โปรแกรม MIB Browser ใช้สำหรับเรียกดูค่าของอุปกรณ์ที่ต้องการตรวจสอบโอไอดี ในการดึงข้อมูลจะต้องใส่หมายเลข IP Address ของเอเจนต์ ใส่ค่าคอมมิวนิตี้สตริงให้ตรงกับที่เอเจนต์กำหนดไว้และเครื่องเอเจนต์จะต้องเปิด Service SNMP โปรแกรม MIN Browser ถึงจะสามารถเข้าไปดึงข้อมูลจากเครื่องเอเจนต์ได้

ผลการเรียกค่าโอไอดีที่ได้จากการทดลอง

1) ค่าปริมาณการใช้งานซีพียู (CPU Usage)

Name : cpmCPUTotal5min

OID : .1.3.6.1.4.1.9.9.109.1.1.1.1.5.1

2) ค่าปริมาณการใช้งานเมมโมรี่ (Memory Usage)

Name : ciscoMemoryPoolUsed

OID : .1.3.6.1.4.1.9.9.48.1.1.1.5.1

3) ค่าข้อมูลรายละเอียดของอุปกรณ์

Name : sysDescr

OID : .1.3.6.1.2.1.1.1.0

4) ค่าของการเปิดใช้งานของอุปกรณ์

Name : sysUpTime

OID : .1.3.6.1.2.1.1.3.0

5) ค่าของการเปิดใช้งานของอุปกรณ์

Name : sysUpTime

OID : .1.3.6.1.2.1.1.3.0

6) ค่าการใช้งาน Traffic ขาเข้า

Name : ifInOctets

OID : .1.3.6.1.2.1.2.2.1.10.1

7) ค่าการใช้งาน Traffic ขาเข้า

Name : ifOutOctets

OID : .1.3.6.1.2.1.2.2.1.16.1

8) ค่าอุณหภูมิของอุปกรณ์

Name : ciscoEnvMonTemperatureStatusValue

OID : .1.3.6.1.4.1.9.9.13.1.3.1.3

**ข้อมูลที่ยังไม่ผ่านการแปลงข้อมูล**

**ตารางที่ 4‑1** ตัวอย่างของข้อมูล System

|  |  |  |
| --- | --- | --- |
| Name/OID | Value | OID |
| sysDescr | Cisco IOS Software, Catalyst 4500 L3 Switch Software (cat4500e-LANBASEK9-M), Version 15.0(2)SG1, RELEASE SOFTWARE (fc4) Technical Support: http://www.cisco.com/techsupport Copyright (c) 1986-2011 by Cisco Systems, Inc. Compiled Thu 25-Aug-11 09:27 by p | .1.3.6.1.2.1.1.1.0 |
| sysUpTime | 1434 hours 44 minutes 38 seconds (516507837) | .1.3.6.1.2.1.1.3.0 |
| sysName | SW4503 | .1.3.6.1.2.1.1.5.0 |
| cpmCPUTotal5min | 5 | .1.3.6.1.4.1.9.9.109.1.1.1.1.5.1 |
| ciscoMemoryPoolUsed | 194108612 | .1.3.6.1.4.1.9.9.48.1.1.1.5.1 |
| ciscoEnvMonTemperatureStatusValue | 26 | 1.3.6.1.4.1.9.9.13.1.3.1.3 |

**ตารางที่ 4‑2** ตัวอย่างของข้อมูล Traffic Inbound

|  |  |  |
| --- | --- | --- |
| Name/OID | Value | OID |
| ifInOctets.1 | 0 | .1.3.6.1.2.1.2.2.1.10.1 |
| ifInOctets.51 | 116575506 | .1.3.6.1.2.1.2.2.1.10.1 |
| ifInOctets.52 | 2789121378 | .1.3.6.1.2.1.2.2.1.10.1 |
| ifInOctets.53 | 87404303 | .1.3.6.1.2.1.2.2.1.10.1 |
| ifInOctets.54 | 320120017 | .1.3.6.1.2.1.2.2.1.10.1 |
| ifInOctets.55 | 211044208 | .1.3.6.1.2.1.2.2.1.10.1 |
| ifInOctets.56 | 0 | .1.3.6.1.2.1.2.2.1.10.1 |
| ifInOctets.57 | 29195075 | .1.3.6.1.2.1.2.2.1.10.1 |
| ifInOctets.58 | 637365144 | .1.3.6.1.2.1.2.2.1.10.1 |
| ifInOctets.59 | 0 | .1.3.6.1.2.1.2.2.1.10.1 |
| ifInOctets.88 | 343565568 | .1.3.6.1.2.1.2.2.1.10.1 |
| ifInOctets.99 | 0 | .1.3.6.1.2.1.2.2.1.10.1 |
| ifInOctets.100 | 0 | .1.3.6.1.2.1.2.2.1.10.1 |
| ifInOctets.10101 | 3302340847 | .1.3.6.1.2.1.2.2.1.10.1 |
| ifInOctets.10102 | 0 | .1.3.6.1.2.1.2.2.1.10.1 |
| ifInOctets.10103 | 382648916 | .1.3.6.1.2.1.2.2.1.10.1 |
| ifInOctets.10104 | 0 | .1.3.6.1.2.1.2.2.1.10.1 |
| ifInOctets.10105 | 4084194297 | .1.3.6.1.2.1.2.2.1.10.1 |
| ifInOctets.10106 | 0 | .1.3.6.1.2.1.2.2.1.10.1 |
| ifInOctets.10107 | 2657447915 | .1.3.6.1.2.1.2.2.1.10.1 |

ตัวอย่างของข้อมูล Traffic Inbound ที่ได้จากการใช้โปรแกรม Mib browser เพื่อทดสอบการดึงค่า Traffic Inbound

**ตารางที่ 4‑3** ตัวอย่างของข้อมูล Traffic Outbound

|  |  |  |
| --- | --- | --- |
| Name/OID | Value | OID |
| ifOutOctets.1 | 0 | .1.3.6.1.2.1.2.2.1.16.1 |
| ifOutOctets.51 | 116656529 | .1.3.6.1.2.1.2.2.1.16.1 |
| ifOutOctets.52 | 1496371872 | .1.3.6.1.2.1.2.2.1.16.1 |
| ifOutOctets.53 | 120398715 | .1.3.6.1.2.1.2.2.1.16.1 |
| ifOutOctets.54 | 362527249 | .1.3.6.1.2.1.2.2.1.16.1 |
| ifOutOctets.55 | 253503850 | .1.3.6.1.2.1.2.2.1.16.1 |
| ifOutOctets.56 | 678 | .1.3.6.1.2.1.2.2.1.16.1 |
| ifOutOctets.57 | 133813753 | .1.3.6.1.2.1.2.2.1.16.1 |
| ifOutOctets.58 | 431768585 | .1.3.6.1.2.1.2.2.1.16.1 |
| ifOutOctets.59 | 0 | .1.3.6.1.2.1.2.2.1.16.1 |
| ifOutOctets.88 | 149157043 | .1.3.6.1.2.1.2.2.1.16.1 |
| ifOutOctets.99 | 0 | .1.3.6.1.2.1.2.2.1.16.1 |
| ifOutOctets.100 | 678 | .1.3.6.1.2.1.2.2.1.16.1 |
| ifOutOctets.10101 | 1995310172 | .1.3.6.1.2.1.2.2.1.16.1 |
| ifOutOctets.10102 | 0 | .1.3.6.1.2.1.2.2.1.16.1 |
| ifOutOctets.10103 | 2832577732 | .1.3.6.1.2.1.2.2.1.16.1 |
| ifOutOctets.10104 | 0 | .1.3.6.1.2.1.2.2.1.16.1 |
| ifOutOctets.10105 | 3431308899 | .1.3.6.1.2.1.2.2.1.16.1 |
| ifOutOctets.10106 | 0 | .1.3.6.1.2.1.2.2.1.16.1 |
| ifOutOctets.10107 | 794595360 | .1.3.6.1.2.1.2.2.1.16.1 |

ตัวอย่างของข้อมูล Traffic Inbound ที่ได้จากการใช้โปรแกรม Mib browser เพื่อทดสอบการดึงค่า Traffic Outbound

**ตารางที่ 4‑4** ตัวอย่างของข้อมูล Interface

|  |  |  |
| --- | --- | --- |
| Name/OID | Value | OID |
| ifDescr.1 | Vlan1 | .1.3.6.1.2.1.2.2.1.2.1 |
| ifDescr.51 | Vlan51 | .1.3.6.1.2.1.2.2.1.2.1 |
| ifDescr.52 | Vlan52 | .1.3.6.1.2.1.2.2.1.2.1 |
| ifDescr.53 | Vlan53 | .1.3.6.1.2.1.2.2.1.2.1 |
| ifDescr.54 | Vlan54 | .1.3.6.1.2.1.2.2.1.2.1 |
| ifDescr.55 | Vlan55 | .1.3.6.1.2.1.2.2.1.2.1 |
| ifDescr.56 | Vlan56 | .1.3.6.1.2.1.2.2.1.2.1 |
| ifDescr.57 | Vlan57 | .1.3.6.1.2.1.2.2.1.2.1 |
| ifDescr.58 | Vlan58 | .1.3.6.1.2.1.2.2.1.2.1 |
| ifDescr.59 | Vlan59 | .1.3.6.1.2.1.2.2.1.2.1 |
| ifDescr.88 | Vlan88 | .1.3.6.1.2.1.2.2.1.2.1 |
| ifDescr.99 | Vlan99 | .1.3.6.1.2.1.2.2.1.2.1 |
| ifDescr.100 | Vlan100 | .1.3.6.1.2.1.2.2.1.2.1 |
| ifDescr.10101 | GigabitEthernet0/1 | .1.3.6.1.2.1.2.2.1.2.1 |
| ifDescr.10102 | GigabitEthernet0/2 | .1.3.6.1.2.1.2.2.1.2.1 |
| ifDescr.10103 | GigabitEthernet0/3 | .1.3.6.1.2.1.2.2.1.2.1 |
| ifDescr.10104 | GigabitEthernet0/4 | .1.3.6.1.2.1.2.2.1.2.1 |
| ifDescr.10105 | GigabitEthernet0/5 | .1.3.6.1.2.1.2.2.1.2.1 |
| ifDescr.10106 | GigabitEthernet0/6 | .1.3.6.1.2.1.2.2.1.2.1 |
| ifDescr.10107 | GigabitEthernet0/7 | .1.3.6.1.2.1.2.2.1.2.1 |

ตัวอย่างของข้อมูล Traffic Inbound ที่ได้จากการใช้โปรแกรม Mib browser เพื่อทดสอบการดึงค่า Interface ว่าอุปกรณ์นี้มีกี่ port และบอกได้ว่ามี port อะไรบ้าง

