****

**ระบบติดตามยานพาหนะด้วย GPS โดยระบบ Iot**

**Development of GPS vehicle tracking system using Iot system**

**โดย**

**นายภัคพล โสตทิพย์**

**รหัสนักศึกษา 6504305001322**

**อาจารย์ประจำรายวิชา**

**ดร. สมพงษ์ ยิ่งเมือง**

**รายงานเล่มนี้เป็นส่วนหนึ่งของรายวิชา SCS0216 วิศวกรรมซอฟต์แวร์**

**หลักสูตรวิทยาศาสตร์บัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์ มหาวิทยาลัยมหาวิทยาลัยราชภัฏสุราษฎร์ธานี ประจำภาคการศึกษาที่ 2 ปีการศึกษา 2567**

**ระบบติดตามยานพาหนะด้วย GPS โดยระบบ Iot**

**Development of GPS vehicle tracking system using Iot system**

**โดย**

**นายภัคพล โสตทิพย์**

**รหัสนักศึกษา 6504305001322**

**อาจารย์ประจำรายวิชา**

**ดร. สมพงษ์ ยิ่งเมือง**

**รายงานเล่มนี้เป็นส่วนหนึ่งของรายวิชา SCS0216 วิศวกรรมซอฟต์แวร์**

**หลักสูตรวิทยาศาสตร์บัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์ มหาวิทยาลัยมหาวิทยาลัยราชภัฏสุราษฎร์ธานี ประจำภาคการศึกษาที่ 2 ปีการศึกษา 2567**

**คำนำ**

การพัฒนาระบบติดตามยานพาหนะด้วย GPS โดยระบบ IoT (Internet of Things) มีความสำคัญอย่างยิ่งในยุคปัจจุบันที่เทคโนโลยีได้มีบทบาทสำคัญในทุกๆ ด้านของชีวิตประจำวัน ระบบติดตามยานพาหนะช่วยเพิ่มความปลอดภัยและความสะดวกสบายในการบริหารจัดการยานพาหนะ โดยเฉพาะในการติดตามตำแหน่งและการตรวจสอบการเคลื่อนที่ในเวลาใกล้เคียงกับเวลาจริง โครงงานนี้มีจุดมุ่งหมายในการพัฒนาและทดสอบระบบติดตามยานพาหนะที่ใช้เทคโนโลยี GPS และ IoT เพื่อให้สามารถติดตามตำแหน่งของยานพาหนะได้ในทุกที่ทุกเวลา รวมถึงการแจ้งเตือนในกรณีที่ยานพาหนะออกจากพื้นที่ที่กำหนดไว้ (Geofencing) หรือเมื่อเกิดปัญหาต่างๆ ในระบบ ในกระบวนการพัฒนานี้ ระบบได้ถูกทดสอบในหลายๆ ระดับ ทั้งในระดับหน่วยย่อยและระดับระบบ เพื่อให้แน่ใจว่าระบบทำงานได้ตามที่คาดหวังและมีประสิทธิภาพสูงสุด การทดสอบที่สำคัญได้แก่การทดสอบการเชื่อมต่อระหว่าง GPS Module และระบบ IoT, การทดสอบการแสดงผลข้อมูลในแอปพลิเคชัน และการทดสอบระบบแจ้งเตือน

ขอขอบคุณอาจารย์ที่ปรึกษาและทุกท่านที่มีส่วนร่วมในการสนับสนุนโครงงานนี้ ไม่ว่าจะเป็นการให้คำแนะนำทางเทคนิคหรือข้อเสนอแนะในการปรับปรุงระบบ เพื่อให้โครงงานนี้บรรลุผลสำเร็จตามที่ตั้งเป้าหมายไว้.

ภัคพล โสตทิพย์

ผู้จัดทำ

**สารบัญ**

|  |  |  |
| --- | --- | --- |
| **เนื้อหา** |  | **หน้า** |
| คำนำ |  | ก |
| สารบัญ |  | ข |
| สารบัญภาพ |  | ง |
| สารบัญตาราง |  | จ |
| **บทที่1** | **บทนำ** | 1 |
|  | 1.1 ความเป็นมาและความสำคัญ | 1 |
|  | 1.2 วัตถุประสงค์ | 1 |
|  | 1.3 ขอบเขต | 1 |
|  | 1.4 ขั้นตอนการดำเนินงาน | 3 |
|  | 1.5 ประโยชน์และผลที่คาดว่าจะได้รับ | 3 |
| **บทที่ 2** | **ทฤษฎี งานวิจัย และเครื่องมือที่เกี่ยวข้อง** | 4 |
|  | 2.1 ทฤษฎีที่เกี่ยวข้อง | 4 |
|  | 2.2 งานวิจัยที่เกี่ยวข้อง | 5 |
|  | 2.3 เครื่องมือที่เกี่ยวข้อง | 7 |
| **บทที่ 3** | **การวิเคราะห์และออกแบบระบบ** | 14 |
|  | 3.1ระบบงาน | 14 |
|  | 3.2 การออกแบบระบบงาน | 15 |
| **บทที่ 4** | **การออกแบบและเครื่องมือที่ใช้ในการพัฒนาระบบ** | 16 |
|  | 4.1 การออกแบบระบบ | 16 |
|  | 4.2 การออกแบบส่วนต่อประสานผู้ใช้ | 17 |
|  | 4.3 เครื่องมือที่ใช้ในการพัฒนาระบบ | 18 |
| **บทที่ 5** | **การทดสอบ** | 20 |
|  | 5.1 การทดสอบระบบ | 20 |
|  | 5.2 กรณีทดสอบ | 20 |
|  | 5.3 ผลการทดสอบ | 21 |
| **บทที่ 6** | **สรุปและเสนอแนะ** | 22 |
|  | **6.1 สรุปผลการดำเนินงาน** | 22 |
|  | 6.2 ข้อเสนอแนะ | 22 |
|  | 6.3 บทสรุป | 23 |
| **บรรณานุกรม** |  | 24 |

