****

**ระบบตรวจจับแก๊สรั่วไหลและการแจ้งเตือนผ่าน Gmail**

**Development Gas leak detection and notification via Gmail**

**โดย**

**นายภัคพล โสตทิพย์**

**รหัสนักศึกษา 6504305001322**

**อาจารย์ประจำรายวิชา**

**ดร. สมพงษ์ ยิ่งเมือง**

**รายงานเล่มนี้เป็นส่วนหนึ่งของรายวิชา SCS0216 วิศวกรรมซอฟต์แวร์**

**หลักสูตรวิทยาศาสตร์บัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์ มหาวิทยาลัยมหาวิทยาลัยราชภัฏสุราษฎร์ธานี ประจำภาคการศึกษาที่ 2 ปีการศึกษา 2567**

**บทที่1**

**บทนำ**

**1.1 ความเป็นมาและความสำคัญ**

ในยุคดิจิทัลที่เทคโนโลยี Internet of Things (IoT) กำลังเติบโตอย่างรวดเร็ว การนำอุปกรณ์ต่าง ๆ มาเชื่อมต่อกันผ่านอินเทอร์เน็ตกลายเป็นสิ่งที่เปลี่ยนแปลงวิถีชีวิตและการทำงานของมนุษย์ หนึ่งในนวัตกรรมที่ได้รับความสนใจอย่างมากคือระบบติดตามยานพาหนะ (Vehicle Tracking System) ที่ผสานการทำงานของเทคโนโลยี GPS (Global Positioning System) กับ IoT

ในปัจจุบันการจัดการและควบคุมยานพาหนะมีบทบาทสำคัญในหลากหลายอุตสาหกรรม เช่น โลจิสติกส์ การขนส่งมวลชน บริการแท็กซี่ และการจัดการยานพาหนะส่วนบุคคล ระบบติดตามยานพาหนะช่วยเพิ่มประสิทธิภาพในการจัดการ เสริมสร้างความปลอดภัย และลดค่าใช้จ่ายที่เกี่ยวข้อง ตัวอย่างเช่น บริษัทขนส่งสามารถใช้ระบบติดตามเพื่อควบคุมการใช้งานเชื้อเพลิง วางแผนเส้นทางเดินรถและติดตามสถานะการขนส่งสินค้าแบบเรียลไทม์

นอกจากนี้การพัฒนาระบบติดตามยานพาหนะยังมีความสำคัญในด้านความปลอดภัย เช่น การป้องกันการโจรกรรม การแจ้งเตือนกรณีเกิดอุบัติเหตุ และการตรวจสอบพฤติกรรมการขับขี่ ระบบดังกล่าวไม่เพียงแต่เพิ่มความสะดวกสบายแก่ผู้ใช้งาน แต่ยังช่วยลดความเสี่ยงและความเสียหายที่อาจเกิดขึ้น

**1.2 วัตถุประสงค์**

1.2.1 เพื่อติดตามตำแหน่งยานพาหนะแบบเรียลไทม์

1.2.2 เพื่อเสริมสร้างความปลอดภัยให้กับยานพาหนะและผู้โดยสาร

1.2.3 เพื่อเก็บข้อมูลประวัติการเดินทางของยานพาหนะ

1.2.4 เพื่อรองรับการเติบโตของเทคโนโลยี IoT

**1.3 ขอบเขต**

1.3.1 เทคโนโลยีที่ใช้ในการพัฒนาระบบ

1.3.1.1 การออกแบบและพัฒนาระบบระบบระบบติดตามยานพาหนะโดยใช้ โดยใช้ Arduino IDE ตัวเซนเซอร์ตรวจจับตำแหน่ง GPS Module และส่งข้อมูลไปยังเซิร์ฟเวอร์การทำงาน การพัฒนาแอปพลิเคชั้นและการเชื่อมต่อฐานข้อมูลของ MySQL

1.3.1.2 GPS Module ใช้สำหรับรับตำแหน่งยานพาหนะ

1.3.1.3 IoT Device ใช้สำหรับส่งข้อมูลไปยัง Server

1.3.1.4 ใช้ระบบฐานข้อมูล MySQL

1.3.1.5 การเชื่อมต่อ Wi-Fi และส่งข้อมูลไปยัง Firebase

1.3.2 ฟังก์ชันการทำงานของระบบ

1.3.2.1 ฟังก์ชันการจัดการข้อมูลเบื้องต้นของระบบแจ้งเตือน

- การตั้งค่าการแจ้งเตือน

- การตรวจจับเหตุการณ์ที่ผิดปกติ

- การบันทึกข้อมูลการแจ้งเตือน

- การส่งการแจ้งเตือน

- การจัดการการแจ้งเตือนย้อนหลัง

- การตั้งค่าและปรับแต่งเกณฑ์การแจ้งเตือน

- การควบคุมการตอบสนองหลังจากการแจ้งเตือน

1.3.2.2 ฟังก์ชันการทำงานส่วนของลูกค้าหรือบุคคลภายนอกองค์กร

- การลงทะเบียนและการเข้าสู่ระบบ (Authentication & Registration)

- การดูตำแหน่งยานพาหนะแบบเรียลไทม์ (Real-time Vehicle Tracking)

- การตั้งค่าและปรับแต่งการแจ้งเตือน (Alert & Notification Settings)

- การดูประวัติการเดินทาง (Trip History)

- การค้นหายานพาหนะ (Vehicle Search)

- การตั้งค่าบัญชีผู้ใช้งาน (Account Settings)

1.3.2.3 ฟังก์ชันการทำงานส่วนของบุคลากรหรือเจ้าหน้าที่ขององค์กร

- ส่วนของผู้ดูแลระบบ เป็นส่วนที่ให้ผู้ดูแลระบบดูแลบริหารจัดการข้อมูลต่าง ๆ ของระบบได้ เจ้าหน้าที่มารถรับการแจ้งเตือนเกี่ยวกับระบบจากแหลงข้อมูลอื่นหรือตรวจสอบความถูกต้องและความน่าเชื่อถือและเจ้าหน้าที่สามารถใช้ข้อมูลที่มีอยู่ในการตัดสินใจในการตอบสนองรวมถึงการส่งข้อความแจ้งเตือนภายในองค์กรณ์

- ส่วนของผู้ใช้งานระบบ ผู้ใช้จะได้รับการแจ้งเตือนเหตุการณ์และตำแหน่งในพื้นที่ที่ผู้ใช้ตั้งค่าไว้และระบบสามารถส่งการแจ้งเตือนตามตำแหน่งที่ผู้ใช้ตั้งค่ารวมถึงการแสดงข้อมูลสถานการณ์ที่เกิดขึ้น

