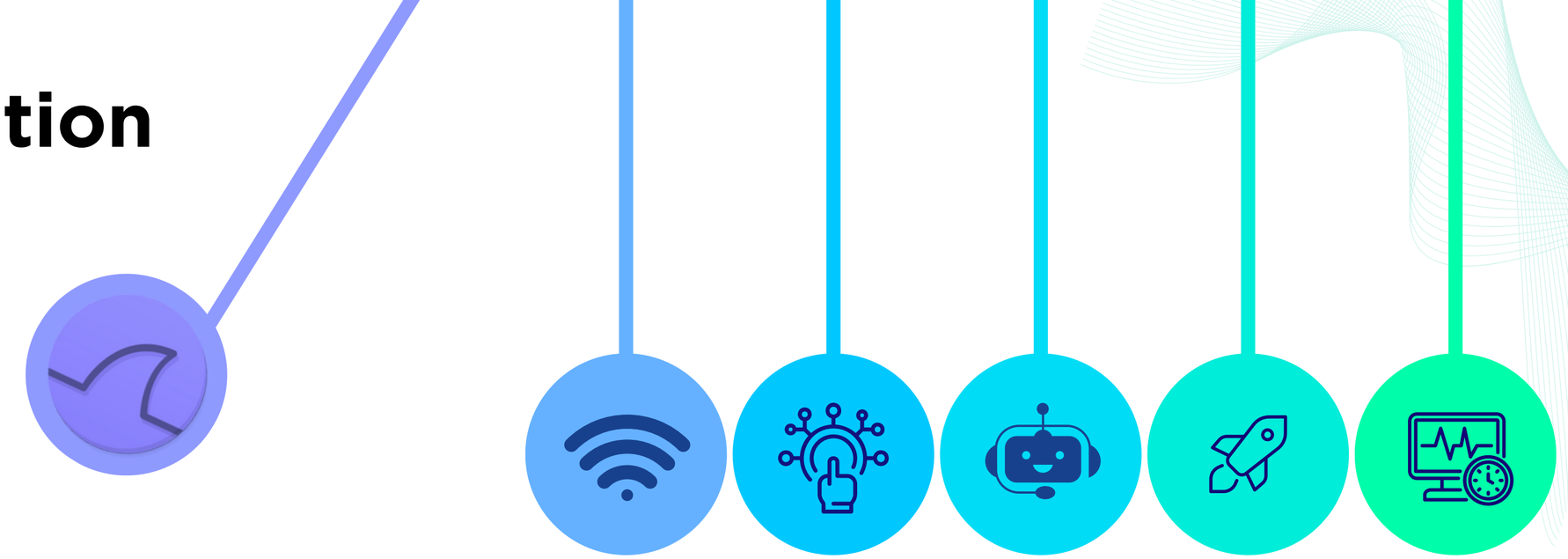
05

Hidden Traffic Patterns

The absence of geospatial maps and flow diagrams obscures critical relationships between devices and traffic sources

Proposed Solution

NetPulse



01

Real-time Packet Capturing via TShark

Direct ingestion of live network traffic using the high-performance TShark engine.

02

Network Metrics Calculation

Automated computation of critical performance data including latency, throughput, and jitter.

03

Interactive Dashboard

A responsive React-based web interface for centralized monitoring and management.

04

AI-Powered Summary

Leveraging Gemini LLM to transform complex technical metrics into plain-language insights.

