



## การพัฒนาเว็บแอปพลิเคชันในร้านขายยาเภสัช

โดย

นาย วรทา มิ่งพรวัฒน์

โครงการพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

วิทยาศาสตร์บัณฑิต

สาขาวิชาวิทยาการคอมพิวเตอร์

คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยธรรมศาสตร์

ปีการศึกษา 2567

ลิขสิทธิ์ของมหาวิทยาลัยธรรมศาสตร์

การพัฒนาเว็บแอปพลิเคชันในร้านขายยาเภสัช

โดย

นาย วรธา มิ่งพรวัฒน์

โครงการพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

วิทยาศาสตร์บัณฑิต

สาขาวิชาวิทยาการคอมพิวเตอร์

คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยธรรมศาสตร์

ปีการศึกษา 2567

ลิขสิทธิ์ของมหาวิทยาลัยธรรมศาสตร์

**Development the web application in the pharmacy**

**BY**

**Worrata Mingponwat**

**A FINAL-YEAR PROJECT REPORT SUBMITTED IN PARTIAL FULFILLMENT OF THE  
REQUIREMENTS FOR THE DEGREE OF BACHELOR OF SCIENCE**

**COMPUTER SCIENCE  
FACULTY OF SCIENCE AND TECHNOLOGY  
THAMMASAT UNIVERSITY  
ACADEMIC YEAR 2024**

**COPYRIGHT OF THAMMASAT UNIVERSITY**

มหาวิทยาลัยธรรมศาสตร์  
คณะวิทยาศาสตร์และเทคโนโลยี

รายงานโครงการพิเศษ

ของ

นาย วรทา มิ่งพรพัฒน์

เรื่อง

การพัฒนาเว็บแอปพลิเคชันในร้านขายยาเภสัช

ได้รับการตรวจสอบและอนุมัติ ให้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

หลักสูตรวิทยาศาสตรบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์

เมื่อ วันที่ 30 พฤษภาคม พ.ศ. 2568

อาจารย์ที่ปรึกษา

นวฤทธิ์ ชลารักษ์

(อ.ดร.นวฤทธิ์ ชลารักษ์)

กรรมการสอบโครงการพิเศษ

จิรา สิริศักดิ์

(ผศ. ดร.จิรา สิริศักดิ์)

กรรมการสอบโครงการพิเศษ

(ผศ. ดร.ปกป้อง ส่องเมือง)

มหาวิทยาลัยธรรมศาสตร์  
คณะวิทยาศาสตร์และเทคโนโลยี

รายงานโครงงานพิเศษ

ของ

นาย วรทา มิ่งพรพัฒน์

เรื่อง

การพัฒนาเว็บแอปพลิเคชันในร้านขายยาเภสัช

ได้รับการตรวจสอบและอนุมัติ ให้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

หลักสูตรวิทยาศาสตรบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์

เมื่อ วันที่ 30 พฤษภาคม พ.ศ. 2568

อาจารย์ที่ปรึกษา

นวฤทธิ์ ชลารักษ์  
(อ.ดร.นวฤทธิ์ ชลารักษ์)

กรรมการสอบโครงงานพิเศษ

จิรา สิริศักดิ์  
(ผศ. ดร.จิรา สิริศักดิ์)

กรรมการสอบโครงงานพิเศษ

  
(ผศ. ดร.ป๋อง ส่องเมือง)

หัวข้อโครงการพิเศษ	การพัฒนาเว็บแอปพลิเคชันในร้านขายยาเภสัช
ชื่อผู้เขียน	นาย วรชา มิ่งพรวัฒน์
ชื่อปริญญา	วิทยาศาสตร์บัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์
สาขาวิชา/คณะ/มหาวิทยาลัย	สาขาวิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยธรรมศาสตร์
อาจารย์ที่ปรึกษาโครงการพิเศษ	อ.ดร.นวกฤษ ชลารักษ์
อาจารย์ที่ปรึกษาโครงการพิเศษร่วม	
ปีการศึกษา	2566

#### บทคัดย่อ

โครงการนี้มีวัตถุประสงค์ในการพัฒนาเว็บแอปพลิเคชันสำหรับร้านขายยาเพื่อแก้ปัญหาเรื่องสินค้าใกล้จะหมดและการแจ้งเตือนว่ายาใกล้หมดอายุ ระบบที่พัฒนาจะช่วยให้ร้านขายยาสามารถจัดการสต็อกได้อย่างมีประสิทธิภาพ ลดการสูญเสียจากสินค้าที่หมดอายุ และเพิ่มความพึงพอใจของลูกค้าด้วยการรับประกันว่ายาที่ได้รับมีความปลอดภัยและมีคุณภาพ

ระบบจะประกอบด้วยฟีเจอร์หลัก ๆ เช่น การแจ้งเตือนเมื่อสินค้าถึงจุดที่ต้องเติมสต็อก การแจ้งเตือนเมื่อยากำลังจะหมดอายุ เพื่อให้ร้านขายยาสามารถดำเนินการอย่างทันที่และป้องกันการสูญหายจากการใช้สินค้าที่หมดอายุ

การพัฒนาเว็บแอปพลิเคชันนี้มีความสำคัญในการเพิ่มประสิทธิภาพการทำงานของร้านขายยา ลดความเสี่ยงในการใช้ยาที่หมดอายุ และสร้างความมั่นใจให้กับลูกค้าว่าจะได้รับยาที่ปลอดภัยและมีคุณภาพ โครงการนี้มีเป้าหมายในการสร้างระบบที่สามารถใช้งานได้ง่ายและมีประโยชน์ต่อการดำเนินงานของร้านขายยาในชีวิตประจำวัน

**คำสำคัญ:** เรื่องการเตือนสินค้าใกล้จะหมด, การแจ้งเตือนว่ายาใกล้หมดอายุ

Thesis Title	Improving the web application in the pharmacy
Author	Worrata mingponwat
Degree	Bachelor of Science
Major Field/Faculty/University	Computer Science Faculty of Science and Technology Thammasat University
Project Advisor	Dr.Nawarerk Chalarak
Academic Years	2024

### Abstract

This project aims to develop a web application for pharmacies to address the issues of low stock alerts and expiration date notifications. The system will help pharmacies manage inventory efficiently, reduce losses caused by expired products, and enhance customer satisfaction by ensuring that medications provided are safe and of high quality.

The system will include key features such as alerts when products reach a low stock threshold and notifications for medications that are nearing expiration. This enables pharmacies to take immediate action and prevent losses caused by selling expired products.

Developing this web application is crucial for improving pharmacy operations, minimizing the risk of using expired medications, and assuring customers that the medications they receive are safe and reliable. The goal of this project is to create a functional system that provides practical benefits to the daily operations of pharmacies.

**Keywords:** Low Stock Alerts, Medication Expiration Notifications

## กิตติกรรมประกาศ

ข้าพเจ้าขอขอบคุณอาจารย์ที่ปรึกษา อ.ดร.นวกฤษ์ ชลารักษ์ ที่ได้ให้คำแนะนำและคำปรึกษาอย่างเต็มใจและอดทนตลอดระยะเวลาการดำเนินโครงการ ขอขอบคุณที่ชี้แนะแนวทางและสนับสนุนในการแก้ไขปัญหาต่าง ๆ ที่เกิดขึ้น รวมถึงการให้ข้อเสนอแนะที่เป็นประโยชน์ต่อการพัฒนาโครงการนี้

นอกจากนี้ ข้าพเจ้าขอขอบคุณครอบครัวที่ให้การสนับสนุนในทุกด้าน ไม่ว่าจะเป็นกำลังใจ คำแนะนำ และการสนับสนุนทางการเงิน ทำให้ข้าพเจ้ามีกำลังใจในการทำงานและสามารถผ่านพ้นอุปสรรคต่าง ๆ ไปได้

สุดท้ายนี้ ข้าพเจ้าขอขอบคุณทุกร้านขายยาและองค์กรที่เกี่ยวข้องที่ได้ให้ข้อมูลและคำแนะนำที่เป็นประโยชน์ต่อการพัฒนาแอปพลิเคชันนี้ ข้อมูลที่ได้รับเป็นสิ่งที่มีความจำเป็นในการสร้างระบบที่ตอบสนองต่อความต้องการและแก้ไขปัญหาที่เกิดขึ้นในทางปฏิบัติจริง

ขอขอบคุณทุกท่านที่มีส่วนร่วมในการทำให้โครงการนี้สำเร็จลุล่วงไปด้วยดี

นาย วรธา มิ่งพรวัฒน์



สารบัญ	หน้า
บทคัดย่อ	1
ABSTRACT	2
กิตติกรรมประกาศ	3
สารบัญ	4
บทที่ 1 บทนำ	5
1.1    ความเป็นมาและความสำคัญของโครงการ	5
1.2    วัตถุประสงค์	6
1.3    ขอบเขตของโครงการ	7
1.4    ประโยชน์ของโครงการ	8
1.5    ข้อจำกัดของโครงการ	9
บทที่ 2    วรรณกรรมและงานวิจัยที่เกี่ยวข้อง	<a href="#">10</a>
2.1    แนวคิดทฤษฎีที่เกี่ยวข้อง	<a href="#">10</a>
2.2    งานวิจัยที่เกี่ยวข้อง	28
บทที่ 3    วิธีการวิจัย	<a href="#">29</a>
3.1    วิธีดำเนินการวิจัย	29
บทที่ 4    ทรัพยากรและแผนการดำเนินงาน	33
4.1    การจัดเตรียมฮาร์ดแวร์และซอฟต์แวร์	<a href="#">34</a>
4.2    แผนการดำเนินงาน	35

## บทนำ

### ความเป็นมาและความสำคัญของโครงการ

ในปัจจุบัน ร้านขายยาามีบทบาทสำคัญในการดูแลสุขภาพของประชาชน โดยการจำหน่ายยาและเวชภัณฑ์ที่มีคุณภาพและปลอดภัย อย่างไรก็ตาม การจัดการสินค้าคงคลังในร้านขายยายังคงมีความท้าทาย โดยเฉพาะปัญหาการขาดแคลนสินค้าและการจำหน่ายยาที่หมดอายุ ซึ่งอาจส่งผลกระทบต่อความปลอดภัยของผู้บริโภคและความเชื่อมั่นในร้านขายยา

ปัญหาหลักที่ร้านขายยาประสบคือการตรวจสอบและจัดการสต็อกสินค้าที่ซับซ้อน เนื่องจากมียาหลากหลายประเภทและอายุการใช้งานที่แตกต่างกัน การจัดการสต็อกที่ไม่เป็นระบบอาจนำไปสู่การสูญเสียจากการทิ้งยาที่หมดอายุหรือการขาดแคลนยาที่จำเป็นในช่วงเวลาสำคัญ การพัฒนาเว็บแอปพลิเคชันที่สามารถแจ้งเตือนเมื่อสินค้าคงคลังใกล้หมดและเมื่อยากำลังจะหมดอายุจึงมีความสำคัญอย่างยิ่งในการแก้ไขปัญหา

เว็บแอปพลิเคชันนี้มีเป้าหมายที่จะช่วยให้ร้านขายยาสามารถจัดการสต็อกได้อย่างมีประสิทธิภาพและทันเวลา โดยระบบจะแจ้งเตือนเมื่อสินค้าถึงจุดที่ต้องเติมสต็อกและเมื่อยากำลังจะหมดอายุ ทำให้ร้านขายยาสามารถวางแผนการสั่งซื้อได้อย่างแม่นยำและป้องกันการสูญเสียจากสินค้าที่หมดอายุ นอกจากนี้ ระบบยังช่วยเพิ่มความมั่นใจให้กับลูกค้าว่าจะได้รับยาที่ปลอดภัยและมีคุณภาพ

การพัฒนาโครงการนี้จึงมีความสำคัญในการเพิ่มประสิทธิภาพการทำงานของร้านขายยา ลดความเสี่ยงที่เกิดจากการใช้ยาที่หมดอายุ และสร้างความพึงพอใจให้กับลูกค้า โดยระบบที่พัฒนานี้จะมีบทบาทสำคัญในการสนับสนุนการดำเนินงานของร้านขายยาในยุคดิจิทัล

## วัตถุประสงค์

1. เพื่อเพิ่มความปลอดภัยและความมั่นใจให้กับลูกค้าในการใช้ยา โดยรับประกันว่ายาที่จำหน่ายมีคุณภาพและยังไม่หมดอายุ
2. เพื่อสนับสนุนการสั่งซื้อสินค้าอย่างมีประสิทธิภาพโดยการบันทึกวันที่รับสินค้าและวันหมดอายุของสินค้า
3. เพื่อสร้างระบบที่ใช้งานง่ายและสะดวกสำหรับผู้ใช้งานในร้านขายยา โดยมีจำนวนคลิกที่น้อยกว่าโปรแกรมอื่นๆ ทำให้การจัดการสต็อกและการตรวจสอบสินค้าคงคลังเป็นเรื่องง่ายและรวดเร็ว
4. เพื่อสร้างระบบแจ้งเตือนยาที่ใกล้หมดอายุ เพื่อป้องกันการใช้ยาที่หมดอายุและลดการสูญเสียจากการทิ้งยาที่ไม่สามารถใช้ได้
5. เพื่อปรับปรุงประสิทธิภาพในการจัดการสินค้าคงคลังของร้านขายยา โดยระบบจะช่วยให้การบันทึกและติดตามสินค้าทั้งหมดที่มีอยู่ในสต็อกอย่างแม่นยำ
6. เพื่อพัฒนาเว็บแอปพลิเคชันที่สามารถแจ้งเตือนเมื่อสินค้าคงคลังใกล้หมด ช่วยให้ร้านขายยาสามารถเติมสต็อกได้ทันเวลา
7. ลดความเสี่ยงจากการขาดแคลนสินค้าที่จำเป็น

## ขอบเขตของโครงการ

### 1. การพัฒนาระบบแจ้งเตือนสินค้าคงคลังใกล้หมด:

- ระบบจะตรวจสอบและแจ้งเตือนเมื่อจำนวนสินค้าคงคลังถึงระดับที่กำหนดล่วงหน้า
- ผู้ใช้สามารถตั้งค่าระดับการแจ้งเตือนสำหรับสินค้าต่าง ๆ ได้
- ระบบจะแจ้งเตือนเป็นalertขึ้นมาบนหน้าเว็บแอปพลิเคชัน

### 2. การพัฒนาระบบแจ้งเตือนยาที่ใกล้หมดอายุ:

- ระบบจะตรวจสอบและแจ้งเตือนเมื่อยากำลังจะหมดอายุภายในระยะเวลาที่กำหนด
- ผู้ใช้สามารถตั้งค่าระยะเวลาในการแจ้งเตือนก่อนยาหมดอายุได้
- ระบบจะแจ้งเตือนเป็นalertขึ้นมาบนหน้าเว็บแอปพลิเคชัน

### 3. การจัดการสินค้าคงคลัง:

- ระบบจะช่วยในการบันทึกและติดตามสินค้าคงคลังทั้งหมดที่มีอยู่ในร้านขายยา
- ผู้ใช้สามารถเพิ่ม แก้ไข และลบข้อมูลสินค้าคงคลังได้ตามต้องการ
- ระบบจะบันทึกข้อมูลการสั่งซื้อและการรับสินค้าใหม่เข้ามาในระบบ

