# ***FlowSensei***

## ***Computer Communication Based Software Development Workshop***

Abstract Project Idea:

- Network Traffic Prioritization Tool

Submitters:

- Asaf Koenigsberg 318654118
- Ofek Markus 318418423

Mentor:

- Dr. Hadar Binsky

Github: FlowSensei

## *Project Description*

Our project aims to develop a versatile tool for managing network traffic priorities during periods of high demand. By dynamically adjusting task priorities based on real-time conditions and predefined policies, the tool ensures the smooth operation of critical network services and activities. Additionally, it informs about hacking attempts and changes the network security level in response to emerging threats or vulnerabilities.

## *The Problem*

During periods of high network load, non-critical activities can overwhelm crucial services, leading to performance degradation, disruptions and reduced productivity. Furthermore, crucial services are exposed to cybersecurity threats.
Our project seeks to address this by focusing on traffic prioritization and protection during high-demand situations, thereby ensuring the efficient management of network resources and the uninterrupted operation of critical services.

## *Approach*

Our approach involves a software defined router, that can be versatility defined via an API, and a GUI for easily configuring it. Using real-time monitoring and dynamic prioritization algorithms, the router optimizes network performance during high demand periods.

Key components include:

1) Real-Time Monitoring: Continuous network monitoring to identify congestion and severe request loads.

2) Adaptive algorithms: Adjust priorities based on real-time conditions and predefined policies.

3) Policy Configuration: Flexible policy settings enable customization based on specific criteria like application type.

## *Other Approaches*

1. Static QoS configurations: configure pre-defined settings and parameters that are manually configured on network devices to prioritize certain types of traffic or allocate bandwidth based on specific criteria
2. Manual traffic shaping: a method of controlling the flow of data across a network by configuring network devices to limit the rate of data transmission according to predetermined policies or rules.

3. Simple priority queuing: a basic form of packet queuing where network packets are assigned to different queues based solely on their assigned priority levels.

   These approaches often lack flexibility, scalability, or adaptability to change network conditions. Our proposed tool distinguishes itself by providing dynamic and adaptive prioritization based on real-time network traffic analysis and user-defined policies.

## *Expected Users*

1. Everyday users: Our tool provides them with a simple and intuitive interface to prioritize network activities based on their preferences, ensuring a seamless and enjoyable online experience.
2. Network Admins in commercial small tech companies: Our tool offers an intuitive interface to prioritize critical tasks, enhancing network performance and reliability for seamless business operations.
3. IT professionals oversee organizational IT infrastructure: Our tool equips them with an intuitive platform to tailor traffic prioritization, ensuring business needs are met and network performance is optimized.
4. System engineers: Our tool provides them with a user-friendly interface to fine-tune traffic management, optimizing performance and resource allocation for robust system operation.