**ตารางที่ 4‑5** ตัวอย่างของข้อมูลสถานะ Interface

|  |  |  |
| --- | --- | --- |
| Name/OID | Value | OID |
| ifOperStatus.1 | down (2) | .1.3.6.1.2.1.2.2.1.8.1 |
| ifOperStatus.51 | down (2) | .1.3.6.1.2.1.2.2.1.8.1 |
| ifOperStatus.52 | up (1) | .1.3.6.1.2.1.2.2.1.8.1 |
| ifOperStatus.53 | up (1) | .1.3.6.1.2.1.2.2.1.8.1 |
| ifOperStatus.54 | up (1) | .1.3.6.1.2.1.2.2.1.8.1 |
| ifOperStatus.55 | up (1) | .1.3.6.1.2.1.2.2.1.8.1 |
| ifOperStatus.56 | down (2) | .1.3.6.1.2.1.2.2.1.8.1 |
| ifOperStatus.57 | up (1) | .1.3.6.1.2.1.2.2.1.8.1 |
| ifOperStatus.58 | up (1) | .1.3.6.1.2.1.2.2.1.8.1 |
| ifOperStatus.59 | down (2) | .1.3.6.1.2.1.2.2.1.8.1 |
| ifOperStatus.88 | up (1) | .1.3.6.1.2.1.2.2.1.8.1 |
| ifOperStatus.99 | down (2) | .1.3.6.1.2.1.2.2.1.8.1 |
| ifOperStatus.100 | down (2) | .1.3.6.1.2.1.2.2.1.8.1 |
| ifOperStatus.10101 | down (2) | .1.3.6.1.2.1.2.2.1.8.1 |
| ifOperStatus.10102 | down (2) | .1.3.6.1.2.1.2.2.1.8.1 |
| ifOperStatus.10103 | up (1) | .1.3.6.1.2.1.2.2.1.8.1 |
| ifOperStatus.10104 | down (2) | .1.3.6.1.2.1.2.2.1.8.1 |
| ifOperStatus.10105 | up (1) | .1.3.6.1.2.1.2.2.1.8.1 |
| ifOperStatus.10106 | down (2) | .1.3.6.1.2.1.2.2.1.8.1 |
| ifOperStatus.10107 | up (1) | .1.3.6.1.2.1.2.2.1.8.1 |

ตัวอย่างของข้อมูล Traffic Inbound ที่ได้จากการใช้โปรแกรม Mib browser เพื่อทดสอบการดึงค่า Interface ว่าอุปกรณ์นี้มีกี่ port และบอกได้ว่ามี port อะไรบ้าง

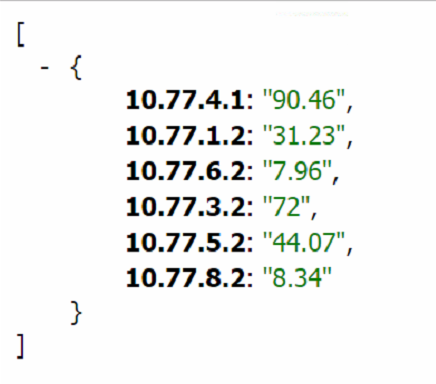
**4.1.2 ขั้นตอนการค้นคว้าข้อมูลเกี่ยวกับการพัฒนาแบบแอปพลิเคชัน**

ขั้นตอนการค้นคว้าข้อมูลเกี่ยวกับการพัฒนาแบบแอปพลิเคชัน ศึกษาค้นคว้าเกี่ยวกับภาษา PHP เพื่อใช้เก็บ Log จากอุปกรณ์  และใช้ PHP เป็นส่วนที่ใช้ในการคำนวณ ประมวลผลเก็บค่า และทำตามคำสั่งต่าง ๆ และเก็บไว้เพื่อนำมาแสดงผลต่อไป  PHP เป็นภาษาที่สามารถศึกษาได้ง่าย และทำงานได้มีประสิทธิภาพ ทำให้เป็นที่นิยมอย่างยิ่งในปัจจุบัน และศึกษาเพิ่มเติมเกี่ยวกับการใช้ ภาษาJava Script เพื่อใช้พัฒนาเว็บในส่วนของหน้าเว็บแอปพลิเคชันที่ใช้แสดงข้อมูล เพื่อให้เว็บไซต์ดูมีการเคลื่อนไหว สามารถตอบสนองผู้ใช้งานได้มากขึ้น และเป็นภาษาที่ใช้ทรัพยากรเครื่องน้อยมาก ใช้งานร่วมกับภาษา HTML  ภาษาหลักที่ใช้ในการเขียนเว็บแอปพลิเคชัน โดยใช้ Tag ในการกำหนดการแสดงผล HTML ย่อมาจากคำว่า Hypertext Markup Language โดย Hypertext หมายถึง ข้อความที่เชื่อมต่อกันผ่านลิงก์ (Hyperlink) Markup language หมายถึงภาษาที่ใช้ Tag ในการกำหนดการแสดงผลสิ่งต่าง ๆ ที่แสดงอยู่บนเว็บเบราว์เซอร์ และศึกษาการใช้ CSS สำหรับตกแต่ง HTML ให้มีหน้าตา สีสัน ตัวอักษร เส้นขอบ พื้นหลัง ระยะห่าง ฯลฯ อย่างที่เราต้องการ ด้วยการกำหนดคุณสมบัติให้กับ Element ต่าง ๆ และใช้ jquery เพื่อให้การเขียน JavaScript นั้นมีความสะดวกและง่ายขึ้น เพราะว่าการนำ JavaScript เอาไปประยุกต์กับงานจำพวกเว็บ (Client-side JavaScript) นั้น เป็นสิ่งที่ยุ่งยาก ไม่ว่าจะเป็นเรื่องความไม่เข้ากันของ **Web Browser**, DOM หรือ API เป็นต้น  jquery จึงรวมเอา Object และ Function ต่าง ๆ ที่จำเป็นมารวบรวมไว้ในรูปแบบของ Library ช่วยให้เขียน Java Script ได้ง่ายและสั้นลง

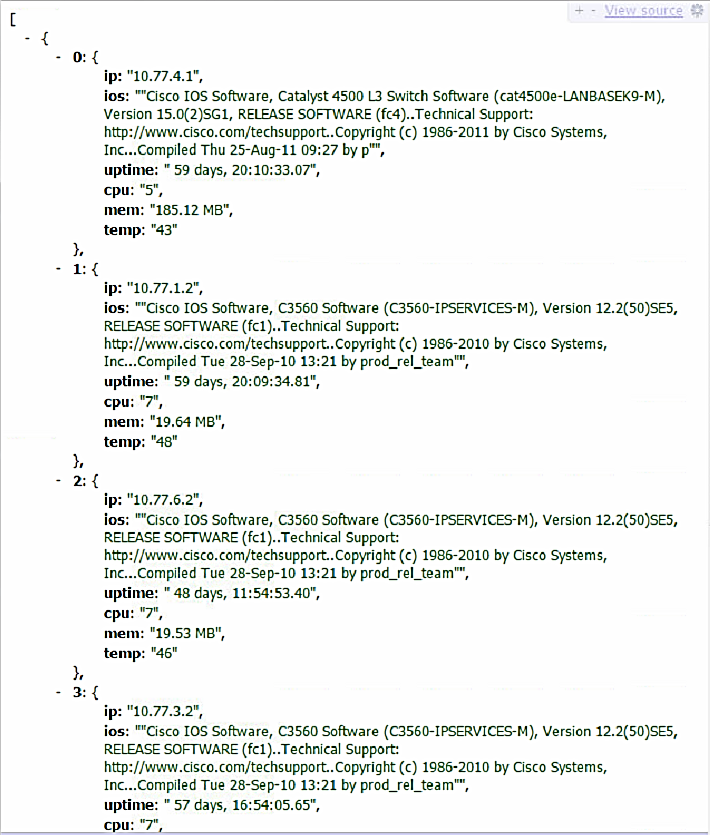
ระบบได้มีการปรับเปลี่ยนโครงสร้างบางอย่างเริ่มจากการเปลี่ยน Server เป็นการใช้ Raspberry Pi เข้ามาแทนเครื่อง server เพื่อลดค่าใช้จ่ายและมีขนาดเล็ก จึงได้มีการศึกษาการใช้งานและการติดตั้ง tool ที่ต้องใช้ มีการปรับเปลี่ยนภาษาที่ใช้พัฒนาระบบให้เหมาะสมกับความสามารถของตัว server จากเดิมที่ใช้ PHP ในการพัฒนาระบบเป็นการใช้ Node Js ในการดึงค่าข้อมูลเพราะมีความสามารถในการทำงานได้ดีกว่า

**4.1.3 ขั้นตอนการวิเคราะห์ข้อมูล**

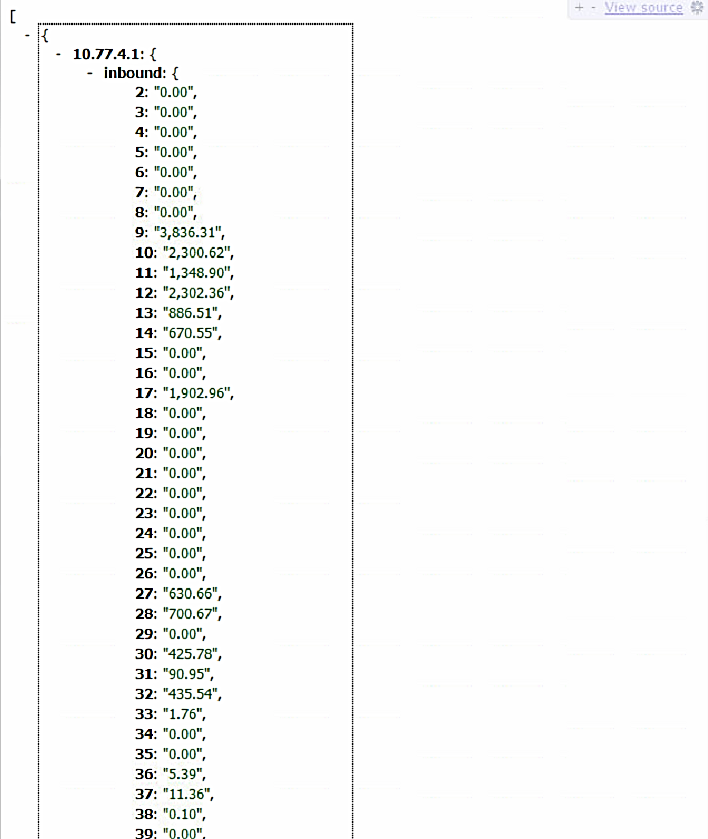
ในขั้นตอนการวิเคราะห์ข้อมูลนั้นได้ศึกษาถึงระบบ Monitoring ในรูปแบบต่าง ๆ ศึกษาถึงข้อดีข้อเสียของระบบ Monitoring แต่ละตัว และปรับเอาข้อดีข้อเสียของระบบ Monitoring ที่มีอยู่แล้วนำมาเป็นแนวทางในการพัฒนาโครงงานพิเศษให้มีประสิทธิภาพสูงสุดและใช้งานได้จริงอย่างสมบูรณ์แบบและมีการทดลองเขียนโปรแกรมในภาษา PHP ในการเก็บข้อมูลต่าง ๆ และทดสอบ API ที่ถูกสร้างขึ้นในรูปแบบ JSON อย่างละเอียด ก่อนนำมาพัฒนาเว็บแอปพลิเคชัน โดยข้อมูลที่ได้จากอุปกรณ์จะถูกวิเคราะห์และคัดเอาข้อมูลที่มีประโยชน์และน่านำมาเสนอหน้าเว็บเบราว์เซอร์