### 4. การค้นหาข้อมูลสินค้าและยา:

- ระบบจะมีฟังก์ชันค้นหาข้อมูลสินค้าและยา เพื่อให้ผู้ใช้สามารถค้นหาข้อมูลได้อย่างรวดเร็ว
- ข้อมูลสินค้าจะประกอบด้วยชื่อสินค้า วันที่ผลิต วันที่หมดอายุ และจำนวนคงคลัง

### 5. การพัฒนาระบบให้ใช้งานง่าย:

- ระบบจะถูกออกแบบให้มีอินเทอร์เฟซที่ใช้งานง่ายและสะดวกสำหรับผู้ใช้

## ประโยชน์ของโครงการ

### 1. เพิ่มประสิทธิภาพในการจัดการสต็อกสินค้า:

- ระบบแจ้งเตือนสินค้าคงคลังใกล้หมดและยาที่ใกล้หมดอายุช่วยให้ร้านขายยาสามารถจัดการสต็อกได้อย่างมีประสิทธิภาพ ลดปัญหาการขาดแคลนสินค้าหรือการมีสินค้าค้างสต็อกเกินจำเป็น

### 2. ลดการสูญเสียจากสินค้าที่หมดอายุ:

- การแจ้งเตือนยาที่ใกล้หมดอายุช่วยป้องกันการขายยาที่หมดอายุ ลดการสูญเสียจากการทิ้งยาที่ไม่สามารถใช้งานได้ และเพิ่มความปลอดภัยในการขายให้กับผู้บริโภค

### 3. ปรับปรุงการวางแผนการสั่งซื้อสินค้า:

- ข้อมูลจากระบบช่วยให้ร้านขายยาสามารถวิเคราะห์และวางแผนการสั่งซื้อสินค้าได้อย่างแม่นยำ ลดความเสี่ยงจากการขาดแคลนสินค้าที่จำเป็น และเพิ่มประสิทธิภาพในการบริหารจัดการสต็อก

### 4. เพิ่มความมั่นใจให้กับลูกค้า:

- ระบบช่วยรับประกันว่าสินค้าและยาที่จำหน่ายมีคุณภาพและยังไม่หมดอายุ เพิ่มความมั่นใจให้กับลูกค้าว่าจะได้รับสินค้าที่ปลอดภัยและมีคุณภาพ

### 5. ปรับปรุงการดำเนินงานของร้านขายยา:

- ระบบช่วยให้การจัดการสต็อกและการตรวจสอบสินค้าคงคลังเป็นเรื่องง่ายและรวดเร็ว ทำให้พนักงานสามารถทำงานได้อย่างมีประสิทธิภาพและลดภาระงานที่ซับซ้อน

### 6. สนับสนุนการตัดสินใจด้วยข้อมูลที่แม่นยำ:

- ระบบสามารถสร้างรายงานและการวิเคราะห์ข้อมูลที่เป็นประโยชน์ต่อการตัดสินใจในด้านการจัดการสต็อก การสั่งซื้อ และการบริหารงานของร้านขายยา

### 7. อนาคตของโครงการ:

- การพัฒนาเว็บแอปพลิเคชันช่วยให้ร้านขายยาสามารถปรับตัวและรองรับการใช้งานในยุคดิจิทัล เพิ่มความสะดวกในการจัดการและติดตามข้อมูลสินค้า

## ข้อจำกัดของโครงการ

### 1. การปรับใช้กับร้านขายยาต่างๆ:

- ระบบอาจต้องมีการปรับแต่งเพื่อให้สอดคล้องกับการดำเนินงานและข้อกำหนดเฉพาะของแต่ละร้านขายยา ซึ่งอาจใช้เวลาและทรัพยากรเพิ่มเติม

### 2. ความเชี่ยวชาญด้านเทคโนโลยีของผู้ใช้งาน:

- ผู้ใช้งานบางคนอาจไม่คุ้นเคยกับการใช้เทคโนโลยีหรือระบบดิจิทัล ซึ่งอาจต้องการการฝึกอบรมเพิ่มเติมเพื่อให้สามารถใช้งานระบบได้อย่างเต็มประสิทธิภาพ

### 3. ข้อจำกัดด้านการเชื่อมต่ออินเทอร์เน็ต:

- ระบบเว็บแอปพลิเคชันต้องการการเชื่อมต่ออินเทอร์เน็ตในการใช้งาน ดังนั้นร้านขายยาที่ตั้งอยู่ในพื้นที่ที่มีการเชื่อมต่ออินเทอร์เน็ตที่ไม่เสถียรอาจประสบปัญหาในการเข้าถึงและใช้งานระบบ

### 4. ความปลอดภัยของข้อมูล:

- การจัดการข้อมูลสินค้าคงคลังและข้อมูลยาในระบบดิจิทัลมีความเสี่ยงต่อการโจมตีทางไซเบอร์ การละเมิดข้อมูล หรือการเข้าถึงข้อมูลโดยไม่ได้รับอนุญาต ซึ่งจำเป็นต้องมีมาตรการรักษาความปลอดภัยที่เหมาะสม

### 5. ข้อจำกัดในการบูรณาการกับระบบอื่น:

- ระบบเว็บแอปพลิเคชันอาจมีข้อจำกัดในการบูรณาการกับระบบจัดการสต็อกหรือระบบบริหารจัดการอื่นๆ ที่ร้านขายยาใช้งานอยู่ ซึ่งอาจต้องมีการปรับแต่งหรือพัฒนาเพิ่มเติม

### 6. ข้อจำกัดด้านงบประมาณและทรัพยากร:

- การพัฒนาและบำรุงรักษาระบบต้องการงบประมาณและทรัพยากรในการดำเนินการ ซึ่งอาจเป็นข้อจำกัดสำหรับร้านขายยาขนาดเล็กหรือมีงบประมาณจำกัด

### 7. ความพร้อมในการรับรองคุณภาพและมาตรฐาน:

- ระบบต้องผ่านการทดสอบและรับรองคุณภาพตามมาตรฐานที่เกี่ยวข้อง ซึ่งอาจต้องใช้เวลาและทรัพยากรในการดำเนินการให้สำเร็จ

## บทที่ 2

### วรรณกรรมและงานวิจัยที่เกี่ยวข้อง

ในการพัฒนาเว็บแอปพลิเคชันในร้านขายยาเภสัชได้ทำการพัฒนาภายใต้แนวคิด ทฤษฎี และงานวิจัยที่เกี่ยวข้อง ซึ่งเป็นแนวทางการวิจัย โดยสามารถแบ่งเป็นหัวข้อดังนี้

#### 2.1 การทบทวนวรรณกรรม

- 2.1.1 แนวความคิดวงจรการพัฒนาซอฟต์แวร์
- 2.1.2 กระบวนการพัฒนาซอฟต์แวร์แบบแอจไจล์
- 2.1.3 การออกแบบฐานข้อมูล
- 2.1.4 Visual Studio code
- 2.1.5 Xampp
- 2.1.6 Node.js
- 2.1.7 xpress.js
- 2.1.8 EJS (Embedded JavaScript)
- 2.1.9 JavaScript
- 2.1.10 CSS (Cascading Style Sheets)
- 2.1.11 ระบบร้านขายยาในปัจจุบัน

### 2.1.1 แนวความคิดวงจรการพัฒนาซอฟต์แวร์

กระบวนการพัฒนาซอฟต์แวร์ คือ ลำดับขั้นตอนการดำเนินการในการพัฒนา ซอฟต์แวร์จนกระทั่งได้ซอฟต์แวร์ที่สำเร็จรูป ภายใต้ระยะเวลาที่กำหนดและได้ซอฟต์แวร์ที่มีคุณภาพ ตามมาตรฐานที่กำหนดไว้ (Kaur, 2015) ซึ่งเป็นที่รู้จักกันในนามของกระบวนการพัฒนาซอฟต์แวร์ แบบขั้นน้ำตก (Waterfall model) โดยกิจกรรมพื้นฐานของกระบวนการพัฒนาซอฟต์แวร์ ประกอบด้วย 5 ขั้นตอนพื้นฐาน ดังนี้

#### 1) ขั้นตอนการเก็บรวบรวมและวิเคราะห์ความต้องการ (requirement)

ซึ่งจะช่วยให้เข้าใจปัญหามากขึ้น

#### 2) ขั้นตอนการออกแบบ (design) เป็นขั้นตอนการวางแผนเพื่อแก้ไขปัญหา

#### 3) ขั้นตอนพัฒนาซอฟต์แวร์ (implement) การพัฒนาซอฟต์แวร์ด้วยการเขียน โปรแกรม

#### 4) ขั้นตอนการทดสอบโปรแกรม (verification) เป็นการทดสอบผลลัพธ์ที่ได้จากการเขียน โปรแกรม

#### 5) ขั้นตอนการบำรุงรักษา (maintenance) เป็นขั้นตอนการใช้งานจริงและ บำรุงรักษาซอฟต์แวร์

### 2.1.2 กระบวนการพัฒนาซอฟต์แวร์แบบแอจไจล์

เป็นกระบวนการพัฒนาซอฟต์แวร์ที่แก้ไขข้อบกพร่องของกระบวนการพัฒนาซอฟต์แวร์ แบบขั้นน้ำตก โดยนำกระบวนการมาแบ่งเป็นส่วนเล็กเพื่อการทำรอบซ้ำ ๆ และเพิ่มจำนวนการพัฒนา แต่ละรอบให้มากขึ้น และเน้นถึงการติดต่อสื่อสารระหว่างผู้มีส่วนได้ส่วนเสีย การปรับปรุงความต้องการ และวางแผนงาน เพื่อให้สามารถส่งมอบซอฟต์แวร์ไปยังลูกค้าให้เร็วที่สุด (Schaller, 2016) โดยผู้ร่วมโครงการเชื่อว่ากระบวนการออกแบบต้องพร้อมที่จะปรับปรุงอยู่เสมอ ด้วยข้อมูลใหม่ที่มีประสิทธิภาพและช่วยลดความเสี่ยงในโครงการที่จะเกิดขึ้น รวมถึงเพิ่มผลสำคัญของผลลัพธ์ที่ได้จากโครงการ (Cois et al., 2014)



### 2.1.3 การออกแบบฐานข้อมูล

การออกแบบฐานข้อมูล (Database Design) ฐานข้อมูลเป็นเรื่องสำคัญสำหรับ ระบบงานสารสนเทศที่ใช้คอมพิวเตอร์ประมวลผลในปัจจุบัน ในการออกแบบระบบสารสนเทศ ปัจจุบันก็มีการพัฒนาเทคโนโลยีกันอย่างมากมาย ฐานข้อมูลแบบสัมพันธ์(Relational Database) หนึ่งฐานข้อมูล สามารถบรรจุได้หลายตาราง

การออกแบบฐานข้อมูลที่ดี หลักการบางอย่างเป็นแนวทางสำหรับขั้นตอนการ ออกแบบฐานข้อมูล หลักการแรกคือข้อมูลที่เกิดขึ้นซ้ำๆ (หรือเรียกว่าข้อมูลที่ซ้ำกัน) เป็นสิ่งไม่ดี เนื่องจากเปลืองพื้นที่และเพิ่มความเสี่ยงที่จะเกิดข้อผิดพลาดและความไม่สอดคล้องกัน

หลักการที่สองคือความถูกต้องและความสมบูรณ์ของข้อมูลเป็นสิ่งสำคัญ ถ้าฐานข้อมูลของคุณมีข้อมูลที่ไม่ถูกต้องรายงานใดๆที่ถึงข้อมูลจากฐานข้อมูลจะมีข้อมูลที่ไม่ถูกต้องเช่นเดียวกัน ส่งผลให้การตัดสินใจได้ที่คุณทำโดยอ้างอิงจากรายงานเหล่านั้นจะเป็นข้อมูลที่ผิดพลาด

การออกแบบฐานข้อมูลที่ดีคือ ฐานข้อมูลที่:

- 1.แบ่งข้อมูลของคุณลงในตารางตามหัวข้อเพื่อลดข้อมูลที่ซ้ำกัน
- 2.มอบการเข้าถึงข้อมูลที่เป็นเพื่อใช้งานข้อมูลในตารางร่วมกันตามจำเป็น
- 3.ช่วยสนับสนุนและแน่ใจในความถูกต้องและความสมบูรณ์ของข้อมูลของคุณ
- 4.เหมาะสำหรับการดำเนินการข้อมูลและความต้องการในการทำรายงานของคุณ

## 2.1.4 Visual Studio Code (VS Code)

**Visual Studio Code** หรือที่เรียกสั้น ๆ ว่า **VS Code** เป็นโปรแกรมแก้ไขโค้ด (Code Editor) ที่ได้รับความนิยมอย่างแพร่หลายในหมู่นักพัฒนาซอฟต์แวร์ทั่วโลก พัฒนาโดย **Microsoft** และออกแบบมาให้รองรับการทำงานที่มีประสิทธิภาพทั้งในระบบปฏิบัติการ **Windows, macOS, และ Linux** จุดเด่นของ **VS Code** คือความเบา (**Lightweight**) และความสามารถในการปรับแต่งได้หลากหลาย **VS Code** รองรับหลายภาษาโปรแกรม เช่น **JavaScript, Python, Java, C++, PHP,** และอีกมากมาย ด้วยปลั๊กอินหรือส่วนเสริม (**Extensions**) ที่สามารถติดตั้งเพิ่มเติมได้จาก **Marketplace** ช่วยเพิ่มขีดความสามารถให้กับเครื่องมือได้ตามความต้องการของนักพัฒนา

**VS Code** มีคุณสมบัติที่ช่วยเพิ่มประสิทธิภาพในการเขียนโค้ดอย่างครบครัน เช่น **IntelliSense** ซึ่งเป็นระบบช่วยเติมคำ (**Autocomplete**) และแนะนำคำสั่งที่เกี่ยวข้องในภาษาโปรแกรมที่ใช้งาน ช่วยลดเวลาและความผิดพลาดในการเขียนโค้ด นอกจากนี้ยังมี **Integrated Terminal** ที่ช่วยให้นักพัฒนาสามารถเรียกใช้คำสั่ง **CLI (Command Line Interface)** ภายในโปรแกรมได้ทันที โดยไม่ต้องสลับหน้าจอ รวมถึง **Git Integration** ที่ฝังมาในตัวโปรแกรม ช่วยให้นักพัฒนาสามารถจัดการกับระบบควบคุมเวอร์ชัน (**Version Control**) ได้อย่างสะดวก เช่น การ **Commit, Push, Pull** หรือดูความเปลี่ยนแปลงในไฟล์

อีกหนึ่งคุณสมบัติเด่นของ **VS Code** คือ **Debugging** ที่ช่วยให้นักพัฒนาสามารถตรวจสอบและแก้ไขข้อผิดพลาดในโค้ดได้อย่างมีประสิทธิภาพ โดยไม่จำเป็นต้องใช้เครื่องมือ **Debugging** ภายนอก **VS Code** ยังรองรับการทำงานร่วมกับ **Remote Development** ซึ่งช่วยให้นักพัฒนาสามารถเชื่อมต่อไปยังเซิร์ฟเวอร์หรือ **Docker Container** ได้โดยตรง และยังสามารถทำงานในสภาพแวดล้อมที่มีทรัพยากรจำกัด เช่น **WSL (Windows Subsystem for Linux)**

