****

**TrafficGuard: Data flow to destination system**

**presentation in a map view**

**FYP-II Report**

**Submitted by**

201018 – Muhammad Arqam Amin

200944 – Muhammad Jawad Aftab

200997 – Abdul Saboor

**Supervised by**

Dr. Ammara Gul

December 24, 2023

Department of Cyber Security

Faculty of Computing and Artificial Intelligence

Air University

**FYP – II (Report) Mark Form**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | | **To Be Filled by Students** | | |
|  | **Student 01** | | **Student 02** | **Student 03** |
| **Registration No** | 201018 | | 200944 | 200997 |
| **Student Names** | M. Arqam amin | | M. Jawad Aftab | Abdul Saboor |
| **Project Title** | Traffic Guard | | | |
| **Supervisor** | Dr. Ammara Gul | | | |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **To Be Filled by Examiners** | | | | | | |  |
|  | | | **Max** | **Student 01** | **Student 02** | **Student 03** | |
| **Project Management (Attendance)** | | | **10** |  |  |  | |
| **Presentation** | | | | | | | |
|  | Layout and Design | | **10** |  |  |  | |
|  | Delivery and Appearance | | **10** |  |  |  | |
|  | Content and Impact | | **10** |  |  |  | |
| **Project Demo** | | | **10** |  |  |  | |
| **FYP – II Report** | | | | | | | |
|  | Literature Review | | **10** |  |  |  | |
|  | Requirement Analysis | | **10** |  |  |  | |
|  | Proposed Design / Methodology | | **10** |  |  |  | |
|  | Implementation / Experiments | | **10** |  |  |  | |
|  | Business Model | | **10** |  |  |  | |
|  | | **Total** | **100** |  |  |  | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Feedback:** | | | | |
|  |  |  |  |  |
| **Supervisor** | **Examiner** | **Moderator** |
|  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Type (Nature of project) | | | [✓] Development [ ] Research | | |
| Area of specialization | | | [✓ ] Desktop App [ ] Mobile App  [ ] AI based [ ] Embedded System | | |
| FYP ID | | | 2316 | | |
| **Project Group Members** | | | | | |
| **Sr.#** | **Reg. #** | **Student Name** | | **Email ID** | **\*Signature** |
| (i) | 201018 | M. Arqam amin | | [201018@students.au.edu.pk](mailto:201018@students.au.edu.pk) |  |
| (ii) | 200944 | M. Jawad Aftab | | [200944@students.au.edu.pk](mailto:200944@students.au.edu.pk) |  |
| (iii) | 200944 | Abdul Saboor | | [200997@students.au.edu.pk](mailto:200997@students.au.edu.pk) |  |

\*The candidates confirm that the work submitted is their own and appropriate credit has been given where reference has been made to work of others

Plagiarism Certificate

I M. Arqam Amin S/D of M. Amin, group leader of FYP under registration no. 201018 at Department of Cyber Security, Air University, declare that my FYP report is checked for plagiarism by my supervisor.

Name of Group Leader: Signature:

Name of Supervisor: Signature:

Designation:

Name of Co-Supervisor: Signature:

Designation:

Chair, Dept pf Cyber Security: Date:

# Table of Contents

[Table of Contents iv](#_Toc154354249)

[List of Figures v](#_Toc154354250)

[List of Tables vi](#_Toc154354251)

[1. Introduction 1](#_Toc154354252)

[1.1 Goals and Objectives 1](#_Toc154354253)

[1.2 Scope 1](#_Toc154354254)

[2. Background: 2](#_Toc154354256)

[3. Methodology 3](#_Toc154354257)

[3.1 Use Case Diagram 3](#_Toc154354258)

[3.2 Methodology Description 9](#_Toc154354259)

[4. Experiments and Results 12](#_Toc154354260)

[4.1 Experiment Performed 12](#_Toc154354261)

[4.2 Results Achieved So Far 12](#_Toc154354262)

[5. Business Model 22](#_Toc154354263)

[6. Project Gantt chart 23](#_Toc154354264)

[7. References 24](#_Toc154354265)

# List of Figures

[Figure 1 - Use Case Diagram of the Project 3](#_Toc122012094)

[Figure 2 - High Level System Components 6](#_Toc122012095)

[Figure 3 - Front End and Back End Communication 7](#_Toc122012096)

[Figure 4 10](#_Toc122012097)

[Figure 5 11](#_Toc122012098)

[Figure 6 11](#_Toc122012099)

[Figure 7 12](#_Toc122012100)

[Figure 8 12](#_Toc122012101)

[Figure 9 13](#_Toc122012102)

[Figure 10 13](#_Toc122012103)

[Figure 11 - Gantt Chart 15](#_Toc122012104)

# List of Tables

[Table 1 - Use case: 00 4](#_Toc121955559)

[Table 2 - Use case: 01 5](#_Toc121955560)

[Table 3 - Use Case: 02 6](#_Toc121955561)

Table 4 – Use case: 03……………………………………………………………………………………...7

# Introduction

In the dynamic realm of cybersecurity, the demand for a unified tool encompassing network traffic monitoring, IP geolocation, and VPN detection has never been more pronounced. Our project addresses this imperative need by developing a versatile desktop application that runs seamlessly on both Windows and Linux operating systems.

