ระบบตรวจสอบสถานะและการตอบสนองเครือข่ายอัตโนมัติ

Automated Network Status and Response Monitoring System

บทคัดย่อ

ในยุคปัจจุบันที่ทุกองค์กรใช้ระบบเครือข่ายดิจิทัลเพื่อการสื่อสาร ความเสถียรและความปลอดภัยของเครือข่ายเป็นสิ่งสำคัญที่ส่งผลต่อประสิทธิภาพการปฏิบัติงานของทุกฝ่าย โดยเฉพาะการตรวจสอบและตอบสนองการทำงานของระบบเครือข่ายแบบเรียลไทม์ส่งผลให้ทำงานได้อย่างต่อเนื่อง งานวิจัยนี้มีวัตถุประสงค์เพื่อพัฒนาระบบตรวจสอบสถานะและการตอบสนองอัตโนมัติของเครือข่าย เพื่อเพิ่มประสิทธิภาพและความเสถียรของเครือข่ายในองค์กรขนาดใหญ่ เครื่องมือที่ใช้ในการพัฒนาระบบ ได้แก่ .NET Framework 4.8.1 และภาษา C# ขอบเขตระบบครอบคลุมการตรวจสอบอุปกรณ์และการเชื่อมต่อระบบ ได้แก่ Access Point และสาย LAN ผ่านแดชบอร์ดที่แสดงผลช่วยให้ผู้ดูแลระบบตรวจจับและแก้ไขปัญหาเครือข่ายได้อย่างรวดเร็ว ผลการทดสอบระบบที่พัฒนา พบว่า ประสิทธิภาพการตรวจสอบระบบเครือข่ายมีความรวดเร็วในการตอบสนอง มีความเสถียร และความปลอดภัยของระบบ

คำสำคัญ – ระบบตรวจสอบเครือข่าย, การตอบสนองอัตโนมัติ, ความเสถียรของเครือข่าย, การตรวจสอบแบบเรียลไทม์

ABSTRACT

In today's era, where every organization relies on digital network systems for communication, network stability and security are crucial factors affecting operational efficiency across all departments. Real-time network monitoring and response play a vital role in ensuring continuous operation. This study aims to develop an automated network status and response monitoring system to enhance the efficiency and stability of networks within large organizations. The tools used for system development include .NET Framework 4.8.1 and C#. The system scope covers monitoring devices and network connections, including Access Points and LAN cables, through a dashboard interface, enabling network administrators to detect and resolve network issues promptly. System testing results show that the developed system provides fast response times, stability, and enhanced network security.

Keywords – network monitoring system, automated response, network stability, real-time monitoring

1. บทนำ

ระบบเครือข่ายเป็นหนึ่งในโครงสร้างพื้นฐานที่มีความสำคัญต่อองค์กรทั้งขนาดเล็กและขนาดใหญ่ โดยเฉพาะในยุคที่เทคโนโลยีดิจิทัลมีความสำคัญในการดำเนินธุรกิจและการให้บริการต่าง ๆ [1] การทำงานขององค์กรในปัจจุบันส่วนใหญ่ต้องมีการสื่อสารและการแลกเปลี่ยนข้อมูลผ่านระบบเครือข่าย ไม่ว่าจะเป็นการเชื่อมต่อระหว่างพนักงานภายในองค์กรเพื่อทำงานร่วมกันหรือการสื่อสารกับคู่ค้าและลูกค้าภายนอก ระบบเครือข่ายทำหน้าที่เป็นทางผ่านสำหรับการเข้าถึงข้อมูลที่สำคัญการดำเนินการทางธุรกิจ [2] เช่น ระบบบัญชี การจัดการลูกค้าและการผลิต ทำให้การทำงานของเครือข่ายต้องมีความเสถียรและปลอดภัย เพื่อรองรับปริมาณข้อมูลที่เพิ่มขึ้นและความต้องการที่ซับซ้อนมากขึ้นในทุกระดับ หากระบบเครือข่ายเกิด

ปัญหา อาจส่งผลต่อการทำงานและประสิทธิภาพขององค์กรอย่างรุนแรง ทำให้การบริหารจัดการระบบเครือข่ายหนึ่งในความท้าทายของทุกองค์กรไม่ควรมองข้าม

สำหรับองค์กรขนาดใหญ่ที่มีระบบเครือข่ายที่ซับซ้อน เนื่องจากต้องรองรับจำนวนอุปกรณ์และผู้ใช้งานจำนวนมาก โดยเฉพาะในองค์กรที่มีหลายสาขาหรือมีโครงสร้างที่ซับซ้อน อุปกรณ์เครือข่ายต่าง ๆ [3] เช่น เซิร์ฟเวอร์ (Server) เราเตอร์ (Router) สวิตช์ (Switch) และอุปกรณ์เชื่อมต่อไร้สาย (Access Point) ต้องทำงานร่วมกันอย่างต่อเนื่องและไร้ข้อผิดพลาดเพื่อให้การทำงานขององค์กรเป็นไปอย่างราบรื่น นอกจากนี้ การทำงานของเครือข่ายภายนอกที่เชื่อมต่อกับลูกค้าและพันธมิตรทางธุรกิจก็มีความสำคัญเช่นกัน การตรวจสอบและบำรุงรักษาเครือข่ายในระบบขนาดใหญ่เหล่านี้ต้องใช้เวลามาก และต้องอาศัยความรู้ทางเทคนิคที่เฉพาะเจาะจง เพื่อตรวจสอบปัญหาที่อาจเกิดขึ้น เช่น การเชื่อมต่อที่ขาดหาย ความเร็วในการส่งข้อมูลที่ลดลง หรือปัญหาด้านความปลอดภัย [4] นอกจากนี้กระบวนการตรวจสอบที่ช้าจะทำให้การระบุและแก้ไขปัญหาที่เกิดขึ้นในระบบเครือข่ายล่าช้า ส่งผลให้เกิดความไม่เสถียรในการใช้งานเครือข่ายขององค์กร