**VS Code** ยังได้รับความนิยมในฐานะเครื่องมือที่รองรับ **การพัฒนาแอปพลิเคชันแบบครบวงจร (Full-stack Development)** ด้วยการสนับสนุนส่วนเสริม เช่น **Live Server** สำหรับการพัฒนาเว็บแอปพลิเคชันแบบไลฟ์, **Prettier** สำหรับจัดระเบียบโค้ด, และ **ESLint** สำหรับตรวจสอบข้อผิดพลาดในการเขียนโค้ดตามมาตรฐาน ทั้งนี้การปรับแต่งส่วนติดต่อผู้ใช้งาน (**User Interface**) เช่น ธีม สีพื้นหลัง และการตั้งค่าคีย์ลัด ยังช่วยให้นักพัฒนาสามารถปรับแต่งเครื่องมือให้เหมาะกับสไตล์การทำงานของตนเอง

ด้วยความยืดหยุ่น ความเบา และฟีเจอร์ที่ครบครัน **VS Code** จึงกลายเป็นเครื่องมือที่ได้รับการยอมรับในวงการพัฒนาซอฟต์แวร์อย่างกว้างขวาง ไม่ว่าจะเป็นนักพัฒนามือใหม่หรือนักพัฒนามืออาชีพ ต่างเลือกใช้ **VS Code** เป็นเครื่องมือหลักในการเขียนโค้ดและพัฒนาโครงการต่าง ๆ ในหลากหลายขนาดและรูปแบบ

## คุณสมบัติของ VS Code

VS Code มีคุณสมบัติที่โดดเด่นหลายประการ เช่น

1. รองรับหลายภาษาโปรแกรม  
สามารถใช้งานได้กับหลายภาษาโปรแกรม เช่น JavaScript, Python, Java, C#, PHP, HTML, CSS เป็นต้น
2. ส่วนขยาย (Extensions)  
ผู้ใช้งานสามารถติดตั้งส่วนขยายเพิ่มเติมเพื่อเพิ่มความสามารถ เช่น การจัดรูปแบบโค้ด, การตรวจสอบข้อผิดพลาด, และการเชื่อมต่อกับเครื่องมืออื่น ๆ เช่น Git
3. เครื่องมือ Debugger  
มีเครื่องมือสำหรับการดีบั๊กที่ใช้งานง่ายและสามารถตรวจสอบการทำงานของโค้ดแบบเรียลไทม์
4. การปรับแต่ง UI  
ผู้ใช้งานสามารถปรับแต่งธีม สี หรือแม้กระทั่งคีย์ลัดเพื่อเพิ่มความสะดวกสบายในการทำงาน
5. การทำงานร่วมกับ Git  
VS Code มีการผนวกรวมกับ Git ช่วยให้ผู้ใช้สามารถจัดการการควบคุมเวอร์ชันของโค้ดได้โดยตรงในโปรแกรม
6. การสนับสนุน Live Share  
ผู้ใช้งานสามารถทำงานร่วมกับทีมในแบบเรียลไทม์ผ่านฟีเจอร์ Live Share

## การใช้งาน VS Code ในโครงการ

ในโครงการนี้ VS Code ถูกนำมาใช้เป็นเครื่องมือหลักสำหรับพัฒนาโปรแกรม เนื่องจากมีความสามารถดังต่อไปนี้

- การพัฒนา Front-End  
VS Code ช่วยให้การเขียน HTML, CSS และ JavaScript สะดวกยิ่งขึ้นผ่านส่วนขยายและการเติมคำอัตโนมัติ (Autocomplete)
- การพัฒนา Back-End  
สนับสนุนการเขียนโค้ด Node.js ซึ่งเป็นเทคโนโลยีที่ใช้ในโครงการ รวมถึงการเชื่อมต่อกับฐานข้อมูล MySQL

- การจัดการโค้ด

ใช้ฟีเจอร์ **Git** และ **GitHub** ในการจัดการเวอร์ชันของโค้ด ช่วยให้ทีมงานสามารถทำงานร่วมกันได้อย่างมีประสิทธิภาพ

### ประโยชน์ของการใช้ **VS Code**

การใช้ **VS Code** ในโครงการมีข้อดีดังนี้

- ลดเวลาในการพัฒนาโปรแกรมด้วยฟีเจอร์ **Autocomplete** และ **Debugger**
- เพิ่มความสะดวกในการทำงานร่วมกันของทีมด้วย **Git** และ **Live Share**
- รองรับส่วนขยายที่ช่วยปรับปรุงการเขียนโค้ดให้เหมาะสมกับโครงการ

จากคุณสมบัติและประโยชน์ข้างต้น **VS Code** จึงเป็นเครื่องมือที่เหมาะสมสำหรับการพัฒนาโครงการในครั้งนี้  
ทั้งในแง่ของประสิทธิภาพ ความสะดวก และการสนับสนุนเทคโนโลยีที่เกี่ยวข้องในโครงการ

### 2.1.5 XAMPP

**XAMPP** เป็นซอฟต์แวร์โอเพ่นซอร์สที่ได้รับความนิยมอย่างแพร่หลายสำหรับการพัฒนาและจำลองเซิร์ฟเวอร์บนเครื่องคอมพิวเตอร์ส่วนบุคคล ถูกออกแบบมาเพื่อช่วยให้นักพัฒนาและผู้ดูแลระบบสามารถสร้างและทดสอบเว็บแอปพลิเคชันในสภาพแวดล้อมที่ปลอดภัยก่อนนำไปใช้งานจริง ชื่อ **XAMPP** มาจากตัวอักษรย่อที่แสดงถึงองค์ประกอบหลักของซอฟต์แวร์ ได้แก่ **X** (Cross-platform) ซึ่งหมายถึงการรองรับหลายระบบปฏิบัติการ เช่น Windows, macOS, และ Linux, **A** (Apache) ซึ่งเป็นเว็บเซิร์ฟเวอร์สำหรับโฮสต์แอปพลิเคชัน, **M** (MariaDB) ซึ่งเป็นระบบจัดการฐานข้อมูล, **P** (PHP) และ **P** (Perl) ซึ่งเป็นภาษาสคริปต์ยอดนิยมสำหรับการพัฒนาแอปพลิเคชันแบบไดนามิก

**XAMPP** ได้รับการออกแบบให้ติดตั้งและใช้งานง่าย ช่วยให้นักพัฒนาสามารถเริ่มต้นเซิร์ฟเวอร์ได้ในไม่กี่คลิกผ่าน **XAMPP Control Panel** โดย Control Panel นี้ช่วยให้สามารถเริ่มต้นหรือหยุดการทำงานของเซิร์ฟเวอร์ เช่น Apache, MariaDB, และ FileZilla ได้อย่างสะดวก นอกจากนี้ยังมาพร้อมกับ **phpMyAdmin** ซึ่งเป็นเครื่องมือที่ช่วยในการจัดการฐานข้อมูลผ่านเว็บเบราว์เซอร์ ทำให้ง่ายต่อการสร้างแก้ไข และดูแลฐานข้อมูล MariaDB หรือ MySQL

**XAMPP** ยังเป็นเครื่องมือสำคัญสำหรับการพัฒนาเว็บแอปพลิเคชันในระยะเริ่มต้นหรือในระดับการทดสอบ เช่น การสร้างและปรับแต่งเว็บไซต์ด้วย WordPress การพัฒนา RESTful API ด้วย PHP หรือการทดสอบฟังก์ชันต่าง ๆ ของระบบก่อนการนำขึ้นเซิร์ฟเวอร์จริง นอกจากนี้ **XAMPP** ยังรองรับการพัฒนาแอปพลิเคชันที่ใช้ภาษา PHP ร่วมกับเฟรมเวิร์กยอดนิยม เช่น Laravel หรือ CodeIgniter ทำให้สามารถสร้างระบบที่มีประสิทธิภาพและตอบสนองความต้องการของผู้ใช้งานได้

อีกหนึ่งข้อได้เปรียบของ **XAMPP** คือการรองรับ การปรับแต่งและการทำงานร่วมกับซอฟต์แวร์อื่น ๆ เช่น การติดตั้งโมดูลเพิ่มเติมสำหรับการใช้งานพิเศษ หรือการปรับเปลี่ยนการตั้งค่าเซิร์ฟเวอร์เพื่อให้เหมาะสมกับโครงการ โดยเฉพาะสำหรับนักพัฒนาที่ต้องการจำลองการทำงานในสภาพแวดล้อมที่ใกล้เคียงกับเซิร์ฟเวอร์จริง นอกจากนี้ **XAMPP** ยังช่วยให้สามารถทดสอบการทำงานของเว็บไซต์ได้ในแบบออฟไลน์ (Localhost) โดยไม่จำเป็นต้องเชื่อมต่ออินเทอร์เน็ต

อย่างไรก็ตาม **XAMPP** มีข้อจำกัดบางประการ เช่น การตั้งค่าความปลอดภัยเริ่มต้นที่ไม่ได้เหมาะสมสำหรับเซิร์ฟเวอร์จริง (Production) ซึ่งต้องมีการปรับเปลี่ยนเพิ่มเติมก่อนนำไปใช้งานจริง นอกจากนี้ ในกรณีที่มีการใช้งานร่วมกับโครงการขนาดใหญ่ หรือโครงสร้างระบบที่ซับซ้อน อาจต้องพิจารณาใช้โซลูชันที่ปรับแต่งได้มากกว่า เช่น Docker หรือ Kubernetes

โดยสรุป **XAMPP** เป็นเครื่องมือที่ทรงพลังสำหรับนักพัฒนาที่ต้องการเริ่มต้นพัฒนาเว็บแอปพลิเคชันในสภาพแวดล้อมที่ง่ายต่อการติดตั้งและใช้งาน ด้วยความสามารถที่ครอบคลุมและความยืดหยุ่นในการปรับแต่ง **XAMPP** ยังคงเป็นตัวเลือกยอดนิยมสำหรับการพัฒนาระบบในหลากหลายรูปแบบ ไม่ว่าจะเป็นเว็บไซต์ขนาดเล็กหรือโครงการที่ต้องการความคล่องตัวในขั้นตอนการทดสอบและพัฒนา

### ลักษณะของ **XAMPP**

- ประกอบด้วย **Apache** สำหรับการทำงานเป็นเว็บเซิร์ฟเวอร์ **MariaDB** (ฐานข้อมูล), **PHP** และ **Perl** (ภาษาสคริปต์)
- รองรับการพัฒนาและทดสอบแอปพลิเคชันบนเซิร์ฟเวอร์จำลอง
- มีอินเทอร์เฟซการใช้งานครึ่งอัตโนมัติ ช่วยให้ผู้ใช้สามารถเริ่มเซิร์ฟเวอร์หรือหยุดการทำงานได้สะดวก
- ใช้งานร่วมกับเครื่องมือและแพลตฟอร์มพัฒนาเว็บได้หลากหลาย เช่น **WordPress**, **Joomla**, และ **Laravel**

### ประโยชน์ของ **XAMPP**

- ช่วยในการพัฒนาและทดสอบเว็บไซต์แบบโลคัล (Localhost) โดยไม่จำเป็นต้องเชื่อมต่อกับเซิร์ฟเวอร์จริง
- สามารถจัดการฐานข้อมูลผ่าน **phpMyAdmin** ซึ่งเป็นเครื่องมือบนเว็บสำหรับบริหารจัดการฐานข้อมูล
- รองรับการพัฒนาแอปพลิเคชันด้วย **PHP** และการใช้งานฐานข้อมูล **MariaDB** อย่างมีประสิทธิภาพ
- ช่วยให้นักพัฒนาสามารถทำการ **Debug** และตรวจสอบข้อผิดพลาดได้ในสภาพแวดล้อมที่ปลอดภัย

### องค์ประกอบสำคัญของ **XAMPP**

- **Apache**: เซิร์ฟเวอร์สำหรับโฮสต์และส่งข้อมูลไปยังเบราว์เซอร์
- **MariaDB**: ระบบจัดการฐานข้อมูลที่ใช้สำหรับเก็บข้อมูล
- **PHP**: ภาษาสคริปต์สำหรับพัฒนาเว็บไซต์แบบไดนามิก
- **phpMyAdmin**: อินเทอร์เฟซที่ใช้จัดการฐานข้อมูลผ่านเว็บ
- **Control Panel**: ส่วนกลางสำหรับควบคุมการทำงานของเซิร์ฟเวอร์

### 2.1.6 Node.js

**Node.js** เป็นแพลตฟอร์มสำหรับการพัฒนาแอปพลิเคชันฝั่งเซิร์ฟเวอร์ที่สร้างขึ้นบน **V8 JavaScript Engine** ซึ่งเป็นเอนจินเดียวกับที่ใช้ใน Google Chrome Node.js ถูกออกแบบมาให้รองรับการทำงานแบบ **Asynchronous** และ **Event-driven** ทำให้สามารถจัดการคำขอจำนวนมากพร้อมกันได้ อย่างมีประสิทธิภาพ โดยไม่ต้องสร้างเธรดใหม่สำหรับแต่ละคำขอ เช่นเดียวกับแพลตฟอร์มดั้งเดิมอื่น ๆ ด้วยเหตุนี้ Node.js จึงเป็นตัวเลือกลอดนิยมสำหรับการพัฒนาแอปพลิเคชันที่ต้องการความเร็ว เช่น ระบบแชทบอท เกมออนไลน์ แอปพลิเคชันเกม หรือบริการ RESTful API

จุดเด่นที่ทำให้ Node.js ได้รับความนิยมอย่างกว้างขวางคือการใช้ **JavaScript** ซึ่งเป็นภาษาที่นักพัฒนาจำนวนมากคุ้นเคยอยู่แล้ว ด้วย Node.js นักพัฒนาสามารถใช้ภาษาเดียวกันสำหรับทั้งฝั่งเซิร์ฟเวอร์และไคลเอนต์ ทำให้กระบวนการพัฒนาระบบเป็นไปอย่างต่อเนื่องและง่ายดาย นอกจากนี้ Node.js ยังมาพร้อมกับ **npm (Node Package Manager)** ซึ่งเป็นเครื่องมือสำหรับจัดการไลบรารีและแพ็คเกจที่ช่วยลดเวลาการพัฒนา โดยสามารถค้นหาและติดตั้งแพ็คเกจที่ต้องการจากคลัง npm ได้อย่างรวดเร็ว

Node.js มีโครงสร้างที่รองรับการทำงานแบบ **Non-blocking I/O** ซึ่งช่วยให้ระบบสามารถจัดการคำขอได้อย่างมีประสิทธิภาพโดยไม่ต้องรอการดำเนินการให้เสร็จก่อน เช่น การอ่านไฟล์จากดิสก์หรือการเชื่อมต่อฐานข้อมูล ข้อดีนี้ทำให้ Node.js เหมาะสำหรับระบบที่ต้องการประมวลผลข้อมูลแบบเรียลไทม์ เช่น การสตรีมวิดีโอหรือการส่งข้อมูลระหว่างอุปกรณ์แบบทันที