**สารบัญภาพ**

|  |  |  |
| --- | --- | --- |
| **เนื้อหา** |  | **หน้า** |
| ภาพที่ 2.1 | โมดูล GPS | **7** |
| ภาพที่ 2.2 | Arduino UNO R3 | **8** |
| ภาพที่ 2.3 | สายไฟจั๊มเปอร์ | **8** |
| ภาพที่ 2.4 | ฐานรองบอร์ด | **9** |
| ภาพที่ 2.5 | Discord | **9** |
| ภาพที่ 2.6 | Arduino IDE | **10** |
| ภาพที่ 2.7 | My Sql | **11** |
| ภาพที่ 3.1 | แผนภาพยูสเคส | **12** |
| ภาพที่ 3.2 | แผนภาพคลาส | **15** |
| ภาพที่ 3.3 | ลำดับเหตุการณ์ | **16** |
| ภาพที่ 3.4 | ความสัมพันธ์ของตาราง | **17** |

**สารบัญตาราง**

|  |  |  |
| --- | --- | --- |
| **เนื้อหา** |  | **หน้า** |
| **ตารางที่ 4.1** | เครื่องมือที่ใช้ในการพัฒนาระบบ | **20** |
| **ตารางที่ 4.2** | ซอฟต์แวร์ที่ใช้ในการพัฒนาระบบ | **21** |

**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

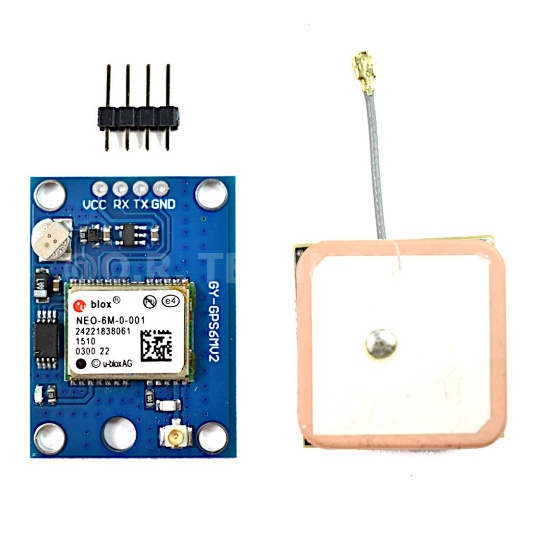
ในปัจจุบัน การวิเคราะห์ข้อมูลถือเป็นสิ่งที่ช่วยให้องค์กร หรือในอุตสาหกรรมสามารถสร้างความได้เปรียบได้ โดยเฉพาะในแง่มุมของธุรกิจและการตลาด ทำให้การวิเคราะห์ข้อมูลมีบทบาทอย่างยิ่งจนเกิดเป็น “วิทยาศาสตร์ข้อมูล” หรือ “Data Science” และการใช้ [“ข้อมูลมหัต” หรือ “Big Data”](https://1stcraft.com/what-is-big-data/)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



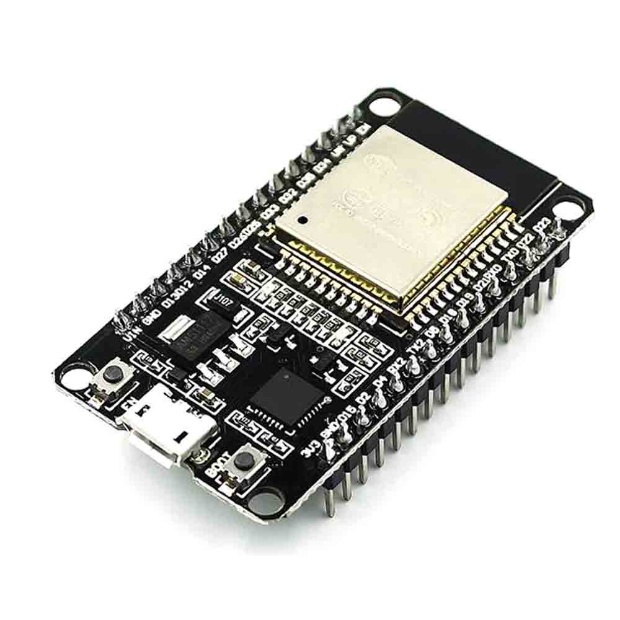
**ภาพที่ 2.1** โมดูล GPS

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

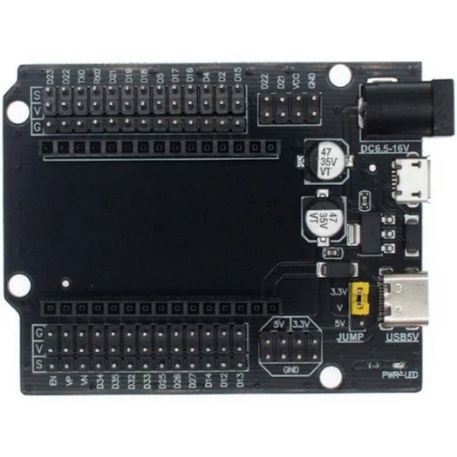
2.3.1.4 สายไฟจั๊มเปอร์สายไฟจั๊มเปอร์ (Jumper wire)