จากปัญหาความซับซ้อนและความท้าทายในการจัดการเครือข่ายขนาดใหญ่ คณะผู้วิจัยจึงมีแนวคิดในการออกแบบระบบที่สามารถช่วยลดภาระในการตรวจสอบสถานะและการทำงานของเครือข่ายได้อย่างมีประสิทธิภาพ โดยมีวัตประสงค์ของการวิจัยเพื่อออกแบบและพัฒนาระบบตรวจสอบสถานะและการตอบสนองเครือข่ายอัตโนมัติ ระบบนี้ถูกออกแบบมาเพื่อให้สามารถตรวจสอบสถานะของอุปกรณ์และการเชื่อมต่อภายในองค์กรอย่างครบถ้วน โดยมีการเฝ้าระวังสถานะการทำงานของเครือข่ายแบบเรียลไทม์ ทำให้สามารถรับมือกับปัญหาได้ทันท่วงที นอกจากนี้ ระบบยังถูกออกแบบมาให้รองรับการตรวจสอบจุดเชื่อมต่อไร้สายและการเชื่อมต่อของอุปกรณ์ที่เกี่ยวข้องในองค์กร ซึ่งจะช่วยลดความจำเป็นในการใช้แรงงานคนในการตรวจสอบเครือข่ายและทำให้สามารถบริหารจัดการเครือข่ายได้อย่างมีประสิทธิภาพ

2. วัตถุประสงค์

2.1 เพื่อออกแบบและพัฒนาระบบตรวจสอบสถานะและการตอบสนองอัตโนมัติของเครือข่าย

3. ทฤษฎีที่เกี่ยวข้อง

3.1 แนวคิดการตรวจสอบระบบเครือข่าย

แนวคิดพื้นฐานของการตรวจสอบระบบเครือข่ายเริ่มจากการตรวจสอบการเชื่อมต่อพื้นฐานระหว่างอุปกรณ์ในเครือข่าย ซึ่งกระบวนการนี้รู้จักกันในชื่อของ Ping Monitoring [5] โดย Ping ทำหน้าที่ในการส่งแพ็กเกจข้อมูล (Packet) จากต้นทางไปยังปลายทางเพื่อวัดค่าความหน่วงของการเชื่อมต่อ (Latency) และตรวจสอบว่าการเชื่อมต่อนั้นยังทำงานได้หรือไม่ การศึกษาโดย Dhillipan และคณะ [6] ได้กล่าวถึงเทคนิคการใช้ Ping Method เพื่อวิเคราะห์ประสิทธิภาพของเครือข่ายในสภาวะแวดล้อมที่แตกต่างกัน และชี้ให้เห็นถึงความสำคัญของการตรวจสอบค่าที่ได้จาก Ping ในการปรับปรุงการบริหารจัดการเครือข่าย อย่างไรก็ตาม การใช้เพียงแค่การตรวจสอบ Ping อย่างเดียวอาจไม่เพียงพอในการเฝ้าติดตามการทำงานของเครือข่ายในสภาพแวดล้อมที่ซับซ้อน จึงมีการพัฒนาเทคนิคการตรวจสอบอื่น ๆ เช่น การวิเคราะห์ค่าความสูญเสียของข้อมูล (Packet Loss) การวัดปริมาณแบนด์วิธ (Throughput) และการวิเคราะห์การทำงานของ Latency [7]

3.2 เครื่องมือสำหรับการตรวจสอบระบบเครือข่าย

เครื่องมือและแพลตฟอร์มที่ถูกพัฒนาเพื่อช่วยในการตรวจสอบเครือข่าย โดยเครื่องมือที่นิยมใช้ในการตรวจสอบเครือข่ายองค์กรขนาดใหญ่มักประกอบด้วย Nagios, Zabbix, และ PRTG Network Monitor [8] ทั้งหมดนี้สามารถตรวจสอบอุปกรณ์และการเชื่อมต่อในเครือข่ายได้แบบเรียลไทม์ และกำหนดการแจ้งเตือนเมื่อเกิดความผิดปกติหรือขัดข้องในเครือข่าย เครื่องมือเหล่านี้ได้รับการยอมรับอย่างกว้างขวางเนื่องจากมีความสามารถในการขยายขอบเขตการตรวจสอบ และสามารถเชื่อมต่อกับอุปกรณ์เครือข่ายหลายประเภท นอกจากนี้ยังมีฟีเจอร์การสร้างรายงานที่ช่วยให้ผู้ดูแลสามารถติดตามผลและวิเคราะห์แนวโน้มการทำงานของเครือข่ายได้อย่างชัดเจน

การนำเครื่องมือที่พัฒนาขึ้นโดยใช้ C# และ NET Framework มาใช้ตรวจสอบเครือข่ายก็มีความยืดหยุ่นสูง C# สามารถเขียนโปรแกรมให้ทำงานร่วมกับ API ต่าง ๆ เช่น Simple Network Management Protocol (SNMP) เพื่อดึงข้อมูลจากอุปกรณ์เครือข่าย [9] เช่น เราเตอร์ หรือสวิตช์ เครื่องมือนี้สามารถใช้ในการตรวจสอบสภาพการทำงานของอุปกรณ์เช่น สถานะของ CPU หน่วยความจำ หรือการใช้งาน

แบนด์วิธแบบละเอียด นอกจากนี้ การพัฒนาโดยใช้ NET Framework ยังสามารถรองรับการประมวลผลข้อมูลขนาดใหญ่ได้ดี ทำให้สามารถใช้ในการตรวจสอบเครือข่ายขนาดใหญ่ที่มีอุปกรณ์จำนวนมากเชื่อมต่อกัน