**1.4 ขั้นตอนการดำเนินงาน**

1.4.1 การวางแผนและกำหนดขอบเขตของระบบ

1.4.2 การศึกษาและออกแบบระบบ

1.4.3 การพัฒนาและติดตั้งฮาร์ดแวร์

1.4.4 การพัฒนาซอฟต์แวร์

1.4.5 การทดสอบระบบ

1.4.6 การปรับปรุงและพัฒนาเพิ่มเติม

1.4.7 การส่งมอบและฝึกอบรมผู้ใช้งาน

1.4.8 การดูแลและบำรุงรักษาระบบ

**1.5 ประโยชน์และผลที่คาดว่าจะได้รับ**

1.5.1 สามารถติดตามยานพาหนะแบบเรียลไทม์

1.5.2 สามารถเพิ่มประสิทธิภาพในการจัดการยานพาหนะ

1.5.3 สนับสนุนการพัฒนาเทคโนโลยี IoT และ GPS

1.5.4 สามารถตรวจสอบตำแหน่งของยานพาหนะได้ผ่านแอปพลิเคชันหรือเว็บไซต์

**บทที่ 2**

**ทฤษฎี งานวิจัย และเครื่องมือที่เกี่ยวข้อง**

การศึกษาระบบ ติดตามยานพาหนะด้วย GPS โดยระบบ Iot ผู้ศึกษาได้ศึกษาแนวคิด ทฤษฎี และงานวิจัย ที่เกี่ยวข้อง ดังมีรายละเอียดต่อไปนี้

**2.1 ทฤษฎีที่เกี่ยวข้อง**

2.1.1 ทฤษฎีระบบกำหนดตำแหน่งบนโลก (Global Positioning System - GPS Theory) GPS เป็นระบบนำทางที่ใช้ดาวเทียมสำหรับกำหนดตำแหน่งบนพื้นโลก โดยมีทฤษฎีพื้นฐาน หลักการทำงานของ GPSดาวเทียม GPS ส่งสัญญาณไปยังผู้รับ (Receiver) โดยสัญญาณดังกล่าวประกอบด้วยข้อมูลตำแหน่งของดาวเทียมและเวลาในการส่งสัญญาณเครื่องรับ GPS จะคำนวณระยะทางจากดาวเทียมโดยใช้เวลาในการเดินทางของสัญญาณ (Time of Flight)การคำนวณตำแหน่งต้องอาศัยดาวเทียมอย่างน้อย 4 ดวง (Trilateration)ข้อดีและข้อจำกัดของ GPS ข้อดี มีความแม่นยำสูง, ครอบคลุมทั่วโลกข้อจำกัด สัญญาณอาจถูกรบกวนในพื้นที่ที่มีสิ่งกีดขวาง เช่น อาคารสูง

2.1.2 ทฤษฎีอินเทอร์เน็ตของสรรพสิ่ง (Internet of Things - IoT Theory) IoT เป็นแนวคิดที่เชื่อมโยงอุปกรณ์ต่าง ๆ เข้าด้วยกันผ่านอินเทอร์เน็ตเพื่อแลกเปลี่ยนข้อมูล องค์ประกอบของ IoT เซ็นเซอร์ (Sensors) ตรวจจับข้อมูลจากสิ่งแวดล้อม เช่น ตำแหน่ง, อุณหภูมิ การสื่อสาร (Communication) ใช้โปรโตคอล เช่น MQTT, HTTP ในการแลกเปลี่ยนข้อมูล การประมวลผล (Processing) ประมวลผลข้อมูลที่ได้รับจากเซ็นเซอร์เพื่อใช้ในการวิเคราะห์ ระบบคลาวด์ (Cloud Computing) จัดเก็บและวิเคราะห์ข้อมูลขนาดใหญ่ แนวคิดพื้นฐานของ IoT เน้นการทำงานร่วมกันของอุปกรณ์ที่หลากหลาย เพื่อเพิ่มความสะดวกและลดความซับซ้อนในการจัดการระบบ

2.1.3 ทฤษฎีระบบสารสนเทศภูมิศาสตร์ (Geographic Information System - GIS Theory) GIS เป็นระบบที่ใช้สำหรับเก็บ จัดการ และแสดงผลข้อมูลเชิงพื้นที่ หลักการของ GIS ใช้ข้อมูลเชิงพื้นที่ (Spatial Data) เช่น ตำแหน่ง GPS เพื่อวิเคราะห์และสร้างแผนที่ การแสดงผลตำแหน่งยานพาหนะบนแผนที่ การคำนวณเส้นทางที่เหมาะสม การใช้งานในระบบติดตามยานพาหนะแสดงตำแหน่งเรียลไทม์วางแผนเส้นทางเพื่อเพิ่มประสิทธิภาพในการเดินทาง

2.1.4 ทฤษฎีการจัดการข้อมูล (Data Management Theory) การจัดการข้อมูลเป็นหัวใจสำคัญในการพัฒนาระบบที่สามารถเก็บ วิเคราะห์ และแสดงผลข้อมูล กระบวนการจัดการข้อมูล การเก็บข้อมูลจากเซ็นเซอร์ การประมวลผลและจัดเก็บในฐานข้อมูล การแสดงผลผ่านแดชบอร์ดหรือแอปพลิเคชัน แนวทางในการจัดการข้อมูล การใช้ฐานข้อมูลเชิงสัมพันธ์ (Relational Database) เช่น MySQL การใช้ฐานข้อมูล NoSQL เช่น MongoDB สำหรับข้อมูลที่ไม่มีโครงสร้าง

**2.2 งานวิจัยที่เกี่ยวข้อง**

2.2.1 ระบบติดตามตำแหน่ง (GPS Tracking System) GPS tracking คือ การระบุตำแหน่งของวัตถุผ่านระบบระบุตำแหน่งบนพื้นโลก (Global Positioning System: GPS) ซึ่งใช้เพื่อติดตามและระบุตำแหน่งของวัตถุนั้นๆจากระยะไกล โดยเทคโนโลยี GPS tracking นี้สามารระบุได้ครอบคลุมถึงพิกัดภูมิศาสตร์ [ละติจูด, ลองจิจูด](https://www.prosoftgps.com/Article/Detail/72143), ความเร็วบนภาคพื้น ทิศทางและเส้นทางการเคลื่อนที่ของวัตถุนั้นๆ ที่เราติดตามอยู่ได้

[จีพีเอส (GPS)](https://www.prosoftgps.com/Article/Detail/70661) เป็นระบบที่มีกลุ่ม[ดาวเทียม](https://www.prosoftgps.com/Article/Detail/71773) 24 ดวงโคจรรอบโลกและมีระยะห่างระหว่างดาวเทียมแต่ละดวงเท่ากัน โดยส่งสัญญาณระบุพิกัดมายังโลกเพื่อให้ผู้คนที่ภาคพื้นดินและมีเครื่องรับสัญญาณสามารถที่จะระบุตำแหน่งทางภูมิศาสตร์ได้ ซึ่งความแม่นยำของตำแหน่งจะคลาดเคลื่อนเพียง 10 ถึง 100 เมตร ขึ้นอยู่กับคุณภาพของอุปกรณ์รับสัญญาณ และอาจจะคาดเคลื่อนได้เพียง 1 เมตร ถ้าเป็นอุปกรณ์รับสัญญาณที่มีความพิเศษ เช่น อุปกรณ์ที่ใช้ในกองทัพ

อุปกรณ์รับสัญญาณจีพีเอส โดยทั่วไปนั้นจะใช้สำหรับงานทางด้านวิทยาศาสตร์เป็นหลัก แต่เนื่องด้วยปัจจุบันต้นทุนในการผลิตอุปกรณ์รับสัญญาณจีพีเอสนั้นถูกลง ทำให้คนทุกๆกลุ่มสามารถเข้าถึงอุปกรณ์นี้ได้ ซึ่งตัวรับสัญญาณจีพีเอสนอกเหนือจากที่จะติดตั้งไว้ในอุปกรณ์สำหรับนำทางแล้ว ยังได้ติดตั้งไว้เป็นอุปกรณ์พื้นฐานในโทรศัพท์เคลื่อนที่และแท็บเล็ตอีกด้วย