**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 โปรแกรม Discord



**ภาพที่ 2.5** Discord

คือแอปสื่อสารฟรีที่ให้คุณแบ่งปันเสียง วิดีโอ และข้อความกับเพื่อนๆ ชุมชนเกม และนักพัฒนา แอปนี้มีผู้ใช้หลายล้านคน ซึ่งเป็นหนึ่งในวิธีการสื่อสารกับผู้คนออนไลน์ที่เป็นที่นิยมมากที่สุด นอกจากนี้ คุณยังสามารถใช้ Discord บนแพลตฟอร์มและอุปกรณ์ที่เป็นที่นิยมได้เกือบทั้งหมด รวมถึง Windows, macOS, Linux, iOS, iPadOS, Android และเว็บเบราว์เซอร์ การแชทด้วยข้อความผ่าน Discord เซิร์ฟเวอร์ Discord มีช่องข้อความได้หลากหลาย ช่องเหล่านี้มักใช้สำหรับให้ชุมชนถามตอบคำถามต่างๆ แชร์มุกตลกและมีม และพูดคุยกันโดยไม่ต้องใช้ไมโครโฟน เซิร์ฟเวอร์ Discord ใหม่จะมาพร้อมกับช่อง "ทั่วไป" (General) แต่ผู้ดูแลสามารถสร้างช่องได้หลากหลายช่องสำหรับวัตถุประสงค์ต่างๆ ซึ่งอาจรวมถึงการประกาศข่าวสารและกฎของเซิร์ฟเวอร์ หรือการแชทที่เกี่ยวข้องกับเกมใดเกมหนึ่งโดยเฉพาะ เช่น ช่องสำหรับ Fortnite ขึ้นอยู่ที่ว่าผู้ดูแลและผู้ใช้เซิร์ฟเวอร์นั้นจะกำหนดอย่างไร คุณสามารถตั้งช่องข้อความเป็นแบบ "อ่านอย่างเดียว" (read-only) ได้ ซึ่งเป็นความคิดที่ดีสำหรับช่องกฎ ช่องข้อความส่วนมากจะเปิดให้ผู้ใช้สามารถใช้งานได้ โดยไม่มีตัวกรองแชทที่จะบล็อกภาษาที่หยาบคาย อย่างไรก็ตาม ระบบมีตัวกรองที่จะพยายามบล็อกการโพสต์รูปภาพที่ไม่เหมาะสม

2.3.2.2 Arduino IDE

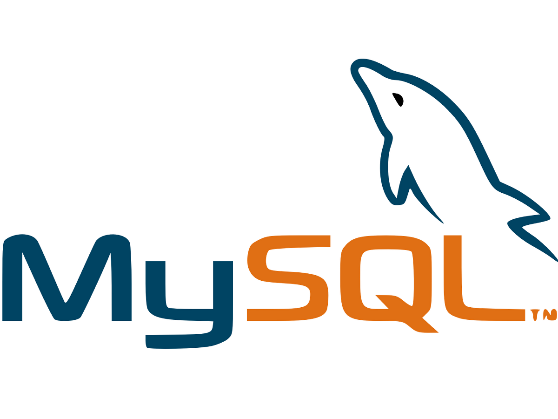


**ภาพที่ 2.6** Arduino IDE

เป็นเครื่องมือสำคัญสำหรับนักประดิษฐ์ นักศึกษาวิศวกรรม อาจารย์ ตลอดจนผู้ที่สนใจในการพัฒนาโครงการอิเล็กทรอนิกส์แบบไมโครคอนโทรลเลอร์ ด้วยความที่ Arduino มีลักษณะเด่นเรื่องความง่ายในการใช้งาน เปิดกว้างให้ผู้คนมากมายเข้าถึงวงจรอิเล็กทรอนิกส์ได้อย่างสะดวกรวดเร็ว และที่สำคัญคือมีคอมมูนิตี้ขนาดใหญ่ที่คอยช่วยกันแบ่งปันความรู้และซอร์สโค้ดไลบรารี่ต่าง ๆ ซึ่งล้วนเอื้ออำนวยให้การเรียนรู้และพัฒนาโปรเจกต์เป็นไปอย่างสนุกและสร้างสรรค์ หลายคนอาจสงสัยว่า “Arduino” คืออะไร คำตอบแบบสรุปสั้น ๆ คือแพลตฟอร์มฮาร์ดแวร์และซอฟต์แวร์โอเพนซอร์ส (Open-source) ซึ่งออกแบบมาเพื่อช่วยให้เราสามารถสร้างโปรเจกต์อิเล็กทรอนิกส์ได้ง่ายขึ้น บอร์ด Arduino รุ่นยอดนิยม เช่น Arduino Uno, Arduino Mega, Arduino Nano และอื่น ๆ ล้วนใช้ชิปไมโครคอนโทรลเลอร์ตระกูล AVR (เช่น ATmega328P) หรือบางรุ่นอาจใช้ ARM Cortex หรือ ESP32 ขึ้นอยู่กับรุ่นและผู้ผลิต แต่หัวใจสำคัญคือ Arduino IDE นี่เองที่ทำหน้าที่เป็นตัวกลางในการเขียนโปรแกรม (Sketch) ทดสอบ แก้ไขโค้ด และอัปโหลดโปรแกรมลงบอร์ด ความโดดเด่นของ Arduino IDE เมื่อเทียบกับโปรแกรมพัฒนาซอฟต์แวร์อื่น ๆ คือ อินเทอร์เฟซที่เข้าใจง่าย มีปุ่มย่อย ๆ ไม่กี่ปุ่ม ได้แก่ ตรวจสอบโค้ด (Verify/Compile) อัปโหลด (Upload) เปิดตัวอย่าง (Examples) และจัดการไลบรารี่ (Library Manager) นอกจากนี้ยังมีการแบ่งพื้นที่ในการแสดงผลสถานะคอมไพล์ (Compiler Messages) เพื่อให้ผู้ใช้เข้าใจได้ทันทีว่าโค้ดของตนมีข้อผิดพลาดหรือไม่ และจุดไหนที่เกิดปัญหา