05

Secure, Scalable & Modern Structure

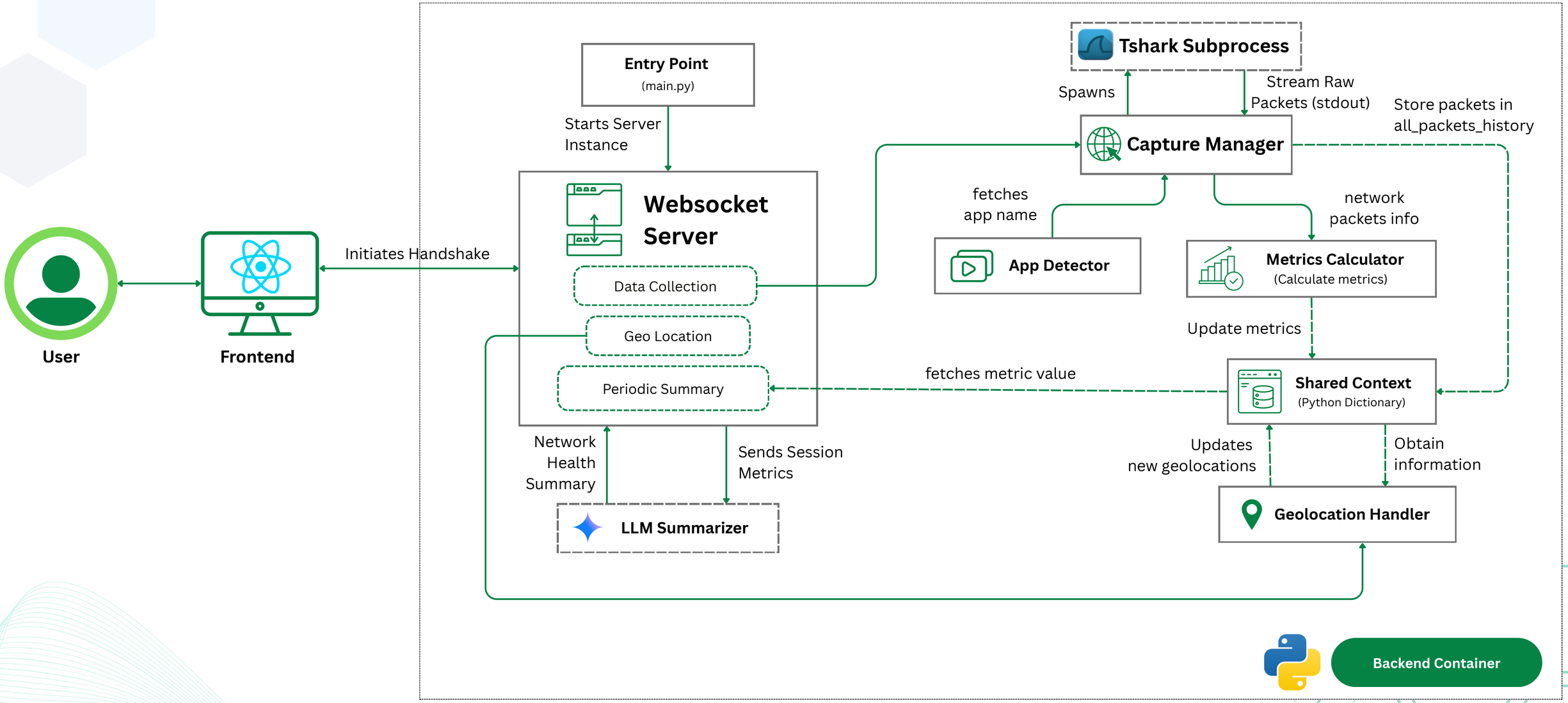
Built on a containerized and modular backend to ensure cross-platform reliability.

06

Live Analysis and Visualizations

Dynamic graphical representation using specialized charting library "Recharts".

System Architecture



KEY FEATURES

Packet Capture Engine

Real-time packet capture, Powered by TShark (Wireshark CLI)

Traffic Analysis

Top outbound conversations,
Protocol distribution,
IPv4 vs IPv6 traffic
Encryption stats

AI-Powered Intelligence

Periodic summaries every 60 seconds and detailed Overall session summary

Geographic Insights

Public IP tracking,
Geolocation map,
App & Host name

Real-Time Network Monitoring

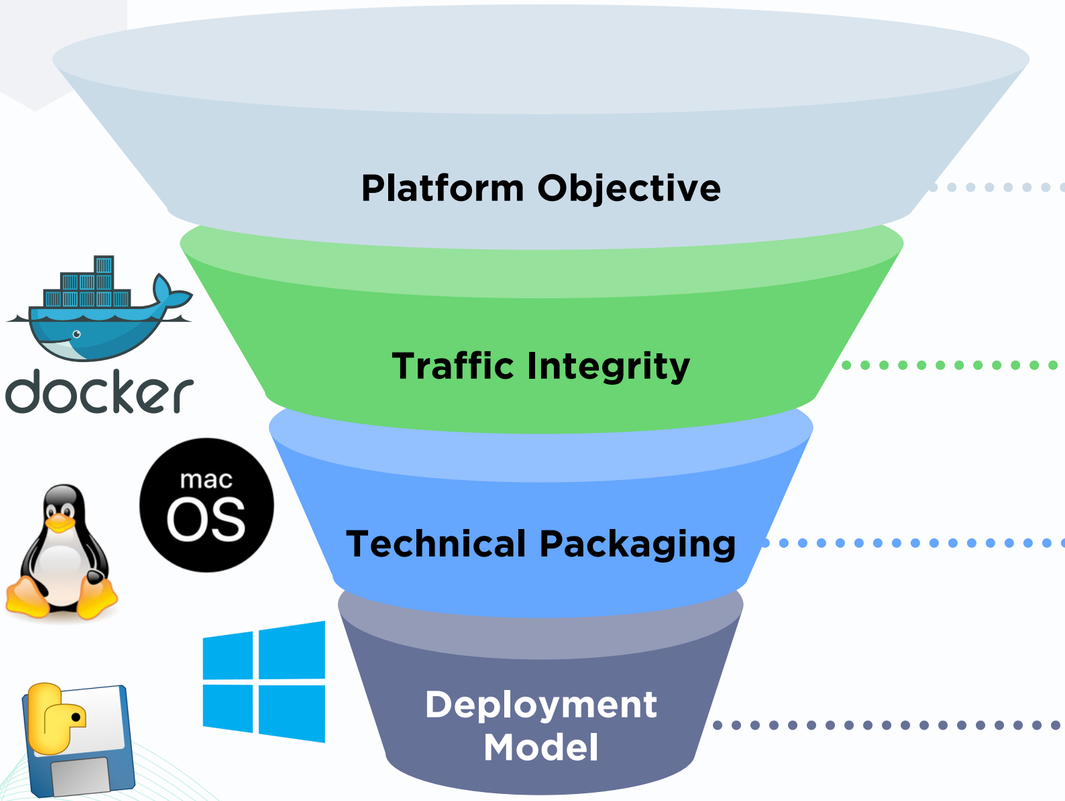
Delivers instant, live-refreshing updates on metrics, instantly reflecting fluctuations