3.3 ความท้าทายในการตรวจสอบเครือข่ายในยุคปัจจุบัน

ระบบเครือข่ายในปัจจุบันมีความซับซ้อนมากขึ้นเนื่องจากการขยายตัวของเครือข่ายในองค์กรและการใช้บริการบนคลาวด์ การตรวจสอบเครือข่ายต้องครอบคลุมทั้งภายในและภายนอกองค์กร และการจัดการเครือข่ายที่มีโครงสร้างกระจายตัวไปยังหลายภูมิภาคทำให้ความต้องการในการตรวจสอบและดูแลระบบเพิ่มมากขึ้น Bravo-Arrabal และคณะ [10] เสนอแนวทางการใช้ Hybrid Network Monitoring เป็นวิธีที่รวมการตรวจสอบภายในองค์กรและเครือข่ายคลาวด์เข้าด้วยกัน การใช้การตรวจสอบแบบไฮบริดนี้ช่วยให้สามารถตรวจสอบสถานะการทำงานของเครือข่ายทั้งหมด

การพัฒนาเครื่องมือโดยใช้ C# และ NET Framework มีการใช้เทคนิคการเข้ารหัสข้อมูล (Encryption) และการรับรองความถูกต้อง (Authentication) [11] เพื่อให้การตรวจสอบเครือข่ายมีความปลอดภัยมากยิ่งขึ้น นอกจากนี้ C# ยังสามารถใช้ในการพัฒนาฟีเจอร์การแจ้งเตือนเมื่อมีเหตุการณ์ที่อาจเกิดความเสี่ยงด้านความปลอดภัย เช่น การโจมตีแบบ DDoS หรือการพยายามเข้าถึงอุปกรณ์เครือข่ายโดยไม่ได้รับอนุญาต ทำให้การตรวจสอบเครือข่ายในปัจจุบันไม่เพียงแค่เน้นการตรวจสอบประสิทธิภาพแต่ยังต้องคำนึงถึงความปลอดภัยของระบบ [12]

4. วิธีการดำเนินงานวิจัย

การดำเนินงานวิจัยเรื่องระบบตรวจสอบสถานะและการตอบสนองของเครือข่ายอัตโนมัติถูกจัดทำตามวงจรการพัฒนาระบบ (System Development Life Cycle: SDLC) โดยแบ่งออกเป็นขั้นตอนต่าง ๆ ดังนี้

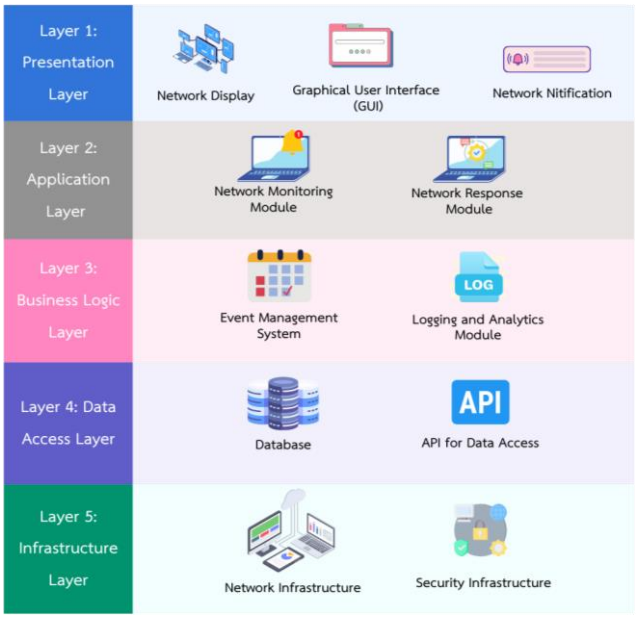
4.1 การวิเคราะห์ความต้องการ (Requirement Analysis)

การรวบรวมความต้องการจากผู้ใช้งาน เช่น ผู้ดูแลเครือข่าย และระบุคุณสมบัติหลักของระบบ เช่น การตรวจสอบ Access Point การเช็คสาย LAN การสแกนอุปกรณ์ที่เชื่อมต่อ การเข้าถึงเครื่องระยะไกลผ่าน PuTTY และการแจ้งเตือนปัญหาเครือข่าย และวิเคราะห์เครื่องมือและเทคโนโลยีที่เหมาะสม เช่น การใช้โปรโตคอล SSH การทำงานผ่านอินเทอร์เฟซกราฟิก และการประมวลผลบนเซิร์ฟเวอร์สำหรับจัดการเครือข่าย

4.2 การออกแบบระบบ (System Design)

4.2.1การออกแบบเชิงสถาปัตยกรรมระบบและ Use Case Diagram

การวางแผนและออกแบบสถาปัตยกรรมระบบที่จะแสดงข้อมูลเครือข่ายและอุปกรณ์ต่าง ๆ แบบเรียลไทม์ ดังภาพที่ 1



ภาพ 1 สถาปัตยกรรมระบบ

ภาพที่ 1 แสดงสถาปัตยกรรมระบบตรวจสอบสถานะและการตอบสนองอัตโนมัติของเครือข่าย แบ่งโครงสร้างเป็นดังนี้

1. Layer 1: ชั้นการแสดงผล (Presentation Layer)

- หน้าจอแสดงผล (User Interface)
  - การแสดงผลข้อมูลเครือข่าย เช่น สถานะของอุปกรณ์ การเชื่อมต่อของ Access Point การสแกนหาอุปกรณ์ที่เชื่อมต่อในวงเครือข่าย
  - การแสดงผลแบบกราฟิก (Graphical User Interface - GUI) ที่ให้ผู้ใช้สามารถดูสถานะเครือข่ายในรูปแบบของโครงสร้างลำดับชั้น (Tree Hierarchy) เช่น การเชื่อมต่อ Access Point แต่ละชั้น ประกอบด้วย Devices หลาย ๆ เครื่อง
  - ผู้ดูแลเครือข่ายสามารถโต้ตอบกับระบบผ่านแดชบอร์ดที่แสดงสถานะเรียลไทม์ การแจ้งเตือนเมื่อเกิดปัญหา และการเข้าถึงฟังก์ชันการตอบสนอง เช่น การรีโมทเข้าเครื่องหรือการตั้งค่าใหม่