## Goals and Objectives

* Create a desktop application that integrates network traffic monitoring, IP geolocation, and VPN detection to offer a unified solution for cybersecurity professionals.
* Provide organizations with a tool that goes beyond traditional security measures, offering real-time insights into network activities and strengthening defenses against a spectrum of cyber threats.
* Design an intuitive and user-friendly interface for the desktop application, ensuring accessibility for cybersecurity professionals with varying levels of technical expertise.
* Ensure seamless operation on both Windows and Linux operating systems, catering to the diverse IT environments of different organizations.

## Scope

**Scope Statement:**

# Our project includes the development of algorithms that are necessary and required for the analyzing network traffic data, displaying active connections, IP'S of source and destination, filtering the output and continuous monitoring as well as the integration of the Google Maps API for location tracking as well as vpn tracing. Additionally, the application will need to be thoroughly evaluated to ensure its functionality and reliability.

**Proposal:**

Addressing the critical need for a comprehensive cybersecurity tool that integrates network traffic monitoring, IP geolocation, and VPN detection in a single, user-friendly application. Creating a desktop application that runs on both Windows and Linux operating systems, ensuring widespread usability and adaptability to diverse organizational IT environments. Recognizing the absence of a standalone tool that seamlessly combines the functionalities of network traffic monitoring, IP geolocation, and VPN detection, providing users with an all-encompassing solution. Enabling organizations to fortify their cybersecurity defenses by not only monitoring and tracing but also blocking traffic from devices using VPNs, a common tactic employed by malicious entities.

**1.3 Project statement:**

There exists a pressing need for a comprehensive, standalone solution that not only monitors both incoming and outgoing network traffic, location tracing of the respective Ip addresses but also offers the crucial capability to trace Virtual Private Network (VPN) connections.

# Background:

One of the distinguishing aspects of our initiative is its approach, combining network traffic monitoring, IP geolocation, and VPN detection within a single desktop application. This integration not only streamlines the user experience but also fills a void in the market – there is currently no standalone tool that comprehensively addresses these three critical aspects of cybersecurity. Furthermore, the decision to create a desktop application ensures accessibility across multiple operating systems, catering to a diverse user base. This approach acknowledges the diversity in organizational IT infrastructures and aims to provide a unified solution that is both versatile and user-friendly.

As organizations grapple with the challenges posed by cyber threats and the evasive tactics of malicious actors, our project aims to deliver a robust, all-encompassing tool. This tool is not merely a technological innovation; it is a response to the evolving threat landscape, designed to empower cybersecurity professionals with a unified solution to fortify their defenses against a spectrum of cyber threats.