## *Main Features*

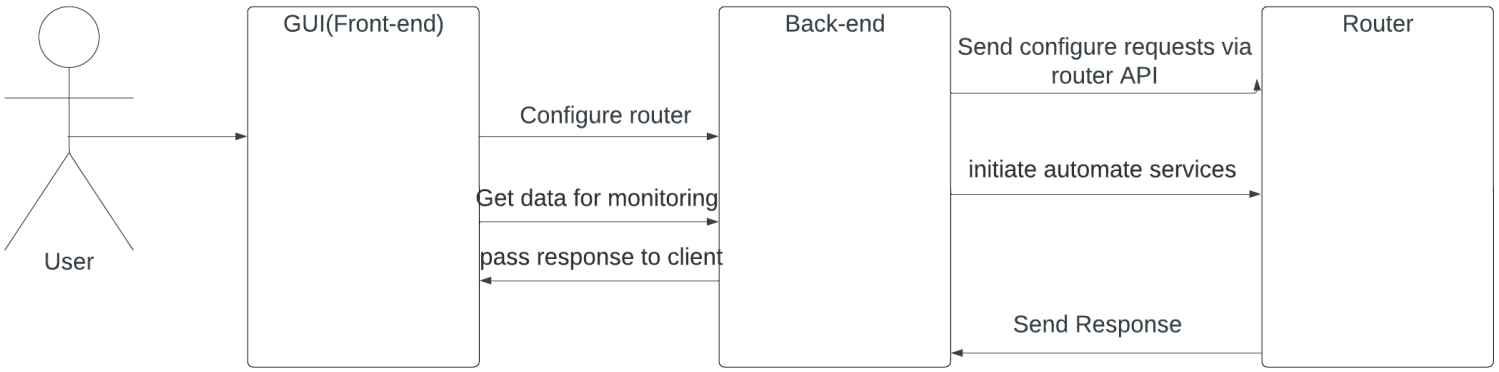
1. Real-Time Monitoring: The tool continuously monitors network conditions to identify periods of increased demand.
2. Dynamic Prioritization: Adaptive algorithms adjust task priorities based on real-time conditions and advanced predefined policies.
3. Policy Configuration: Users can define and modify policies to tailor traffic prioritization according to their specific requirements.
4. Critical Service Protection: Prioritizes network traffic for critical services and activities, ensuring their uninterrupted operation during high-demand situations.
5. Changing Network Security Level: Configure different security levels based on network requirements, each with distinct firewall rules and access control policies.
6. Notify About Hacking Attempts: Monitor network traffic for suspicious activity, while configuring logging to record security events like firewall hits or login failures.

# *User Flows*

1. Real-Time Monitoring:
    a. Scenario: A network administrator wants to ensure optimal performance of critical applications during peak usage hours.
    b. Use Case: The administrator utilizes the real-time monitoring feature to identify periods of increased demand and adjust network resources accordingly to maintain service quality.
    c. Scenario: A regular user experiences slow internet speeds while streaming videos at home.
    d. Use Case: The user contacts their internet service provider (ISP) regarding the slow speeds. The ISP's network team uses real-time monitoring tools to identify congestion points and adjust bandwidth allocations to improve the user's streaming experience.
2. Dynamic Prioritization:
    a. Scenario: An organization has multiple departments with varying network usage priorities.
    b. Use Case: The IT manager configures adaptive algorithms and predefined policies within the tool to dynamically prioritize network tasks based on real-time conditions, ensuring critical departmental operations receive sufficient bandwidth during high-demand periods.

3. Policy Configuration:
    a. Scenario: A company needs to enforce specific traffic prioritization rules tailored to its business requirements.
    b. Use Case: Network administrators use the tool's policy configuration capabilities to define and modify policies according to the organization's needs, ensuring that essential services receive appropriate prioritization under different network conditions.
4. Critical Service Protection:
    a. Scenario: In a corporate environment, there are critical services such as email servers, database servers, and internal communication platforms that are vital for daily operations. The IT team must ensure that these services remain operational and responsive, even during periods of high network demand or unexpected events.
    b. Use Case: IT professionals utilize the tool's Critical Service Protection feature to ensure those vital daily operations
5. Changing Network Security Level:
    a. Scenario: System engineers are responsible for managing network security in a financial institution.
    b. Use Case: In response to a heightened cybersecurity threat, the system engineers use the network management tool to elevate the network security level, implementing stricter firewall rules and access control policies to protect sensitive financial data from potential breaches.

6. <u>Notify About Hacking Attempts:</u>
    a. Scenario: An organization wants to promptly respond to potential security breaches.
    b. Use Case: When a hacking attempt is detected, the tool sends alerts to administrators via notification services, enabling rapid response and mitigation actions to protect the network from further compromise.



## *External Dependencies*

1. Software defined router
2. Router API
3. Frontend framework
4. Backend framework
5. Monitoring & Logging system

## *Development Workflow[28 weeks]*

1. Scouting potential software defined routers- 3 weeks
2. Designing the backend architecture- 4 weeks
3. Developing the backend- 5 weeks
4. Testing the backend- 4 weeks
5. Designing the frontend architecture- 4 weeks
6. Developing the frontend- 4 weeks
7. Testing the frontend- 4 weeks