2. Layer 2: ชั้นการประมวลผลแอปพลิเคชัน (Application Layer)

- ระบบการตรวจสอบเครือข่าย (Network Monitoring Module)
  - ประมวลผลและตรวจสอบสถานะของอุปกรณ์เครือข่าย เช่น Access Point สาย LAN และการเชื่อมต่ออินเทอร์เน็ต
  - ทำการสแกนอุปกรณ์ที่เชื่อมต่อในเครือข่าย พร้อมทั้งแสดงข้อมูล IP Address MAC Address และชื่ออุปกรณ์
  - วิเคราะห์การใช้แบนด์วิธ (Bandwidth Monitoring) เพื่อตรวจสอบการใช้งานที่ผิดปกติ
- ระบบตอบสนองอัตโนมัติ (Network Response Module):
  - ทำหน้าที่ตอบสนองต่อเหตุการณ์ที่เกิดขึ้นในเครือข่าย เช่น การแจ้งเตือนผ่านอีเมลหรือข้อความเมื่อเกิดปัญหาขึ้นในเครือข่าย
  - ฟังก์ชันการเข้าถึงเครื่องระยะไกล (Remote PC) ผ่านโปรโตคอล RDP หรือการใช้ PuTTY เพื่อเข้าถึงเซิร์ฟเวอร์หรืออุปกรณ์เครือข่ายผ่าน SSH/Telnet
  - ฟังก์ชันอัปเดตเฟิร์มแวร์และการตั้งค่าอุปกรณ์จากระยะไกล
  - 3. Layer 3: ชั้นกฎเกณฑ์การทำงาน (Business Logic Layer)
- ระบบตอบสนองอัตโนมัติ (Network Response Module)
  - กำหนดกฎการตอบสนองต่อสถานะเครือข่าย เช่น เมื่อเกิดข้อผิดพลาดในสาย LAN หรือ Access Point ระบบจะส่งการแจ้งเตือนทันที
  - ควบคุมการตรวจสอบสถานะเครือข่ายแบบเรียลไทม์ และแสดงผลผ่านแดชบอร์ดให้ผู้ใช้เห็นปัญหาและสถานะในเครือข่าย
- ระบบบันทึกและวิเคราะห์ (Logging and Analytics Module)
  - บันทึกเหตุการณ์ที่เกิดขึ้นในเครือข่าย เช่น การเชื่อมต่ออุปกรณ์ การขาดการเชื่อมต่อ และปัญหาที่พบในระบบ
  - วิเคราะห์ข้อมูลย้อนหลังเพื่อตรวจสอบปัญหาที่อาจเกิดขึ้นซ้ำในอนาคต

4. Layer 4: ชั้นการเข้าถึงข้อมูล (Data Access Layer)

- ฐานข้อมูล (Database)
  - เก็บข้อมูลทั้งหมดที่เกี่ยวข้องกับเครือข่าย เช่น ข้อมูลอุปกรณ์ (Device Information) ข้อมูลการเชื่อมต่อ (Connection Logs) และข้อมูลการแจ้งเตือน (Alert History)
  - ฐานข้อมูลรองรับการเก็บข้อมูลแบบกระจาย (Distributed Database) เพื่อรองรับการตรวจสอบเครือข่ายในหลายสถานที่พร้อมกัน
- การเข้าถึงข้อมูลผ่าน API (API for Data Access)
  - ใช้ API สำหรับการเข้าถึงข้อมูลเครือข่ายจากฐานข้อมูลเพื่อนำข้อมูลไปแสดงผลและประมวลผลในระบบตรวจสอบสถานะและตอบสนอง

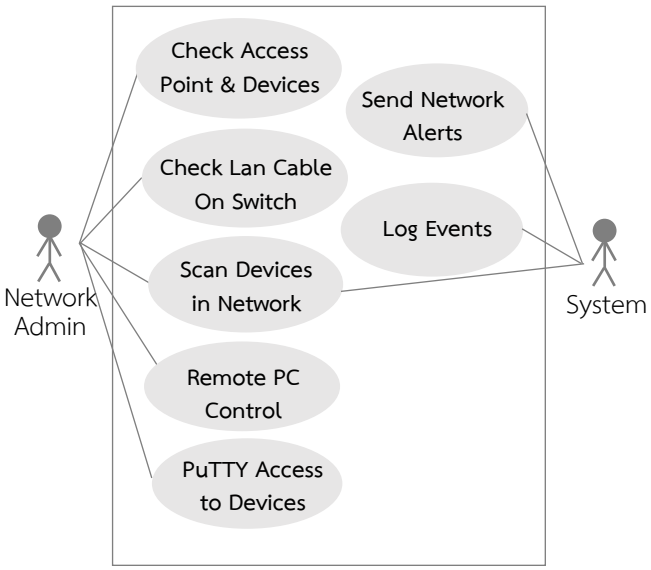
5. Layer 5: ชั้นโครงสร้างพื้นฐาน (Infrastructure Layer)

- ระบบเครือข่าย (Network Infrastructure)
  - ประกอบด้วยเซิร์ฟเวอร์และอุปกรณ์เครือข่ายที่รองรับการตรวจสอบและควบคุมอุปกรณ์ทั้งหมดภายในองค์กร เช่น Access Point สวิตช์ เราเตอร์ และเซิร์ฟเวอร์
  - รองรับการตรวจสอบอุปกรณ์ในหลายสาขาขององค์กรและการทำงานแบบกระจายตัว (Distributed System)
- ระบบความปลอดภัย (Security Infrastructure)
  - ปกป้องข้อมูลเครือข่ายและระบบตรวจสอบจากการบุกรุก โดยการเข้ารหัสข้อมูลระหว่างการส่งผ่านระบบ เช่น การใช้ SSH สำหรับการเข้าถึงอุปกรณ์ระยะไกล
  - ใช้ระบบการยืนยันตัวตนแบบหลายขั้นตอน (Multi-factor Authentication) เพื่อป้องกันการเข้าถึงที่ไม่ได้รับอนุญาต