# Methodology

## Use Case Diagram

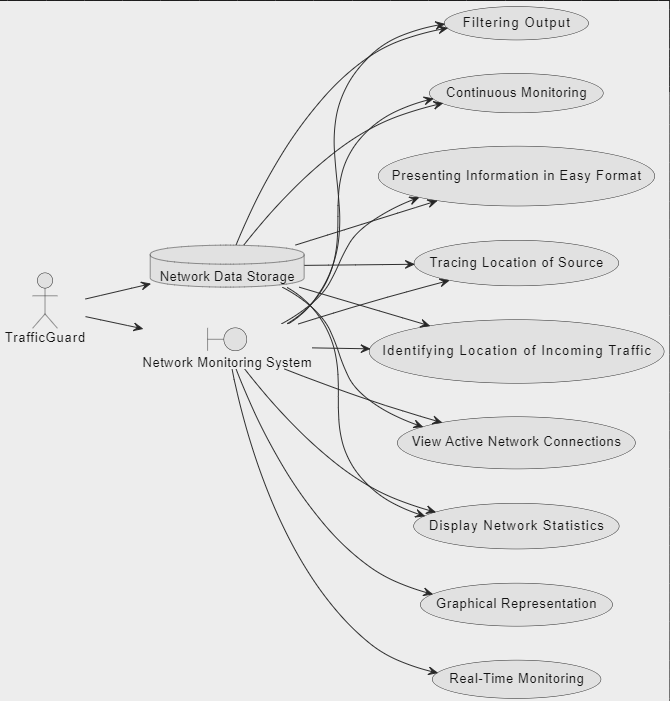


Figure 1 - Use Case Diagram of the Project

**Use Case ID: 00**

|  |  |
| --- | --- |
| **Use Case Name:** | **View Active Network Connections** |
| **Primary Actor:** | TrafficGuard |
| **Goal In Context:** | To provide the TrafficGuard with real-time information about active network connections. |
| **Preconditions:** | The network monitoring system is operational, and the TrafficGuard has access to the system. |
| **Postconditions:** | The TrafficGuard successfully views the active network connections on the dashboard. |
| **Priority:** | High, critical for monitoring network activity. |
| **Frequency of Use:** | Frequently, as the TrafficGuard needs to monitor network connections in real-time. |
| **Normal Course of Events:** | * The TrafficGuard accesses the network monitoring system. * The system displays the dashboard with active network connections. * The TrafficGuard reviews the information on active network connections. |
| **Alternative Courses:** | None |
| **Scenario:** | * The TrafficGuard logs into the system. * The system presents a real-time dashboard with active network connections. * The TrafficGuard navigates through the dashboard to view specific details. * The system updates the display continuously to reflect real-time changes. |
| **Exceptions:** | If there is an issue with system availability, the TrafficGuard may not be able to view real-time data. |
| **Assumption:** | The network monitoring system is configured correctly, and there are no connectivity issues during the monitoring period. |

Table 1 - Use case: 00

**Use Case ID: 01**

|  |  |
| --- | --- |
| **Use Case Name:** | **Display Network Statistics** |
| **Primary Actor:** | TrafficGuard |
| **Goal In Context:** | To provide the TrafficGuard with real-time information about active network connections. |
| **Preconditions:** | To provide the TrafficGuard with detailed statistics for different network protocols. |
| **Postconditions:** | The TrafficGuard successfully views detailed statistics for network protocols on the dashboard. |
| **Priority:** | High, crucial for understanding network performance. |
| **Frequency of Use:** | Frequently, as the TrafficGuard needs to monitor network statistics regularly. |
| **Normal Course of Events:** | * The TrafficGuard accesses the network monitoring system. * The system displays the dashboard with detailed network statistics. * The TrafficGuard reviews the information on network protocols. |
| **Alternative Courses:** | None |
| **Scenario:** | * The TrafficGuard logs into the system. * The system presents a dashboard with detailed network statistics. * The TrafficGuard explores different protocols and reviews relevant information. |
| **Exceptions:** | If there is an issue with system availability, the TrafficGuard may not be able to view real-time data. |
| **Assumption:** | The network monitoring system is configured correctly, and there are no connectivity issues during the monitoring period. |

Table 2 - Use case: 01

**Use Case ID: 02**

|  |  |
| --- | --- |
| **Use Case Name:** | **Continuous Monitoring** |
| **Primary Actor:** | TrafficGuard |
| **Goal In Context:** | To provide the TrafficGuard with continuous real-time updates on network traffic, ensuring prompt identification of changes. |
| **Preconditions:** | The network monitoring system is operational, and the TrafficGuard has access to the system. |
| **Postconditions:** | The TrafficGuard successfully monitors network traffic continuously, with the dashboard updating at regular intervals. |
| **Priority:** | High, critical for identifying and responding to changes in network activity promptly. |
| **Frequency of Use:** | Continuous, as the TrafficGuard requires real-time updates during the entire monitoring session. |
| **Normal Course of Events:** | * The TrafficGuard accesses the network monitoring system. * The system displays the dashboard with continuous real-time updates on network traffic. * The TrafficGuard regularly reviews the updated information for changes in network connections or statistics. |
| **Alternative Courses:** | None |
| **Scenario:** | * The TrafficGuard logs into the system. * The system presents a real-time dashboard with continuous updates on network traffic. * The TrafficGuard remains engaged in monitoring activities, with the dashboard updating at regular intervals. |
| **Exceptions:** | If there is an issue with system availability, continuous monitoring may be interrupted, and real-time updates may be delayed. |
| **Assumption:** | The network monitoring system is configured correctly, and there are no connectivity issues during the continuous monitoring session. |

