**บทที่ 1**

**บทนำ**

1. **ความเป็นมาและความสำคัญของโครงงาน**

เนื่องจากเหตุการณ์โรคระบาด ส่งผลให้ธุรกิจขนส่งอาหารเติบโตมากขึ้น หลากหลายมากขึ้น สะดวกสบายมากขึ้น ทำให้ผู้ประกอบการจำนวนมากหันมาสนใจใน platform เหล่านี้ และเกิดเป็นธุรกิจร้านอาหารแบบใหม่ ที่ไม่มีโต๊ะอาหาร ไม่มีหน้าร้าน รับคำสั่งซื้อผ่านระบบ online delivery อย่างเดียว

โครงการ “ระบบจัดการร้านอาหารเดลิเวอรี่” มุ่งเป้าไปที่การสร้างระบบหลังบ้านที่สามารถจัดการร้านอาหารที่ใช้ระบบส่งออนไลน์เป็นหลักได้อย่างมีประสิทธิภาพ สามารถทำบัญชีและจัดการเงินเข้าออกจากแอพพลิเคชั่นต่างๆได้ง่าย เก็บบันทึกสถิติการขายเมนูต่างๆของทุกแอพพลิเคชั่น เพื่อทำแบบวิเคราะห์การขายและออกโปรโมชั่น โดยในเบื้องต้น โครงงานนี้จะใช้โมเดลธุรกิจของร้าน ‘สมายดอง’ เป็นหลักในการพัฒนา

ในปัจจุบัน platform การส่งอาหารออนไลน์ในประเทศไทยมีหลากหลายเป็นอย่างมาก เช่น Grab Food, Lineman Wongnai, ShopeeFood, Robinhood โดยแต่ละบริษัทก็มีระบบการให้บริการที่แตกต่างกันไป ส่งผลให้การทำบัญชี การคิดรายรับ รายจ่าย กำไร ต้นทุน หรือบัญชีการเงินต่างๆในร้านค้าเกิดความซับซ้อนมากกว่าการขายหน้าร้านปกติทั่วไป สิ่งเหล่านี้สามารถแก้ไขได้เมื่อมีการเพิ่ม custom rule set ขึ้นมา ซึ่งถือเป็น feature สำคัญของระบบนี้

1. **วัตถุประสงค์ของโครงงาน**
2. มีระบบบริการร้านอาหารเดลิเวอรี่ครบวงจร ที่สามารถทำบัญชี และรายงานสถิติได้
3. มีระบบที่สามารถช่วยให้กิจกรรมรายวัน (เช่น จัดการบัญชีเงิน นับสต๊อค) ง่ายขึ้น
4. มีระบบที่สามารถ customize ส่วนต่างๆได้ด้วยตัวผู้ใช้เอง
5. **ขอบเขตของโครงงาน**
   1. **ด้านประชากร**

ระบุกลุ่มประชากรผู้ใช้งานซอฟต์แวร์ให้ชัดเจน ยกตัวอย่าง เช่น

1. เจ้าของร้านและพนักงานร้าน ‘สมายดอง’
   1. **ด้านเทคโนโลยีที่ใช้ในการพัฒนาโครงงาน**
      1. **Hardware**

- Desktop Computer (RAM 16GB, AMD FX-8320E, NVIDIA GeForce GTX 1050 Ti)

- Smart Phone (Xiaomi 10T pro)

* + 1. **Software**

- Microsoft Windows 10 Pro build 19043

- MongoDB 5.0

- Express.js 4.0

- React 18.0.0

- NodeJS 16.14.2 LTS

- Visual Studio Code 1.66.2

- Notepad++ 8.3.3

* + 1. **Technology**

- Model View Controller

- REST API

- Cloud Database (MongoDB)

- Node.JS

- Framework (Express, React)

1. **ขอบเขตของโครงงาน**

ระบุเป็นข้อๆ โดยขอบเขตของโครงงาน คือ ความสามารถของซอฟต์แวร์ที่พัฒนา เพื่อระบุว่าในโครงงานนี้ ประกอบไปด้วยโมดูลย่อยอะไรบ้าง และแต่ละโมดูลย่อยนั้น ๆ มีความสามารถอะไรบ้าง เช่น