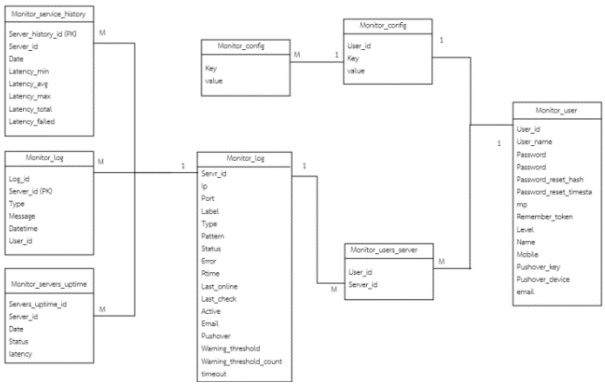
การออกแบบ Use Case Diagram สำหรับระบบตรวจสอบสถานะและการตอบสนองของเครือข่ายอัตโนมัติสำหรับผู้ดูแลระบบเครือข่าย (Network Administrator) ในการจัดการและตรวจสอบสถานะของอุปกรณ์ในเครือข่าย โดยระบบจะช่วยในการตรวจสอบปัญหาที่เกิดขึ้นและตอบสนองต่อเหตุการณ์ต่าง ๆ ดังภาพที่ 2

4.2.2 การออกแบบฐานข้อมูล

การออกแบบฐานข้อมูลทั้งหมด 8 ตารางเพื่อเก็บข้อมูลเกี่ยวกับการตั้งค่าอุปกรณ์ ผู้ใช้งาน การกระทำของผู้ใช้ อุปกรณ์เครือข่าย สถานะการเชื่อมต่อ และการบันทึกเหตุการณ์ที่เกิดขึ้นในระบบ ภาพแสดง ER-Diagram ดังภาพที่ 3



ภาพ 2 Use Case Diagram



ภาพ 3 ER-Diagram

4.2.3 การออกแบบอินเทอร์เฟซผู้ใช้ (UI/UX)

การออกแบบอินเทอร์เฟซที่ง่ายต่อการใช้งาน เพื่อให้ผู้ดูแลเครือข่ายสามารถตรวจสอบและควบคุมเครือข่ายได้อย่างสะดวก เช่น การแสดงผลผ่านแดชบอร์ด การแจ้งเตือนปัญหา และการสั่งงานรีโมทผ่านระบบ

4.3 การพัฒนาและเขียนโปรแกรม (Development)

- เครื่องมือที่ใช้ในการพัฒนาระบบได้แก่

1. .NET Framework 4.8.1 เป็นแพลตฟอร์มการพัฒนาซอฟต์แวร์ที่ Microsoft พัฒนาขึ้นมาเพื่อสร้างและรันแอปพลิเคชันต่างๆ บน Windows โดยเฉพาะแอปพลิเคชันที่ใช้ภาษา คือ C#

2. Visual Studio Code (VS Code) เป็นโปรแกรมแก้ไขข้อความ (Text Editor) และเครื่องมือสำหรับพัฒนาโปรแกรม

- การพัฒนาระบบเซิร์ฟเวอร์และเครือข่าย การพัฒนาระบบตามการออกแบบที่ได้วางไว้ โดยการเขียนโปรแกรมที่รองรับการตรวจสอบสถานะอุปกรณ์ การควบคุมผ่านเครือข่าย และการเชื่อมต่ออุปกรณ์ผ่าน SSH หรือ Telnet
- การพัฒนาอินเทอร์เฟซผู้ใช้ พัฒนาอินเทอร์เฟซกราฟิก (Graphical User Interface: GUI) ที่สามารถแสดงสถานะของเครือข่ายแบบเรียลไทม์ พร้อมกับสร้างฟังก์ชันการตอบสนอง เช่น การแจ้งเตือนอัตโนมัติเมื่อเกิดปัญหา

4.4 การทดสอบระบบ (Testing)

การทดสอบระบบ แบ่งเป็น 3 ประเภท ได้แก่ การทดสอบหน่วย (Unit Testing) เพื่อทดสอบการทำงานของแต่ละฟังก์ชัน การทดสอบการทำงานร่วมกัน (Integration Testing) เพื่อ ทดสอบการเชื่อมโยงระหว่างส่วนต่างๆ ของระบบ และการทดสอบความปลอดภัย (Security Testing) เพื่อตรวจสอบระบบความปลอดภัย การป้องกันข้อมูล และการเข้าถึงระบบเครือข่ายโดยไม่ได้รับอนุญาต ตารางที่ 1 แสดงผลการทดสอบระบบตรวจสอบสถานะและการตอบสนองอัตโนมัติของเครือข่ายบนพื้นฐานของสถาปัตยกรรมระบบในภาพที่ 1

ตาราง 1 ผลการทดสอบระบบตามสถาะปัตยกรรมระบบที่ออกแบบ

Test Case	ผลลัพธ์การทดสอบ
1. Presentation Layer	
Test Case 1: ตรวจสอบการแสดงผลของสถานะเครือข่าย • ประภทการทดสอบ: Integration Testing • เงื่อนไขการทดสอบ: ผู้ใช้เข้าสู่แดชบอร์ดและดูสถานะเครือข่าย • คาดหวังผลลัพธ์: ระบบแสดงสถานะของอุปกรณ์ที่เชื่อมต่อ สถานะ Access Point สถานะการเชื่อมต่อของแต่ละอุปกรณ์ในโครงสร้างแบบ Tree Hierarchy	ผ่านตามเงื่อนไข
Test Case 2: การแจ้งเตือนเมื่อเกิดปัญหาเครือข่าย • ประภทการทดสอบ: Integration Testing • เงื่อนไขการทดสอบ: ปิดการทำงานของ Access Point หรืออุปกรณ์ใดๆ ในเครือข่าย • คาดหวังผลลัพธ์: ระบบแสดงการแจ้งเตือนบนแดชบอร์ดแบบเรียลไทม์และส่งข้อความแจ้งเตือน	ผ่านตามเงื่อนไข