GPS tracking เป็นอุปกรณ์ที่มีความจำเป็นอย่างยิ่งสำหรับเจ้าหน้าที่ตำรวจ, เจ้าหน้าที่ดับเพลิง, การใช้งานในกองทัพและธุรกิจเกี่ยวกับการขนส่งสินค้า ซึ่งจากหน่วยงานที่กล่าวมาจะใช้ระบบการติดตามตำแหน่งรถยนต์หรือยานพาหนะ[(AVL: Automatic Vehicle Location)](https://www.prosoftgps.com/Article/Detail/70805) ทั้งสิ้น ซึ่งระบบติดตามยานพาหนะหรือที่ติดตั้งในรถยนต์นั้นโดยทั่วไปแล้วจะประกอบไปด้วยอุปกรณ์รับสัญญาณวิทยุหรือสัญญาณโทรศัพท์, อุปกรณ์รับสัญญาณจีพีเอส รวมทั้งเสาอากาศเพื่อรับสัญญาณจีพีเอส โดยระบบเน็ตเวิร์คจะเชื่อมต่อผ่านระบบสัญญาณวิทยุหรือโทรศัพท์ไปยังระบบคอมพิวเตอร์ที่มีหน้าที่แสดงผลตำแหน่งของรถยนต์หรือยานพาหนะนั้นเพื่อให้ทราบว่ารถยนต์หรือยานพาหนะนั้นอยู่ที่ตำแหน่งใด โดยจีพีเอสจะมีระบบการวิเคราะห์และจะไปแสดงตำแหน่งให้สอดคล้องกับแผนที่โลก

ซึ่งระบบติดตามยานพาหนะเป็นอีกระบบหนึ่งที่สามารถใช้เพื่อเพิ่มความรับผิดชอบของบุคลากรและเพิ่มประสิทธิภาพของขั้นตอนการจัดส่งสินค้าหรือบริการต่างๆ ของบริษัท โดยระบบการติดตามรถยนต์หรือยานพาหนะที่เรียกกันว่า GPS tracking จะทำให้การบริหารและจัดการงานเหล่านี้มีประสิทธิภาพ ทั้งยังลดรายจ่ายที่ไม่จำเป็นออกไปได้อีกด้วย

2.2.2 ระบบสารสนเทศภูมิศาสตร์ (GIS) ระบบสารสนเทศภูมิศาสตร์ หรือ Geographic Information System : GIS คือกระบวนการทำงานเกี่ยวกับข้อมูลในเชิงพื้นที่ด้วยระบบคอมพิวเตอร์ ที่ใช้กำหนดข้อมูลและสารสนเทศ ที่มีความสัมพันธ์กับตำแหน่งในเชิงพื้นที่ เช่น ที่อยู่ บ้านเลขที่ สัมพันธ์กับตำแหน่งในแผนที่ ตำแหน่ง เส้นรุ้ง เส้นแวง ข้อมูลและแผนที่ใน GIS เป็นระบบข้อมูลสารสนเทศที่อยู่ในรูปของตารางข้อมูล และฐานข้อมูลที่มีส่วนสัมพันธ์กับข้อมูลเชิงพื้นที่ (Spatial Data) ซึ่งรูปแบบและความสัมพันธ์ของข้อมูลเชิงพื้นที่ทั้งหลาย จะสามารถนำมาวิเคราะห์ด้วย GIS และทำให้สื่อความหมายในเรื่องการเปลี่ยนแปลงที่สัมพันธ์กับเวลาได้ เช่น การแพร่ขยายของโรคระบาด การเคลื่อนย้าย ถิ่นฐาน การบุกรุกทำลาย การเปลี่ยนแปลงของการใช้พื้นที่ ฯลฯ ข้อมูลเหล่านี้ เมื่อปรากฏบนแผนที่ทำให้สามารถแปลและสื่อความหมาย ใช้งานได้ง่าย  
GIS เป็นระบบข้อมูลข่าวสารที่เก็บไว้ในคอมพิวเตอร์ แต่สามารถแปลความหมายเชื่อมโยงกับสภาพภูมิศาสตร์อื่นๆ สภาพท้องที่ สภาพการทำงานของระบบสัมพันธ์กับสัดส่วนระยะทางและพื้นที่จริงบนแผนที่ ข้อแตกต่างระหว่าง GIS กับ MIS นั้นสามารถพิจารณาได้จากลักษณะของข้อมูล คือ ข้อมูลที่จัดเก็บใน GIS มีลักษณะเป็นข้อมูลเชิงพื้นที่ (Spatial Data) ที่แสดงในรูปของภาพ (graphic) แผนที่ (map) ที่เชื่อมโยงกับข้อมูลเชิงบรรยาย (Attribute Data) หรือฐานข้อมูล (Database)การเชื่อมโยงข้อมูลทั้งสองประเภทเข้าด้วยกัน จะทำให้ผู้ใช้สามารถที่จะแสดงข้อมูลทั้งสองประเภทได้พร้อมๆ กัน เช่นสามารถจะค้นหาตำแหน่งของจุดตรวจวัดควันดำ - ควันขาวได้โดยการระบุชื่อจุดตรวจ หรือในทางตรงกันข้าม สามารถที่จะสอบถามรายละเอียดของ จุดตรวจจากตำแหน่งที่เลือกขึ้นมา ซึ่งจะต่างจาก MIS ที่แสดง ภาพเพียงอย่างเดียว โดยจะขาดการเชื่อมโยงกับฐานข้อมูลที่เชื่อมโยงกับรูปภาพนั้น เช่นใน CAD (Computer Aid Design) จะเป็นภาพเพียงอย่างเดียว แต่แผนที่ใน GIS จะมีความสัมพันธ์กับตำแหน่งในเชิงพื้นที่ทางภูมิศาสตร์ คือค่าพิกัดที่แน่นอน ข้อมูลใน GIS ทั้งข้อมูลเชิงพื้นที่และข้อมูลเชิงบรรยาย สามารถอ้างอิงถึงตำแหน่งที่มีอยู่จริงบนพื้นโลกได้โดยอาศัยระบบพิกัดทางภูมิศาสตร์ (Geocode) ซึ่งจะสามารถอ้างอิงได้ทั้งทางตรงและทางอ้อม ข้อมูลใน GIS ที่อ้างอิงกับพื้นผิวโลกโดยตรง หมายถึง ข้อมูลที่มีค่าพิกัดหรือมีตำแหน่งจริงบนพื้นโลกหรือในแผนที่ เช่น ตำแหน่งอาคาร ถนน ฯลฯ สำหรับข้อมูล GIS ที่จะอ้างอิงกับข้อมูลบนพื้นโลกได้โดยทางอ้อมได้แก่ ข้อมูลของบ้าน(รวมถึงบ้านเลขที่ ซอย เขต แขวง จังหวัด และรหัสไปรษณีย์) โดยจากข้อมูลที่อยู่ เราสามารถทราบได้ว่าบ้านหลังนี้มีตำแหน่งอยู่ ณ ที่ใดบนพื้นโลก เนื่องจากบ้านทุกหลังจะมีที่อยู่ไม่ซ้ำกัน