สำหรับการพัฒนาแอปพลิเคชันด้วย Node.js นักพัฒนาสามารถใช้เฟรมเวิร์กและไลบรารีมากมาย เช่น **Express.js** ซึ่งช่วยเพิ่มความสะดวกในการจัดการเส้นทาง (Routing) และสร้าง RESTful API, **Socket.IO** สำหรับการพัฒนาแอปพลิเคชันแบบเรียลไทม์ และ **Mongoose** สำหรับการเชื่อมต่อกับฐานข้อมูล MongoDB Node.js ยังรองรับการทำงานร่วมกับฐานข้อมูลประเภทต่าง ๆ เช่น MySQL, PostgreSQL, และ Redis ทำให้สามารถพัฒนาแอปพลิเคชันที่ซับซ้อนได้อย่างยืดหยุ่น

นอกจากนี้ Node.js ยังสนับสนุนการพัฒนาระบบแบบ **Microservices** ซึ่งช่วยให้นักพัฒนาสามารถแบ่งโครงการขนาดใหญ่ให้เป็นบริการขนาดเล็กที่ทำงานแยกกัน แต่สื่อสารกันผ่าน API ทำให้การจัดการระบบง่ายขึ้นและเพิ่มความสามารถในการปรับขยาย (Scalability) ได้ง่าย

แม้ Node.js จะมีข้อดีมากมาย แต่ก็มีข้อจำกัดบางประการ เช่น การทำงานที่ซับซ้อนในโค้ดแบบ Asynchronous อาจทำให้โค้ดอ่านยาก และไม่เหมาะสำหรับการประมวลผลหนัก ๆ เช่น การคำนวณทาง

คณิตศาสตร์ที่ซับซ้อน อย่างไรก็ตาม ด้วยชุมชนที่ใหญ่และเครื่องมือสนับสนุนที่หลากหลาย Node.js ยังคงเป็นตัวเลือกที่ยอดเยี่ยสำหรับการพัฒนาแอปพลิเคชันในยุคปัจจุบันที่เน้นความเร็วและประสิทธิภาพ

โดยสรุป Node.js เป็นแพลตฟอร์มที่มีประสิทธิภาพสูงสำหรับการพัฒนาแอปพลิเคชันที่หลากหลาย ด้วยการทำงานที่รวดเร็ว ความยืดหยุ่น และความสามารถในการรองรับคำขอจำนวนมาก Node.js จึงเป็นหนึ่งในเครื่องมือที่นักพัฒนาซอฟต์แวร์เลือกใช้มากที่สุดในยุคดิจิทัล

### ลักษณะของ Node.js

- ใช้ JavaScript เป็นภาษาหลักในการพัฒนา
- รองรับการทำงานแบบ **Asynchronous** (ไม่บล็อกการทำงานอื่นระหว่างรอคำสั่งที่ยังไม่เสร็จ)
- มีโมดูลและแพ็คเกจเสริมมากมายผ่าน **npm (Node Package Manager)**
- ออกแบบมาเพื่อการพัฒนาแอปพลิเคชันแบบเรียลไทม์ เช่น แชท แอปพลิเคชันสตรีมมิ่ง และ RESTful API

### ประโยชน์ของ Node.js

- ประสิทธิภาพสูง: Node.js สามารถจัดการกับคำขอจำนวนมากได้พร้อมกันโดยใช้ทรัพยากรเซิร์ฟเวอร์น้อย
- รองรับการพัฒนาเรียลไทม์: ใช้งานได้ดีในระบบที่ต้องการการอัปเดตข้อมูลแบบเรียลไทม์ เช่น ระบบแชทหรือเกม
- ข้ามแพลตฟอร์ม: สามารถพัฒนาแอปพลิเคชันที่รองรับหลายระบบปฏิบัติการได้ เช่น Windows, macOS, และ Linux
- เครื่องมือเสริมมากมาย: มีไลบรารีและแพ็คเกจสำเร็จรูปมากมายใน **npm** ที่ช่วยให้การพัฒนาง่ายและรวดเร็วยิ่งขึ้น

### องค์ประกอบสำคัญของ Node.js

- **Event Loop:** กลไกการจัดการคำสั่งแบบไม่บล็อกที่ช่วยให้ Node.js จัดการงานหลายงานพร้อมกันได้
- **npm (Node Package Manager):** ตัวจัดการแพ็คเกจที่ช่วยติดตั้ง ไลบรารี และเครื่องมือที่จำเป็น
- **Modules:** โมดูลที่ใช้ซ้ำได้ เช่น **fs** สำหรับการจัดการไฟล์ และ **http** สำหรับการสร้างเซิร์ฟเวอร์
- **Callback และ Promises:** วิธีการจัดการงานที่ทำงานแบบไม่บล็อก



## การใช้งานในระบบพัฒนาแอปพลิเคชัน

Node.js ถูกนำไปใช้งานอย่างแพร่หลายในระบบที่ต้องการประสิทธิภาพและความรวดเร็ว เช่น

- การพัฒนา RESTful API สำหรับเชื่อมต่อกับฐานข้อมูลและบริการอื่น ๆ
- การสร้างแอปพลิเคชันแบบเรียลไทม์ เช่น ระบบสนทนาหรือเกม
- การจัดการงานเบื้องหลัง (Backend tasks) เช่น การประมวลผลข้อมูลหรือการจัดเก็บข้อมูล

### 2.1.7 Express.js

Express.js เป็นเฟรมเวิร์กที่ทำงานร่วมกับ Node.js ซึ่งถูกออกแบบมาเพื่อช่วยให้นักพัฒนาสามารถสร้างเว็บแอปพลิเคชันและ API ได้อย่างง่ายดายและรวดเร็ว ด้วยความสามารถในการจัดการคำขอ HTTP, การกำหนดเส้นทาง (Routing), และการใช้งาน Middleware Express.js ได้กลายเป็นเฟรมเวิร์กยอดนิยมสำหรับการพัฒนาแอปพลิเคชันแบบ Backend ในยุคปัจจุบัน

จุดเด่นของ Express.js คือความเรียบง่ายและยืดหยุ่น นักพัฒนาสามารถสร้างแอปพลิเคชันด้วยโค้ดเพียงไม่กี่บรรทัด อีกทั้งยังรองรับการเพิ่มฟังก์ชันการทำงานเพิ่มเติมผ่าน Middleware ซึ่งช่วยให้สามารถจัดการคำขอ (Request) และคำตอบ (Response) ได้อย่างมีประสิทธิภาพ ตัวอย่างการใช้งาน Middleware เช่น การตรวจสอบสิทธิ์ผู้ใช้งาน (Authentication), การบันทึกกิจกรรมในระบบ (Logging), และการจัดการข้อผิดพลาด (Error Handling) นอกจากนี้ Express.js ยังรองรับการพัฒนา RESTful API ที่สามารถเชื่อมต่อและทำงานร่วมกับฐานข้อมูลหรือบริการอื่น ๆ ได้อย่างยืดหยุ่น

หนึ่งในคุณสมบัติที่โดดเด่นของ Express.js คือระบบการกำหนดเส้นทาง (Routing) ซึ่งช่วยให้นักพัฒนาสามารถจัดการ URL ต่าง ๆ ในแอปพลิเคชันได้อย่างง่ายดาย ตัวอย่างเช่น การกำหนดเส้นทางสำหรับแสดงข้อมูลผู้ใช้หรือการจัดการคำขอสำหรับการเพิ่ม แก้ไข หรือลบข้อมูล โดยการใช้งานร่วมกับ Node.js ทำให้ Express.js มีความสามารถในการจัดการคำขอหลายรายการพร้อมกันด้วยความเร็วสูง

นอกจากนี้ Express.js ยังรองรับการใช้งานร่วมกับ Template Engines เช่น EJS, Pug, หรือ Handlebars ซึ่งช่วยให้นักพัฒนาสามารถสร้างหน้า HTML แบบไดนามิกที่ดึงข้อมูลจากฐานข้อมูลมาแสดงผลได้อย่างสะดวก Express.js ยังสามารถเชื่อมต่อกับฐานข้อมูลได้หลากหลาย ไม่ว่าจะเป็น SQL เช่น MySQL หรือ PostgreSQL หรือ NoSQL เช่น MongoDB โดยใช้ไลบรารีเสริม เช่น Sequelize หรือ Mongoose

Express.js ยังเป็นเครื่องมือที่เหมาะสมสำหรับการพัฒนาแอปพลิเคชันที่ต้องการ Realtime Communication เมื่อใช้งานร่วมกับ Socket.IO นักพัฒนาสามารถสร้างแอปพลิเคชันที่มีการสื่อสารแบบเรียลไทม์ เช่น ระบบแชทหรือแอปพลิเคชันเกม ด้วยความสามารถในการประมวลผลคำขอที่รวดเร็วและการรองรับปริมาณข้อมูลจำนวนมาก

อย่างไรก็ตาม Express.js มีข้อจำกัดในเรื่องของการจัดโครงสร้างโค้ด โดยไม่มีระบบโครงสร้างที่ชัดเจนเหมือนเฟรมเวิร์กขนาดใหญ่ เช่น Laravel หรือ Django ดังนั้นนักพัฒนาต้องออกแบบและจัดการโครงสร้างโค้ดด้วยตนเองเพื่อให้เหมาะสมกับขนาดของโครงการ

โดยสรุป Express.js เป็นเฟรมเวิร์กที่มีประสิทธิภาพสูงสำหรับการพัฒนาเว็บแอปพลิเคชันและ API มีความเรียบง่าย ยืดหยุ่น และรองรับการปรับแต่งได้หลากหลาย ด้วยการใช้งานร่วมกับ Node.js และเครื่องมือเสริมอื่น ๆ Express.js ได้กลายเป็นตัวเลือกยอดนิยมสำหรับนักพัฒนาที่ต้องการสร้างแอปพลิเคชันที่มีความเร็วและประสิทธิภาพในยุคดิจิทัล

### คุณสมบัติเด่นของ Express.js

- ความเรียบง่ายและยืดหยุ่น: Express.js ออกแบบมาให้มีโครงสร้างที่เรียบง่าย รองรับการปรับแต่งตามความต้องการ
- รองรับ Middleware: สามารถเพิ่ม Middleware เพื่อตอบสนองคำขอ (Request) หรือจัดการคำตอบ (Response) ได้
- จัดการ Routing ได้ง่าย: ช่วยให้การกำหนดเส้นทางของ API และหน้าเว็บเป็นไปอย่างมีประสิทธิภาพ
- รองรับการพัฒนา API: สนับสนุนการพัฒนา RESTful API อย่างเต็มรูปแบบ
- เครื่องมือเสริมหลากหลาย: มีไลบรารีเสริมมากมายที่ช่วยเพิ่มประสิทธิภาพในการพัฒนา

### ประโยชน์ของ Express.js

1. ช่วยลดเวลาการพัฒนา: Express.js มีฟังก์ชันที่พร้อมใช้งาน ทำให้นักพัฒนาไม่ต้องเริ่มต้นจากศูนย์
2. ประสิทธิภาพสูง: สามารถจัดการคำขอจำนวนมากพร้อมกันได้โดยไม่กระทบต่อประสิทธิภาพ
3. รองรับแอปพลิเคชันหลากหลาย: สามารถใช้พัฒนาเว็บแอปพลิเคชัน, API, และบริการฝั่งเซิร์ฟเวอร์อื่น ๆ
4. เหมาะกับทีมพัฒนา: มีโครงสร้างที่เข้าใจง่าย ทำให้ทำงานร่วมกับทีมได้อย่างมีประสิทธิภาพ

### 2.1.7 EJS (Embedded JavaScript)

#### ภาพรวมของ EJS

EJS (Embedded JavaScript) เป็น Template Engine ที่ใช้สำหรับสร้างหน้าเว็บแบบไดนามิก โดยทำงานร่วมกับ Node.js และ Express.js ได้อย่างมีประสิทธิภาพ EJS ช่วยให้สามารถแทรกโค้ด JavaScript ลงใน HTML ได้อย่างง่ายดายและยืดหยุ่น ซึ่งเหมาะสำหรับการพัฒนาเว็บที่ต้องการส่งข้อมูลจากฝั่งเซิร์ฟเวอร์ไปยังหน้า HTML

#### คุณสมบัติเด่นของ EJS

- การแทรก JavaScript ใน HTML: สามารถใช้โค้ด JavaScript เพื่อจัดการข้อมูลภายในไฟล์ HTML ได้
- ง่ายต่อการเรียนรู้และใช้งาน: โครงสร้างที่เรียบง่ายทำให้เหมาะกับผู้พัฒนาที่เริ่มต้น
- รองรับ Partial Templates: สามารถแยกส่วนของเทมเพลตเพื่อใช้ซ้ำได้ เช่น Header, Footer
- สนับสนุนการแสดงผลแบบไดนามิก: สามารถส่งข้อมูลจากเซิร์ฟเวอร์เพื่อแสดงในหน้าเว็บแบบเรียลไทม์

#### ประโยชน์ของ EJS

1. ช่วยในการสร้างหน้า HTML แบบไดนามิก:
  - สามารถแสดงผลข้อมูลที่ได้จากฐานข้อมูลในรูปแบบ HTML ได้โดยตรง
  - ใช้โค้ด JavaScript เพื่อควบคุมการแสดงผล เช่น การวนลูปหรือการตรวจสอบเงื่อนไข
2. ลดความซับซ้อนในการจัดการเทมเพลต:
  - สามารถแยกส่วนโครงสร้าง HTML เช่น Header และ Footer เพื่อใช้ซ้ำในหลายหน้า
3. รองรับการทำงานร่วมกับ Express.js:
  - ใช้งานง่ายเมื่อทำงานร่วมกับเฟรมเวิร์ก Express.js ในการส่งข้อมูลไปยังไฟล์ EJS

## 2.1.8 JavaScript

### ภาพรวมของ JavaScript

JavaScript เป็นภาษาโปรแกรมที่ถูกออกแบบมาเพื่อพัฒนาเว็บแอปพลิเคชัน โดยเฉพาะการเพิ่มความสามารถแบบโต้ตอบ (Interactive) ให้กับหน้าเว็บ เป็นภาษาแบบไดนามิกที่ทำงานได้ทั้งบนฝั่งไคลเอนต์ (Client-side) และฝั่งเซิร์ฟเวอร์ (Server-side) JavaScript เป็นหนึ่งในเทคโนโลยีหลักของเว็บสมัยใหม่ ร่วมกับ HTML และ CSS

### คุณสมบัติเด่นของ JavaScript

- ภาษาแบบไดนามิก: สามารถเปลี่ยนแปลงโครงสร้างของหน้าเว็บหรือข้อมูลได้ทันทีโดยไม่ต้องรีเฟรชหน้า
- รองรับการทำงานแบบ Event-driven: ทำให้สามารถตอบสนองต่อการกระทำของผู้ใช้งาน เช่น การคลิกหรือการป้อนข้อมูล
- เข้าถึงและจัดการ DOM (Document Object Model): ช่วยให้สามารถเปลี่ยนแปลงเนื้อหาและโครงสร้าง HTML ได้
- รองรับการทำงาน Asynchronous: ช่วยให้แอปพลิเคชันสามารถดำเนินงานอื่นๆ ได้ระหว่างรอคำสั่งที่ยังไม่เสร็จ
- ข้ามแพลตฟอร์ม: ทำงานได้บนทุกเว็บเบราว์เซอร์ที่รองรับมาตรฐานสมัยใหม่