Test Case	ผลลัพธ์การทดสอบ
2. Application Layer	
Test Case 3: การตรวจสอบสถานะของอุปกรณ์เครือข่าย ● ประภทการทดสอบ: ● เงื่อนไขการทดสอบ: ตรวจสอบสถานะของ Access Point และการเชื่อมต่ออินเทอร์เน็ต ● คาดหวังผลลัพธ์: ระบบแสดงข้อมูลสถานะที่ถูกต้อง เช่น IP Address, MAC Address, ชื่ออุปกรณ์	ผ่านตามเงื่อนไข
Test Case 4: การตอบสนองต่อการใช้งานแบนด์วิธที่ผิดปกติ ● ประภทการทดสอบ: Unit Testing ● เงื่อนไขการทดสอบ: ส่งข้อมูลผ่านเครือข่ายให้เกินขีดจำกัดแบนด์วิธ ● คาดหวังผลลัพธ์: ระบบแจ้งเตือนผ่านอีเมลเมื่อการใช้แบนด์วิธผิดปกติ	ผ่านตามเงื่อนไข
Test Case 5: ฟังก์ชันการเข้าถึงเครื่องระยะไกล ● ประภทการทดสอบ: Unit Testing ● เงื่อนไขการทดสอบ: เข้าถึงเครื่องระยะไกลผ่าน SSH ● คาดหวังผลลัพธ์: สามารถเข้าถึงเซิร์ฟเวอร์และควบคุมอุปกรณ์ได้อย่างปลอดภัย	ผ่านตามเงื่อนไข
3. Business Logic Layer	
Test Case 6: การแจ้งเตือนเมื่อเกิดข้อผิดพลาดในระบบ ● ประภทการทดสอบ: Integration Testing ● เงื่อนไขการทดสอบ: ปิดการทำงานของสาย LAN ● คาดหวังผลลัพธ์: ระบบส่งการแจ้งเตือนทันทีผ่านแดชบอร์ดและอีเมล	ผ่านตามเงื่อนไข
Test Case 7: ระบบบันทึกและวิเคราะห์เหตุการณ์เครือข่าย ● ประภทการทดสอบ: Integration Testing ● เงื่อนไขการทดสอบ: ทำการเชื่อมต่อและขาดการเชื่อมต่ออุปกรณ์หลายครั้ง ● คาดหวังผลลัพธ์: ระบบบันทึกและแสดงผลเหตุการณ์ย้อนหลังเพื่อให้สามารถวิเคราะห์ได้	ผ่านตามเงื่อนไข
4. Data Access Layer	
Test Case 8: การเข้าถึงข้อมูลอุปกรณ์จากฐานข้อมูล ● ประภทการทดสอบ: Unit Testing ● เงื่อนไขการทดสอบ: ดึงข้อมูลการเชื่อมต่อจากฐานข้อมูลผ่าน API ● คาดหวังผลลัพธ์: สามารถเข้าถึงข้อมูลการเชื่อมต่อและแสดงผลได้อย่างถูกต้อง	ผ่านตามเงื่อนไข
Test Case 9: การรองรับการทำงานหลายสถานที่พร้อมกัน ● ประภทการทดสอบ: Integration Testing ● เงื่อนไขการทดสอบ: ตรวจสอบการเข้าถึงฐานข้อมูลเครือข่ายจากหลายสถานที่ ● คาดหวังผลลัพธ์: ระบบสามารถเข้าถึงข้อมูลได้พร้อมกันหลายจุดโดยไม่มีความล่าช้า	ผ่านตามเงื่อนไข
5. Infrastructure Layer	
Test Case 10: การทำงานแบบกระจายตัวในหลายแผนก ● ประภทการทดสอบ: Integration Testing ● เงื่อนไขการทดสอบ: ตรวจสอบการเชื่อมต่อจากหลายแผนกขององค์กร ● คาดหวังผลลัพธ์: ระบบสามารถควบคุมและตรวจสอบเครือข่ายในหลายแผนกได้พร้อมกัน	ผ่านตามเงื่อนไข
Test Case 11: การใช้ระบบความปลอดภัยในการเข้าถึงอุปกรณ์ ● ประภทการทดสอบ: Security Testing ● เงื่อนไขการทดสอบ: เข้าถึงอุปกรณ์ผ่าน SSH พร้อม Multi-factor Authentication ● คาดหวังผลลัพธ์: ระบบสามารถยืนยันตัวตนได้และเข้าถึงอุปกรณ์อย่างปลอดภัย	ผ่านตามเงื่อนไข

4.5 การปรับใช้งาน (Deployment)

การติดตั้งระบบบนเครือข่ายจริง คณะผู้วิจัยทำการติดตั้งระบบบนเครือข่ายที่ใช้งานจริงขององค์กร พร้อมกับทดสอบการทำงานและการตอบสนองในสภาพแวดล้อมจริง

4.6 การบำรุงรักษา (Maintenance)

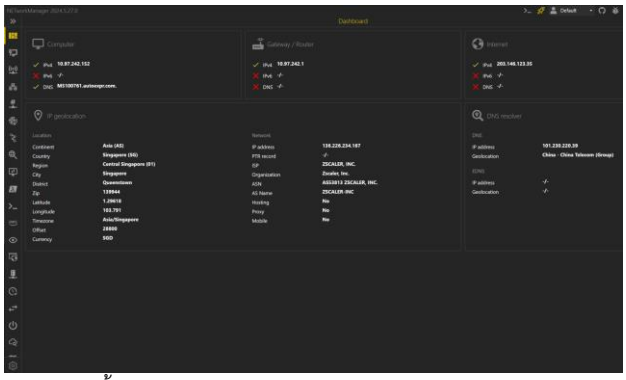
การตรวจสอบและปรับปรุงระบบ หลังจากที่มีการใช้งานระบบจริง จะมีการตรวจสอบการทำงานของระบบอย่างสม่ำเสมอ รวมถึงการปรับปรุงหรือแก้ไขข้อผิดพลาดที่พบและอาจมีการเพิ่มการสนับสนุนอุปกรณ์เครือข่ายใหม่ ๆ