2.2.3 การจัดการข้อมูลและการวิเคราะห์ การวิเคราะห์ข้อมูล (Data Analysis) คือ กระบวนการนำข้อมูลมาเรียบเรียง จัดกลุ่ม/แยกประเภทชุดข้อมูล หาความสัมพันธ์ของชุดข้อมูลแต่ละชุดในรูปแบบต่างๆ เพื่อหาความหมาย หรือคำตอบตามเป้าหมาย หรือวัตถุประสงค์ต่างๆ จนได้ออกมาเป็น “ข้อมูลเชิงลึก” (insight) หรือ “ข้อสรุป” (conclusion) ที่ช่วยให้เข้าใจสถานการณ์ หาสาเหตุ ช่วยสนับสนุนการตัดสินใจ ฯลฯ ได้

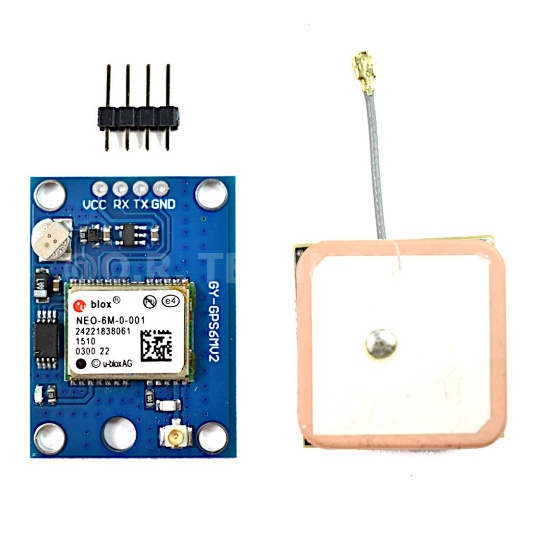
ในปัจจุบัน การวิเคราะห์ข้อมูลถือเป็นสิ่งที่ช่วยให้องค์กร หรือในอุตสาหกรรมสามารถสร้างความได้เปรียบได้ โดยเฉพาะในแง่มุมของธุรกิจและการตลาด ทำให้การวิเคราะห์ข้อมูลมีบทบาทอย่างยิ่งจนเกิดเป็น “วิทยาศาสตร์ข้อมูล” หรือ “Data Science” และการใช้ [“ข้อมูลมหัต” หรือ “Big Data”](https://1stcraft.com/what-is-big-data/) เพื่อให้ธุรกิจและองค์กรได้องค์ความรู้มาใช้ในการดำเนินธุรกิจและขับเคลื่อนองค์กร

**2.3 เครื่องมือที่เกี่ยวข้อง**

การพัฒนาระบบติดตามยานพาหนะด้วย GPS โดยระบบ IoT ต้องใช้เครื่องมือหลากหลายประเภทในทุกขั้นตอน ตั้งแต่การออกแบบฮาร์ดแวร์ ซอฟต์แวร์ ไปจนถึงการจัดการข้อมูลและแสดงผลต่อผู้ใช้งาน เครื่องมือที่เกี่ยวข้องสามารถแบ่งออกได้ดังนี้

2.3.1 เครื่องมือสำหรับฮาร์ดแวร์ (Hardware Tools)

2.3.1.1 โมดูล GPS Ublox NEO-6M



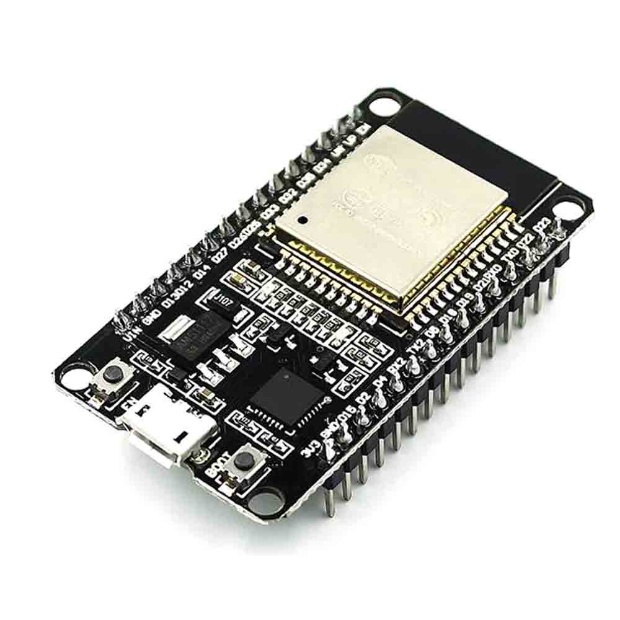
**ภาพที่ 2.1** โมดูล GPS

เป็นโมดูล U-blox รุ่น NEO-6M ซึ่งเป็นที่นิยมกันอย่างมาก สามารถเชื่อมต่อได้กับไมโครคอนโทรเลอร์หลายหลายประเภทไม่ว่าจะเป็น Arduino, NodeMCU, Raspberry Pi, ESP32 ผ่านทาง Serial UART ความเร็วที่ 9600 สามารถอัพได้และตำแหน่งอัพเดทตลอดทุก 1 วินาทีสามารถตั้งค่าให้เร็วกว่านี้ได้ การทำงานเมื่อตัวโมดูลจับสัญญาณได้จะขึ้นไฟสีเขียวกระพริบ ตัวโมดูลมีแบตเตอรี่เก็บตำแหน่งล่าสุดและคอนฟิกต่างๆความแม่นยำในการระบุตำแหน่ง ประมาณ 2.5 เมตร (CEP)ความเร็วสูงสุดที่รองรับ 500 เมตร/วินาทีอัตราการอัปเดตข้อมูล สูงสุด 5 Hzความสามารถในการจับสัญญาณ Cold Start ประมาณ 27 วินาที Hot Start ประมาณ 1 วินาที

ตัวอย่างการเชื่อมต่อ Ublox NEO-6M กับ Arduino

* VCC เชื่อมต่อกับ 5V ของ Arduino
* GND เชื่อมต่อกับ GND ของ Arduino
* TX (Transmit) เชื่อมต่อกับ RX ของ Arduino
* RX (Receive) เชื่อมต่อกับ TX ของ Arduino (ควรใช้ตัวต้านทานลดแรงดัน หาก Arduino ใช้ 5V)

2.3.1.2บอร์ดควบคุมไมโครคอนโทรลเลอร์



**ภาพที่ 2.2** Arduino UNO R3

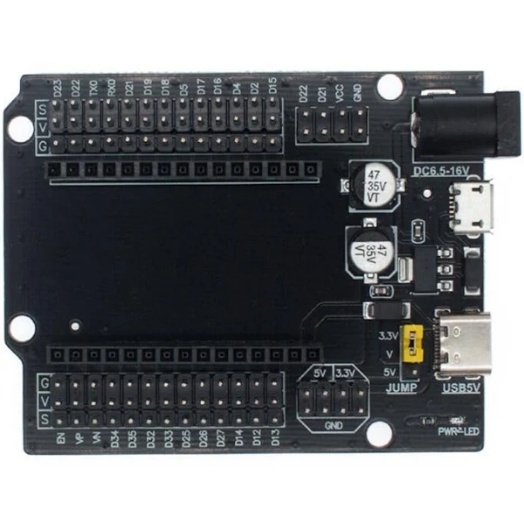
Arduino Unoคือไมโครคอนโทรเลอร์แบบ Open Source คือ เปิดเผยวงจรและวิธีการผลิตทั้งหมด ทุกคนสามารถนำแบบวงจรนี้ไปผลิตหรือต่อยอดได้ภายใต้ข้อกำหนดของ Open Source สามารถใช้โปรแกรม Arduino IDE ในการเขียนโปรแกรมภาษา C ลงบอร์ด ด้วยความง่ายในการเขียนโปรแกรมไม่กี่บรรทัด เสียบสาย USB กับบอร์ดก็อัพโหลดโค้ดลงบอร์ดได้แล้ว บอร์ดมีให้เลือกใช้หลายรุ่นมาก ๆ จึงเหมาะสำหรับงานเกือบทุกชนิด จุดเด่น ง่ายต่อการใช้งานและมีราคาไม่แพง