### ประโยชน์ของ JavaScript

1. เพิ่มความโต้ตอบให้กับหน้าเว็บ
  - สามารถสร้างแอปพลิเคชันที่มีการตอบสนองแบบเรียลไทม์ เช่น แชทออนไลน์หรือการอัปเดตข้อมูลสด
2. ลดภาระของเซิร์ฟเวอร์
  - การประมวลผลบางส่วนสามารถทำได้บนฝั่งไคลเอนต์ เช่น การตรวจสอบความถูกต้องของฟอร์ม

### 3. สนับสนุนการพัฒนา Single Page Application (SPA)

- ร่วมกับเฟรมเวิร์ก เช่น React.js, Angular.js หรือ Vue.js

### 4. รองรับการพัฒนาแอปพลิเคชันฝั่งเซิร์ฟเวอร์

- ด้วยแพลตฟอร์มอย่าง Node.js

## การใช้งาน JavaScript ในระบบปัจจุบัน

### 1. การพัฒนา Frontend:

- ใช้ในเว็บเบราว์เซอร์เพื่อจัดการ UI และการโต้ตอบของผู้ใช้
- ทำงานร่วมกับเฟรมเวิร์ก เช่น React.js, Angular, และ Vue.js

### 2. การพัฒนา Backend:

- ใช้ Node.js สำหรับพัฒนาเซิร์ฟเวอร์และ API
- รองรับฐานข้อมูล เช่น MongoDB, MySQL

### 3. การพัฒนาแอปพลิเคชันแบบ Cross-platform:

- ใช้เฟรมเวิร์กอย่าง Electron.js สำหรับแอปพลิเคชันเดสก์ท็อป
- ใช้ React Native สำหรับแอปพลิเคชันมือถือ

## ข้อจำกัดของ JavaScript

- การทำงานบางอย่างอาจขึ้นอยู่กับเบราว์เซอร์ ซึ่งอาจเกิดปัญหาความเข้ากันได้
- การทำงานบนฝั่งไคลเอนต์อาจไม่ปลอดภัย หากไม่มีการป้องกันที่เหมาะสม
- การจัดการโค้ดที่มีขนาดใหญ่และซับซ้อนอาจยากหากไม่มีโครงสร้างที่ดี

### 2.1.10 CSS (Cascading Style Sheets)

#### ภาพรวมของ CSS

CSS (Cascading Style Sheets) เป็นภาษาที่ใช้สำหรับกำหนดลักษณะการแสดงผลของหน้าเว็บ โดยทำงานร่วมกับ HTML เพื่อจัดรูปแบบ เช่น สี ขนาดฟอนต์ ระยะห่าง และการจัดวางเลย์เอาต์ CSS ช่วยให้เว็บไซต์มีความสวยงามและเป็นระเบียบ อีกทั้งยังแยกส่วนโครงสร้างเนื้อหา (HTML) และการออกแบบ (CSS) ออกจากกัน

#### คุณสมบัติเด่นของ CSS

- ความยืดหยุ่น: สามารถกำหนดรูปแบบการแสดงผลให้กับเอกสาร HTML ได้หลากหลาย
- ความสามารถในการควบคุมเลย์เอาต์: รองรับการจัดวางองค์ประกอบ เช่น Flexbox, Grid
- การกำหนดรูปแบบซ้ำได้: สไตล์ที่กำหนดใน CSS สามารถนำไปใช้ซ้ำในหลายหน้าได้
- การรองรับ Responsive Design: ปรับการแสดงผลของเว็บไซต์ให้เหมาะสมกับหน้าจอขนาดต่าง ๆ

#### ประโยชน์ของ CSS

1. เพิ่มความสวยงามให้กับเว็บไซต์
  - ช่วยกำหนดสี ขนาดฟอนต์ การจัดเรียง และรูปแบบต่าง ๆ ให้เว็บไซต์ดูเป็นมืออาชีพ
2. ลดความซับซ้อนของโค้ด HTML
  - แยกส่วนของการออกแบบออกจากโครงสร้างเนื้อหา
3. ช่วยประหยัดเวลา
  - ใช้ไฟล์ CSS เดียวกันกับหลายหน้า HTML ทำให้การแก้ไขทำได้สะดวก
4. รองรับการออกแบบ Responsive
  - ปรับเปลี่ยนการแสดงผลอัตโนมัติตามอุปกรณ์ เช่น สมาร์ทโฟนหรือแท็บเล็ต

### 2.1.11 ระบบร้านขายยาในปัจจุบัน

ภาพรวมของระบบร้านขายยาในปัจจุบัน

ระบบร้านขายยาในปัจจุบันมีความสำคัญอย่างมากต่อการให้บริการด้านสุขภาพ เนื่องจากเป็นจุดเชื่อมต่อระหว่างผู้ป่วยและยาที่ได้รับจากการจ่ายของแพทย์หรือเภสัชกร ร้านขายยานอกจากจะเป็นสถานที่จำหน่ายยาแล้ว ยังมีบทบาทในการให้คำแนะนำเกี่ยวกับการใช้ยา รวมถึงผลิตภัณฑ์สุขภาพอื่น ๆ อย่างถูกต้องและปลอดภัย

#### รูปแบบการจัดการร้านขายยา

ปัจจุบันร้านขายยามีการจัดการทั้งในรูปแบบดั้งเดิม (Manual) และแบบดิจิทัล (Digital) โดยแบ่งออกได้ดังนี้:

##### 1. ร้านขายยาแบบดั้งเดิม (Traditional Pharmacy)

- ใช้กระบวนการบันทึกข้อมูลและจัดการสต็อกด้วยระบบเอกสาร
- การค้นหาข้อมูลยาหรือการเช็คสต็อกทำด้วยแรงงานคน ซึ่งอาจเกิดความผิดพลาดได้ง่าย
- เหมาะสำหรับร้านขายยาขนาดเล็กที่มีรายการสินค้าและจำนวนลูกค้าไม่มาก

##### 2. ร้านขายยาแบบดิจิทัล (Digital Pharmacy)

- ใช้ระบบจัดการร้านขายยา (Pharmacy Management System) เพื่อช่วยบันทึกข้อมูลการขาย การสั่งซื้อ และการจัดการสต็อก
- มีระบบแจ้งเตือนเมื่อสต็อกยาต่ำหรือยากำลังหมดอายุ
- เชื่อมต่อกับระบบฐานข้อมูลเพื่ออัปเดตข้อมูลยาใหม่หรือข้อกำหนดของหน่วยงานที่เกี่ยวข้อง
- รองรับการขายออนไลน์ เช่น การสั่งซื้อผ่านเว็บไซต์หรือแอปพลิเคชัน

#### เทคโนโลยีที่ใช้ในระบบร้านขายยา

ระบบร้านขายยาในปัจจุบันได้นำเทคโนโลยีต่าง ๆ มาใช้เพื่อเพิ่มความสะดวกและลดข้อผิดพลาด ได้แก่:

##### 1. ระบบจัดการฐานข้อมูล (Database Management System)

- ช่วยในการบันทึกข้อมูลยา ผู้ผลิต ผู้จำหน่าย และประวัติการขาย
- ทำให้การค้นหาข้อมูลง่ายและรวดเร็ว



## 2. ระบบแจ้งเตือนอัตโนมัติ (Automatic Alerts)

- แจ้งเตือนสต็อกต่ำหรือยาที่ใกล้หมดอายุ
- ช่วยลดความสูญเสียจากยาหมดอายุที่ไม่สามารถขายได้

## 3. ระบบขายหน้าร้าน (Point of Sale: POS)

- เชื่อมต่อกับระบบจัดการสต็อกเพื่ออัปเดตข้อมูลแบบเรียลไทม์
- รองรับการชำระเงินแบบดิจิทัล เช่น QR Code, บัตรเครดิต

## 4. ระบบออนไลน์

- การขายยาและผลิตภัณฑ์สุขภาพผ่านเว็บไซต์หรือแอปพลิเคชัน
- รองรับการจัดส่งยาถึงบ้าน เพิ่มความสะดวกให้ลูกค้า

ปัญหาและข้อจำกัดของระบบร้านขายยาในปัจจุบัน

แม้ระบบร้านขายยาจะพัฒนาไปมาก แต่ยังคงมีปัญหาและข้อจำกัดในบางส่วน ได้แก่:

- การปรับตัวของร้านขายยาแบบดั้งเดิมสู่ระบบดิจิทัลที่ต้องใช้ต้นทุนสูง
- การบันทึกข้อมูลที่ไม่เป็นระบบ อาจเกิดข้อผิดพลาดในการจ่ายยา
- ปัญหาด้านความปลอดภัยและความเป็นส่วนตัวของข้อมูลผู้ป่วย

## 2.2 งานวิจัยที่เกี่ยวข้อง

### การทิ้งและกำจัดยาที่ถูกต้อง

อาจารย์ ดร.ภก.บรรมพจน์ พฤตวินาสัณห์ ภาควิชาเภสัชเคมี คณะเภสัชศาสตร์ มหาวิทยาลัยมหิดล

ปัญหาขยะตกค้างในแหล่งธรรมชาติต่างๆ เช่น แม่น้ำ ลำคลอง ทะเลสาบ หรือพื้นดิน สามารถพบเจอได้ทั่วโลก ผลกระทบของยาตกค้างในธรรมชาติที่ชัดเจน คือ การทำลายระบบนิเวศของสิ่งมีชีวิตที่อาศัยในธรรมชาติ นั้น ซึ่งสามารถส่งผลกระทบที่เป็นลูกโซ่ต่อมายังมนุษย์ได้ สาเหตุหนึ่งที่สำคัญของปัญหาดังกล่าว คือ การทิ้งและทำลายยาอย่างไม่ถูกวิธี ในอดีตเรามักจะได้ยินว่าการทิ้งยาสามารถทำได้โดยการบดยาให้เป็นผงและทิ้งลงถังขยะเพื่อนำไปฝังกลบหรือนำไปละลายน้ำและทิ้งลงสู่ระบบบำบัดน้ำเสีย แต่ในปัจจุบันพบว่า การกระทำดังกล่าวเป็นสาเหตุที่ทำให้เกิดปัญหาขยะตกค้างในแหล่งธรรมชาติอย่างชัดเจน เนื่องจากยาบางชนิดสามารถผ่านระบบบำบัดและเล็ดลอดสู่ธรรมชาติได้โดยไม่ถูกกำจัดทิ้ง สำหรับยาที่ถูกฝังกลบจะละลายออกมาจากพื้นดินลงสู่แหล่งน้ำตามธรรมชาติได้ต่อไปหากพิจารณาแล้ว ยา คือ สารเคมีทั่วไปที่มีฤทธิ์ในการรักษาโรค ดังนั้น การกำจัดยาก็สามารถใช้หลักการการกำจัดของเสียสารเคมีซึ่งนิยมใช้การเผาที่อุณหภูมิสูงร่วมกับการฝังกลบอย่างถูกวิธี ปัจจุบัน ศูนย์พัฒนานโยบายแห่งชาติด้านสารเคมี กองแผนงานและวิชาการ สำนักงานคณะกรรมการอาหารและยา สำนักอนามัยสิ่งแวดล้อม กรมอนามัย และ คณะเภสัชศาสตร์ มหาวิทยาลัยมหิดล ได้ร่วมจัดทำแนวปฏิบัติการจัดการของเสียทางการแพทย์ที่เป็นสารเคมี ซึ่งมุ่งเน้นไปยังของเสีย 3 ชนิดหลัก ได้แก่ ยาและเวชภัณฑ์ ของเสียที่เกิดจากสารเคมีในห้องปฏิบัติการทางการแพทย์ และของเสียพวกเครื่องมือแพทย์ที่มีปริมาณโลหะหนักสูง แนวปฏิบัติดังกล่าวได้แนะนำการทำลายยาและเวชภัณฑ์ที่หมดอายุโดยการเผาทำลายทิ้งที่อุณหภูมิตั้งแต่ 850 ถึง 1,600 องศาเซลเซียส โดยอุณหภูมิที่ใช้จะขึ้นกับประเภทและความเป็นอันตรายของยา หลังจากนั้นจึงค่อยนำกากของเสียหลังจากการเผาไหม้ไปฝังกลบอย่างถูกวิธี

## บทที่ 3

### วิธีดำเนินการวิจัย

#### วิธีการดำเนินการวิจัย

การวิจัยนี้เป็นการพัฒนาซอฟต์แวร์เพื่อการจัดการร้านขายยา โดยมีขั้นตอนการดำเนินการวิจัยที่แบ่งออกเป็น 7 ขั้นตอนหลัก ดังนี้:

#### 3.1 การออกแบบฐานข้อมูล



ตารางที่สำคัญในระบบร้านขายยา:

#### 1. ตาราง **products** – เก็บข้อมูลสินค้าทั้งหมดในร้าน

Attribute	ความหมาย
product_id	รหัสสินค้า (Primary Key) ใช้ระบุสินค้าแต่ละตัวแบบไม่ซ้ำกัน
product_name	ชื่อของสินค้า/ยา
ingredients	ส่วนประกอบของยา เช่น Paracetamol, Ibuprofen
price	ราคาขายต่อหน่วยของสินค้า
stock_quantity	จำนวนสินค้าที่เหลือในคลัง
expiry_date	วันที่หมดอายุของสินค้า

Attribute	ความหมาย
-----------	----------

warning_conditions	เงื่อนไขการใช้จ่าย เช่น ห้ามใช้กับผู้ป่วยความดันสูง
--------------------	---

## 2. ตาราง customers – เก็บข้อมูลลูกค้า

Attribute	ความหมาย
-----------	----------

customer_id	รหัสลูกค้า (Primary Key)
-------------	--------------------------

first_name	ชื่อลูกค้า
------------	------------

last_name	นามสกุลลูกค้า
-----------	---------------

email	อีเมลของลูกค้า
-------	----------------

phone	เบอร์โทรศัพท์ลูกค้า
-------	---------------------

membership_id	รหัสสมาชิก (Foreign Key อ้างถึง memberships)
---------------	--

## 3. ตาราง orders – เก็บข้อมูลการสั่งซื้อ

Attribute	ความหมาย
-----------	----------

order_id	รหัสคำสั่งซื้อ (Primary Key)
----------	------------------------------

customer_id	รหัสลูกค้าที่สั่งซื้อ (Foreign Key)
-------------	-------------------------------------

order_date	วันที่ทำการสั่งซื้อ
------------	---------------------

total_amount	ยอดรวมทั้งหมดของคำสั่งซื้อ
--------------	----------------------------

payment_method	วิธีการชำระเงิน เช่น เงินสด, QR code, บัตรเครดิต
----------------	--

status	สถานะของคำสั่งซื้อ เช่น เสร็จสิ้น, รอดำเนินการ, ยกเลิก
--------	--

## 4. ตาราง order\_items – รายการสินค้าที่อยู่ในคำสั่งซื้อ

Attribute	ความหมาย
-----------	----------

Attribute	ความหมาย
-----------	----------

order_item_id	รหัสของรายการสินค้า (Primary Key)
---------------	-----------------------------------

order_id	รหัสคำสั่งซื้อที่รายการนี้เป็นของ (Foreign Key)
----------	---

product_id	รหัสสินค้าที่สั่งซื้อ (Foreign Key)
------------	-------------------------------------

quantity	จำนวนที่ซื้อ
----------	--------------

unit_price	ราคาต่อหน่วยของสินค้านั้น ณ เวลาที่ซื้อ
------------	---

total_price	ราคารวมของสินค้านี้ (quantity × unit_price)
-------------	---

## 5. ตาราง memberships – เก็บข้อมูลประเภทสมาชิก

Attribute	ความหมาย
-----------	----------

membership_id	รหัสประเภทสมาชิก (Primary Key)
---------------	--------------------------------

membership_level	ระดับสมาชิก เช่น Silver, Gold, Platinum
------------------	---

discount_rate	เปอร์เซ็นต์ส่วนลดที่ได้รับ เช่น 0.10 (แปลว่า 10%)
---------------	---