5. ผลการดำเนินงานวิจัย

5.1 ผลการพัฒนาระบบตรวจสอบสถานะและการตอบสนองอัตโนมัติของเครือข่าย

5.1.1 หน้าหลักของระบบ

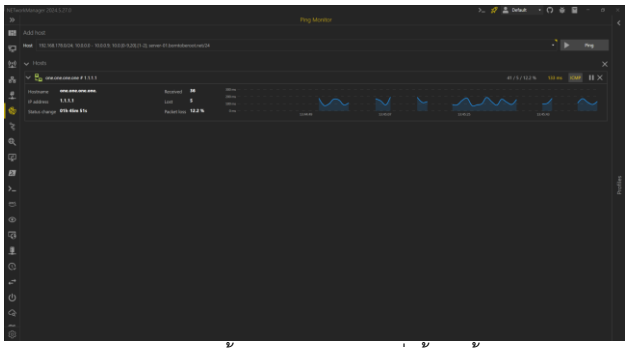
หน้าหลักของระบบตรวจสอบสถานะและการตอบสนองอัตโนมัติของเครือข่าย แบ่งการแสดงผลเป็น 3 ส่วน ได้แก่ การทำงานของคอมพิวเตอร์ เกตเวย์และเราเตอร์ อินเทอร์เน็ต โดยสามารถตรวจสอบสถานะได้ทั้ง IPV4, IPV6, และ DNS การระบุตำแหน่งของระบบเครือข่าย และรายละเอียด DNS ดังภาพที่ 4



ภาพ 4 หน้าหลักของระบบตรวจสอบสถานะและการตอบสนองอัตโนมัติ

5.1.2 หน้าทดสอบกราฟทราฟฟิคของการทำงาน

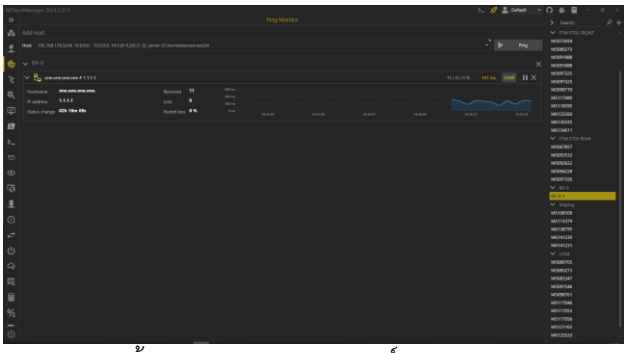
หน้าทดสอบการทำงานของกราฟทราฟฟิค (Traffic Graph) ถูกดำเนินการเพื่อตรวจสอบประสิทธิภาพของค่าการดาวน์โหลดและอัพโหลด โดยประเมินค่าต่ำสุดและสูงสุดที่สามารถเกิดขึ้นได้ ซึ่งสามารถรายงานผลเป็นรายนาทีได้โดยการเลือกจากแถบรายการกราฟแสดงผลการทำงาน ทั้งนี้ผู้ใช้งานสามารถปรับการแสดงผลรายงานให้เป็นแบบรายนาทีหรือรายชั่วโมงตามความต้องการ ดังภาพที่ 5



ภาพ 5 หน้าหลักของโปรแกรมสั่งซื้อสินค้า

5.1.3 หน้าทดสอบการดูอินเตอร์เฟสระดับกลุ่ม

หน้าทดสอบการตรวจสอบอินเทอร์เฟซ (Interface) ในระดับกลุ่มหรือแผนกถูกออกแบบมาเพื่อประเมินการทำงานของอินเทอร์เฟซที่ได้ถูกจัดกลุ่มไว้ล่วงหน้า โดยผู้ใช้สามารถเลือกกลุ่มที่ต้องการตรวจสอบ ได้อย่างง่ายดาย รวมถึงสามารถเลือกโฮสต์ที่จัดอยู่ในกลุ่มนั้นเพื่อแสดงผลการทำงานของแต่ละพอร์ตในกลุ่มที่เกี่ยวข้อง ตัวอย่างในการตรวจสอบการทำงานของสายการผลิตและอุปกรณ์ มีขั้นตอนที่ต้องปฏิบัติตามอย่างชัดเจน โดยยกตัวอย่างดังนี้ ขั้นแรกคือการเลือกสายการผลิตที่ต้องการตรวจสอบ ในที่นี้คือ "RX-0" ซึ่งเป็นการกำหนดขอบเขตของการตรวจสอบการทำงาน หลังจากนั้นในขั้นตอนที่สอง เลือกอุปกรณ์ที่ต้องการประเมิน ซึ่งตัวอย่างอุปกรณ์นี้คือ "RX-0-1" เพื่อระบุอุปกรณ์เฉพาะที่จะได้รับการตรวจสอบ จากนั้นจึงเข้าสู่ขั้นตอนที่สาม ซึ่งเป็นการเลือกดูกราฟการทำงานของพอร์ตอินเทอร์เฟซโดยสามารถเลือกพอร์ตที่ต้องการตรวจสอบ ดังภาพที่ 6



ภาพ 6 หน้าทดสอบการเลือกอินเตอร์เฟสของการทำงาน

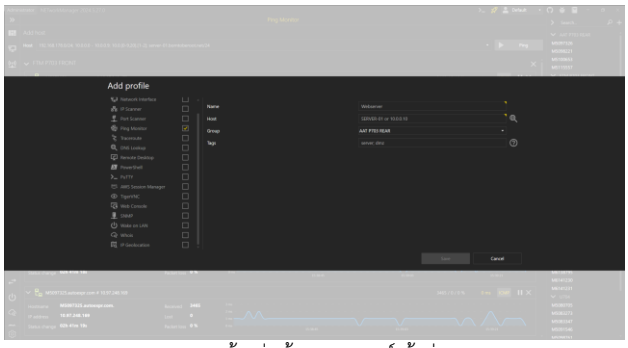
5.1.4 หน้าเชื่อมของอุปกรณ์และเซิร์ฟเวอร์

หน้าทดสอบการแสดงภาพการเชื่อมต่อของอุปกรณ์เครือข่ายต่าง ๆ และเซิร์ฟเวอร์ถูกออกแบบมาเพื่อความสะดวกในการตรวจสอบและค้นหาตำแหน่งของอุปกรณ์ที่ต้องการ อุปกรณ์แต่ละชิ้นจะแสดงรายละเอียดที่ชัดเจน เช่น ตำแหน่งที่ตั้งและข้อมูลที่เกี่ยวข้อง เพื่อให้สามารถระบุและประเมินสถานะของอุปกรณ์ได้อย่างง่ายดายและแม่นยำ ดังภาพที่ 7