2.3.1.4 สายไฟจั๊มเปอร์สายไฟจั๊มเปอร์ (Jumper wire)



**ภาพที่ 2.3** สายไฟจั๊มเปอร์

เป็นสายไฟที่ใช้ในการเชื่อมต่ออุปกรณ์อิเล็กทรอนิกส์หรือคอมพิวเตอร์เข้าด้วยกัน โดยมักใช้ในการเชื่อมต่อบอร์ดพัฒนาอย่าง Arduino, Raspberry Pi, หรืออุปกรณ์อื่น ๆ ที่มีขาเชื่อมต่อต่าง ๆ เพื่อทําการเชื่อมต่อสัญญาณไฟฟ้าหรือสัญญาณข้อมูลระหว่างอุปกรณ์ต่าง ๆสายไฟจั๊มเปอร์มักมีลักษณะเป็นสายที่มีหัวปลั๊กสองด้านเพื่อให้ง่ายต่อการต่อและหลีกเลี่ยงการต้องใช้เชื่อมต่อพวงมาลัยหรือตัดสายไฟเอง เคาะแบบเดียวกันหรือหลายเส้นรวมกันในแบบเส้นทึบหรือแบบแผง สายไฟจั๊มเปอร์มักมีความยืดหยุ่นเพียงพอเพื่อให้ง่ายต่อการใช้งานและการเคลื่อนย้ายอุปกรณ์สายไฟจั๊มเปอร์มีความสําคัญในการสร้างโปรเจกต์อิเล็กทรอนิกส์หรือการทดลองเชื่อมต่ออุปกรณ์ต่าง ๆ โดยเฉพาะในงานที่ต้องการการเชื่อมต่อแบบชั่วคราวหรือทดสอบโครงสร้างของวงจรอิเล็กทรอนิกส์อย่างสะดวกและรวดเร็ว

2.3.1.5 ฐานรองบอร์ด

**ภาพที่ 2.4** ฐานรองบอร์ด

ฐานรองบอร์ด (หรือบอร์ดเมาท์) สำหรับ ESP32 นั้นสามารถใช้ได้หลายประเภท ขึ้นอยู่กับการใช้งานและการเชื่อมต่อที่ต้องการ โดยทั่วไปแล้วฐานรองบอร์ด ESP32 จะช่วยให้คุณเชื่อมต่อ ESP32 เข้ากับอุปกรณ์ต่างๆ ได้ง่ายขึ้น และเพิ่มความสะดวกในการโปรแกรม ฐานรองบอร์ด ESP32 มักจะมีคุณสมบัติหลักๆ เช่นช่องเสียบสำหรับบอร์ด ESP32 มีช่องสำหรับยึดบอร์ด ESP32 เพื่อให้ง่ายต่อการเชื่อมต่อกับสายไฟหรือส่วนประกอบอื่นๆขา GPIO ที่สามารถเข้าถึงได้ บางฐานรองบอร์ดมีการจัดเรียงขา GPIO ของ ESP32 ไว้ในลักษณะที่สะดวกต่อการใช้งาน ฟีเจอร์เพิ่มเติม เช่น ปุ่มรีเซ็ต, ปุ่มดาวน์โหลด (สำหรับการโปรแกรมบอร์ด), ช่องต่อพลังงาน, และอุปกรณ์เสริมอื่นๆ

**2.3.2** ซอฟต์แวร์ (Software)

2.3.2.1 Blynk



**ภาพที่ 2.5** โปรแกรม Blynk

เป็นชื่อโดยรวมของการบริการให้ผู้ใช้งานได้ใช้งานเครื่องแม่ข่าย คือ Blynk Server ที่เป็น IoTCloud ซึ่ง ถูกพัฒนามาจากภาษา Java ทําให้สามารถทํางานภายใต้ระบบปฏิบัติการที่หลากหลายเช่น Windows, Mac หรือ Linux โดยเครื่องแม่ข่าย (Blynk Server) พัฒนาเป็นแบบเปิด (open-source) ภายใต้ลิขสิทธิ์แบบ GNU ทําให้เราสามารถนํา Blynk ไปใช้งานประกอบการสร้างนวัตกรรมเพื่อการค้า แก้ไข ดัดแปลงเผยแพร่ หรือแจกจ่ายได้

Blynk App คือ แอปพลิเคชันสําเร็จรูปที่ใช้สําหรับงานที่เกี่ยวกับอินเทอร์เน็ตของสรรพสิ่ง (Internet ofThings, IoT) ที่ทําให้เราสามารถเชื่อมต่ออุปกรณ์ต่าง ๆ เข้ากับอินเทอร์เน็ตในลักษณะการเชื่อมต่อเครื่องแม่ข่าย(Server) ไปยังอุปกรณ์ลูกข่าย (Client) เช่น Arduino, ESP-8266, ESP-32, NodeMCU และ Raspberry Pi ซึ่งแอปพลิเคชัน Blynk สามารถใช้งานได้ฟรีและใช้งานได้ทั้งบนระบบปฏิบัติการ IOS และ Android แสดงภาพรายการอุปกรณ์ต่าง ๆ ที่สามารถเชื่อมต่อ แสดงผล และ/หรือ ควบคุมด้วย Blynk App ได้ โดย เริ่มต้นหลังจากสมัครเข้าใช้งาน เราจะได้รับ “Energy” ซึ่งเปรียบเสมือนเงินในโปรแกรมนี้ ในการเรียกใช้งานอุปกรณ์แต่ละตัว เราจะต้องแลกด้วย “Energy” และหาก “Energy”

2.3.2.2 Google Maps API



**ภาพที่ 2.6** โปรแกรม GPS

Google Maps API เป็นชุด API ของ Google สำหรับพัฒนา web application และ mobile application (Android, iOS)ไว้สำหรับเรียกใช้แผนที่และชุด service ต่าง ๆ ของ Google เพื่อพัฒนา Application ได้เหมือนกับที่ Google โดยแผนที่ยัง features ต่าง ๆ มากมายให้เรียกใช้

- การปรับแต่งแผนที่ (Styled Map)  
- ชุดควบคุมแผนที่ (Map Control)  
- ชุดเครื่องมือวาดภาพบนแผนที่ (Drawing)  
- การนำทางจากจุดหนึ่งไปยังอีกจุดหนึ่ง (Directions Service)  
- การคำนวณความสูงของจุดพิกัด (Elevation Service)  
- การแปลงที่อยู่เป็นพิกัด Lattitude เเละ Longtitude (GeoCoding Service)  
- การดึงข้อมูล POI (Point of Interest) คือข้อมูลสถานที่ต่าง ๆ ที่ Google รวบรวมไว้ให้ เช่น โรงแรม ห้างสรรพสินค้า โรงเรียน -สถานที่ราชการต่างๆ เเละอื่นๆ อีกมากมาย (Places API) มาใช้งานใน   application เรา  
- Street View