## 6. ตาราง alerts – เก็บคำเตือนเกี่ยวกับสินค้า

Attribute	ความหมาย
-----------	----------

alert_id	รหัสคำเตือน (Primary Key)
----------	---------------------------

product_id	รหัสสินค้าที่เกิดการแจ้งเตือน (Foreign Key)
------------	---

alert_type	ประเภทการแจ้งเตือน เช่น “สินค้าใกล้หมด”, “ใกล้หมดอายุ”
------------	--

alert_date	วันที่ที่มีการสร้างคำเตือน
------------	----------------------------

ความสัมพันธ์ระหว่างตาราง:

- **customers** มีความสัมพันธ์ **many-to-one** กับ **memberships** (ลูกค้าสามารถมีระดับสมาชิกหนึ่งรายการ)
- **orders** มีความสัมพันธ์ **many-to-one** กับ **customers** (ลูกค้าสามารถมีคำสั่งซื้อหลายคำสั่ง)
- **order\_items** มีความสัมพันธ์ **many-to-one** กับ **orders** และ **products** (คำสั่งซื้อสามารถมีหลายรายการสินค้า, และแต่ละรายการสินค้าเชื่อมกับสินค้าในตาราง **products**)
- **products** มีความสัมพันธ์ **many-to-many** กับ **alerts** (สินค้าสามารถมีหลายคำเตือน เช่น หมดอายุ หรือใกล้หมดอายุ)

### 3.2 การพัฒนาส่วน Backend

#### 1. ใช้ Node.js และ Express.js:

- ออกแบบโครงสร้าง **RESTful API** เพื่อรองรับการเชื่อมต่อกับฐานข้อมูลและการทำงานของฟังก์ชันต่าง ๆ
- ใช้ **MySQL** เป็นระบบจัดการฐานข้อมูล โดยสร้างตารางข้อมูลสำหรับสินค้า, การขาย, และสมาชิก

#### 2. ฟังก์ชันการทำงานของระบบ:

- การเพิ่มสินค้าใหม่, อัปเดตข้อมูลสินค้า และลบสินค้า
- การจัดการข้อมูลการขาย รวมถึงการคำนวณยอดขายแบบเรียลไทม์
- การตรวจสอบและแจ้งเตือนสินค้าใกล้หมดหรือใกล้หมดอายุผ่าน **Cron Jobs**

#### 3. การควบคุมคุณภาพซอร์สโค้ด:

- ใช้ **Git** และ **GitHub** สำหรับจัดการเวอร์ชันและการเปลี่ยนแปลงซอร์สโค้ด
- ใช้ **SonarQube** เพื่อตรวจสอบคุณภาพของซอร์สโค้ด หาข้อบกพร่องและปรับปรุงให้โค้ดมีคุณภาพ

### 3.3 การพัฒนาส่วน Frontend

#### 1. ใช้ EJS (Embedded JavaScript):

- สร้างหน้า HTML แบบไดนามิก เพื่อแสดงข้อมูลสินค้าหรือรายงานการขาย
- เชื่อมต่อกับ Backend ผ่าน API เพื่อดึงข้อมูลที่ต้องการแสดงผล

#### 2. การออกแบบ UI/UX:

- ใช้ CSS และ Bootstrap เพื่อออกแบบหน้าเว็บให้ใช้งานง่ายและรองรับการแสดงผลบนอุปกรณ์ต่างๆ (Responsive Design)
- เพิ่มฟังก์ชันการค้นหาสินค้าและการกรองข้อมูลที่ใช้งานง่าย

#### 3. การทดสอบการทำงานของหน้าเว็บ:

- ทดสอบฟังก์ชันการทำงานของปุ่มต่างๆ เช่น เพิ่มสินค้า, ลบสินค้า และการแสดงข้อมูลในตาราง

### 3.4 การจัดการฐานข้อมูล

#### 1. การออกแบบโครงสร้างฐานข้อมูล:

- สร้างตารางที่จำเป็น เช่น products, sales, members, และ notifications
- ออกแบบความสัมพันธ์ระหว่างตารางเพื่อให้สามารถดึงข้อมูลแบบซับซ้อนได้

#### 2. การตรวจสอบความถูกต้องของข้อมูล:

- ใช้ SQL Scripts เพื่อตรวจสอบความถูกต้องของข้อมูล เช่น ความสมบูรณ์ของ Schema ระหว่างฐานข้อมูลพัฒนา (Development) และฐานข้อมูลใช้งานจริง (Production)

3.5 การทดสอบระบบ

รายงานการทดสอบ Unit Testing

โครงการ: ระบบร้านขายยา

ผู้จัดทำ: วรทา มิ่งพรวัฒน์

วันที่ทดสอบ: 30 เมษายน 2025

1. วัตถุประสงค์

เพื่อทดสอบการทำงานของแต่ละหน่วย (Unit) ของระบบร้านขายยา ว่าสามารถทำงานได้ถูกต้องตามที่ออกแบบไว้ ครอบคลุมทุกฟังก์ชันหลัก เช่น ระบบขายสินค้า ระบบตรวจสอบสต็อก ระบบสมาชิก ระบบแจ้งเตือนสินค้าใกล้หมดอายุ เป็นต้น

2. รายการ Unit ที่ทดสอบ

ลำดับ ชื่อ Unit/Module	คำอธิบาย
1 ระบบค้นหาสินค้า	ค้นหาชื่อสินค้า/ยาจากฐานข้อมูล
2 ระบบเพิ่มสินค้าลงตะกร้า	เพิ่มสินค้าที่เลือกจากหน้าค้นหาลงในตะกร้า
3 ระบบคิดเงิน	คำนวณราคารวม คิดส่วนลดตามระดับสมาชิก
4 ระบบบันทึกการขาย	บันทึกข้อมูลการขายลงฐานข้อมูล orders และ order_items
5 ระบบแจ้งเตือนสต็อกต่ำ/สินค้าใกล้หมดอายุ	แจ้งเตือนถ้าสต็อกต่ำกว่าเกณฑ์ หรือสินค้าใกล้หมดอายุ
6 ระบบตรวจสอบการแพ้ยา	ตรวจสอบยาในตะกร้าว่ามีข้อห้ามตามโรคประจำตัวหรือไม่
7 ระบบแนะนำยาใกล้เคียง	แนะนำยาที่มีส่วนผสมคล้ายกับสินค้าที่ค้นหา
8 ระบบจัดการสมาชิก	เพิ่ม/แก้ไข/ค้นหาข้อมูลสมาชิก



### 3. สรุปผลการทดสอบ

#### 1. ชื่อ Unit/Module: ระบบค้นหาสินค้า

คำอธิบาย: ค้นหาซื้อสินค้า/ยาจากฐานข้อมูล

Input: คำค้นหาจากผู้ใช้

Process: ค้นหาข้อมูลในตาราง **products** โดยใช้คำค้นแบบ **LIKE**

Output: แสดงรายการสินค้าที่ตรงกับคำค้น

ค้นหา

ค้นหา

หน้าหลัก เด็ดยา แสดงยา เพิ่มสมาชิก แสดงสมาชิก แจ้งเตือน

ผลการค้นหา:

tylenol

จำนวน: 

เพิ่มลงในรถเข็น

ยาที่มีส่วนผสมเหมือนกัน::

akamol - ส่วนผสม: paracetamol

akamol - ส่วนผสม: paracetamol

sara - ส่วนผสม: paracetamol

sara - ส่วนผสม: paracetamol

sara - ส่วนผสม: paracetamol

#### 2. ชื่อ Unit/Module: ระบบเพิ่มสินค้าลงตะกร้า

คำอธิบาย: เพิ่มสินค้าที่เลือกจากหน้าค้นหาลงในตะกร้า

Input: **product\_id** และจำนวนสินค้าที่ผู้ใช้เลือก

Process: ดึงข้อมูลสินค้าจากฐานข้อมูล แล้วเก็บไว้ใน **session**

Output: แสดงรายการสินค้าในตะกร้าของผู้ใช้

ชื่อสินค้า	ราคาต่อหน่วย	จำนวน	ราคารวม
tylenol	50.00	1	50

ไปยังหน้าชำระเงิน

### 3. ชื่อ Unit/Module: ระบบคิดเงิน

คำอธิบาย: คำนวณราคารวมพร้อมส่วนลดจากระดับสมาชิก

Input: รายการสินค้าในตะกร้า, `membership_id` ของลูกค้า

Process: รวมราคาสินค้า คำนวณส่วนลดจากตาราง `memberships`

Output: แสดงราคารวมหลังหักส่วนลด

หน้าจ่ายเงิน

ชื่อสินค้า	จำนวน	ราคา	รวม
tylenol	1	50.00	50

ยอดรวมทั้งหมด: 50 บาท

ค่านำลูกค้า:

กรณียกเลิก

ส่งส่วนลด | แก้ไขใบเสร็จ

ตรวจสอบรายการในตะกร้าของคุณ

ส่วนลด: 0%

ยอดสุทธิหลังหักส่วนลด: 50 บาท

เงินโอน: 50.00 บาท

จำนวนเงินที่ใส่ไว้:

100

เมื่อกรณียกเลิกชำระเงิน:

เงินสด

ดำเนินการเสร็จ

### 4. ชื่อ Unit/Module: ระบบบันทึกการขาย

คำอธิบาย: บันทึกข้อมูลการขายลงในฐานข้อมูล

Input: รายการสินค้า, `customer_id`, วิธีชำระเงิน

Process: บันทึกคำสั่งซื้อใน `orders` และ `order_items`

Output: แจ้งเตือนว่า "บันทึกคำสั่งซื้อสำเร็จ"

✓ ขอขอบคุณสำหรับการสั่งซื้อ!

คำสั่งซื้อของคุณถูกบันทึกเรียบร้อยแล้ว

กลับไปหน้าแรก

5. ชื่อ Unit/Module: ระบบแจ้งเตือนสต็อกต่ำ/สินค้าใกล้หมดอายุ

คำอธิบาย: แจ้งเตือนเมื่อสินค้าคงเหลือน้อยหรือใกล้หมดอายุ

Input: - (ระบบทำงานอัตโนมัติ)

Process: ตรวจสอบสต็อกจาก products เมื่อปริมาณต่ำกว่ากำหนด หรือใกล้หมดอายุ

Output: รายการแจ้งเตือนสินค้าที่ควรเติมหรือกำลังหมดอายุ

หน้าหลัก เดิมยา แสดงยา เพิ่มสมาชิก แสดงสมาชิก แจ้งเตือน		
ไอดีสินค้า	ชนิดการแจ้งเตือน	วันที่แจ้งเตือน
11	สินค้าใกล้หมด	2025-06-08T09:54:11.682Z
11	สินค้าใกล้หมดอายุ	2025-06-08T09:54:11.683Z
13	สินค้าใกล้หมดอายุ	2025-06-08T09:54:11.685Z

6. ชื่อ Unit/Module: ระบบตรวจสอบการแพ้ยา

คำอธิบาย: ตรวจสอบว่าสินค้าในตะกร้ามีข้อห้ามตามโรคประจำตัวของลูกค้าหรือไม่

Input: customer\_id และรายการ product\_id ที่เลือก

Process: เปรียบเทียบ chronic\_diseases จากลูกค้ากับ warning\_conditions ของยา

Output: แจ้งเตือนหากพบว่ายาสอดคล้องกับโรคประจำตัว

หน้าชำระเงิน

ชื่อสินค้า	จำนวน	ราคา	รวม
sara	1	50.00	50

ยอดรวมทั้งหมด: 50 บาท

คืนภาษีมูลค่า:

ยอดรวม

ส่วนลด

ส่วนลดภาษีมูลค่าเพิ่ม

▲ "sara" ห้ามใช้กับ: ความดัน

ส่วนลด: 0%

ยอดสุทธิหลังหักส่วนลด: 50 บาท

เงินทอน: 50.00 บาท

จำนวนเงินที่ใส่ได้:

100

เลือกวิธีการชำระเงิน:

เงินสด

ยืนยันการสั่งซื้อ

## 7. ชื่อ Unit/Module: ระบบแนะนำยาใกล้เคียง

คำอธิบาย: แนะนำยาที่มีส่วนผสมคล้ายกับสินค้าที่ค้นหา

Input: product\_id หรือ keyword ที่ค้นหา

Process: ดึง ingredients จากสินค้าที่ค้นหา แล้วหาหาอื่นที่มี ingredient เดียวกัน

Output: แสดงรายการยาที่มีส่วนผสมใกล้เคียง

**ค้นหา**

[หน้าหลัก](#) [ค้นหา](#) [แสดงยา](#) [เพิ่มสมาชิก](#) [แสดงสมาชิก](#) [แจ้งเดือน](#)

**ผลการค้นหา:**

- tylenol

**ยาที่มีส่วนผสมเหมือนกัน:**

- akamol - ส่วนผสม: paracetamol
- akamol - ส่วนผสม: paracetamol
- sara - ส่วนผสม: paracetamol
- sara - ส่วนผสม: paracetamol
- sara - ส่วนผสม: paracetamol

## 8. ชื่อ Unit/Module: ระบบจัดการสมาชิก

คำอธิบาย: เพิ่ม แก้ไข และค้นหาข้อมูลสมาชิก

Input: ข้อมูลสมาชิก เช่น ชื่อ เบอร์โทร ระดับสมาชิก ฯลฯ

Process: บันทึกหรือแก้ไขข้อมูลในตาราง customers

Output: สถานะการจัดเก็บข้อมูล หรือแสดงข้อมูลสมาชิกที่ค้นหา

[หน้าหลัก](#) [ค้นหา](#) [แสดงยา](#) [เพิ่มสมาชิก](#) [แสดงสมาชิก](#) [แจ้งเดือน](#)

ชื่อจริง

นามสกุล

เบอร์โทร

อีเมล

ที่อยู่

อาชีพ

น้ำหนัก

เพศ

รหัสประจำตัว

รหัสสมาชิก:

เลือกสมาชิก

## บทที่ 4

### ผลการดำเนินงาน

ในโครงการนี้ได้ทำการพัฒนาและทดสอบระบบ **Pharmacy Management System** ที่เน้นการบริหารจัดการข้อมูลยาและการคำนวณโดสยาโดยอิงจากน้ำหนักของลูกค้า ระบบนี้มีการทำงานที่ครอบคลุมในหลายฟังก์ชันที่สำคัญ ได้แก่ การจัดการข้อมูลลูกค้า การคำนวณยอดรวมหลังหักส่วนลด และการตรวจสอบยาเพื่อหลีกเลี่ยงอันตรายจากโรคประจำตัวของลูกค้า