****

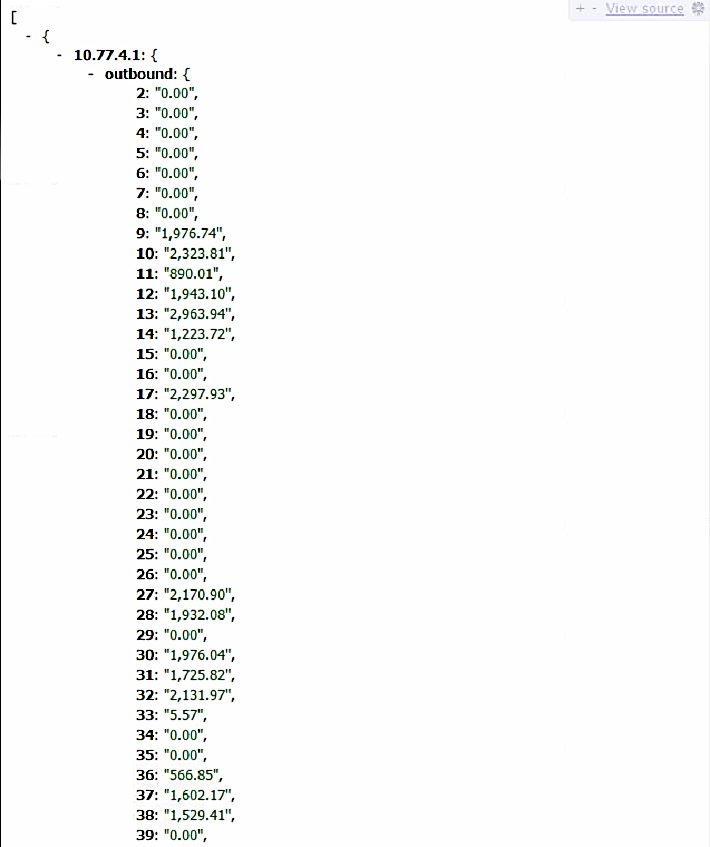
**ภาพที่ 4-1** ข้อมูลการใช้งาน Traffic ของทุกอุปกรณ์ ที่ผ่านการแปลงข้อมูลเป็น API



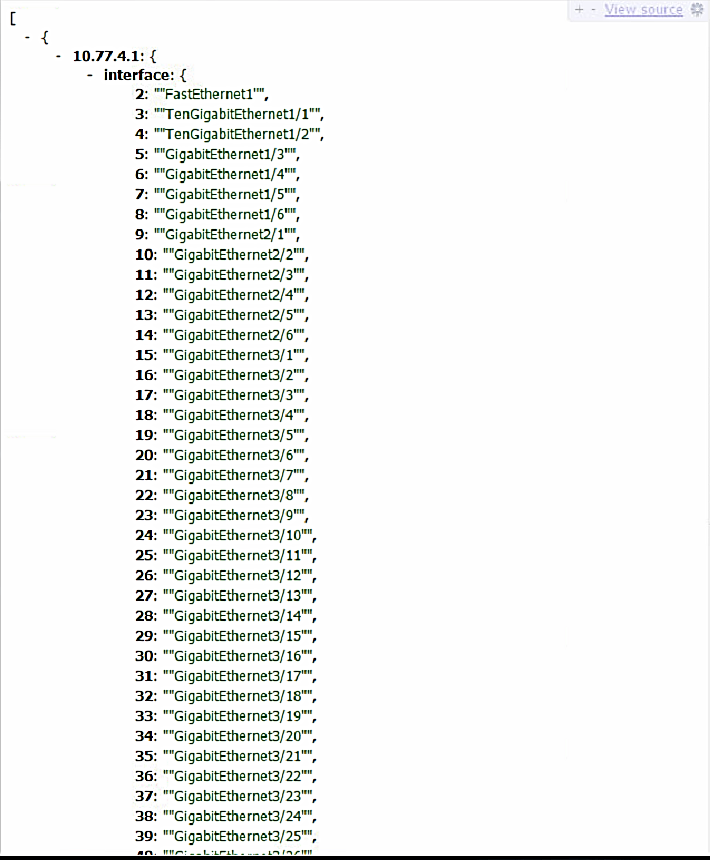
**ภาพที่ 4-2** ตัวอย่างข้อมูล System ที่ผ่านการแปลงข้อมูลเป็น API



**ภาพที่ 4-3** ตัวอย่างข้อมูล Traffic Inbound ที่ผ่านการแปลงข้อมูลเป็น API

****

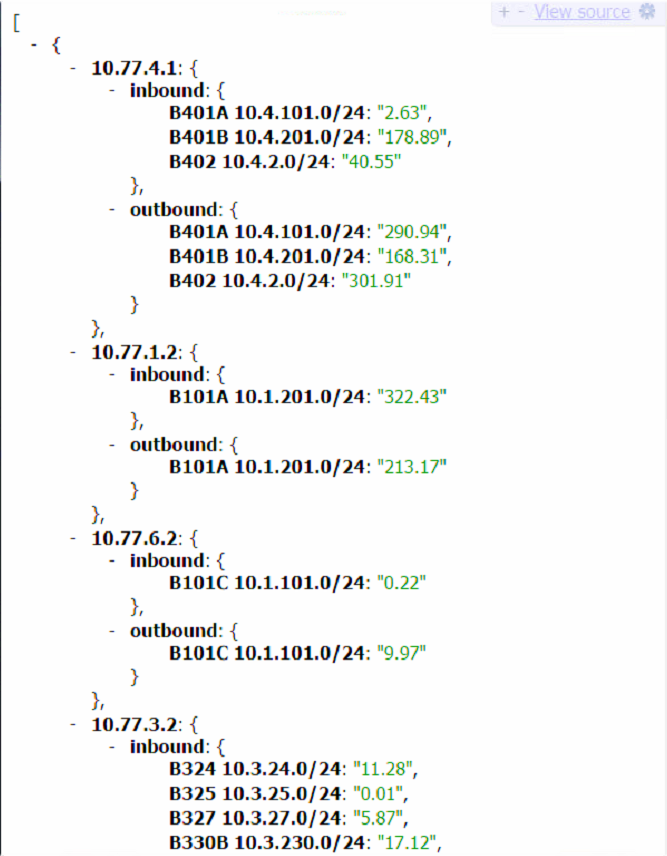
**ภาพที่ 4-4** ตัวอย่างข้อมูล Traffic Outbound ที่ผ่านการแปลงข้อมูลเป็น API

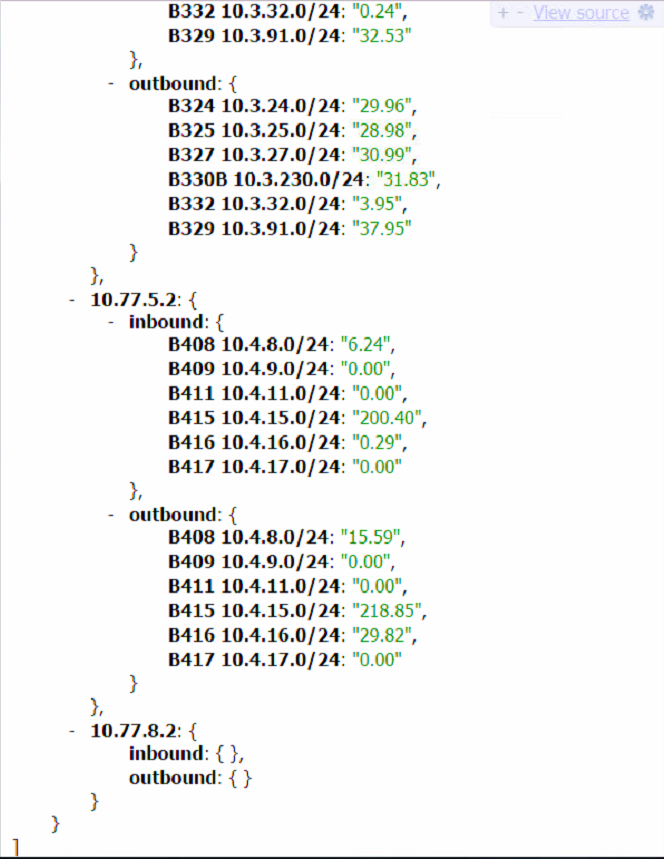


**ภาพที่ 4-5** ตัวอย่างข้อมูล Interface ที่ผ่านการแปลงข้อมูลเป็น API

****

**ภาพที่ 4-6** ตัวอย่างข้อมูลสถานะ Interface ที่ผ่านการแปลงข้อมูลเป็น API

****

****

**ภาพที่ 4-7** ข้อมูลการใช้งาน Traffic ที่แบ่งตาม Network ID ที่ผ่านการแปลงข้อมูลเป็น API

**4.1.4 วิธีการทำงานของระบบ**

ขั้นตอนการออกแบบระบบนั้นเริ่มจากการออกแบบเมนู ฟังก์ชันการใช้งานที่มีประโยชน์สูงสุดต่อผู้ดูแลระบบ และให้มีส่วนติดต่อกับผู้ใช้อย่างเข้าใจง่าย ด้วยการแทนความหมายด้วยรูปภาพและสีที่แตกต่างอย่างชัดเจนเป็นหลัก จากนั้นเป็นการออกแบบการคำนวณค่าของ Traffic และค่าสมรรถนะการทำงานของอุปกรณ์ให้มีความถูกต้อง จากนั้นเป็นการออกแบบโครงสร้างการเขียนโปรแกรมเพื่อพัฒนาเว็บแอปพลิเคชันให้มีการทำงานสอดคล้องกับสิ่งที่ความคาดหวังของโครงงานพิเศษได้