Table 3 - Use Case: 02

**Use Case ID: 03**

|  |  |
| --- | --- |
| **Use Case Name:** | **Tracing Location of Source** |
| **Primary Actor:** | TrafficGuard |
| **Goal In Context:** | To trace and display the geographical location of the source of incoming network traffic using Google Maps. |
| **Preconditions:** | The network monitoring system is operational, and the TrafficGuard has access to the system. |
| **Postconditions:** | The TrafficGuard successfully traces and views the geographical location of the source on Google Maps. |
| **Priority:** | High, crucial for identifying the origin of incoming network traffic. |
| **Frequency of Use:** | As needed, particularly when investigating the source of suspicious network activity. |
| **Normal Course of Events:** | * The TrafficGuard accesses the network monitoring system. * The system displays the dashboard with options for tracing the geographical location of the source. * The TrafficGuard selects a specific source or incoming traffic. * The system integrates with Google Maps to display the geographical location of the source. |
| **Alternative Courses:** | None |
| **Scenario:** | * The TrafficGuard logs into the system. * The system presents a dashboard with options for tracing the location of the source. * The TrafficGuard selects a specific source or incoming traffic. |
| **Exceptions:** | If there is an issue with Google Maps integration or system availability, the geographical location may not be accurately displayed. |
| **Assumption:** | The network monitoring system is configured correctly, and there are no connectivity issues during the tracing process. |

Table 4 - Use Case: 03

## Methodology Description

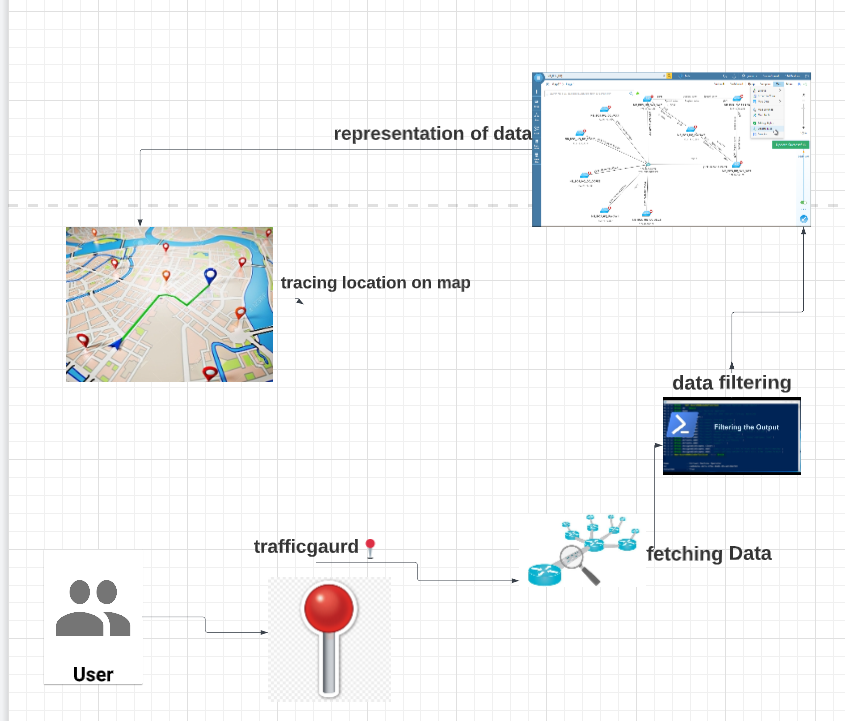


Figure 2 - High Level System Components

**Fetching Data:**

Iterate through the captured packets, and for each packet, access the relevant information. pyshark provides a convenient way to access different layers of the OSI model, including data link, network, transport, and application layers.Extract information such as source and destination IP addresses, protocol type, service in use, timestamps, and more.

**Analysis and Filtering:**Apply filters and conditions to analyze specific aspects of the traffic. For example:Filter packets based on specific protocols (e.g., TCP, UDP).

**Displaying:**Depending on your requirements, you can choose to Display real-time or analyzed information in a user interface.

**Location tracing:**

integration with IP-API:Utilize an external geolocation service like IP-API to obtain detailed information about the identified IP addresses.IP-API is a web service that provides geolocation data based on IP addresses. You can make HTTP requests to their API endpoint with the target IP addresses.