ภาพ 7 หน้าเชื่อมของอุปกรณ์และเซิร์ฟเวอร์

5.1.5 หน้าเพิ่มข้อมูลอุปกรณ์เข้าสู่ระบบ

หน้าเพิ่มข้อมูลเข้าสู่ระบบได้โดยการป้อน IP และชื่อของคอมพิวเตอร์เข้าไป พร้อมทั้งใส่แท็ก (Tag) เพื่อระบุตำแหน่งหรือสถานที่ของคอมพิวเตอร์นั้นๆ การเพิ่มข้อมูลนี้ช่วยให้ระบบสามารถแสดงข้อมูลได้อย่างถูกต้องและสะดวกในการติดตามหรือค้นหาสถานที่ของคอมพิวเตอร์แต่ละเครื่อง ดังภาพที่ 8



ภาพ 8 หน้าเพิ่มข้อมูลอุปกรณ์เข้าสู่ระบบ

5. สรุปผลการวิจัย

โครงการวิจัยนี้มีวัตถุประสงค์เพื่อพัฒนาระบบตรวจสอบสถานะและการตอบสนองเครือข่ายอัตโนมัติ โดยระบบจะช่วยตรวจสอบสถานะของอุปกรณ์เครือข่ายแบบเรียลไทม์ ทำให้สามารถจัดการปัญหาเครือข่ายได้ทันที ระบบยังรองรับการตรวจสอบอุปกรณ์หลายประเภท รวมถึง Access Point และ LAN สายต่าง ๆ ผ่านแดชบอร์ดที่แสดงสถานะเครือข่ายโดยละเอียด ซึ่งช่วยลดภาระงานของผู้ดูแลในการติดตามและแก้ไขปัญหาเครือข่าย ระบบได้รับการทดสอบประสิทธิภาพเริ่มจากการแสดงผลสถานะเครือข่าย การแจ้งเตือนเมื่อเกิดปัญหา ไปจนถึงการตรวจสอบความปลอดภัย ผลการทดสอบแสดงให้เห็นว่าสามารถทำงานได้ตามที่ออกแบบ ทั้งในด้านความเสถียร ความเร็วในการตอบสนอง และความปลอดภัย ข้อเสนอแนะงานวิจัยครั้งต่อไปมุ่งเน้นการใช้ปัญญาประดิษฐ์เพื่อให้ระบบตรวจสอบเครือข่ายคาดการณ์และตอบสนองต่อปัญหาการทำงานของระบบเครือข่ายอย่างแม่นยำมากขึ้น และพัฒนาการเชื่อมต่อกับคลาวด์เพื่อปรับปรุงใหม่ให้ระบบมีความปลอดภัยมากขึ้น และการประเมิน

เอกสารอ้างอิง

[1] W. Yoon, J. Jeong, and K. W. Park, “Informal network structure and knowledge sharing in organizations: An empirical study of a Korean paint manufacturing company,” *Adm. Sci.*, vol. 11, no. 2, p. 52, 2021.

[2] N. Sangperm, “The effect of business networking on the business performance through the technology capability and innovation in the transportation business of Thailand,” *MUT J. Bus. Adm.*, vol. 19, no. 2, pp. 66–88, 2022.

[3] D. Xia, Q. Li, Y. Lei, X. Shen, M. Qian, and C. Zhang, “Extreme vulnerability of high-order organization in complex networks,” *Phys. Lett. A*, vol. 424, no. 127829, p. 127829, 2022.

[4] J. Merrill, M. Caldwell, M. L. Rockoff, K. Gebbie, K. M. Carley, and S. Bakken, “Findings from an organizational network analysis to support local public health management,” J. Urban Health, vol. 85, no. 4, pp. 572–584, 2008.

[6] J. Dhillipan, N. Vijayalakshmi, and S. Suriya, “Network monitoring system using ping methodology and GUI,” in *Intelligent Systems Reference Library*, Cham: Springer International Publishing, 2020, pp. 13–22.

[7] A. A. El-Saleh, A. Alhammadi, I. Shayea, W. H. Hassan, M. S. Honnurvali, and Y. I. Daradkeh, “Measurement analysis and performance evaluation of mobile broadband cellular networks in a populated city,” *Alex. Eng. J.*, vol. 66, pp. 927–946, 2023.

[8] W. Song et al., “A software Deep Packet Inspection system for network traffic analysis and anomaly detection,” *Sensors (Basel)*, vol. 20, no. 6, p. 1637, 2020.

[8] โปรแกรมตรวจสอบเซิร์ฟเวอร์ – เลือกสุดยอด. [ออนไลน์] 2566. [สืบค้นเมื่อ 20 สิงหาคม 2567] จาก

https://tsplus.net/th/server-monitoring-software-a-pick-of-the-best/

[9] Simple Network Management Protocol (SNMP). [ออนไลน์] 2566. [สืบค้นเมื่อ 20 สิงหาคม 2567] จาก https://www.geeksforgeeks.org/simple-network-management-protocol-snmp/

[10] J. Bravo-Arrabal, J. J. Fernandez-Lozano, J. Serón, J. A. Gomez-Ruiz, and A. García-Cerezo, "Development and implementation of a hybrid wireless sensor network of low power and long range for urban environments," *Sensors (Basel)*, vol. 21, no. 2, p. 567, 2021.

[11] A Comprehensive Guide to Secure Coding in C#. [ออนไลน์] 2567. [สืบค้นเมื่อ 20 สิงหาคม 2567] จาก https://www.c-sharpcorner.com/article/a-comprehensive-guide-to-secure-coding-in-c-sharp/

[12] R. V. Deshmukh and K. K. Devadkar, "Understanding DDoS attack & its effect in cloud environment," *Procedia Comput. Sci.*, vol. 49, pp. 202–210, 2015.