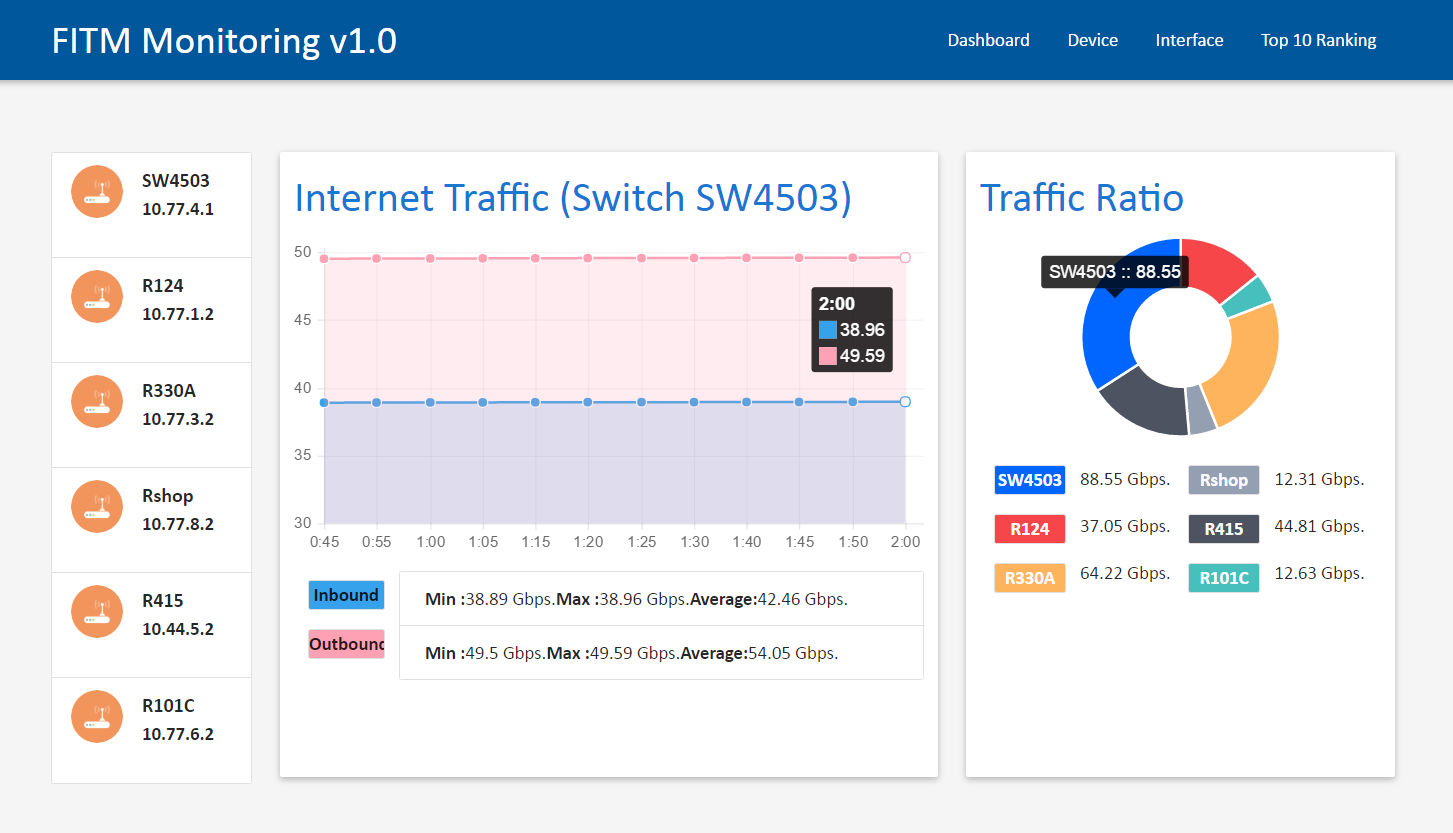
**4.1.5** **รายละเอียดการทำงานของหน้าเว็บแอปพลิเคชัน**

เว็บแอปพลิเคชันสามารถทำงานได้ดังต่อไปนี้

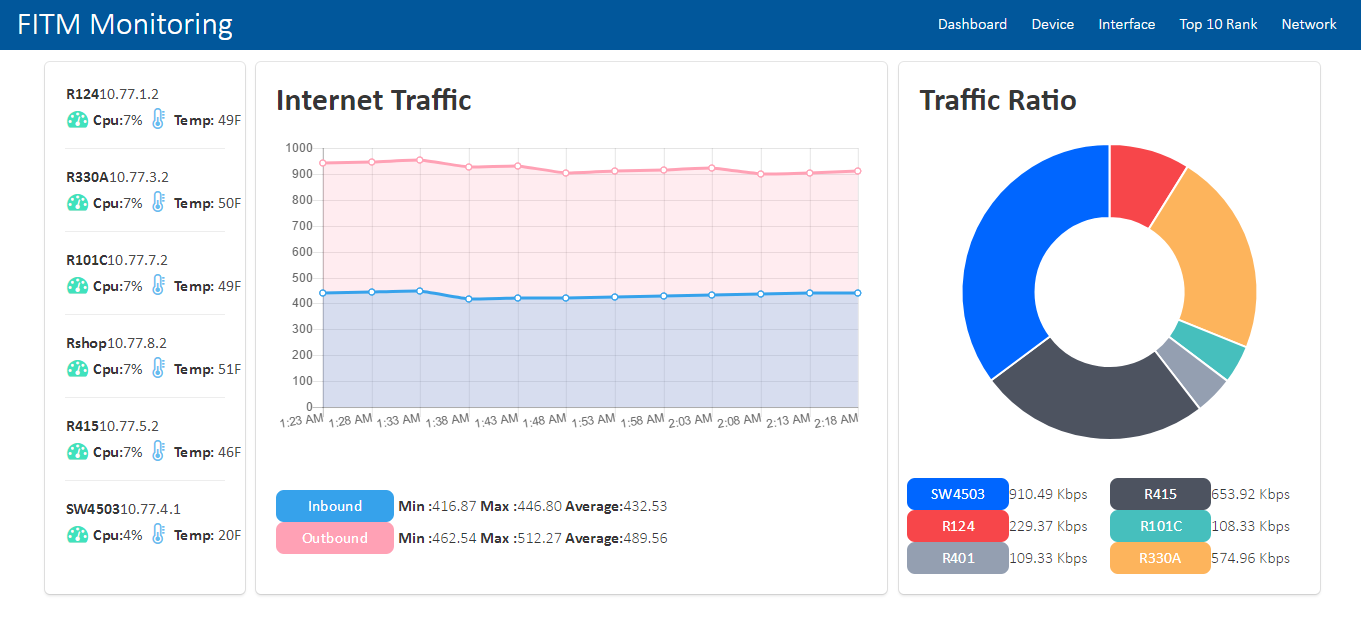
**1. หน้าจอเมนู Dashboard**

เป็นหน้าจอเว็บแอปพลิเคชันแรก เมื่อเปิดเว็บเข้ามาจะพบข้อมูลที่แสดงภาพรวมของสถานะของเครือข่าย โดยจะมีรายละเอียดดังนี้ เมนูด้านบน เป็นเมนูที่จะนำไปสู่หน้าต่างอื่น ๆ เพื่อแสดงข้อมูลโดยละเอียดมากขึ้น โดยจะแบ่งเมนูออกเป็น 3 เมนูหลัก โดยแบ่งเป็นเมนูทางด้านซ้ายจะแสดงรายชื่ออุปกรณ์พร้อม แสดงหมายเลขไอพีของแต่ละอุปกรณ์ ทั้งหมด 6 อุปกรณ์ และแสดงรูปอุปกรณ์ชัดเจน ส่วนกลางเป็นข้อมูล Traffic โดยรวมของเครือข่าย โดยจะแสดงข้อมูลในรูปแบบของกราฟเส้น แบ่งเป็น ข้อมูล Inbound ที่แสดงเป็นเส้นสีฟ้า และ Outbound แสดงเป็นเส้นสีแดงเพื่อให้ดูง่ายยิ่งขึ้น ด้านล่างกราฟจะแสดงข้อมูล Traffic สูงสุดต่ำสุดและ ค่าเฉลี่ยของ Traffic ต่อมาทางด้านขวาเป็นส่วนที่แสดง Traffic Ratio หรือ อัตราส่วนของการใช้ Traffic เพื่อแสดงเน็ตเวิร์กแต่ละเน็ตเวิร์กมีการใช้งานเป็นอย่างไร โดยแสดงผลออกมาในรูปแบบของกราฟวงกลม และมีการใช้สีแทนแต่ละ Network โดยมีการแสดงค่า Traffic โดยรวมของแต่ละอุปกรณ์