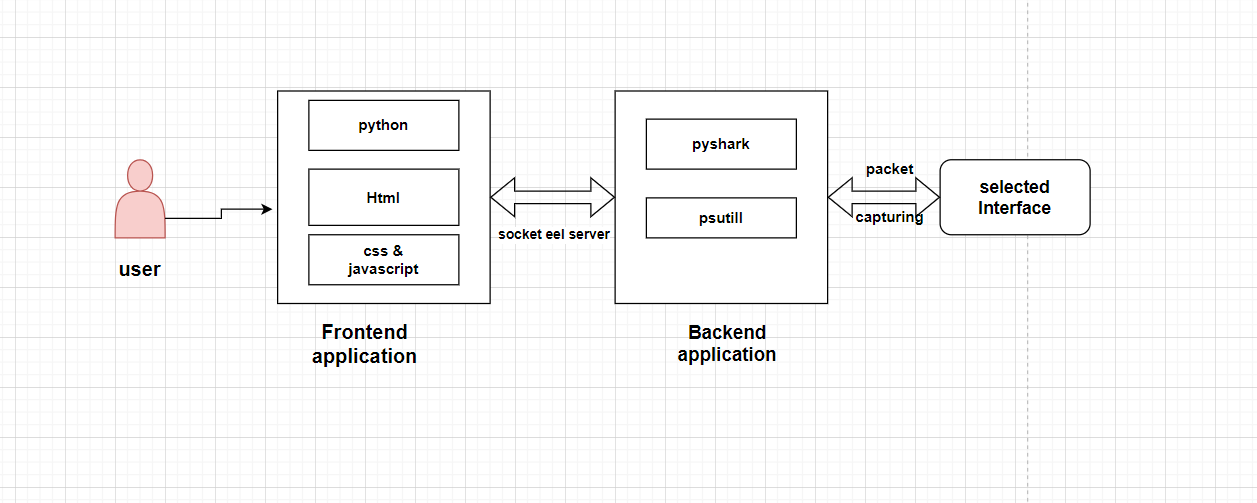


Figure 3 - Front End and Back End Communication

**User**

A user would be an application user who wants to view active connections and get information about them, furthermore who needs to trace location.

**Front End Application**

The frontend of the application is created using standard web technologies, such as HTML, CSS, and JavaScript. This allows for flexibility in designing the user interface and ensures compatibility with web standards. Eel acts as a bridge between the Python backend and the frontend. It provides a simple way to expose Python functions to JavaScript and vice versa. Eel is initialized in the Python script, specifying the folder containing the frontend files. This folder structure typically separates HTML, CSS, and JavaScript files. Python functions that need to be accessible from JavaScript are decorated with @eel.expose. This makes these functions callable from the frontend. JavaScript functions in the frontend can call the exposed Python functions using the eel object. This allows for bidirectional communication between the frontend and backend. The HTML, CSS, and JavaScript files define the layout and behavior of the user interface. Designing the UI involves creating a visually appealing and user-friendly experience. The Eel application is started by calling the eel.start function in the Python script, specifying the main HTML file to display. This opens a web window that renders the frontend.

Eel facilitates communication by providing a mechanism for passing data between Python and JavaScript. This allows for dynamic updates and interactions in the user interface based on backend processing.

**Back End Application**

We Developed the backend logic of the application in Python, leveraging PyShark for packet capturing. Use Python's capabilities to process, analyze, and store the extracted data. We Extracted relevant information from the captured packets using PyShark. This information includes details such as source and destination IP addresses, protocols, and payload data and time stamps etc. Integrate PSUtil, a Python library, for geolocation tracing based on IP addresses. PSUtil can be used to retrieve information about the system, including network-related details such as active connections and IP addresses. Map the IP addresses obtained from the captured packets to geolocation information using PSUtil. PSUtil provide details such as the country, region, city, and ISP associated with each IP address.we Utilized Python Eel to facilitate communication between the backend and the frontend. Exposed backend functions to the frontend to enable bidirectional communication. We Designed the frontend using HTML, CSS, and JavaScript to display the captured network data and geolocation information. We Created an intuitive and user-friendly interface for presenting the information.

# Experiments and Results

## Experiment Performed

|  |  |
| --- | --- |
| **Activities** | **Status** |
| Study of the Related work | Completed |
| Designing Framework | Completed |
| Defining Resources | Completed |
| Graphical User Interface | Completed. |
| Analysis of the proposed scheme | In progress… |
| Location tracing | Need more working… |
| Vpn tracing | Pending… |
| Testing | Pending… |

## Results Achieved So Far

|  |  |  |
| --- | --- | --- |
| **Activities** | **Status** | **Findings** |
| Study of the Related work | Completed | Networking, python. |
| Designing Framework | Completed | System diagrams, Use cases |
| Defining Resources | Completed | Systems,python,JavaScript etc. |
| Graphical User Interface | Completed. | User-friendly GUI |
| Analysis of the proposed scheme | In progress… | Backend working |
| Location tracing | Need more working | Geolocation of the respective ip addresses. |
| Vpn tracing | Pending… | Connections that are using vpn’s. |
| Testing | Pending… | Testing of the code |

**Dashboard:**

Currently, this is appearance of our dashboard.