2.3.3 ฐานข้อมูล (Database)

2.3.3.1 ฐานข้อมูลเชิงสัมพันธ์ MySQL



**ภาพที่ 2.6** โปรแกรม MySQL

คือ ระบบจัดการฐานข้อมูล หรือ Database Management System (DBMS) แบบข้อมูลเชิงสัมพันธ์ หรือ Relational Database Management System (RDBMS) ซึ่งเป็นระบบฐานข้อมูลที่จัดเก็บรวบรวมข้อมูลในรูปแบบตาราง โดยมีการแบ่งข้อมูลออกเป็นแถว (Row) และในแต่ละแถวแบ่งออกเป็นคอลัมน์ (Column) เพื่อเชื่อมโยงระหว่างข้อมูลในตารางกับข้อมูลในคอลัมน์ที่กำหนด แทนการเก็บข้อมูลที่แยกออกจากกัน โดยไม่มีความเชื่อมโยงกัน ซึ่งประกอบด้วยข้อมูล (Attribute) ที่มีความสัมพันธ์เชื่อมโยงกัน (Relation) โดยใช้ RDBMS Tools สำหรับการควบคุมและจัดเก็บฐานข้อมูลที่จำเป็น ทำให้นำไปประยุกต์ใช้งานได้ง่าย ช่วยเพิ่มประสิทธิภาพในการทำงานให้มีความยืดหยุ่นและรวดเร็วได้มากยิ่งขึ้น รวมถึงเชื่อมโยงข้อมูล ที่จัดแบ่งกลุ่มข้อมูลแต่ละประเภทได้ตามต้องการ จึงทำให้ MySQL เป็นโปรแกรมระบบจัดฐานข้อมูลที่ได้รับความนิยมสูง

**บทที่ 3**

**การวิเคราะห์และออกแบบระบบ**

**3.1 ระบบงาน**

3.1.1 กลุ่มผู้ใช้งาน

3.1.1.1 หน่วยงานภาครัฐและองค์กรที่เกี่ยวข้อง

3.1.1.2 ชุมชนและประชาชนทั่วไป

3.1.1.3 ผู้ประกอบการและอุตสาหกรรม

3.1.1.4 นักพัฒนาและนักวิจัย

3.1.2 ความต้องการ

3.1.2.1 ความต้องการที่กำหนดหน้าที่ (Functional Requirements)

- การตรวจจับตำแหน่งของยาพาหนะ

- การตำแหน่งแบบเรียลไทม์

- ระบบการแจ้งเตือน

- การเก็บข้อมูล

3.1.2.2 ความต้องการที่ไม่กำหนดหน้าที่ (Non-Functional Requirements)

- ความน่าเชื่อถือ

- ความเร็วในการตอบสนอง

- ความปลอดภัย

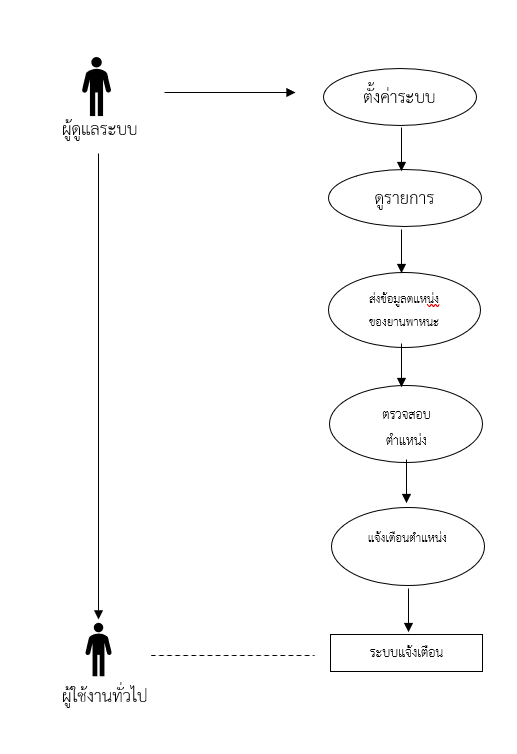
- ความทนทาน

- ประสิทธิภาพ

- ต้นทุน

**3.2 การออกแบบระบบงาน**

3.2.1 แผนภาพยูสเคส (Use Case Diagram)



**ภาพที่ 3.1** แผนภาพยูสเคส

3.2.2 คำอธิบายยูสเคส (Use Case Description)

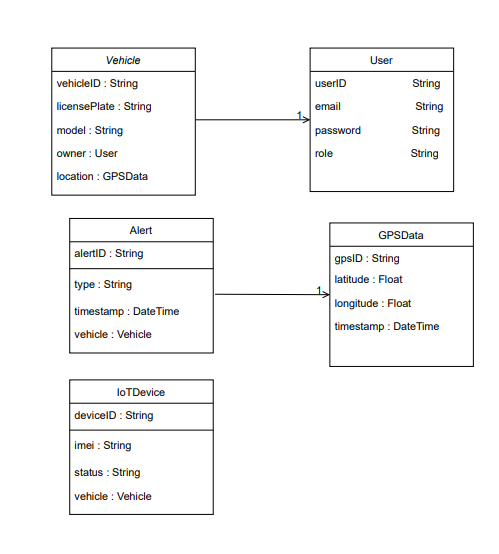
ผู้ใช้งาน (Actors)

* เจ้าของยานพาหนะ (Vehicle Owner) ผู้ที่ต้องการติดตามสถานะและตำแหน่งของยานพาหนะ
* ระบบ IoT (IoT System) ระบบที่รับข้อมูลจาก GPS และส่งข้อมูลให้ผู้ใช้งานGPS Tracker
* - อุปกรณ์ที่ติดตั้งในยานพาหนะเพื่อรับข้อมูลตำแหน่ง (Latitude, Longitude)เซิร์ฟเวอร์ (Server)
* - เก็บข้อมูลและประมวลผลข้อมูลจาก GPS

Use Cases (กรณีการใช้งาน)

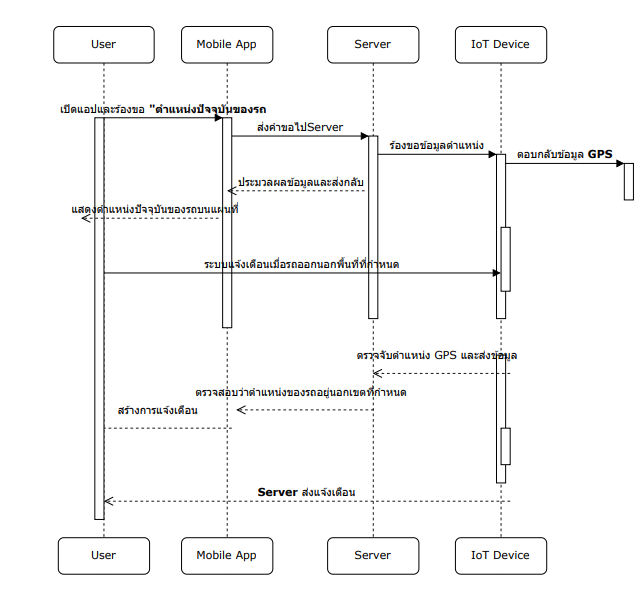
* ติดตามตำแหน่งยานพาหนะ (Track Vehicle Location) ดูตำแหน่งปัจจุบันของรถ
* แจ้งเตือนเมื่อออกนอกพื้นที่ (Geofencing Alert) ระบบแจ้งเตือนเมื่อรถออกนอกพื้นที่ที่กำหนด
* ดูประวัติการเดินทาง (View Travel History) – ดูเส้นทางที่ยานพาหนะเคยเดินทาง
* ตั้งค่าการแจ้งเตือน (Set Notifications) – ตั้งค่าให้ระบบแจ้งเตือนเมื่อรถเคลื่อนที่หรือหยุด
* รายงานสถานะของยานพาหนะ (Vehicle Status Report) – ดูสถานะแบตเตอรี่หรือเซ็นเซอร์ต่างๆ
* จัดการผู้ใช้และอุปกรณ์ (Manage Users & Devices) – ผู้ดูแลระบบสามารถเพิ่ม/ลบผู้ใช้ และจัดการอุปกรณ์ IoT