**ส่วนแสดงหน้าจอเมนู Dashboard**

****

**ภาพที่ 4-8** หน้าจอ Dashboard ของเว็บ

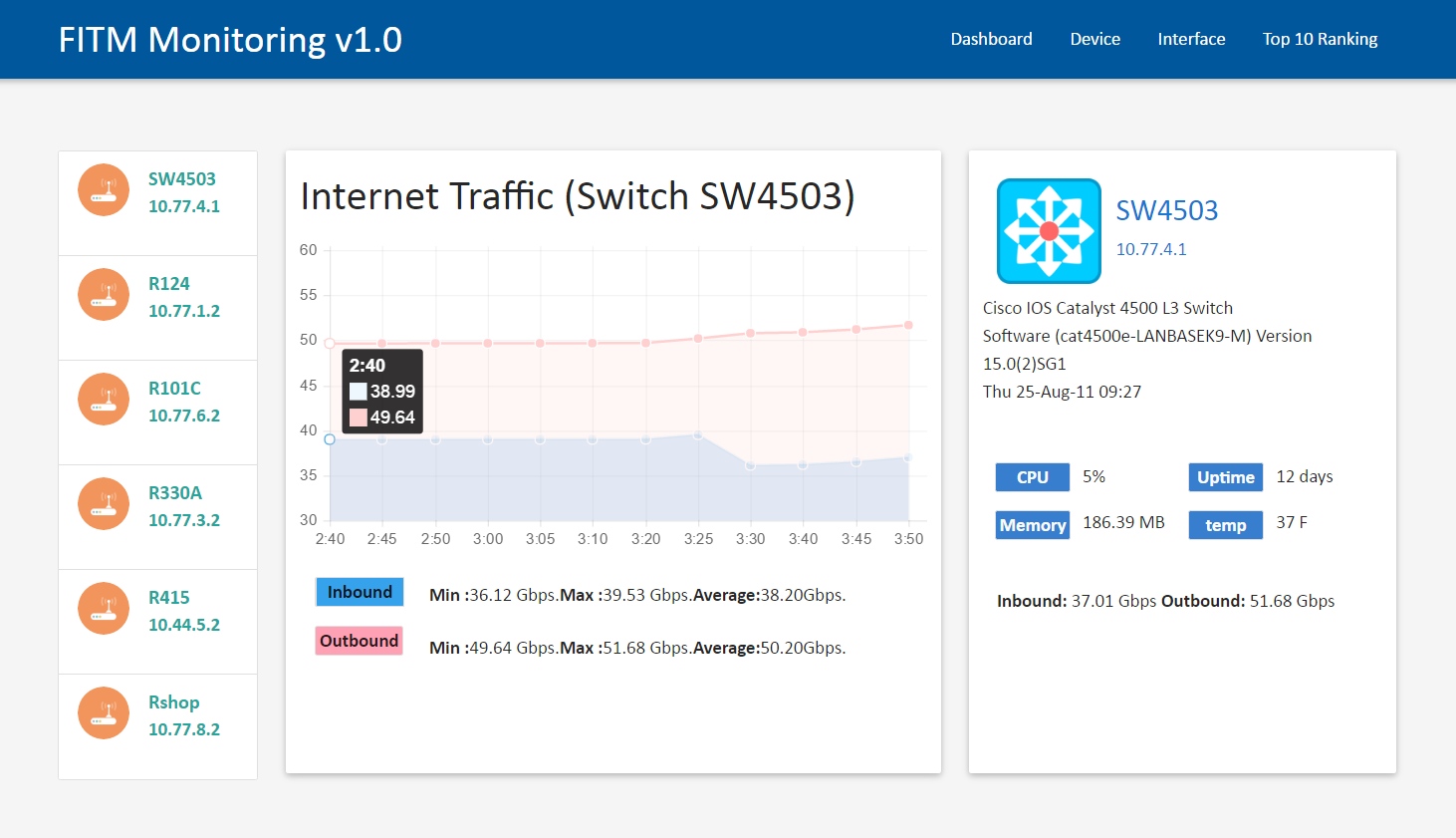


**ภาพที่ 4-9** หน้าจอ Dashboard ของเว็บ แบบใหม่

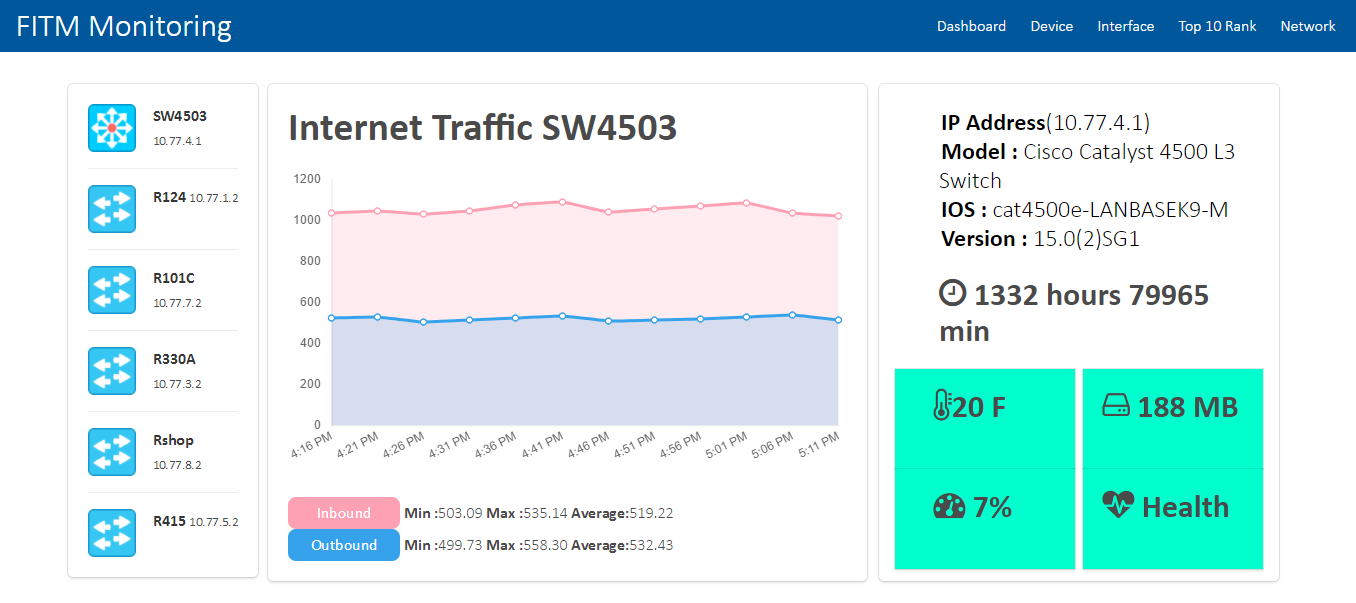
**2. หน้าจอเมนู Device**

หน้า Device แสดงผลจำเพาะของอุปกรณ์เครือข่ายได้ อาทิเช่น ชื่ออุปกรณ์ รุ่นของอุปกรณ์ รายละเอียดของอุปกรณ์ และหมายเลขไอพีของอุปกรณ์ และยังสามารถแสดงข้อมูลทางด้านฮาร์ดแวร์ของอุปกรณ์ได้ เช่น ปริมาณการใช้งานของหน่วยประมวลผลหน่วยความจำ อุณหภูมิ ค่าเวลาตั้งแต่เปิดอุปกรณ์ และสามารถแสดงกราฟและปริมาณ Traffic