1. ระบบตรวจสอบสิทธิในการเข้าใช้งานซอฟต์แวร์ (Authentication or Login)
2. การจัดการวัตถุดิบ
   1. เพิ่มรายการวัตถุดิบ (ชื่อ, ราคา, จำนวน, หน่วย)
   2. แก้ไขรายการวัตถุดิบ
   3. ลบรายการวัตถุดิบ
3. การจัดการบรรจุภัณฑ์
   1. เพิ่มรายการบรรจุภัณฑ์ (ชื่อ, ราคา, จำนวน, หน่วย)
   2. แก้ไขรายการบรรจุภัณฑ์
   3. ลบรายการบรรจุภัณฑ์
4. การจัดการสูตรอาหาร
   1. เพิ่มสูตรอาหาร (ชื่ออาหาร, วัตถุดิบ/บรรจุภัณฑ์+จำนวนที่ใช้, ค่าแรง)
   2. แก้ไขสูตรอาหาร
   3. ลบสูตรอาหาร
5. การจัดการเมนูอาหาร
   1. เพิ่มเมนูอาหาร (ชื่อเมนู/ชุดเซ็ต, อาหารในเมนู, ราคา)
   2. แก้ไขเมนูอาหาร
   3. ลบเมนูอาหาร
6. การจัดการระบบส่งอาหาร
   1. เพิ่มระบบส่งอาหาร (ชื่อ, วิธีการคิดบัญชี)
      1. วิธีการคิดบัญชีจะประกอบด้วย ค่าธรรมเนียมตามออเดอร์ ค่าสมาชิกรายเดือน ฯลฯ
   2. แก้ไขระบบส่งอาหาร (ชื่อ, วิธีการคิดบัญชี, ยอดเงินคงเหลือ)
   3. ลบระบบส่งอาหาร
7. ระบบจัดการรายรับรายจ่าย
   1. ส่วนจัดการหมวดหมู่ดรายรับรายจ่าย
      1. เพิ่มหมวดหมู่ (เลือกว่าเป็นรายรับหรือรายจ่าย, ชื่อหมวด, เลือกหมวด Parent (ถ้ามี))
      2. แก้ไขหมวดหมู่ (ชื่อหมวด, หมวด Parent)
      3. ลบหมวดหมู่
   2. รายจ่าย
      1. เมื่อเลือกหมวดวัตถุดิบจะต้องเลือกวัตถุดิบจากรายชื่อที่เพิ่มไว้แล้ว จากนั้นใส่ราคาที่ซื้อ จำนวนที่ได้ และที่มา
      2. เมื่อเลือกหมวดบรรจุภัณฑ์จะต้องเลือกบรรจุภัณฑ์จากรายชื่อที่เพิ่มไว้แล้วจากนั้นใส่ราคาที่ซื้อและจำนวนที่ได้ และที่มา
      3. เมื่อเลือกหมวดอื่นๆจะต้องใส่ ชื่อ ราคา และที่มา(มีหรือไม่มีก็ได้)
      4. ด้านล่างของส่วน input จะเป็นรายการรายจ่ายล่าสุด n รายการ (ตามแต่ผู้ใช้เลือกจาก dropdown) เมื่อคลิกที่รายการล่าสุดเหล่านั้น ระบบจะคัดลอกข้อมูลทั้งหมดขึ้นไปใน input ด้านบน เพื่อเพิ่มรายการอย่างรวดเร็ว
   3. รายรับ
      1. เมื่อมีออเดอร์เกิดขึ้น ผู้ใช้จะต้องเพิ่มรายการโดย
         1. เลือกช่องทางระบบส่งอาหาร
         2. เลือกเมนูในออเดอร์ที่ได้รับ
         3. ใส่จำนวนของแต่ละเมนู
      2. เมนูเหล่านั้นจะถูกเก็บเป็นสถิติการขายทันที
      3. ยอดรวมของแต่ละออเดอร์จะถูกคำนวณโดยอิงจากกฏการคำนวณของแต่ละระบบส่งอาหาร
      4. ยอดรวมของแต่ละออเดอร์จะถูกเพิ่มเข้าไปในกระเป๋าเงินของระบบส่งอาหารนั้นๆ
      5. แต่ละออเดอร์ที่ถูกเพิ่มจะมีปุ่มที่ท้ายรายการเพื่อยืนยันว่าออเดอร์นั้นๆได้รับเงินจากระบบส่งอาหารแล้ว
      6. ออเดอร์ต้องได้รับการยืนยันรับเงินก่อนถึงจะเพิ่มเป็นรายรับของวันที่ยืนยัน
      7. เมื่อออเดอร์ได้รับการยืนยันรับเงิน ระบบจะหักยอดออกจากกระเป๋าเงินระบบส่งอาหารนั้นๆ
8. ระบบคำนวณต้นทุน แสดงผลเป็นตารางตามการเลือกของผู้ใช้
   1. แสดงผลของแต่ละเมนู ประกอบด้วย
      1. ระบบส่งอาหาร
      2. ชื่อเมนู
      3. ราคาขาย
      4. ต้นทุนวัตถุดิบ
      5. ต้นทุนบรรจุภัณฑ์
      6. ต้นทุนแรงงาน
      7. ต้นทุนน้ำมัน
      8. ต้นทุนทั้งหมด (iii + iv + v +vi)
      9. ราคาขาย
      10. ยอดเงินที่จะได้รับ (ราคาขาย - ค่าธรรมเนียม)
      11. กำไรสุทธิ (ยอดเงินที่จะได้รับ - ต้นทุนทั้งหมด)
   2. ผู้ใช้มีสิทธิ์เลือก
      1. ระบบส่งอาหาร หากไม่เลือกจะแสดงทั้งหมด
      2. วิธีการคำนวณ เมื่อเลือกแล้วจะถูกบันทึกไว้เพื่อใช้แสดงผลครั้งต่อไปและการแสดงผลบนหน้า Dashboard (ค่าเริ่มต้นคือ ใช้ราคาล่าสุด)
         1. ใช้ราคาวัตถุดิบ/บรรจุภัณฑ์ล่าสุด
         2. ใช้ราคาวัตถุดิบ/บรรจุภัณฑ์เฉลี่ยจาก 1 อาทิตย์ที่ผ่านมา
         3. ใช้ราคาวัตถุดิบ/บรรจุภัณฑ์เฉลี่ยจาก 1 เดือนที่ผ่านมา
   3. เมนูที่มีกำไรสุทธิติดลบจะแสดงผลด้วยพื้นหลังที่แตกต่างจากรายการอื่น
9. หน้า Dashboard
   1. แสดงผลเงินคงเหลือในกระเป๋าของแต่ละระบบส่งอาหาร
   2. แสดงผลบัญชีรายวัน
      1. ค่าแรง
      2. กำไร
      3. ต้นทุน
      4. ยอดที่ควรได้รับ
   3. แสดงผลค่าน้ำมันคงเหลือ และปุ่มกด เมื่อกดที่ปุ่มจะเป็นการบอกระบบว่าจะใช้เงินตรงส่วนนี้ ระบบจะถามจำนวน (Default คือค่าน้ำมันคงเหลือทั้งหมด) เมื่อกดยืนยัน ระบบจะหักค่าน้ำมันคงเหลือออกและเพิ่มเป็นรายจ่ายค่าน้ำมันโดยอัตโนมัติ
10. หน้าแสดงสถิติการขายแบบ Responsive แสดงผลเป็น
    1. กราฟและตาราง
    2. ตามรายการอาหารหรือตามรายการเมนู
    3. รายวันหรือรายอาทิตย์หรือรายเดือนหรือรายปี
11. **กรอบแนวคิดการพัฒนาโครงงาน**
    1. **แบบจำลองกระบวนการ (Software Process Model)** เลือกใช้ Iterative Model เนื่องจากผู้พัฒนาได้รับสิทธิ์ให้เข้าถึงข้อมูลจริงของผู้ใช้ตลอดเวลา ผู้พัฒนาจึงสามารถทดสอบระบบได้เท่าที่ต้องการ การพัฒนาตามโมเดลนี้จะช่วยให้ระบบมีความถูกต้อง และตรงตามความต้องการอยู่เสมอ  
       Diagram