3.2.3 แผนภาพคลาส (Class Diagram)



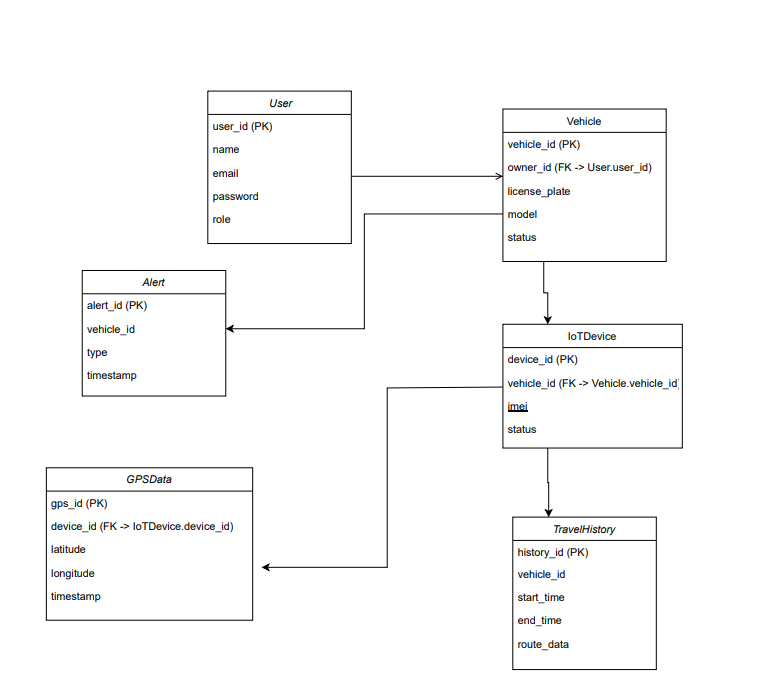
**ภาพที่ 3.2** แผนภาพคลาส

3.2.4 แผนภาพลำดับเหตุการณ์ (Sequence Diagram)



**ภาพที่ 3.4** ลำดับเหตุการณ์

3.2.5 แผนภาพความสัมพันธ์ของตาราง



**ภาพที่ 3.5** ความสัมพันธ์ของตาราง

**บทที่ 4**

**การออกแบบและเครื่องมือที่ใช้ในการพัฒนาระบบ**

**4.1 การออกแบบระบบ**

4.1.1 ส่วนตรวจวัดข้อมูล (Sensing & Data Collection)

4.1.1.1 อุปกรณ์ที่ใช้ใน

- GPS Tracker – ใช้ในการตรวจวัดตำแหน่งของยานพาหนะ (ละติจูด, ลองจิจูด)

- ความเร็ว (Speed Sensor) – ใช้ในการวัดความเร็วของยานพาหนะ

- เซ็นเซอร์การเคลื่อนไหว (Motion Sensor) – ตรวจจับการเคลื่อนไหวของยานพาหนะ (ช่วยในการติดตามการหยุดนิ่ง)

- เซ็นเซอร์อุณหภูมิ (Temperature Sensor) – ใช้ในการตรวจวัดอุณหภูมิในรถ เช่น สำหรับการขนส่งสินค้าที่ต้องการควบคุมอุณหภูมิ

- เซ็นเซอร์แบตเตอรี่ (Battery Sensor) – ใช้ในการตรวจสอบสถานะแบตเตอรี่ของอุปกรณ์ GPS หรือของยานพาหนะ

4.1.1.2 การเก็บรวบรวมข้อมูล

ข้อมูลที่ถูกเก็บรวม ได้แก่

* ตำแหน่ง GPS (Latitude, Longitude)
* ความเร็วของยานพาหนะ
* สถานะการเคลื่อนไหว (เคลื่อนที่/หยุดนิ่ง)
* สถานะแบตเตอรี่ของอุปกรณ์ GPS
* สถานะของเครื่องยนต์ (ON/OFF)
* อุณหภูมิในยานพาหนะ

4.1.2 ส่วนประมวลผลและเก็บข้อมูล (Processing & Data Storage)

- การรับข้อมูล (Data Reception) ข้อมูลจากอุปกรณ์ GPS หรือเซ็นเซอร์จะถูกส่งมายัง Gateway หรือ Cloud Server ผ่าน เครือข่ายต่าง ๆ เช่น GSM, Wi-Fi หรือ LoRa (ขึ้นอยู่กับประเภทของการเชื่อมต่อ) ข้อมูลที่ส่งมาจะมีลักษณะเป็น ข้อความ JSON หรือ XML เพื่อให้สามารถส่งข้อมูลได้หลายประเภท เช่น ตำแหน่ง GPS, ความเร็ว, สถานะเครื่องยนต์

- การตรวจสอบข้อมูล (Data Validation) ระบบต้องทำการ ตรวจสอบความสมบูรณ์ ของข้อมูลที่ได้รับจากอุปกรณ์ เช่น การตรวจสอบว่า ตำแหน่ง GPS ไม่ผิดปกติ หรือ ข้อมูลจากเซ็นเซอร์ (เช่น ความเร็ว) มีค่าที่เป็นไปได้ หากข้อมูลไม่ถูกต้อง ระบบจะทำการ ส่งสัญญาณเตือน เพื่อแจ้งว่าเกิดข้อผิดพลาดหรืออุปกรณ์มีปัญหา

- การประมวลผลข้อมูล (Data Processing)ระบบจะทำการ คำนวณ หรือ ประมวลผล ข้อมูลจากเซ็นเซอร์ต่าง ๆ เพื่อให้ได้ข้อมูลที่มีประโยชน์ เช่นคำนวณเส้นทางการเคลื่อนที่ จากตำแหน่ง GPS ที่เก็บไว้ การคำนวณความเร็ว จากการเปลี่ยนแปลงตำแหน่งตามเวลา การตรวจจับการออกนอกพื้นที่ (Geo-fencing) โดยการเปรียบเทียบตำแหน่ง GPS กับ เขตพื้นที่ที่กำหนด การแจ้งเตือน หากมีพฤติกรรมการขับขี่ที่ไม่เหมาะสม เช่น ขับเกินความเร็ว, หยุดนิ่งนานเกินไป ฯลฯ