**2.1 แสดงหน้าจออุปกรณ์ที่มีการทำงานปกติ**

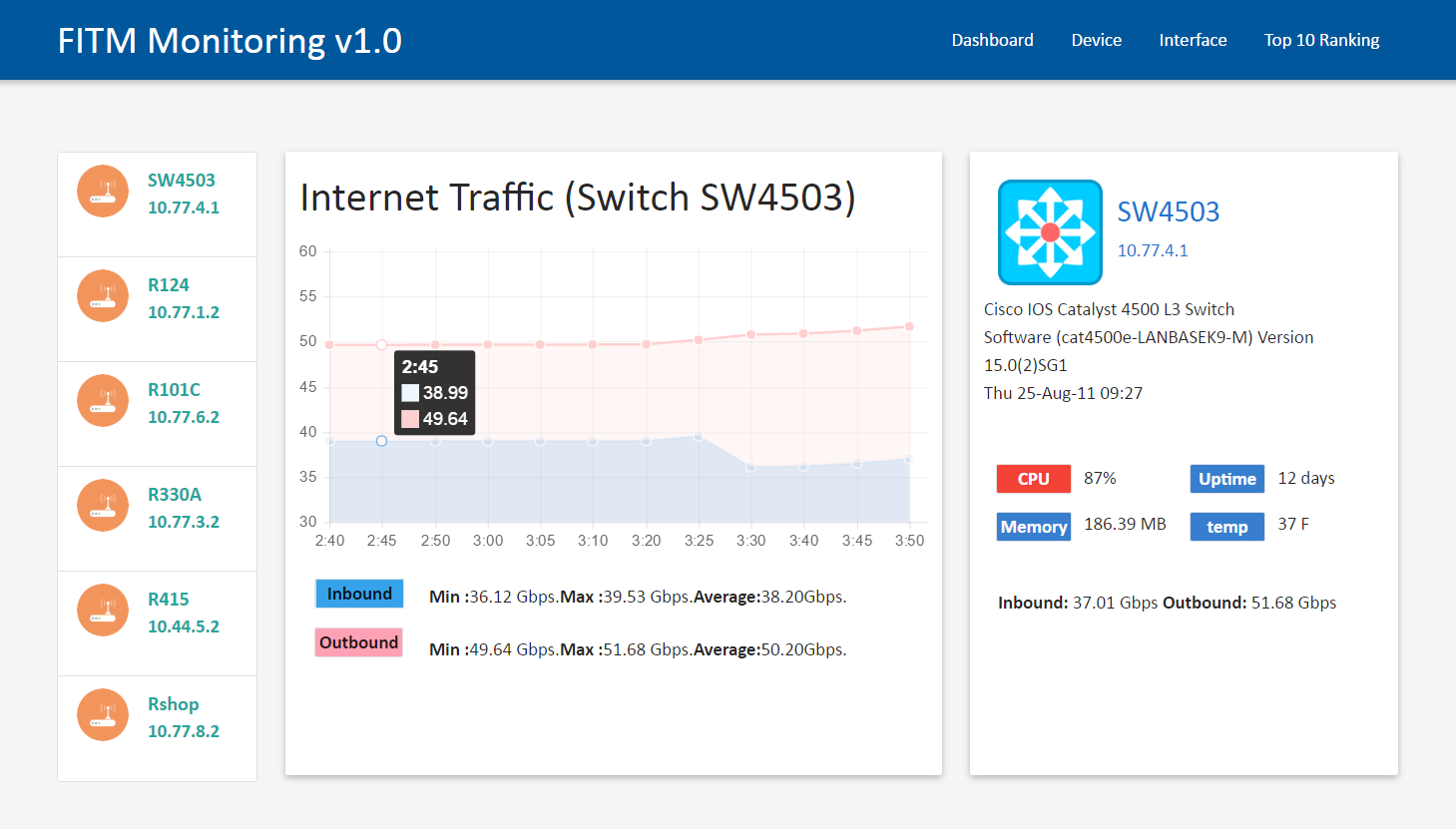
****

**ภาพที่ 4-10** หน้าจอ Device ที่อุปกรณ์ทำงานปกติของเว็บ

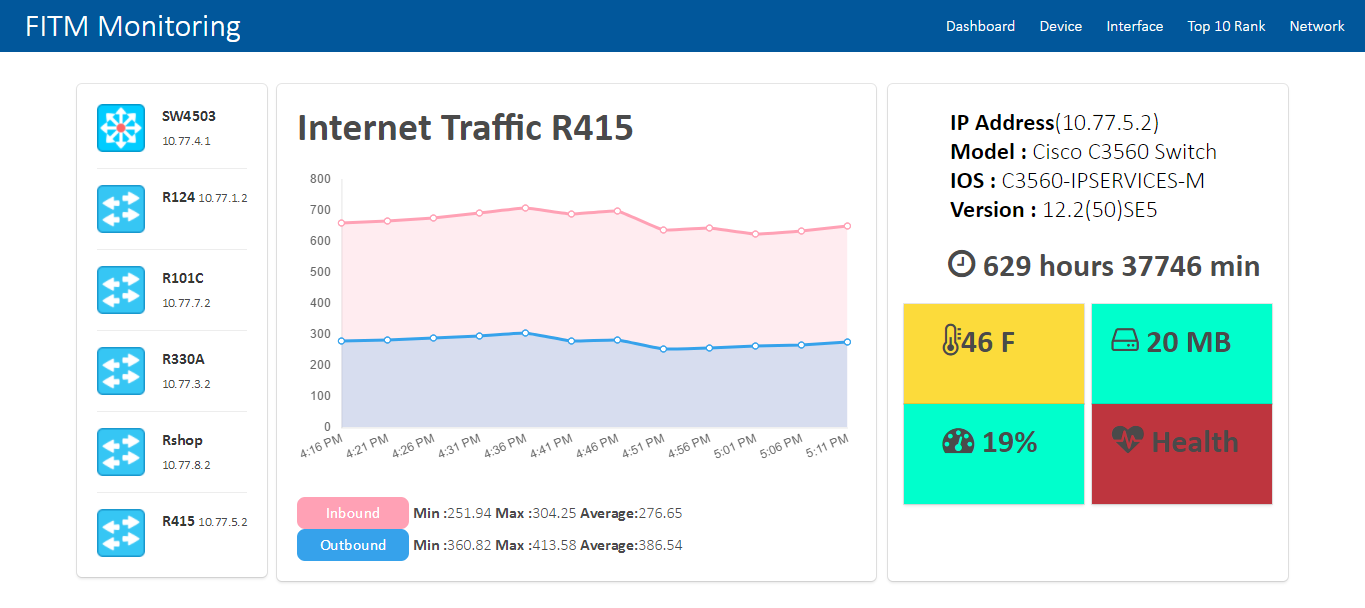


**ภาพที่ 4-11** หน้าจอ Device ที่อุปกรณ์ทำงานปกติของเว็บแบบปรับปรุง

**2.2 แสดงหน้าจออุปกรณ์ที่มีการทำงานผิดปกติ**



**ภาพที่ 4-12** หน้าจอ Device ที่อุปกรณ์ทำงานผิดปกติของเว็บแอปพลิเคชัน

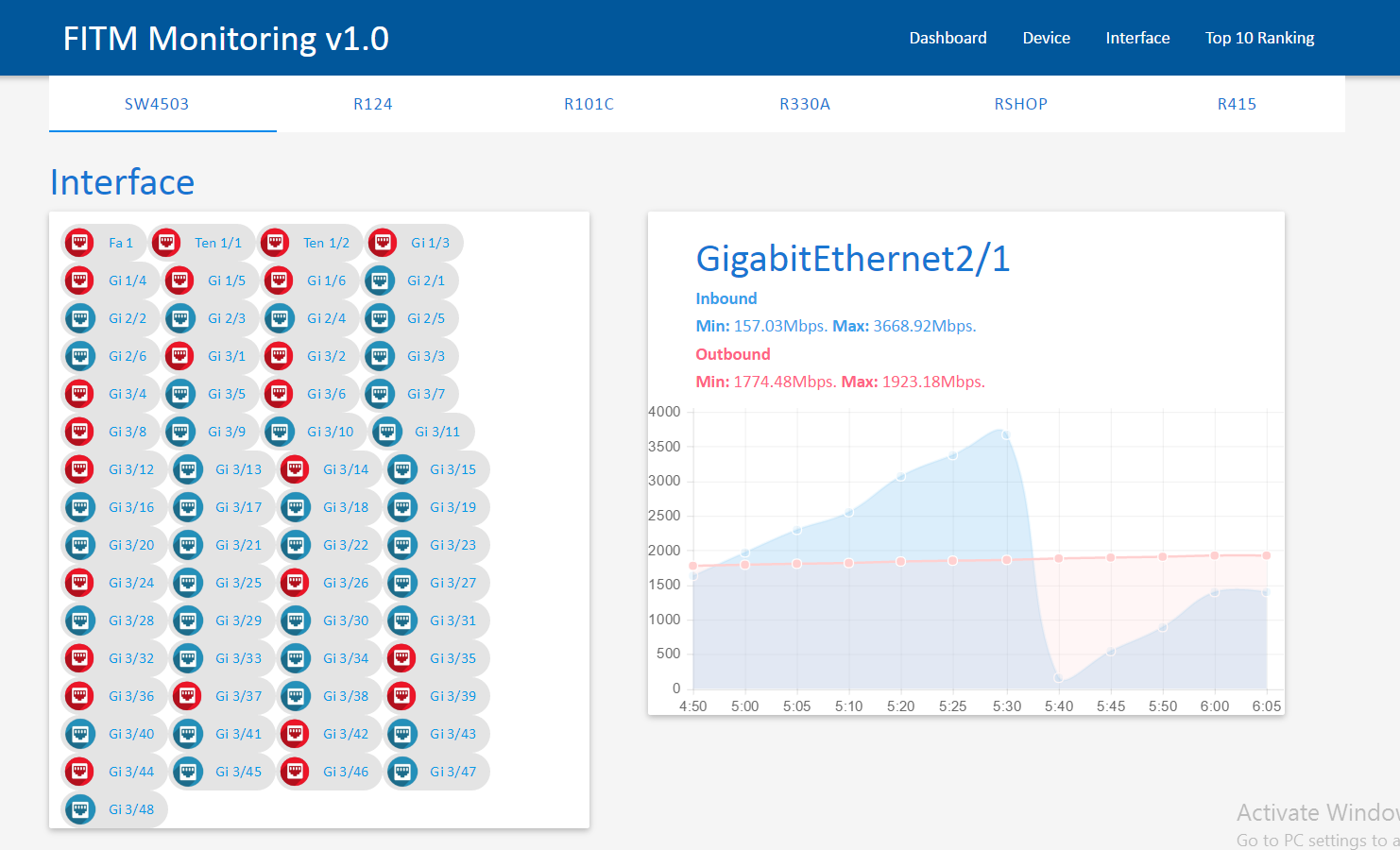
****

**ภาพที่ 4-13** หน้าจอ Device ที่อุปกรณ์ทำงานผิดปกติของเว็บแอปพลิเคชัน แบบใหม่

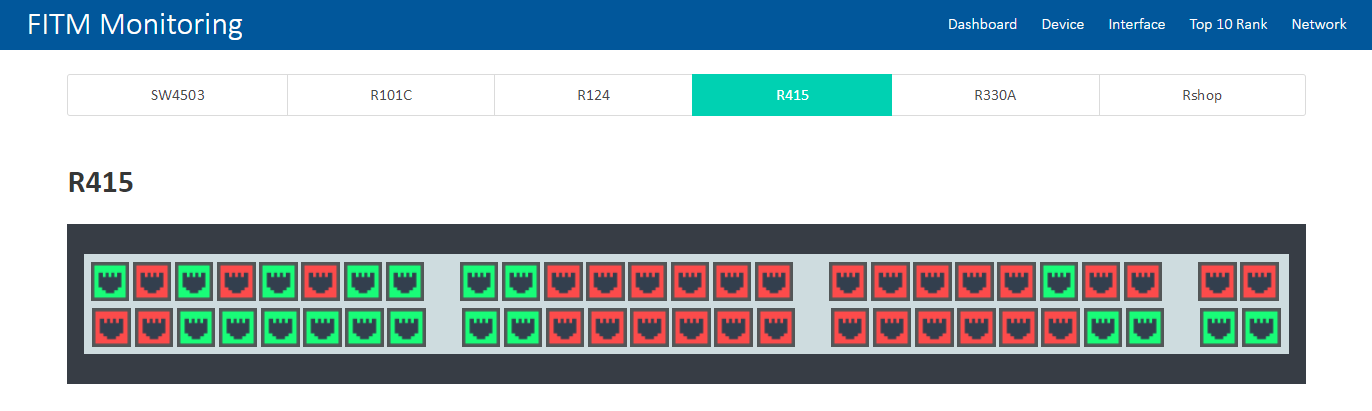
**3. หน้าจอเมนู Interface**

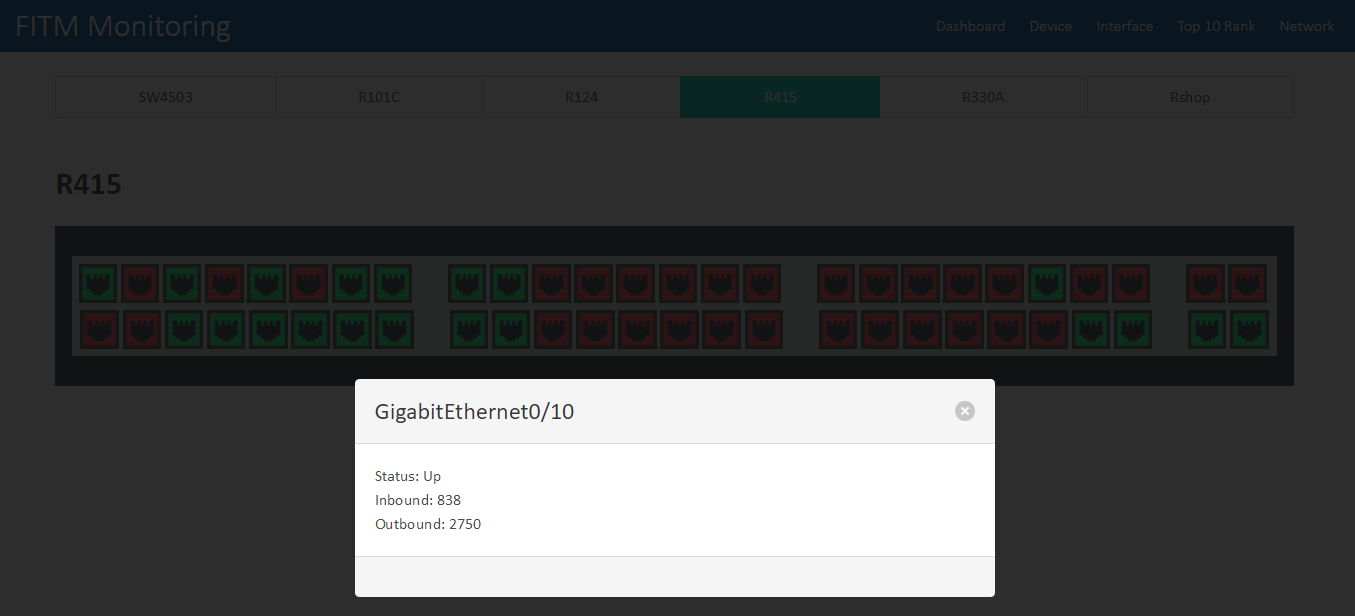
หน้าเมนู Interface สามารถแสดงข้อมูลจำเพาะของอินเตอร์เฟสได้ อาทิเช่น ชื่ออินเตอร์เฟส สถานะอินเตอร์เฟส แสดงข้อมูลที่มีประโยชน์ต่อผู้ดูแลระบบในการดูข้อมูลปริมาณ Traffic ภายใน 1 ชั่วโมงที่ผ่านมาและสามารถดูค่าที่เปลี่ยนแปลงไปได้