2.3.3 ฐานข้อมูล (Database)

2.3.3.1 MySQL



**ภาพที่ 2.7** โปรแกรมฐานข้อมูล

MySQL (มายเอสคิวแอล) เป็น[ระบบจัดการฐานข้อมูลเชิงสัมพันธ์](https://th.wikipedia.org/wiki/%E0%B8%A3%E0%B8%B0%E0%B8%9A%E0%B8%9A%E0%B8%88%E0%B8%B1%E0%B8%94%E0%B8%81%E0%B8%B2%E0%B8%A3%E0%B8%90%E0%B8%B2%E0%B8%99%E0%B8%82%E0%B9%89%E0%B8%AD%E0%B8%A1%E0%B8%B9%E0%B8%A5%E0%B9%80%E0%B8%8A%E0%B8%B4%E0%B8%87%E0%B8%AA%E0%B8%B1%E0%B8%A1%E0%B8%9E%E0%B8%B1%E0%B8%99%E0%B8%98%E0%B9%8C) (Relational Database Management System) โดยใช้ภาษา [SQL](https://th.wikipedia.org/wiki/SQL) แม้ว่า MySQL เป็นซอฟต์แวร์[โอเพนซอร์ส](https://th.wikipedia.org/wiki/%E0%B9%82%E0%B8%AD%E0%B9%80%E0%B8%9E%E0%B8%99%E0%B8%8B%E0%B8%AD%E0%B8%A3%E0%B9%8C%E0%B8%AA) แต่แตกต่างจากซอฟต์แวร์โอเพนซอร์สทั่วไป โดยมีการพัฒนาภายใต้บริษัท MySQL AB ใน[ประเทศสวีเดน](https://th.wikipedia.org/wiki/%E0%B8%9B%E0%B8%A3%E0%B8%B0%E0%B9%80%E0%B8%97%E0%B8%A8%E0%B8%AA%E0%B8%A7%E0%B8%B5%E0%B9%80%E0%B8%94%E0%B8%99) โดยจัดการ MySQL ทั้งในแบบที่ให้ใช้ฟรี และแบบที่ใช้ในเชิงธุรกิจ เมื่อปี ค.ศ. 2008 บริษัท[ซันไมโครซิสเต็มส์](https://th.wikipedia.org/wiki/%E0%B8%8B%E0%B8%B1%E0%B8%99%E0%B9%84%E0%B8%A1%E0%B9%82%E0%B8%84%E0%B8%A3%E0%B8%8B%E0%B8%B4%E0%B8%AA%E0%B9%80%E0%B8%95%E0%B9%87%E0%B8%A1%E0%B8%AA%E0%B9%8C) (Sun Microsystems, Inc.) เข้าซื้อกิจการของ MySQL AB และ บริษัท[ออราเคิลคอร์ปอเiชัน](https://th.wikipedia.org/wiki/%E0%B8%AD%E0%B8%AD%E0%B8%A3%E0%B8%B2%E0%B9%80%E0%B8%84%E0%B8%B4%E0%B8%A5%E0%B8%84%E0%B8%AD%E0%B8%A3%E0%B9%8C%E0%B8%9B%E0%B8%AD%E0%B9%80%E0%B8%A3%E0%B8%8A%E0%B8%B1%E0%B8%99) (OracleCorporation) ได้เข้าซื้อกิจการของบริษัทซันไมโครซิสเต็มส์ อีกทอดหนึ่ง เมื่อปี ค.ศ. 2010 MySQL สร้างขึ้นโดยชาวสวีเดน 2 คน และชาวฟินแลนด์ ชื่อ David Axmark, Allan Larsson และ Michael "Monty" Widenius ชื่อ "MySQL" อ่านออกเสียงว่า "มายเอสคิวเอล" หรือ "มายเอสคิวแอล" (ในการอ่านอักษร [L](https://th.wikipedia.org/wiki/L) ในภาษาไทย) ซึ่งทางซอฟต์แวร์ไม่ได้อ่าน มายซีเควล หรือ มายซีควล เหมือนกับซอฟต์แวร์จัดการฐานข้อมูลตัวอื่นนของผลิตภัณฑ์นั้นแบ่งออกมาได้สามสายการผลิต ได้แก่ เวอร์ชันใช้ฟรี เวอร์ชันการค้า และเวอร์ชันที่สนับสนุนกับผลิตภัณฑ์ [SAP](https://th.wikipedia.org/wiki/SAP) (MAX DB) ความแตกต่างคือเวอร์ชันคอมมิวนิตี้นั้นสามารถนำไปใช้งานได้ฟรีแต่ขาดการสนับสนุนหรือการช่วยเหลือเมื่อมีปัญหาเกิดขึ้น เวอร์ชันที่เป็นคอมเมอร์เชียลนั้นให้บริการด้านความสนับสนุนเมื่อมีปัญหา (ซื้อบริการ) สรุปคร่าว ๆ ประเภทดาต้าเบสให้เลือกใช้ดังนี้