Interactive Dashboard UI

Interactive charts,
Filterable view,
Protocol stats

Deployment Strategy

Native & Containerized Multi-Platform Distribution for Enterprise Networking



Platform Objective

Single-click, dependency-free execution.
No Python, Node.js, or Vite requirements.
Unified GitHub release for Win, Mac, and Linux.

Traffic Integrity

Direct host NIC access in promiscuous mode.
Zero virtualization overhead for real-time monitoring.
Native Npcap (Win) and libpcap (macOS) integration.

Technical Packaging

PyInstaller: Bundled Python, React assets, & backend code.
Docker Hub: Distributed pre-built backend & frontend images.
Runtime Validation: Automatic dependency path checks.

Deployment Model

Desktop (Win/Mac): Standalone .exe and binary executables.
Enterprise (Linux): docker-compose with network_mode: host.

Conclusion

01 Comprehensive network monitoring platform

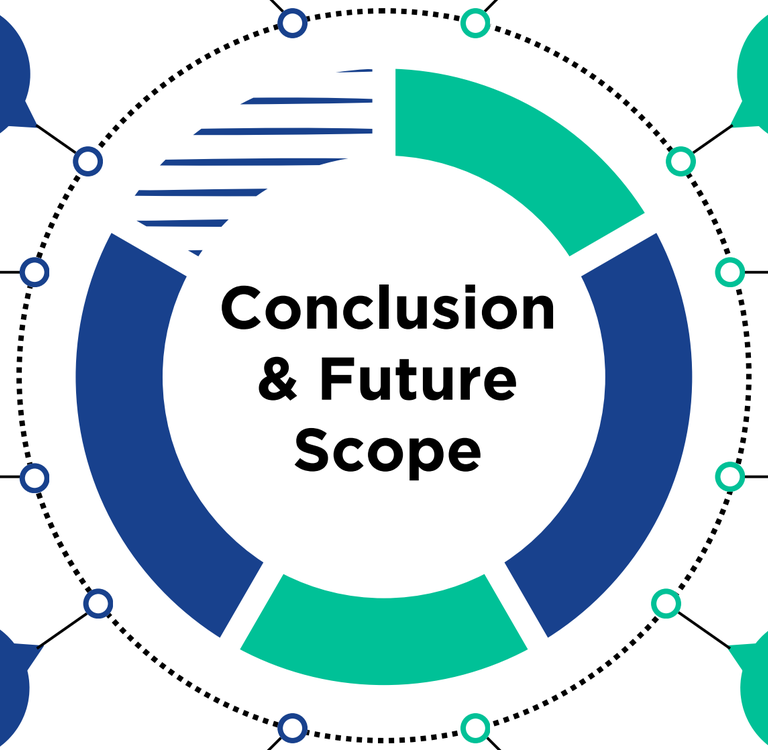
02 Packet-level inspection with protocol analysis

03 Real-time traffic visualization & metrics

04 AI-driven network summaries for insights

05 Interactive flow and top talkers visualization

06 Beginner-friendly network analysis



Future Scope

01 Historical traffic storage in database

02 Predictive analytics for anomaly detection

03 Real-time alerts on network events

04 Deep Application Analysis and Traffic Profiling.

05 Advanced visualizations: Sankey, heatmaps, maps

06 AI-powered chatbot for query insights

Thank You