1. การคำนวณโดสยา: ระบบสามารถดึงข้อมูลน้ำหนักของลูกค้าจากชื่อที่กรอกและคำนวณโดสยาที่เหมาะสมได้อย่างถูกต้อง โดยระบบได้ทำการทดสอบในกรณีที่ลูกค้ากรอกน้ำหนักที่มีในระบบฐานข้อมูล ระบบสามารถแจ้งเตือนและให้คำแนะนำได้อย่างแม่นยำโดยอ้างอิงจากสมการ  $\text{Dose (mg)} = \text{Dose per kg (mg/kg)} \times \text{Weight (kg)}$
2. การดึงข้อมูลส่วนลด: ฟังก์ชันการดึงส่วนลดจากฐานข้อมูลตามชื่อของลูกค้าได้รับการทดสอบอย่างสมบูรณ์ โดยระบบจะตรวจสอบข้อมูลสมาชิกและคำนวณส่วนลดที่ลูกค้าจะได้รับตามประเภทของสมาชิก จากนั้นแสดงผลลัพธ์ให้ผู้ใช้งานเห็นในหน้า **Checkout**
3. การตรวจสอบยาและโรคประจำตัว: ระบบสามารถตรวจสอบยาที่อาจมีอันตรายหรือไม่เหมาะสมกับลูกค้าโดยการเปรียบเทียบโรคประจำตัวของลูกค้ากับคำเตือนในยาได้อย่างถูกต้อง หากพบยาใดที่มีข้อห้ามสำหรับโรคประจำตัว ระบบจะแจ้งเตือนผู้ใช้งานทันที
4. การทำงานร่วมกับฐานข้อมูล: ระบบได้เชื่อมต่อกับฐานข้อมูล **MySQL** เพื่อดึงข้อมูลที่จำเป็น เช่น ข้อมูลลูกค้า ข้อมูลสินค้า ข้อมูลส่วนลด และข้อมูลคำแนะนำโดสยา ระบบสามารถดึงข้อมูลเหล่านี้และแสดงผลได้ในระยะเวลาที่เหมาะสม โดยมีการจัดการฐานข้อมูลอย่างมีประสิทธิภาพ
5. ประสิทธิภาพของระบบ: ระบบสามารถทำงานได้อย่างรวดเร็วและไม่พบปัญหาในกระบวนการหลัก เช่น การดึงข้อมูลจากฐานข้อมูล การคำนวณยอดรวมหลังหักส่วนลด หรือการตรวจสอบยาและโรคประจำตัว แม้ว่าจะมีข้อจำกัดบางประการที่ต้องพิจารณาในการพัฒนาระบบให้ดียิ่งขึ้นในอนาคต

โดยรวมแล้ว โครงการนี้มีความสำเร็จในด้านการพัฒนาระบบที่ตอบโจทย์การบริหารจัดการข้อมูลยาและบริการที่เกี่ยวข้องกับลูกค้าได้ดี มีการพัฒนาให้ระบบทำงานอย่างครบถ้วนและตอบสนองต่อความต้องการของผู้ใช้งานได้อย่างมีประสิทธิภาพ

	แอปพลิเคชันของ โครงการนี้	bsPOS	Hygeia Pharmacy
จำนวนคลิกที่ใช้ในการ ซื้อสินค้าใน1ครั้ง	7	7	8
มีฟังก์ชันคำนวณโดสยา จากน้ำหนักตัว	มี	ไม่มี	ไม่มี
มีฟังก์ชันการแจ้งเตือนยา ที่ต้องระวังกับผู้ป่วยที่มี โรคประจำตัว	มี	ไม่มี	ไม่มี
มีฟังก์ชันการแนะนำยา ใกล้เคียงหรือมีส่วนผสม เดียวกัน	มี	ไม่มี	ไม่มี

#### แบบประเมินความพึงพอใจ

	ผู้ใช้งานจริงบุคคลที่1	ผู้ใช้งานจริงบุคคลที่2	ผู้ใช้งานจริงบุคคลที่3
ความสะดวกสบายในการใช้ งาน	6	7	6
ความสวยงาม	3	4	3
ความหลากหลายของฟังก์ชัน การใช้งาน	8	9	7
ความถูกต้องของฟังก์ชันการใช้ งาน	10	9	7
ความพึงพอใจในการใช้งาน	7	7	6

# Flowการทำงาน

## 1.การเติมยาเข้าระบบ

ค้นหา

ค้นหา

หน้าหลัก เติมนยา แสดงยา เพิ่มสมาชิก แสดงสมาชิก แจ้งเตือน

ผลการค้นหา:

ไม่มียาที่มีส่วนเหมือนกัน

เริ่มที่หน้าหลักเราจะทำการกดไปที่ปุ่ม เติมนยา จะเปลี่ยนไปหน้าที่ให้กรอกข้อมูลยาดังต่อไปนี้

หน้าหลัก เติมนยา แสดงยา เพิ่มสมาชิก แสดงสมาชิก แจ้งเตือน

ชื่อยา

คำบรรยาย

ส่วนผสม

ข้อควรระวัง

หมวดหมู่

ราคา

จำนวน

วันหมดอายุ

๓๓/๑๑/๒๒๒๒

๒

บาร์โค้ด

ชื่อจำกัด

วันที่ผลิต

๓๓/๑๑/๒๒๒๒

๒

ส่งข้อมูล

ให้เรากรอกชื่อยา คำบรรยาย ส่วนผสม ข้อควรระวัง หมวดหมู่ ราคา จำนวน วันหมดอายุ บาร์โค้ด ชื่อจำกัด วันที่ผลิตแล้วกดส่งข้อมูล

2.การแสดงผลยาที่มีอยู่ในระบบ

ค้นหา

หน้าหลัก [เดมยา](#) [แสดงยา](#) [เพิ่มสมาชิก](#) [แสดงสมาชิก](#) [แจ้งเดือน](#)

ผลการค้นหา:  
ไม่มียาที่มีส่วนเหมือนกัน

เริ่มที่หน้าหลัก เรากดที่ปุ่ม แสดงยา จะเปลี่ยนไปหน้าดังต่อไปนี้

หน้าหลัก [เดมยา](#) [แสดงยา](#) [เพิ่มสมาชิก](#) [แสดงสมาชิก](#) [แจ้งเดือน](#)

รหัสสินค้า	ชื่อสินค้า	คำอธิบาย	ส่วนผสม	วิธีการบรรจุ	หมวดหมู่	ราคา	จำนวน	วันหมดอายุ	บาร์โค้ด	วันที่ผลิต
1	akamol	paracetamol	paracetamol	-	ยาสามัญประจำบ้าน	50.00	498	Wed Mar 18 2026 00:00:00 GMT+0700 (เวลาอินโดจีน)	111111111111111111111111	Tue Mar 18 2025 00:00:00 GMT+0700 (เวลาอินโดจีน)
3	tylenol	paracetamol	paracetamol	-	ยาสามัญประจำบ้าน	50.00	500	Wed Mar 18 2026 00:00:00 GMT+0700 (เวลาอินโดจีน)	111111111111111111111111	Tue Mar 18 2025 00:00:00 GMT+0700 (เวลาอินโดจีน)
4	akamol	paracetamol	paracetamol	-	ยาสามัญประจำบ้าน	50.00	500	Wed Jul 08 2026 00:00:00 GMT+0700 (เวลาอินโดจีน)	111111111111111111111111	Tue Mar 18 2025 00:00:00 GMT+0700 (เวลาอินโดจีน)
8	sara	ยาแก้แพ้	paracetamol	ความดัน	ยาแก้แพ้	50.00	462	Wed Jun 24 2026 00:00:00 GMT+0700 (เวลาอินโดจีน)	0	Wed Apr 30 2025 00:00:00 GMT+0700 (เวลาอินโดจีน)
11	sara	ยาแก้แพ้	paracetamol	ความดัน	ยาแก้แพ้	50.00	29	Thu Jun 05 2025 00:00:00 GMT+0700 (เวลาอินโดจีน)	0	Mon May 05 2025 00:00:00 GMT+0700 (เวลาอินโดจีน)
12	sara	ยาแก้แพ้	paracetamol	ความดัน	ยาแก้แพ้	50.00	500	Wed May 06 2026 00:00:00 GMT+0700 (เวลาอินโดจีน)	0	Tue May 06 2025 00:00:00 GMT+0700 (เวลาอินโดจีน)
13	sara	ยาแก้แพ้	parasetamal	ความดัน		25.00	400	Wed Jan 29 2025 00:00:00 GMT+0700 (เวลาอินโดจีน)		Invalid Date

3. การเพิ่มสมาชิกใหม่ในระบบ

ค้นหา

หน้าหลัก [เดมยา](#) [แสดงยา](#) [เพิ่มสมาชิก](#) [แสดงสมาชิก](#) [แจ้งเดือน](#)

ผลการค้นหา:  
ไม่มียาที่มีส่วนเหมือนกัน



เริ่มที่หน้าหลังเราดที่ปุ่ม เพิ่มสมาชิก จะเปลี่ยนไปหน้าดังต่อไปนี้

หน้าหลัก เดิมยา แสดงยา เพิ่มสมาชิก แสดงสมาชิก แจ้งเตือน

ชื่อจริง

นามสกุล

เบอร์โทร

อีเมล

ที่อยู่

อายุ

น้ำหนัก

เพศ

โรคประจำตัว

จัดสมาชิก:  
[เลือก ๗]

เพิ่ม

ให้เรากรอกข้อมูลต่อไปนี้ ชื่อจริง นามสกุล เบอร์โทร อีเมล ที่อยู่ อายุ น้ำหนัก เพศ โรคประจำตัว และเลือกระดับสมาชิก  
ต่างๆได้แก่ **Bronze silver gold**

4.การแสดงผลสมาชิกในระบบ

ค้นหา

กรอกชื่อยา

ค้นหา

หน้าหลัก เดิมยา แสดงยา เพิ่มสมาชิก แสดงสมาชิก แจ้งเตือน

ผลการค้นหา:

ไม่มียาที่มีส่วนผสมเหมือนกัน

Show desktop

เริ่มมาที่หน้าแรกให้เรากดไปที่ปุ่มแสดงสมาชิก จะเปลี่ยนไปหน้าดังต่อไปนี้

หน้าหลัก เดิมยา แสดงยา เพิ่มสมาชิก แสดงสมาชิก แจ้งเตือน

จณัสมารัก	ชื่อจริง	นามสกุล	เบอร์โทร	อีเมล	ที่อยู่	โรคประจำตัว	ระดับสมาชิก
1	worata	Mingponwat	0994495416	worata2545@hotmail.com	Dk	เบาหวาน,ความดัน	3
3	pasakom	mingponwat	0994495414	pmj1999@gmail.com	-		3
4	pattarakit	mingponwat	0635475363	pattarakit@hotmail.com	-		2
17		mingponwat	0994495416	petchu@gmail.com	dk	ความดัน	3

4.การแจ้งเตือน

ค้นหา

กรอกชื่อยา

ค้นหา

หน้าหลัก เดิมยา แสดงยา เพิ่มสมาชิก แสดงสมาชิก แจ้งเตือน

ผลการค้นหา:

ใบยาที่มีส่วนสมเือนกัน

เริ่มที่หน้าหลัก เราจะกดไปที่ปุ่ม แจ้งเตือน จะเปลี่ยนไปหน้าดังต่อไปนี้

หน้าหลัก เดิมยา แสดงยา เพิ่มสมาชิก แสดงสมาชิก แจ้งเตือน		
ใบยีสืบค้น	ชนิดการแจ้งเตือน	วันที่แจ้งเตือน
11	สืบค้นใบยีสืบค้น	2025-06-04T02:26:03.949Z
11	สืบค้นใบยีสืบค้น	2025-06-04T02:26:03.950Z
13	สืบค้นใบยีสืบค้น	2025-06-04T02:26:03.950Z

## 5.การสั่งจ่ายยา

ค้นหา

ค้นหา

หน้าหลัก เดิมยา แสดงยา เพิ่มสมาชิก แสดงสมาชิก แจ้งเตือน

ผลการค้นหา:

ไม่มียาที่มีส่วนผสมเหมือนกัน

เริ่มมาที่หน้าแรก ให้เรากรอกชื่อยาที่ต้องการที่ช่อง ค้นหา แล้วกดปุ่ม ค้นหา

ค้นหา

ค้นหา

หน้าหลัก เดิมยา แสดงยา เพิ่มสมาชิก แสดงสมาชิก แจ้งเตือน

ผลการค้นหา:

tylenol

จำนวน:

เพิ่มลงตะกร้า

ยาที่มีส่วนผสมเหมือนกัน::

akamol - ส่วนผสม: paracetamol

akamol - ส่วนผสม: paracetamol

sara - ส่วนผสม: paracetamol

sara - ส่วนผสม: paracetamol

sara - ส่วนผสม: paracetamol

จะแสดงยาที่เราต้องการแล้วให้เพิ่มจำนวนที่ต้องการจะซื้อแล้วกดปุ่ม เพิ่มลงตะกร้า โดยจะเปลี่ยนไปหน้าดังต่อไปนี้

(ข้างใต้จะแสดงยาที่มีส่วนผสมเหมือนกับยาที่เราต้องการจะซื้อ)

ชื่อสินค้า	ราคาต่อหน่วย	จำนวน	ราคารวม
tylenol	50.00	1	50

ไปยังหน้าชำระเงิน

หน้านี้จะเป็นหน้าแสดงตะกร้าของเรา แสดงชื่อยาที่เราจะซื้อ จำนวน และราคา

เมื่อกดหน้าชำระเงินจะเปลี่ยนเป็นหน้าดังต่อไปนี้

หน้าชำระเงิน

ชื่อสินค้า	จำนวน	ราคา	รวม
tylenol	1	50.00	50

ยอดรวมทั้งหมด: 50 บาท

ค้นหาเลขค่า:

สมัครสมาชิก

ชำระเงิน

ตรวจสอบรายการในตะกร้าสินค้า

ส่วนลด: 0%

ยอดสุทธิถึงหน้าส่วนลด: 50 บาท

เงินทอน: 0 บาท

จำนวนเงินที่ได้รับ:

กรณีสถานการณ์

เลือกวิธีการชำระเงิน:

เงินสด

ยืนยันการสั่งซื้อ

ให้เรากรอกจำนวนเงินที่เราได้รับ แล้ว กดปุ่ม ยืนยันการสั่งซื้อ

## 6.การสั่งจ่ายยาโดยมีการใช้ส่วนลด

ค้นหา

ค้นหา

หน้าหลัก เดิมยา แสดงยา เพิ่มสมาชิก แสดงสมาชิก แจ้งเตือน

ผลการค้นหา:  
ไม่มียาที่มีส่วนผสมเหมือนกัน

เริ่มมาที่หน้าแรก ให้เรากรอกชื่อยาที่ต้องการที่ช่อง ค้นหา แล้วกดปุ่ม ค้นหา

ค้นหา

ค้นหา

หน้าหลัก เดิมยา แสดงยา เพิ่มสมาชิก แสดงสมาชิก แจ้งเตือน

ผลการค้นหา:

• tylenol

จำนวน:

เพิ่มลงตะกร้า

ยาที่มีส่วนผสมเหมือนกัน::