* MySQL เอนเทอร์ไพรส์ [Enterprise](https://th.wikipedia.org/w/index.php?title=Enterprise&action=edit&redlink=1)
* MySQL [คลัสเตอร์](https://th.wikipedia.org/w/index.php?title=%E0%B8%84%E0%B8%A5%E0%B8%B1%E0%B8%AA%E0%B9%80%E0%B8%95%E0%B8%AD%E0%B8%A3%E0%B9%8C&action=edit&redlink=1) [Cluster](https://th.wikipedia.org/w/index.php?title=Cluster&action=edit&redlink=1)
* MySQL [Embedded](https://th.wikipedia.org/w/index.php?title=Embedded&action=edit&redlink=1)
* MySQL Community ([opensource](https://th.wikipedia.org/wiki/Opensource) เวอร์ชัน)

ในเวอร์ชัน 5.0 มีความสามารถหลายอย่างที่สำคัญสำหรับระบบฐานข้อมูลขนาดใหญ่หรือระดับองค์กร ([EnterPrise](https://th.wikipedia.org/w/index.php?title=EnterPrise&action=edit&redlink=1) Feature) เช่น [Store Procedure](https://th.wikipedia.org/wiki/Store_Procedure), [database trigger](https://th.wikipedia.org/w/index.php?title=Database_trigger&action=edit&redlink=1), [database view](https://th.wikipedia.org/w/index.php?title=Database_view&action=edit&redlink=1), [database schema](https://th.wikipedia.org/w/index.php?title=Database_schema&action=edit&redlink=1) ซึ่งได้มีการปรับเพิ่มประสิทธิภาพในส่วนของตารางและการทำ[ดัชนี](https://th.wikipedia.org/wiki/%E0%B8%94%E0%B8%B1%E0%B8%8A%E0%B8%99%E0%B8%B5) (index) ขึ้นมาอีก ปัจจุบันเวอร์ชัน community หรือเวอร์ชันที่เสถียร (stable) 5.0 และเวอร์ชันทดสอบคือ 5.1 beta release และ 5.2 Alpha ตั้งแต่เวอร์ชัน 5.1 เริ่มสนับสนุนการทำ [Parttion Database](https://th.wikipedia.org/w/index.php?title=Parttion_Database&action=edit&redlink=1) ตารางเวลาสำหรับเหตุการณ์ต่าง ๆ

**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

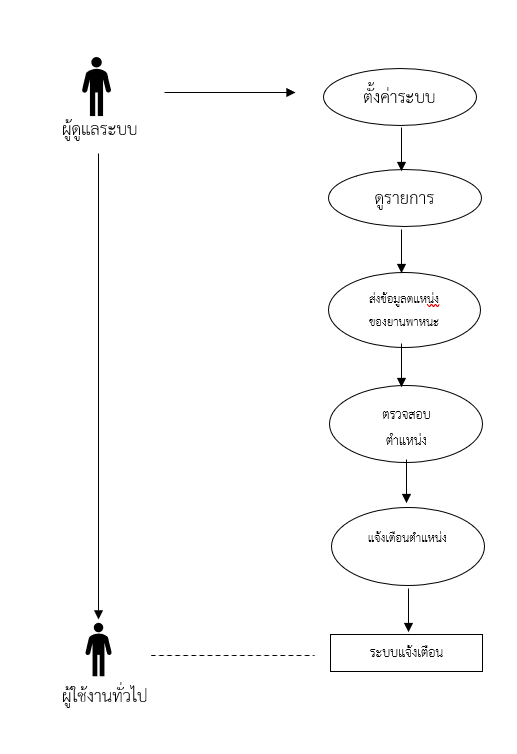
- ความทนทาน

- ประสิทธิภาพ

- ต้นทุน

**3.2 การออกแบบระบบงาน**

3.2.1 แผนภาพยูสเคส (Use Case Diagram)



**ภาพที่ 3.1** แผนภาพยูสเคส

3.2.2 คำอธิบายยูสเคส (Use Case Description)

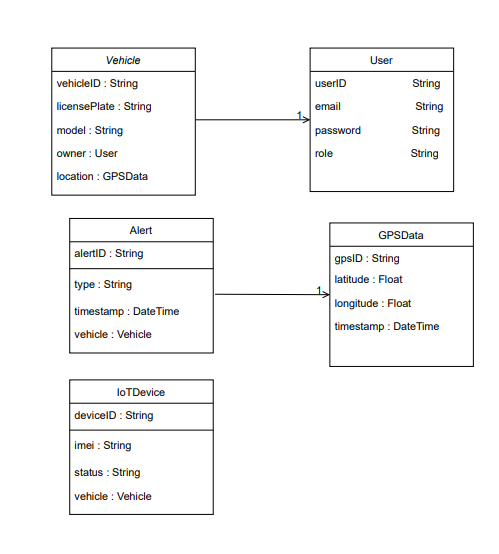
ผู้ใช้งาน (Actors)

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

Use Cases (กรณีการใช้งาน)