**3.1 แสดงหน้าจอ Interface ที่ขา Interface ใช้งานปกติ**

****

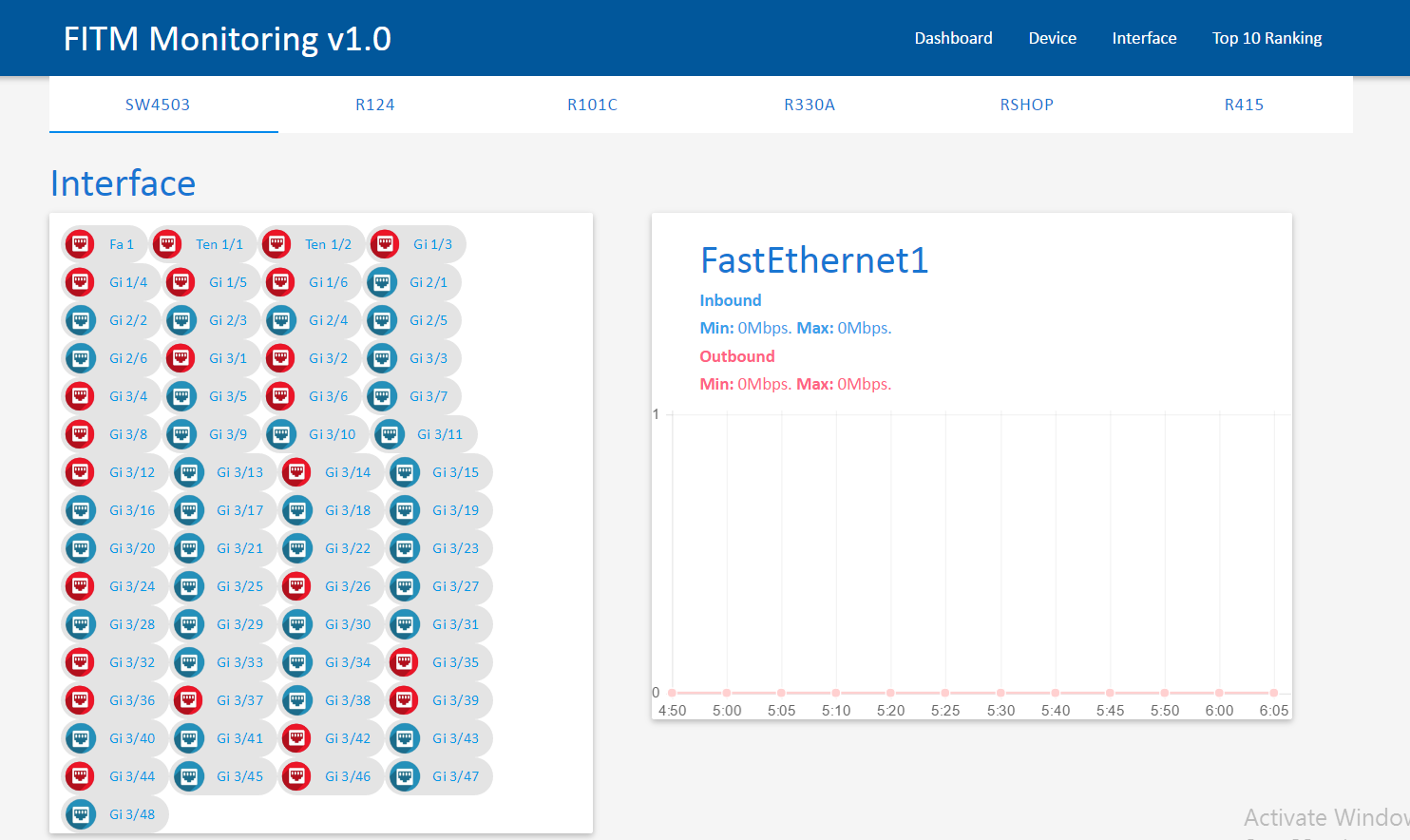
**ภาพที่ 4-14** แสดงหน้าจอ Interface ที่ขา Interface ใช้งานปกติ

****

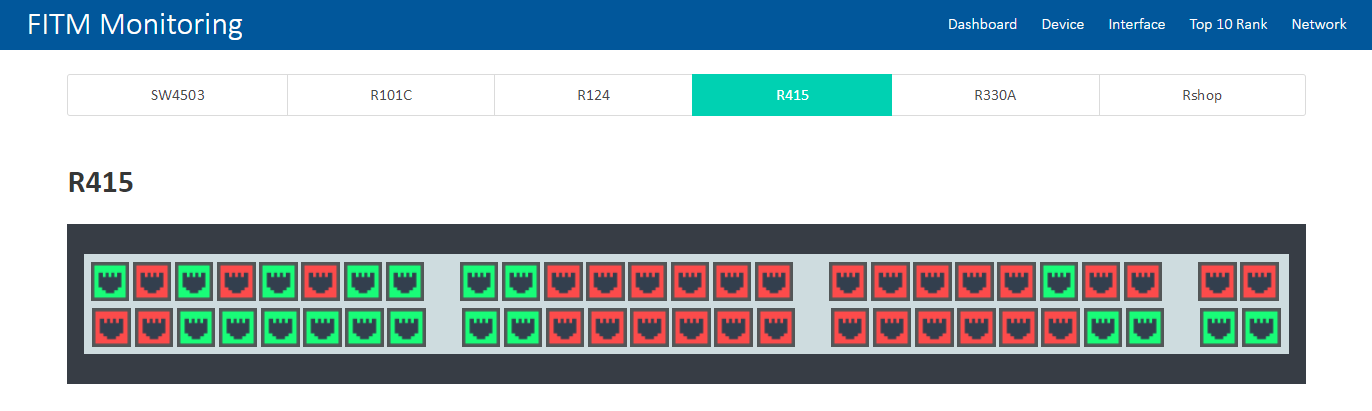
****

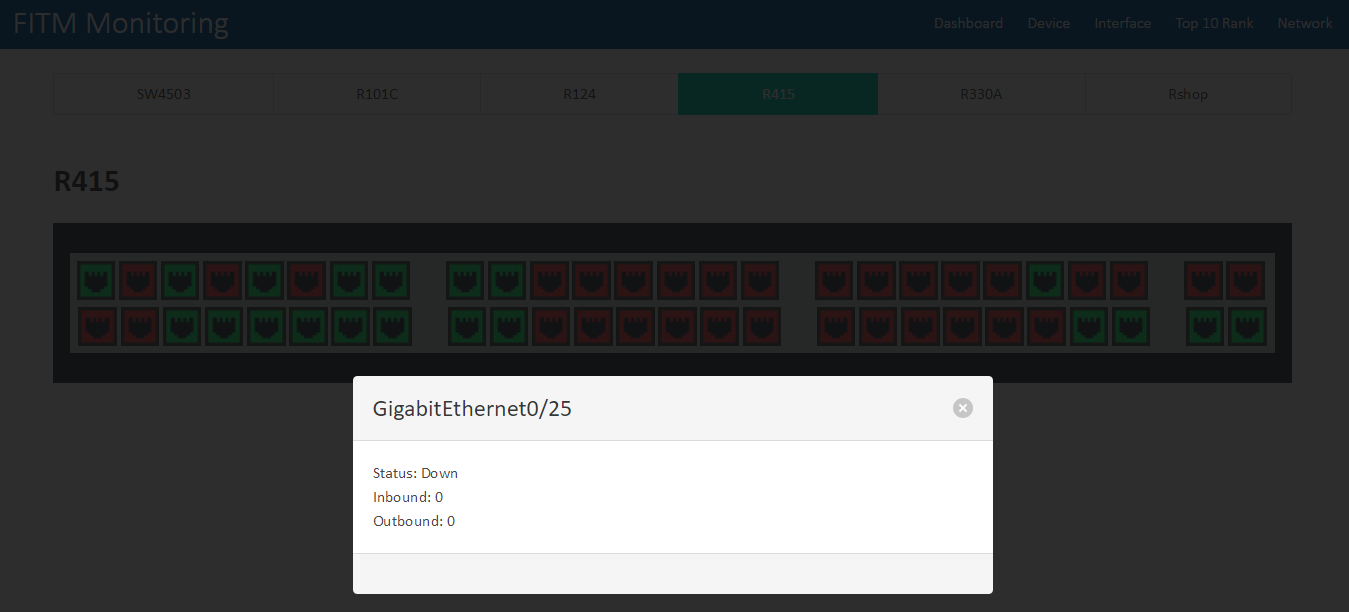
**ภาพที่ 4-15** แสดงหน้าจอ Interface ที่ขา Interface ใช้งานปกติ แบบใหม่