• akamol - ส่วนผสม: paracetamol

akamol - ส่วนผสม: paracetamol

sara - ส่วนผสม: paracetamol

sara - ส่วนผสม: paracetamol

sara - ส่วนผสม: paracetamol

จะแสดงยาที่เราต้องการแล้วให้เพิ่มจำนวนที่ต้องการจะซื้อแล้วกดปุ่ม เพิ่มลงตะกร้า โดยจะเปลี่ยนไปหน้าดังต่อไปนี้

(ข้างใต้จะแสดงยาที่มีส่วนผสมเหมือนกับยาที่เราต้องการจะซื้อ)

ชื่อสินค้า	ราคาต่อหน่วย	จำนวน	ราคารวม
tylenol	50.00	1	50

ไปยังหน้าชำระเงิน

หน้านี้จะเป็นหน้าแสดงตะกร้าของเรา แสดงชื่อยาที่เราจะซื้อ จำนวน และราคา

เมื่อกดหน้าชำระเงินจะเปลี่ยนเป็นหน้าดังต่อไปนี้

หน้าชำระเงิน

ชื่อสินค้า	จำนวน	ราคา	รวม
tylenol	1	50.00	50

ยอดรวมทั้งหมด: 50 บาท

ค้นหาลูกค้า:

ชื่อส่วนต่อ

แนะนำโฆษณา

ตรวจสอบรายการในตะกร้ากับสมาชิก

ส่วนลด: 0%

ยอดสุทธิหลังหักส่วนลด: 50 บาท

เงินทอน: 0 บาท

จำนวนเงินที่ได้รับ:

กรณีสถานการเงิน

เลือกวิธีการชำระเงิน:

เงินสด

ยืนยันการสั่งซื้อ

ให้เรากรอกชื่อสมาชิกที่ช่อง ค้นหาลูกค้า แล้วกดปุ่ม ดึงส่วนลด แล้วกรอกจำนวนเงินที่เราได้รับ แล้ว กดปุ่ม ยืนยันการสั่งซื้อ

หน้าชำระเงิน

ชื่อสินค้า	จำนวน	ราคา	รวม
tylenol	1	50.00	50

ยอดรวมทั้งหมด: 50 บาท

ค้นหาลูกค้า:

wordstat

ชื่อส่วนต่อ

แนะนำโฆษณา

ตรวจสอบรายการในตะกร้ากับสมาชิก

ส่วนลด: 15%

ยอดสุทธิหลังหักส่วนลด: 42.50 บาท

เงินทอน: 57.50 บาท

จำนวนเงินที่ได้รับ:

100

เลือกวิธีการชำระเงิน:

เงินสด

ยืนยันการสั่งซื้อ

## 7.การส่งจ่ายยาโดยต้องการให้แนะนำโดสยาที่ต้องรับประทานแต่กสมาชิก

ค้นหา

ค้นหา

หน้าหลัก เดิมยา แสดงยา เพิ่มสมาชิก แสดงสมาชิก แจ้งเตือน

ผลการค้นหา:

ไม่มียาที่มีส่วนผสมเหมือนกัน

เริ่มมาที่หน้าแรก ให้เรากรอกชื่อยาที่ต้องการที่ช่อง ค้นหา แล้วกดปุ่ม ค้นหา

ค้นหา

ค้นหา

หน้าหลัก เดิมยา แสดงยา เพิ่มสมาชิก แสดงสมาชิก แจ้งเตือน

ผลการค้นหา:

• tylenol

จำนวน:

เพิ่มลงตะกร้า

ยาที่มีส่วนผสมเหมือนกัน::

• akamol - ส่วนผสม: paracetamol

akamol - ส่วนผสม: paracetamol

sara - ส่วนผสม: paracetamol

sara - ส่วนผสม: paracetamol

sara - ส่วนผสม: paracetamol

จะแสดงยาที่เราต้องการแล้วให้เพิ่มจำนวนที่ต้องการจะซื้อแล้วกดปุ่ม เพิ่มลงตะกร้า โดยจะเปลี่ยนไปหน้าดังต่อไปนี้

(ข้างใต้จะแสดงยาที่มีส่วนผสมเหมือนกับยาที่เราต้องการจะซื้อ)

ชื่อสินค้า	ราคาต่อหน่วย	จำนวน	ราคารวม
tylenol	50.00	1	50

ไปยังหน้าชำระเงิน

หน้านี้นี้จะเป็นหน้าแสดงตะกร้าของเรา แสดงชื่อยาที่เราจะซื้อ จำนวน และราคา

เมื่อกดหน้าชำระเงินจะเปลี่ยนเป็นหน้าดังต่อไปนี้

หน้าชำระเงิน

ชื่อสินค้า	จำนวน	ราคา	รวม
tylenol	1	50.00	50

ยอดรวมทั้งหมด: 50 บาท

ค้นหาลูกค้า:

ชื่อส่วนต่อ

แนะนำโดสยา

ตรวจสอบรายการในหมายเหตุสมาชิก

ส่วนลด: 0%

ยอดสุทธิหลังหักส่วนลด: 50 บาท

เงินทอน: 0 บาท

จำนวนเงินที่ได้รับ:

กรณีสถานการเงิน

เลือกวิธีการชำระเงิน:

เงินสด

ดำเนินการสั่งซื้อ

ให้เรากรอกชื่อสมาชิกที่ช่อง ค้นหาลูกค้า แล้วกดปุ่ม แนะนำโดสยา แล้วกรอกจำนวนเงินที่เราได้รับ แล้ว กดปุ่ม ยืนยันการสั่งซื้อ

หน้าชำระเงิน

ชื่อสินค้า	จำนวน	ราคา	รวม
tylenol	1	50.00	50

ยอดรวมทั้งหมด: 50 บาท

ค้นหาลูกค้า:

ชื่อส่วนต่อ

แนะนำโดสยา

คำแนะนำโดสยา: 1.5 mg

ตรวจสอบรายการในหมายเหตุสมาชิก

ส่วนลด: 15%

ยอดสุทธิหลังหักส่วนลด: 42.50 บาท

เงินทอน: 57.50 บาท

จำนวนเงินที่ได้รับ:

100

เลือกวิธีการชำระเงิน:

เงินสด

ดำเนินการสั่งซื้อ



## 8.การสั่งจ่ายยาโดยการตรวจว่ายานี้สมาชิกทานได้ไหม

ค้นหา

ค้นหา

หน้าหลัก เดิมยา แสดงยา เพิ่มสมาชิก แสดงสมาชิก แจ้งเตือน

ผลการค้นหา:  
ไม่มียาที่มีส่วนผสมเหมือนกัน

เริ่มมาที่หน้าแรก ให้เรากรอกชื่อยาที่ต้องการที่ช่อง ค้นหา แล้วกดปุ่ม ค้นหา

ค้นหา

ค้นหา

หน้าหลัก เดิมยา แสดงยา เพิ่มสมาชิก แสดงสมาชิก แจ้งเตือน

ผลการค้นหา:

• 

tylenol

จำนวน: 

เพิ่มลงตะกร้า

ยาที่มีส่วนผสมเหมือนกัน::

• 

akamol - ส่วนผสม: paracetamol

• 

akamol - ส่วนผสม: paracetamol

• 

sara - ส่วนผสม: paracetamol

• 

sara - ส่วนผสม: paracetamol

• 

sara - ส่วนผสม: paracetamol

จะแสดงยาที่เราต้องการแล้วให้เพิ่มจำนวนที่ต้องการจะซื้อแล้วกดปุ่ม เพิ่มลงตะกร้า โดยจะเปลี่ยนไปหน้าดังต่อไปนี้

(ข้างใต้จะแสดงยาที่มีส่วนผสมเหมือนกับยาที่เราต้องการจะซื้อ)

ชื่อสินค้า	ราคาต่อหน่วย	จำนวน	ราคารวม
tylenol	50.00	1	50

ไปยังหน้าชำระเงิน

หน้านี้นี้จะเป็นหน้าแสดงตะกร้าของเรา แสดงชื่อยาที่เราจะซื้อ จำนวน และราคา

เมื่อกดหน้าชำระเงินจะเปลี่ยนเป็นหน้าดังต่อไปนี้

หน้าชำระเงิน

ชื่อสินค้า	จำนวน	ราคา	รวม
tylenol	1	50.00	50

ยอดรวมทั้งหมด: 50 บาท

ค้นหาลูกค้า:

กรณีสั่งลูกค้า

ชื่อส่วนต่อ | เลขประจำตัวประชาชน

ตรวจสอบรายการในหมายเหตุกับสมาชิก

ส่วนลด: 0%

ยอดสุทธิหลังหักส่วนลด: 50 บาท

เงินทอน: 0 บาท

จำนวนเงินที่ได้รับ:

กรณีส่งเงินจริง

เลือกวิธีการชำระเงิน:

เงินสด

ดำเนินการสั่งซื้อ

ให้เรากรอกชื่อสมาชิกที่ช่อง ค้นหาลูกค้า แล้วกดปุ่ม ตรวจสอบยาที่ไม่เหมาะสมกับสมาชิก แล้วกรอกจำนวนเงินที่เราได้รับ แล้ว กดปุ่ม ยืนยันการสั่งซื้อ

หน้าชำระเงิน

ชื่อสินค้า	จำนวน	ราคา	รวม
tylenol	1	50.00	50

ยอดรวมทั้งหมด: 50 บาท

ค้นหาลูกค้า:

พจนานุกรม

ชื่อส่วนต่อ | เลขประจำตัวประชาชน

ตรวจสอบรายการในหมายเหตุกับสมาชิก

☒ โบนัสยาที่ซื้อกับโรงพยาบาลของลูก

ส่วนลด: 0%

ยอดสุทธิหลังหักส่วนลด: 50 บาท

เงินทอน: 50.00 บาท

จำนวนเงินที่ได้รับ:

100

เลือกวิธีการชำระเงิน:

เงินสด

ดำเนินการสั่งซื้อ

### คำแนะนำจากผู้ใช้งานจริง

1. ในส่วนของหน้าแสดง **stock** สินค้า อยากให้เพิ่มหน่วยของสินค้าเช่น จำนวนกล่อง จำนวนแผง จำนวน กระปุก จำนวนชิ้น
2. ในส่วนตะกร้าสินค้าอยากให้รองรับการใช้งานได้ที่หลายตะกร้า ในกรณีที่ต้องรับลูกค้าหลายคนพร้อมกัน สามารถบันทึกสินค้าของแต่ละคนแยกกันทิ้งไว้ได้และแก้ไขรายการสินค้าในตะกร้าได้
3. ในระบบสมาชิกอยากให้มีการเก็บข้อมูลเลขประจำตัวประชาชน เชื้อชาติ โรคประจำตัว ประวัติการแพ้ยา ประวัติการดื่มเหล้า สูบบุหรี่ มีบันทึกประวัติการซื้อยาของสมาชิกทุกคน
4. มีการระบุราคาทุน ราคาขาย ทำสรุปยอดขาย กำไร และคำนวณภาษี
5. มีระบบรองรับ **smart shelf** คือระบุตำแหน่งสินค้าและจำนวนใน **shelf** จำนวนใน **stock** จำนวนรวม และแยกจำนวนสินค้าตามวันหมดอายุ
6. รองรับการพิมพ์บัตรแพ้ยา
7. การฟสรุปยอดขาย รายวัน รายสัปดาห์ รายเดือน รายปี และสามารถดูความถี่การสั่งจ่ายยาเป็นชนิด หรือเทียบยอดขายผู้ขายแต่ละคน

## บทที่ 5

### สรุปผลการทดลอง

#### อภิปรายผลการทดลองและสรุปผลการดำเนินงาน

ในโครงการนี้ เราได้พัฒนาระบบ Pharmacy Management System ซึ่งเป็นระบบที่ใช้ในการบริหารจัดการข้อมูลสินค้า ยา และการคำนวณโดสยา โดยระบบสามารถดึงข้อมูลน้ำหนักของลูกค้ำและแนะนำโดสยาได้ตามน้ำหนักที่ระบุในฐานข้อมูล รวมถึงสามารถดึงข้อมูลส่วนลดและตรวจสอบยาที่อาจมีอันตรายต่อผู้ป่วยจากโรคประจำตัวได้ ระบบได้รับการทดสอบในหลากหลายสถานการณ์และสามารถทำงานได้ตามที่คาดหวังไว้ในส่วนหลักๆ เช่น การคำนวณยอดรวม การแนะนำโดสยา และการตรวจสอบยาที่ไม่เหมาะสมกับผู้ป่วย

#### อภิปรายผลการทดลอง

จากการทดสอบระบบพบว่า ระบบสามารถดึงข้อมูลจากฐานข้อมูลได้อย่างถูกต้องและแสดงผลได้อย่างแม่นยำในทุกฟังก์ชันที่ทดสอบ รวมถึงการแนะนำโดสยา โดยการคำนวณจากน้ำหนักของลูกค้ำก็สามารถทำได้ตามที่ตั้งเป้าหมายไว้ อย่างไรก็ตามยังมีข้อจำกัดบางประการ เช่น การเชื่อมต่อกับฐานข้อมูลในบางครั้งอาจเกิดความล่าช้า และบางครั้งระบบแสดงข้อผิดพลาดเมื่อกรอกข้อมูลไม่ถูกต้องในฟอร์ม ซึ่งสามารถปรับปรุงในส่วนนี้ได้

#### แนวทางการต่อยอดและปรับปรุง

เพื่อพัฒนาระบบให้สมบูรณ์ยิ่งขึ้นในอนาคต สามารถพิจารณาปรับปรุงในด้านต่างๆ เช่น:

1. เพิ่มฟังก์ชันการตรวจสอบข้อมูลเพิ่มเติม: เช่น การตรวจสอบข้อมูลยาให้ครอบคลุมยิ่งขึ้น โดยสามารถให้คำแนะนำเกี่ยวกับยาแนะนำที่มีประสิทธิภาพในการรักษาโรค
2. พัฒนาระบบการเชื่อมต่อฐานข้อมูล: เพิ่มประสิทธิภาพในการเชื่อมต่อฐานข้อมูลให้เร็วยิ่งขึ้น โดยใช้เทคนิคการแคช (caching) หรือปรับปรุงการทำงานของ SQL Query เพื่อลดเวลาในการดึงข้อมูล
3. การพัฒนา UI/UX: เพิ่มประสิทธิภาพในการใช้งานระบบ โดยทำให้หน้าเว็บดูสะอาดตา ใช้งานง่าย และรองรับการเข้าถึงจากอุปกรณ์หลากหลายมากขึ้น เช่น มือถือหรือแท็บเล็ต
4. เสริมการตรวจสอบความปลอดภัย: ควรมีระบบการเข้ารหัสข้อมูลเพื่อเพิ่มความปลอดภัยในด้านต่างๆ เช่น ข้อมูลของลูกค้ำและการชำระเงิน

#### สรุป

ในโครงการนี้ ระบบที่พัฒนาขึ้นสามารถทำงานได้ตามเป้าหมายที่ตั้งไว้ และมีความสำเร็จในการดำเนินงานในหลายๆ ด้าน อย่างไรก็ตาม ยังมีโอกาสในการปรับปรุงและพัฒนาระบบต่อไปเพื่อให้สามารถใช้งานได้อย่างสมบูรณ์และมีประสิทธิภาพสูงสุดในอนาคต