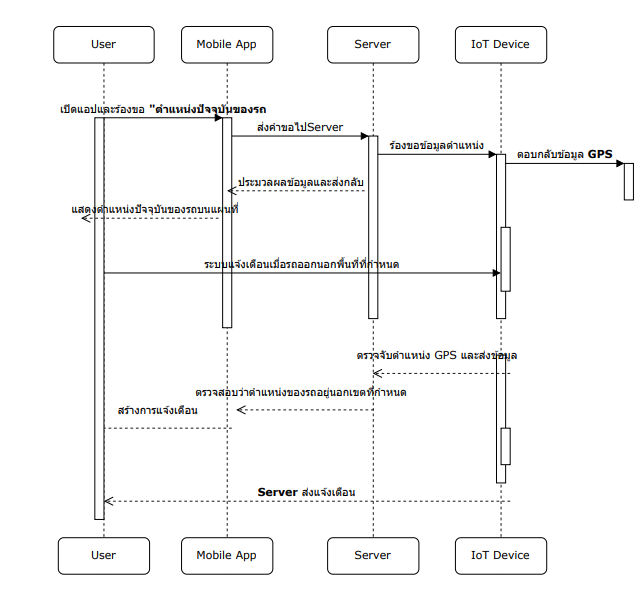
* ติดตามตำแหน่งยานพาหนะ (Track Vehicle Location) ดูตำแหน่งปัจจุบันของรถ
* แจ้งเตือนเมื่อออกนอกพื้นที่ (Geofencing Alert) ระบบแจ้งเตือนเมื่อรถออกนอกพื้นที่ที่กำหนด
* ดูประวัติการเดินทาง (View Travel History) – ดูเส้นทางที่ยานพาหนะเคยเดินทาง
* ตั้งค่าการแจ้งเตือน (Set Notifications) – ตั้งค่าให้ระบบแจ้งเตือนเมื่อรถเคลื่อนที่หรือหยุด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

3.2.3 แผนภาพคลาส (Class Diagram)



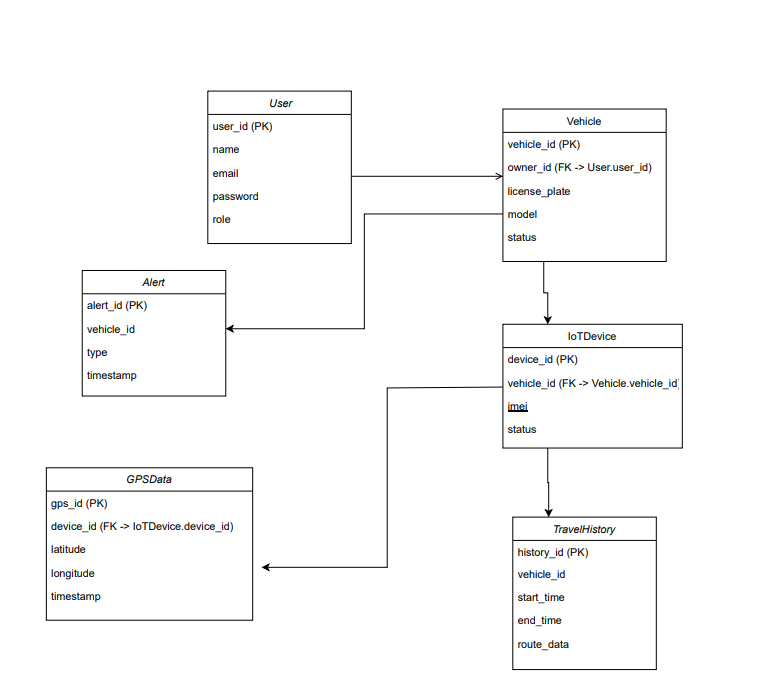
**ภาพที่ 3.2** แผนภาพคลาส

3.2.4 แผนภาพลำดับเหตุการณ์ (Sequence Diagram)



**ภาพที่ 3.3** ลำดับเหตุการณ์

3.2.5 แผนภาพความสัมพันธ์ของตาราง



**ภาพที่ 3.4** 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

**ตารางที่ 4.1** เครื่องมือที่ใช้ในการพัฒนาระบบ

|  |  |
| --- | --- |
| **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 |

4.3.2 ซอฟต์แวร์และเครื่องมือที่ใช้ในการพัฒนา

**ตารางที่ 4.2** ซอฟต์แวร์ที่ใช้ในการพัฒนาระบบ

|  |  |
| --- | --- |
| **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 |

**บทที่ 5**

**การทดสอบ**

* 1. **การทดสอบระบบ**

การทดสอบระบบเป็นขั้นตอนสำคัญในการพัฒนาระบบติดตามยานพาหนะด้วย GPS โดยระบบ IoT ซึ่งจะช่วยตรวจสอบว่าแต่ละส่วนของระบบทำงานได้ตามที่คาดหวังหรือไม่ การทดสอบจะแบ่งออกเป็นสองระดับ ได้แก่ การทดสอบในระดับหน่วยย่อย และการทดสอบในระดับระบบ

5.1.1 การทดสอบในระดับหน่วยย่อย การทดสอบในระดับหน่วยย่อยจะมุ่งเน้นไปที่การทดสอบแต่ละฟังก์ชันหรือส่วนประกอบของระบบ เช่น การทดสอบ GPS Module: ตรวจสอบว่า GPS Module สามารถรับข้อมูลตำแหน่งจากดาวเทียมได้ถูกต้องและแม่นยำ การทดสอบการเชื่อมต่อกับ IoT Platform: ทดสอบว่าเซ็นเซอร์ GPS สามารถส่งข้อมูลตำแหน่งที่ได้ไปยังแพลตฟอร์ม IoT ได้อย่างถูกต้องและทันเวลา การทดสอบการรับข้อมูลจากเซ็นเซอร์: ตรวจสอบว่าเซ็นเซอร์ที่ติดตั้งในยานพาหนะสามารถส่งข้อมูลเกี่ยวกับตำแหน่งและสถานะการเคลื่อนที่ไปยังฐานข้อมูลได้อย่างสมบูรณ์ การทดสอบการแสดงผลในแอปพลิเคชันหรือเว็บแพลตฟอร์ม: ทดสอบว่าแอปพลิเคชันหรือเว็บไซต์สามารถแสดงตำแหน่งของยานพาหนะในเวลาจริงได้อย่างแม่นยำ