       Description automatically generated
    2. **แผนภูมิแกนต์ (Gantt Chart)**

Chart

Description automatically generated

1. **ประโยชน์ที่ได้รับ**
   1. ร้านค้ามีบัญชีรายรับรายจ่ายที่ถูกต้องชัดเจนและสามารถตรวจสอบได้
   2. ร้านค้ามีสถิติการขายเพื่อใช้ในการวิเคราะห์สินค้าและโปรโมชั่น
   3. ร้านค้ามีสถิติการซื้อเพื่อใช้ในการวิเคราะห์ต้นทุนและสต๊อควัตถุดิบ
   4. ร้านค้าทราบถึงต้นทุนที่แท้จริงของสินค้าเพื่อกำหนดราคาอย่างเหมาะสม
   5. ร้านค้าสามารถจัดการเงินเข้าออกไปยังบัญชีต่างๆได้โดยง่าย
   6. ร้านค้าสามารถจัดการระบบภายในร้านได้เองโดยไม่ต้องพึ่งผู้พัฒนาระบบ
2. **นิยามศัพท์เฉพาะ**

คำศัพท์เฉพาะ ที่ใช้ในการดำเนินการพัฒนาโครงงานนี้เท่านั้น เช่น

* 1. ผู้ใช้ คือ ผู้จัดการหรือเจ้าของร้านสมายดอง
  2. สูตรอาหาร คือการเก็บข้อมูลอาหารในร้านว่าประกอบด้วยวัตถุดิบหรือบรรจุภัณฑ์ใดบ้าง เช่น สูตรอาหาร ‘ข้าวญี่ปุ่น’ ต้องมีส่วนประกอบคือ ข้าวญี่ปุ่น, น้ำเปล่า, งาดำ, ไฟฟ้า, ถ้วยข้าว, ฝาถ้วย
  3. เมนูอาหาร คือการเก็บข้อมูลเมนูต่างๆในร้าน ว่าเมนูนั้นๆประกอบด้วยอาหารใดบ้าง เช่น เมนู “กุ้งดองซีอิ๊ว” ประกอบด้วย กุ้งดอง, น้ำจิ้มซีฟู๊ด, พริก, กระเทียม
  4. ระบบส่งอาหาร คือ platform สำหรับการสั่งอาหารต่างๆเช่น Grab, Lineman, Foodpanda, ShopeeFood, Robinhood