*Figure 4*



Figure 5

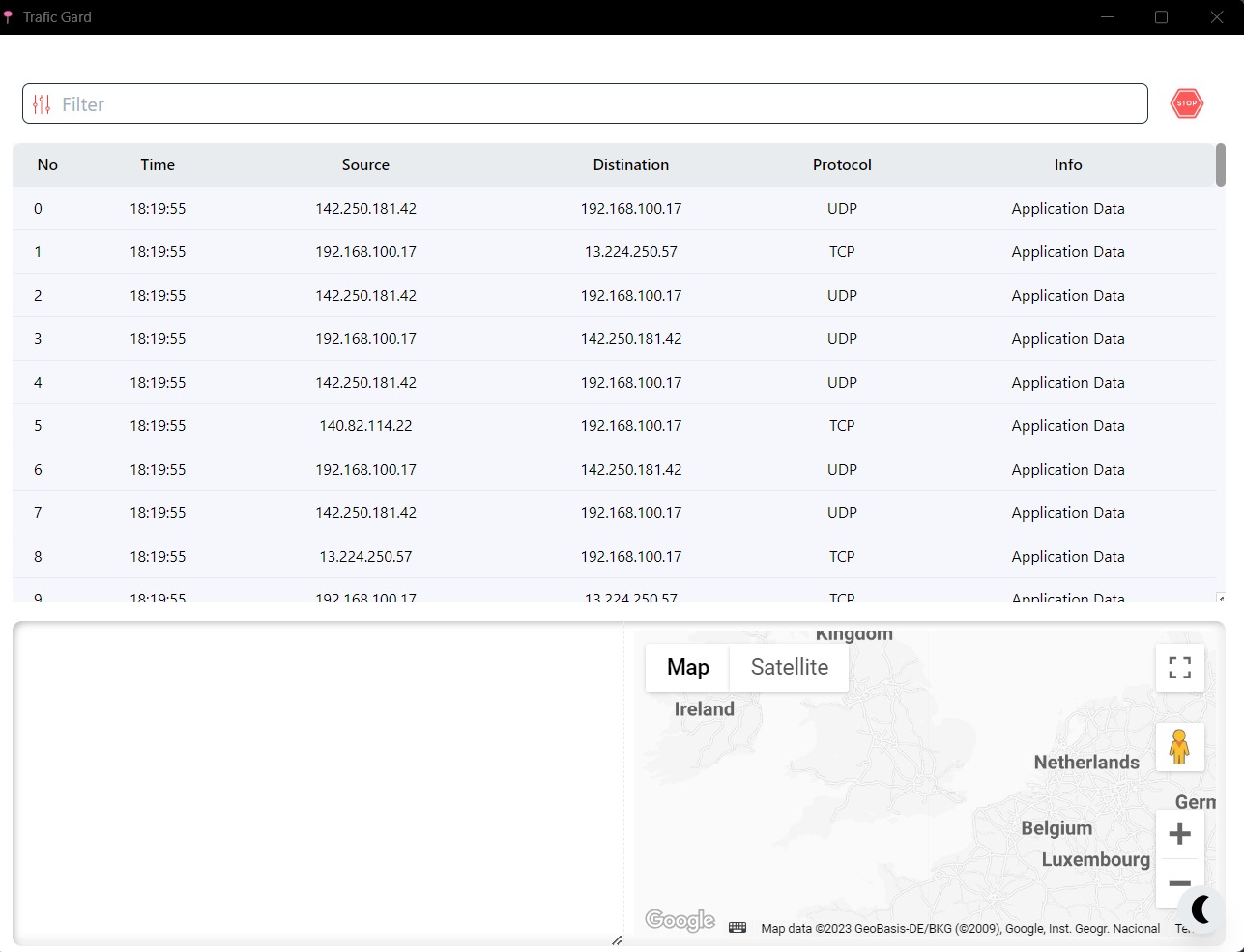