5.1.2 การทดสอบในระดับระบบ (System Testing) การทดสอบในระดับระบบจะตรวจสอบการทำงานรวมของระบบทั้งหมด โดยจะรวมถึง การทดสอบการเชื่อมโยงระหว่าง GPS Module, IoT Platform, และแอปพลิเคชัน: ทดสอบว่าเมื่อระบบทำงานในสภาพแวดล้อมจริง เซ็นเซอร์ GPS สามารถส่งข้อมูลได้อย่างราบรื่นและไม่มีปัญหาการเชื่อมต่อ การทดสอบการอัปเดตตำแหน่งแบบ Real-time: ตรวจสอบว่าเมื่อยานพาหนะเคลื่อนที่ ระบบสามารถติดตามตำแหน่งได้ในเวลาใกล้เคียงกับเวลาจริง (real-time) โดยไม่มีการดีเลย์ การทดสอบการแจ้งเตือน: ทดสอบการทำงานของระบบแจ้งเตือนเมื่อยานพาหนะเข้าใกล้พื้นที่ที่กำหนด (Geofencing) หรือเมื่อมีการขัดข้องในระบบการทดสอบการใช้งานของแอปพลิเคชัน: ทดสอบว่าแอปพลิเคชันหรือแพลตฟอร์ม IoT สามารถรับข้อมูลและแสดงผลข้อมูลได้อย่างถูกต้อง รวมถึงตรวจสอบความเร็วในการแสดงผลของตำแหน่งในเวลาใกล้เคียงกับจริง

* 1. **กรณีทดสอบ**

การเตรียมกรณีทดสอบจะต้องครอบคลุมทุกสถานการณ์ที่ระบบอาจต้องเผชิญ รวมถึงการทดสอบในสภาพแวดล้อมที่แตกต่างกัน เช่นกรณีทดสอบการเดินทางในพื้นที่ต่างๆ ทดสอบการติดตามยานพาหนะในพื้นที่ที่มีสัญญาณ GPS อ่อน เช่น ในอาคารหรือในพื้นที่ที่มีสิ่งกีดขวาง กรณีทดสอบการขัดข้องของเครือข่าย: ทดสอบว่าเมื่อระบบไม่มีการเชื่อมต่ออินเทอร์เน็ต ระบบยังสามารถทำงานได้ตามปกติ หรือบันทึกข้อมูลไว้ในหน่วยความจำชั่วคราวและส่งข้อมูลเมื่อเชื่อมต่ออินเทอร์เน็ตได้ กรณีทดสอบการทำงานของระบบแจ้งเตือน: ทดสอบระบบแจ้งเตือนเมื่อยานพาหนะออกจากพื้นที่ที่กำหนด หรือเมื่อมีการเคลื่อนที่ผิดปกติ

**5.3 ผลการทดสอบ**

ผลการทดสอบจะได้รับการวิเคราะห์เพื่อดูว่าระบบทำงานได้ตามข้อกำหนดหรือไม่ และระบุข้อบกพร่องหรือปัญหาที่พบระหว่างการทดสอบ เช่น การทดสอบ GPS: ผลการทดสอบพบว่า GPS Module สามารถรับสัญญาณได้ในพื้นที่ส่วนใหญ่ แต่บางครั้งเกิดปัญหาการสูญเสียสัญญาณในอาคาร การทดสอบการเชื่อมต่อกับ IoT Platform: การเชื่อมต่อระหว่าง GPS Module และแพลตฟอร์ม IoT ทำงานได้ดี ไม่มีปัญหาการส่งข้อมูลการทดสอบการแสดงผลตำแหน่ง: ผลการทดสอบพบว่าแอปพลิเคชันสามารถแสดงตำแหน่งของยานพาหนะได้แม่นยำและทันเวลาในสภาวะส่วนใหญ่การทดสอบการแจ้งเตือน: ระบบสามารถส่งการแจ้งเตือนเมื่อยานพาหนะออกจากพื้นที่ที่กำหนดหรือมีการเคลื่อนที่ผิดปกติอย่างถูกต้อง

**บทที่ 6**

**สรุปผลและเสนอแนะ**

**6.1 สรุปผลการดำเนินงาน**

การพัฒนาระบบติดตามยานพาหนะด้วย GPS โดยระบบ IoT มีวัตถุประสงค์หลักเพื่อสร้างระบบที่สามารถติดตามตำแหน่งของยานพาหนะในเวลาจริงและสามารถส่งข้อมูลตำแหน่งผ่านระบบ IoT ไปยังผู้ใช้งานได้อย่างแม่นยำและทันเวลา ผลการดำเนินงานของโครงงานนี้พบว่า ระบบทำงานได้ตามที่คาดหวังในหลายๆ ด้าน เช่น:

6.1.1 ระบบ GPS สามารถรับข้อมูลตำแหน่งจากดาวเทียมและส่งข้อมูลไปยัง IoT Platform ได้อย่างมีประสิทธิภาพ