- การตัดสินใจและการแจ้งเตือน (Decision Making & Alerting)เมื่อข้อมูลถูกประมวลผลแล้ว ระบบจะสามารถ สร้างการแจ้งเตือน (alert) เมื่อพบเหตุการณ์ที่ไม่ปกติ เช่น รถออกนอกพื้นที่ (Geo-fencing Alert) เครื่องยนต์หยุดทำงาน หรือ แบตเตอรี่ต่ำ พฤติกรรมการขับขี่ที่ไม่ปลอดภัย เช่น ความเร็วเกินกว่าที่กำหนด การแจ้งเตือนจะถูกส่งไปยัง แอปมือถือ หรือ อีเมล ของผู้ใช้งาน

4.1.3 ส่วนแจ้งเตือนและแสดงผล (Notification & Visualization)

ชนิดของการแจ้งเตือน (Types of Alerts)

* Geo-fencing Alerts แจ้งเตือนเมื่อยานพาหนะออกจากพื้นที่ที่กำหนดไว้ ตัวอย่างหากยานพาหนะขับออกจากเขตที่กำหนด เช่น บริเวณลานจอดรถหรือพื้นที่ที่ห้ามเข้า
* Speeding Alerts แจ้งเตือนเมื่อยานพาหนะขับขี่เกินความเร็วที่กำหนด ตัวอย่าง หากรถขับเกินความเร็วที่ปลอดภัย เช่น 120 กม./ชม.
* Stationary Alerts แจ้งเตือนเมื่อยานพาหนะหยุดนิ่งนานเกินไป ตัวอย่าง หากรถหยุดอยู่ในจุดเดิมเกินระยะเวลาที่กำหนด เช่น หยุดอยู่ในที่เดียวเกิน 30 นาที
* Battery Alerts แจ้งเตือนเมื่อระดับพลังงานของอุปกรณ์ GPS หรือแบตเตอรี่ของรถต่ำ ตัวอย่าง หากแบตเตอรี่ของอุปกรณ์ GPS ต่ำกว่า 20%
* Engine Status Alerts แจ้งเตือนเมื่อเครื่องยนต์ของยานพาหนะมีปัญหาหรือหยุดทำงาน ตัวอย่างหากเครื่องยนต์ปิดขณะขับขี่ หรือมีการสตาร์ทเครื่องยนต์ที่ผิดปกติ
* Temperature Alerts: แจ้งเตือนเมื่ออุณหภูมิในยานพาหนะสูงหรือต่ำเกินไปตัวอย่าง หากอุณหภูมิภายในรถสูงเกิน 30 องศาเซลเซียส สำหรับการขนส่งสินค้าที่ต้องการควบคุมอุณหภูมิ

ช่องทางการแจ้งเตือน (Alert Channels)

* Mobile App Notification: การแจ้งเตือนผ่านแอปมือถือที่ผู้ใช้ติดตั้ง
* Email Notification: การส่งอีเมลแจ้งเตือน
* SMS Notification: การส่งข้อความแจ้งเตือนผ่าน SMS
* Push Notification: การแจ้งเตือนทันทีบนหน้าจอมือถือหรือเว็บไซต์เมื่อมีเหตุการณ์ที่สำคัญ
* Web Dashboard Notification: การแสดงการแจ้งเตือนบนแดชบอร์ดออนไลน์ที่ผู้ใช้สามารถดูสถานะการติดตามได้แบบเรียลไทม์

**4.2 การออกแบบส่วนต่อประสานผู้ใช้**

* + 1. แอปพลิเคชันมือถือ (Mobile Application)

เครื่องมือที่ใช้

* Flutter + Firebase พัฒนาแอปสำหรับ Android และ iOS 2)
* Blynk หรือ MIT App Inventor → ใช้สร้างแอปควบคุม ESP32

คุณสมบัติของแอป

* แสดงตำแหน่งเรียลไทม์
* แจ้งเตือนผ่านเว็บ
* รองรับการแจ้งเตือน
* ดูประวัติของตำแหน่งย้อนหลัง
  + 1. การแจ้งเตือนผ่าน Blynk
* Blynk แสดงทำแหน่งบนหนแว็บ

**4.3 เครื่องมือที่ใช้ในการพัฒนาระบบ**

4.3.1 ฮาร์ดแวร์ที่ใช้ในการพัฒนาระบบ

|  |  |
| --- | --- |
| **อุปกรณ์** | **หน้าที่** |
| ไมโครคอนโทรลเลอร์ (Microcontroller) | 1. ทำหน้าที่รับข้อมูลจากเซ็นเซอร์ต่าง ๆ และทำการประมวลผล 2. ควบคุมการสื่อสารระหว่างอุปกรณ์ เช่น การส่งข้อมูลไปยังคลาวด์ผ่าน Wi-Fi หรือ GSM 3. ประมวลผลข้อมูลตำแหน่ง GPS และส่งต่อให้ผู้ใช้ |
| โมดูล GPS (GPS Module) | 1. ใช้สำหรับระบุตำแหน่งของยานพาหนะ (Latitude, Longitude) 2. สามารถตรวจจับความเร็วของยานพาหนะ 3. ส่งข้อมูลพิกัดให้กับไมโครคอนโทรลเลอร์เพื่อนำไปประมวลผล |
| โมดูลวัดความเร็ว | 1. ตรวจจับความเร็วของยานพาหนะโดยอาศัยแม่เหล็กหรือเซ็นเซอร์อ่านค่าจากล้อหรือเครื่องยนต์ 2. ใช้ร่วมกับ OBD-II เพื่ออ่านค่าความเร็วจาก ECU ของรถ |
| เซ็นเซอร์ตรวจจับการเคลื่อนไหว | 1. ตรวจจับการเคลื่อนไหวของยานพาหนะ (กำลังเคลื่อนที่ หรือ หยุดนิ่ง) 2. ตรวจจับแรงกระแทก กรณีที่เกิดอุบัติเหตุ 3. ใช้สำหรับวิเคราะห์พฤติกรรมการขับขี่ เช่น การเร่งกะทันหัน การเบรกแรง |

**ตารางที่ 4.1** เครื่องมือที่ใช้ในการพัฒนาระบบ

4.3.2 ซอฟต์แวร์และเครื่องมือที่ใช้ในการพัฒนา

|  |  |
| --- | --- |
| **เครื่องมือ** | **หน้าที่** |
| Arduino IDE | 1. รองรับไลบรารีที่เกี่ยวข้องกับ GPS, GSM, และเซ็นเซอร์ต่าง ๆ 2. มีตัวเลือก Serial Monitor เพื่อตรวจสอบข้อมูลจากอุปกรณ์ |
| MicroPython & Thonny IDE | การเขียนโค้ดด้วยภาษา **Python** บนไมโครคอนโทรลเลอร์ เช่น ESP32 |
| Firebas | เก็บและเรียกใช้ข้อมูลจากอุปกรณ์ IoT แบบเรียลไทม์ |
| Google Maps API / Leaflet.js | แสดงตำแหน่ง GPS บนแผนที่ |
| Node.js + Express.js | พัฒนา Backend API เพื่อเชื่อมต่อข้อมูล IoT |
| PostgreSQL / MySQL / MongoDB | เก็บข้อมูลตำแหน่ง GPS, สถานะของรถ, การแจ้งเตือน |

**ตารางที่ 4.2** ซอฟต์แวร์ที่ใช้ในการพัฒนาระบบ