**3.2 แสดงหน้าจอ Interface ที่ขา Interface ไม่มีการใช้งาน**

****

**ภาพที่ 4-16** แสดงหน้าจอ Interface ที่ขา Interface ไม่มีการใช้งาน

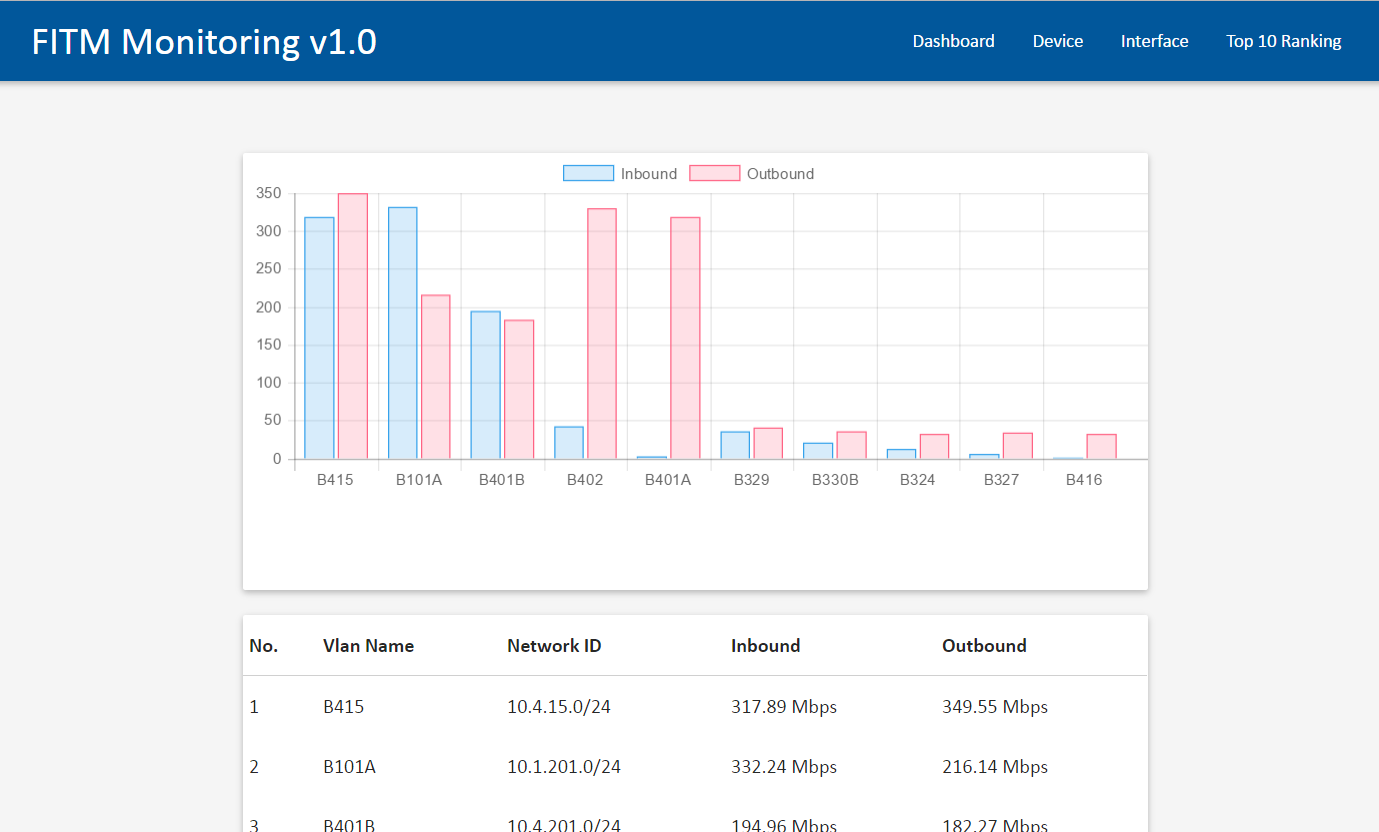
****

****

**ภาพที่ 4-17** แสดงหน้าจอ Interface ที่ขา Interface ไม่มีการใช้งาน แบบใหม่

**4. หน้าจอเมนู Top 10 Ranking**

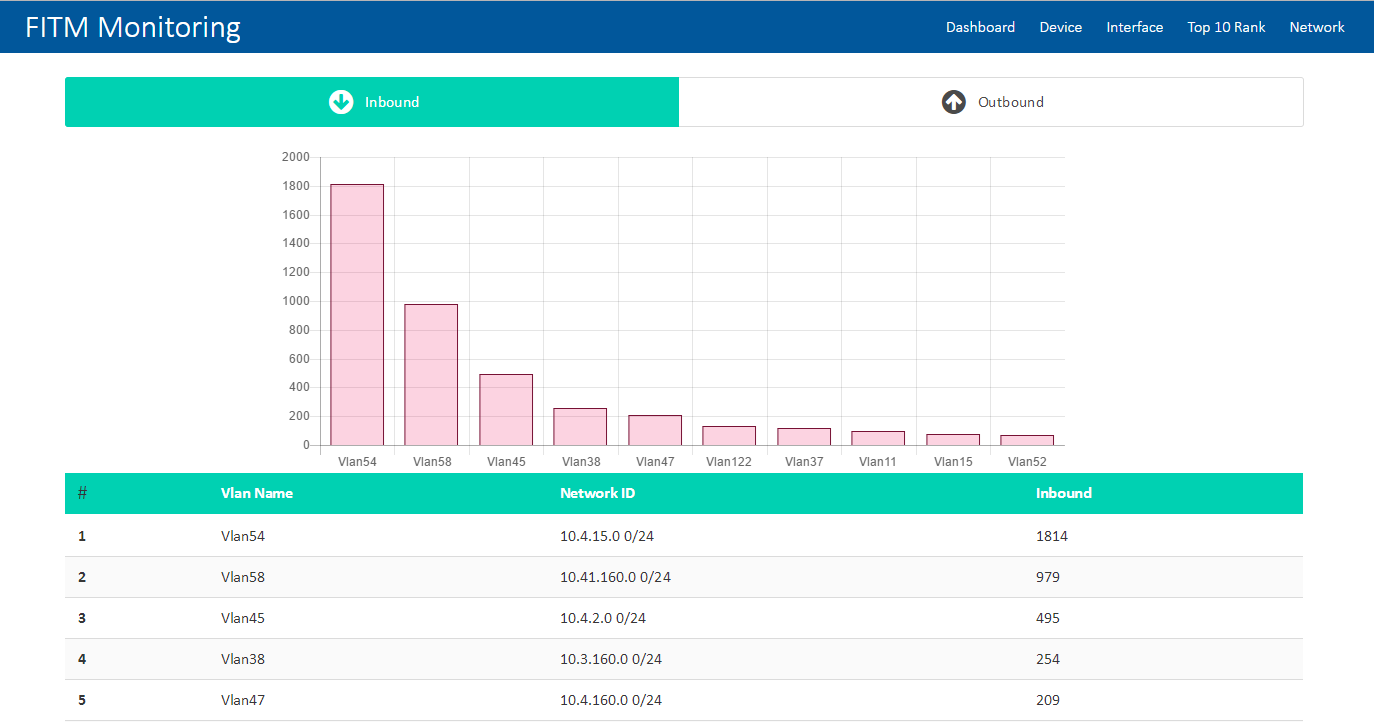
หน้าเว็บสามารถแสดงข้อมูลของห้องที่มีการใช้งานสูงสุดได้ สามารถระบุ VLAN ID Network ID Inbound และ Outbound ทำให้ง่ายและสะดวกต่อการตรวจสอบ และสามารถรายงานผลให้ดูง่ายขึ้นด้วยแผนภูมิแท่ง เป็นประโยชน์มากต่อผู้ดูแลระบบในการตรวจสอบ VLAN ที่มีการใช้งานสูงสุด



**ภาพที่ 4-18** หน้าจอ Top 10 Ranking ของเว็บ

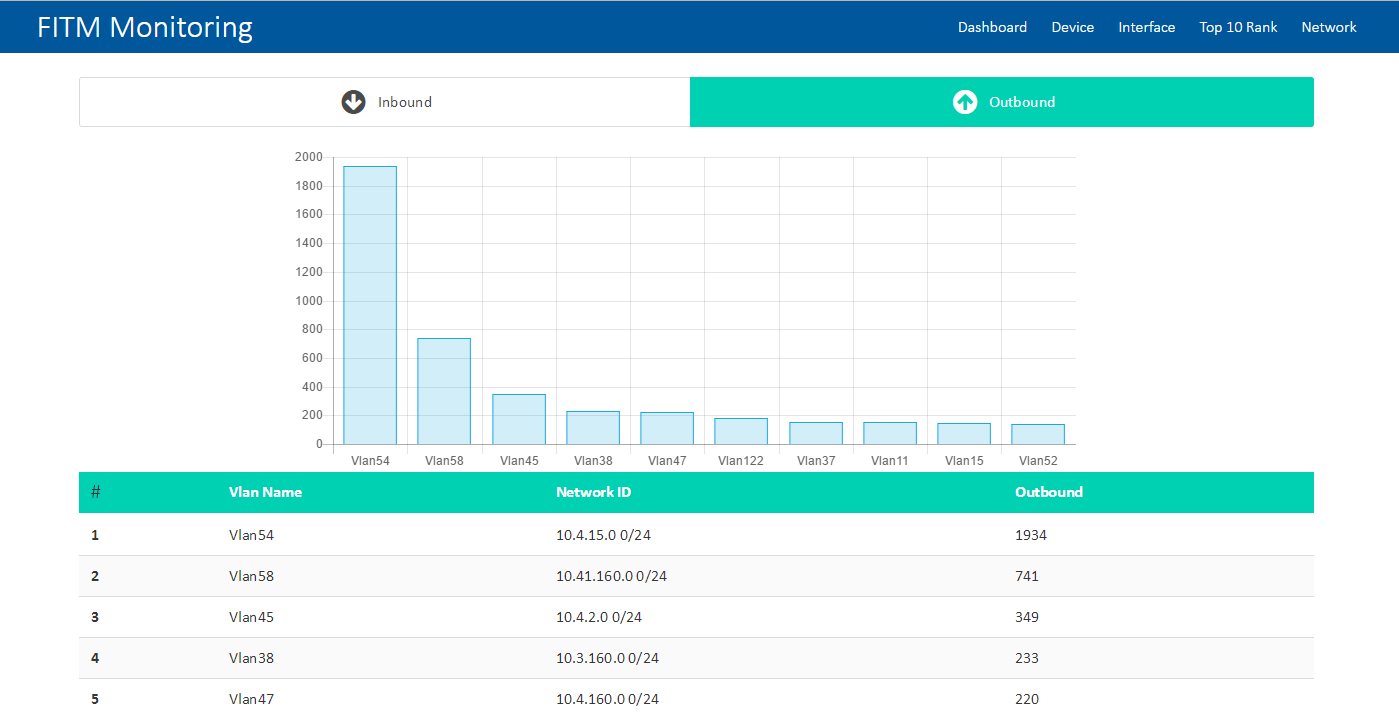
ส่วนของการแสดงผลหน้า Top 10 Rank เป็นการแสดงข้อมูล Vlan ที่มีการใช้งานสูงสุด 10 อันดับ มีการแยกส่วนการแสดงผลให้เป็น 2 ส่วน คือ ส่วนการแสดงข้อมูลของ traffic ขาเข้า และ ส่วนการแสดงข้อมูลของ traffic ขาออก

**ส่วนการแสดงข้อมูลของ traffic ขาเข้า**



**ภาพที่ 4-19** ส่วนการแสดงผลข้อมูล Traffic ขาเข้า แบบใหม่

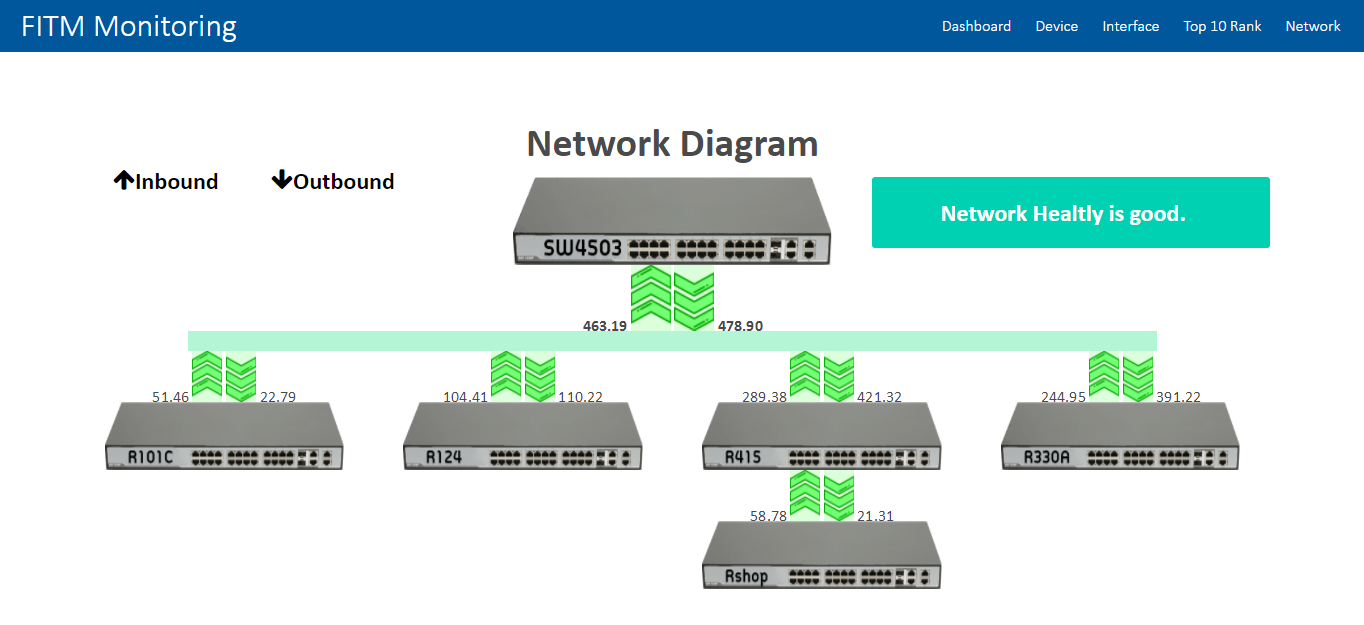
**ส่วนการแสดงข้อมูลของ traffic ขาออก**



**ภาพที่ 4-20** ส่วนการแสดงผลข้อมูล Traffic ขาออก แบบใหม่

**5. หน้าจอเมนู Network**

หน้าเว็บสามารถแสดง Network Diagram ตามจริงสามารถดูประมาณ Traffic ขาเข้าและออกได้จากหน้านี้ และยังสามารถดูการทำงานของเครือข่ายได้



**ภาพที่ 4-21** ส่วนการแสดงผล Network Diagram