6.1.2 ข้อมูลตำแหน่งที่ส่งไปยังแอปพลิเคชันหรือเว็บแพลตฟอร์มมีความถูกต้องและทันเวลา (Real-time)

6.1.3 ระบบสามารถแจ้งเตือนเมื่อยานพาหนะออกจากพื้นที่ที่กำหนด (Geofencing) หรือเมื่อมีการเคลื่อนที่ผิดปกติ

6.1.4 การทดสอบระบบในระดับหน่วยย่อยและระดับระบบพบว่าระบบทำงานได้ดี และไม่มีปัญหาการเชื่อมต่อหรือความผิดปกติในการทำงาน

**6.2 ข้อเสนอแนะ**

แม้ว่าระบบนี้จะทำงานได้ตามที่คาดหวัง แต่ยังมีพื้นที่ที่สามารถปรับปรุงและพัฒนาได้ เพื่อให้ระบบมีประสิทธิภาพสูงขึ้นและสามารถตอบสนองความต้องการของผู้ใช้งานได้ดีขึ้น ดังนี้

6.2.1 การปรับปรุงระบบ การปรับปรุงการรับสัญญาณ GPS: ในบางพื้นที่ที่มีสิ่งกีดขวางหรือในอาคาร ระบบ GPS อาจสูญเสียสัญญาณหรือมีความผิดพลาดในการคำนวณตำแหน่ง สามารถพัฒนาการใช้เซ็นเซอร์เพิ่มเติม หรือเทคโนโลยีที่ช่วยในการรับสัญญาณที่ดีขึ้นในสภาพแวดล้อมที่ท้าทาย การปรับปรุงการแจ้งเตือนเพิ่มฟังก์ชันการแจ้งเตือนที่สามารถปรับแต่งได้ตามความต้องการ เช่น การแจ้งเตือนเมื่อยานพาหนะมีความเร็วสูงเกินกว่าที่กำหนด หรือเมื่อมีการเข้าใกล้พื้นที่ที่ต้องการป้องกัน การปรับปรุงประสิทธิภาพของแอปพลิเคชัน: ควรพัฒนาแอปพลิเคชันให้สามารถรองรับผู้ใช้งานจำนวนมากได้อย่างมีประสิทธิภาพ รวมถึงปรับปรุง UI/UX เพื่อความสะดวกในการใช้งาน

6.2.2 การนำไปใช้ในอนาคต การนำไปใช้ในภาคธุรกิจ: ระบบนี้สามารถนำไปใช้ในการติดตามยานพาหนะในภาคธุรกิจ เช่น การขนส่งสินค้า, การจัดการฟลีท (Fleet Management), และการติดตามรถยนต์ของลูกค้าในกรณีของธุรกิจให้เช่ารถ การพัฒนาระบบเพิ่มเติม: สามารถต่อยอดการใช้งานโดยการเพิ่มฟังก์ชันใหม่ๆ เช่น การตรวจจับอุบัติเหตุ (Accident Detection), การติดตามสภาพของยานพาหนะ (Vehicle Health Monitoring), หรือการรวมกับระบบการวิเคราะห์ข้อมูลเพื่อเพิ่มประสิทธิภาพในการบริหารจัดการ การใช้งานในด้านการขนส่งสาธารณะ: ระบบนี้สามารถนำไปใช้ในการติดตามรถเมล์, แท็กซี่, หรือยานพาหนะในระบบขนส่งสาธารณะเพื่อเพิ่มความปลอดภัยและการจัดการที่ดีขึ้น

**6.3 บทสรุป**

จากการพัฒนาและทดสอบระบบติดตามยานพาหนะด้วย GPS โดยระบบ IoT พบว่าเป็นระบบที่มีความสามารถในการติดตามตำแหน่งของยานพาหนะได้อย่างแม่นยำและรวดเร็ว ระบบนี้ช่วยให้ผู้ใช้งานสามารถติดตามตำแหน่งของยานพาหนะได้ในเวลาจริงผ่านแอปพลิเคชันหรือแพลตฟอร์มต่างๆ และสามารถส่งการแจ้งเตือนได้ทันทีเมื่อเกิดเหตุการณ์ที่ไม่ปกติ

ถึงแม้ว่าจะมีบางจุดที่สามารถปรับปรุงได้ แต่ระบบนี้ถือว่าเป็นแนวทางที่ดีในการนำเทคโนโลยี GPS และ IoT มาประยุกต์ใช้ในการติดตามยานพาหนะ ซึ่งสามารถนำไปใช้ได้จริงทั้งในภาคธุรกิจและการใช้งานส่วนบุคคลในอนาคต.

**บรรณานุกรม**

Lee, J. & Kim, S. (2019). "Vehicle Tracking System using GPS and IoT". Journal of Internet of Things, 5(3), 150-160.

สมศักดิ์, ก. (2563). การพัฒนาระบบติดตามยานพาหนะด้วย GPS โดยระบบ IoT. กรุงเทพฯ: สำนักพิมพ์เทคโนโลยี.

วิชัย, ศ. (2561). "การพัฒนาระบบติดตามยานพาหนะด้วย GPS". วารสารวิศวกรรมและเทคโนโลยี, 12(4), 67-78.

ธนา, พ. (2564). การออกแบบและพัฒนาระบบติดตามยานพาหนะผ่านเทคโนโลยี IoT. มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ.

Zhang, H. (2020). "IoT-Based Vehicle Tracking and Management System". International Journal of Transportation Engineering, 8(1), 35-42.