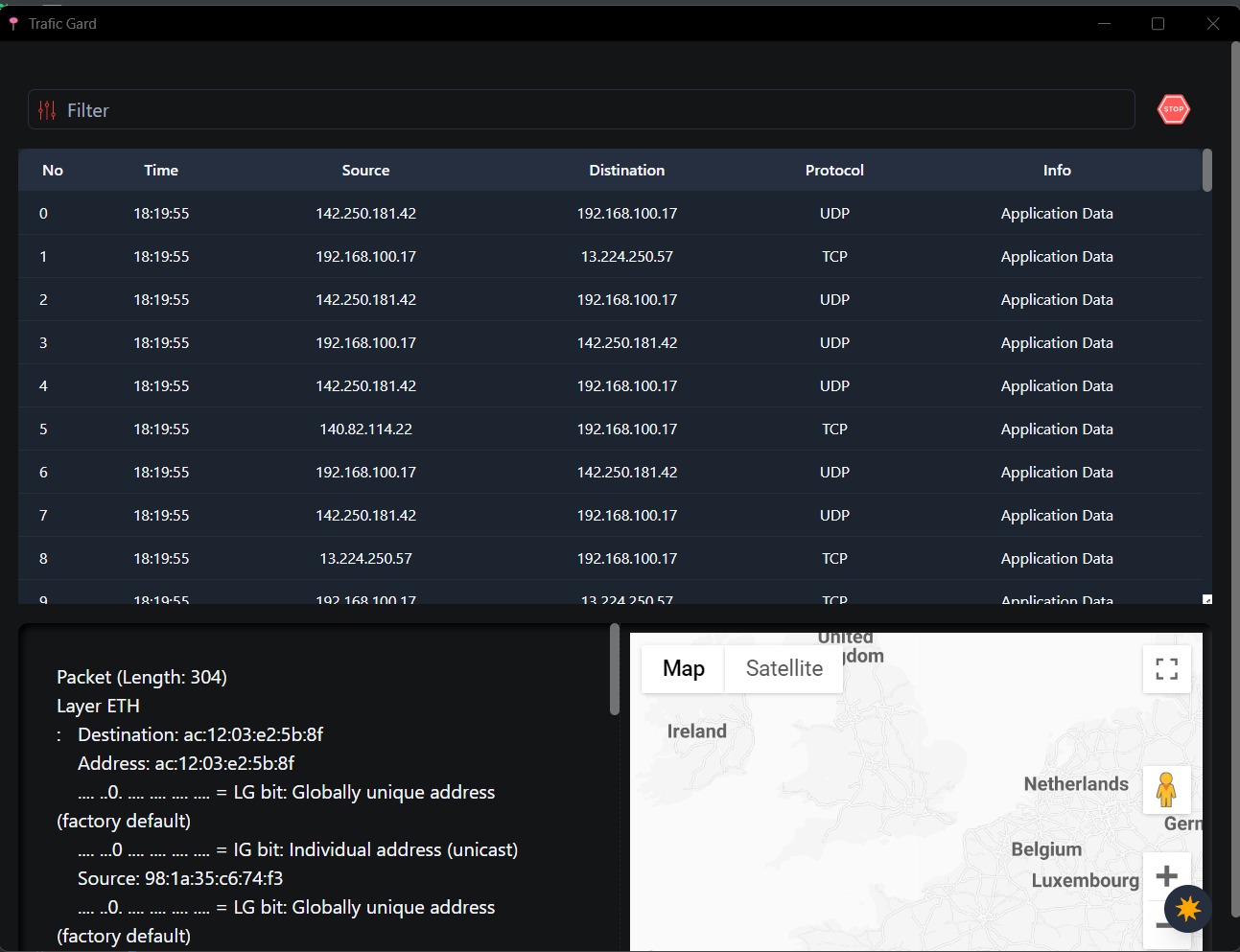


Figure 6



*Figure 7*

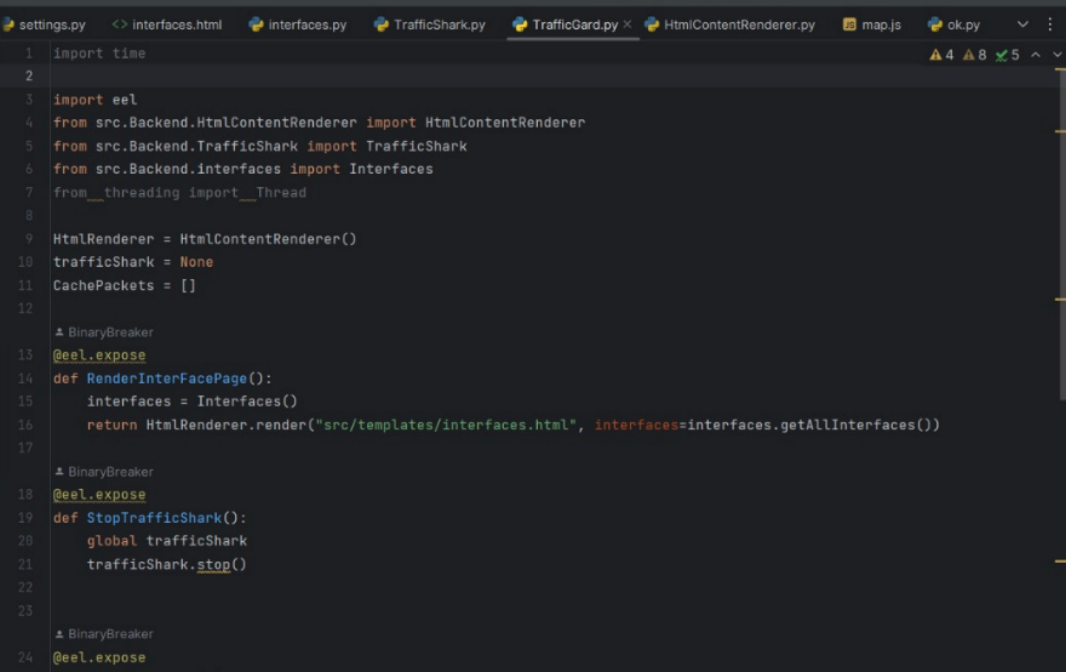
A screenshot of a computer

Description automatically generated

*Figure 8*

A screen shot of a computer

Description automatically generated



*Figure 9*

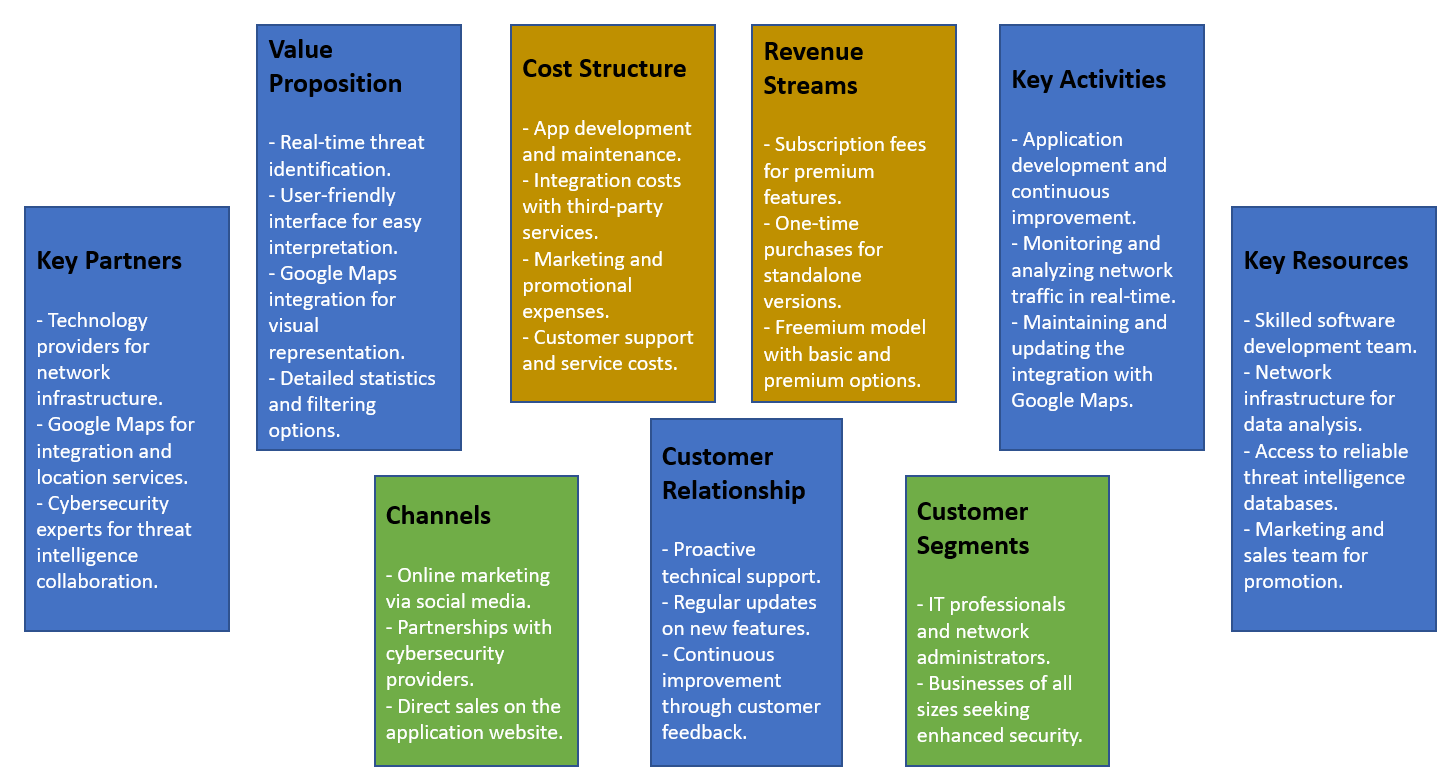
A screenshot of a computer program

Description automatically generated

*Figure 10*

# Business Model

Launching a desktop application focused on monitoring network traffic, tracing locations of active connections, and addressing VPN usage can be an exciting venture. Here's a comprehensive business plan that you can consider for the launch of your application:



# Project Gantt chart

Chart, histogram

Description automatically generated

Figure 11 - Gantt Chart

# References

<https://thepythoncode.com/article/make-a-network-usage-monitor-in-python>

<https://www.geeksforgeeks.org/python-script-to-monitor-network-connection-and-saving-into-log-file/>

<https://www.pythoncentral.io/automate-location-tracking-processes-with-python/>