โครงงานระบบจัดการร้านคาเฟ่ Café Management System

สมาชิก

นายวีรภัทร	แก้วคำลา	663380024-9	Sec 1
นายพสิษฐ์	ผลวิเศษพรสุข	663380020-7	Sec 1
นายธนทัต	ภู่แก้ว	663380211-0	Sec 1
นายวิสิษฎ์	ศรีอดิศักดิ์	663380234-8	Sec 1
นางสาวอรอาภา	เหล่าชัย	663380245-3	Sec 2

เสนอ

รศ. ดร.ปัญญาพล หอระตะ

โครงงานนี้เป็นส่วนหนึ่งของรายวิชา CP353002 : Principles of Software Design and Development

สาขาวิทยาการคอมพิวเตอร์ วิทยาลัยการคอมพิวเตอร์ มหาวิทยาลัยขอนแก่น

ปีการศึกษา 1/2568

ชื่อหัวข้อโครงงาน : ระบบจัดการร้านคาเฟ่ Café Management System
ชื่อผู้จัดทำ : นายวีรภัทร แก้วคำลา 663380024-9
นายพสิษฐ์ ผลวิเศษพรสุข 663380020-7
นายธนทัต ภู่แก้ว 663380211-0
นายวิสิษฎ์ ศรีอดิศักดิ์ 663380234-8
นางสาวอรอาภา เหล่าชัย 663380245-3

ที่ปรึกษา : รศ. ดร.ปัญญาพล หอระตะ
สาขาวิชา : วิทยาการคอมพิวเตอร์ วิทยาลัยการคอมพิวเตอร์
มหาวิทยาลัยขอนแก่น

ปีการศึกษา : 1/2568

บทคัดย่อ

โครงงานนี้เป็นการพัฒนาระบบจัดการร้านคาเฟ่ (Café Management System) เพื่อช่วยบริหารจัดการร้านคาเฟ่ให้มีประสิทธิภาพมากขึ้น ระบบถูกออกแบบมาให้สามารถจัดการรายการสินค้า คำสั่งซื้อ และการชำระเงินได้อย่างเป็นระบบ โดยมีการบันทึกข้อมูลเจ้าของร้านและลูกค้า รวมถึงการสร้างรายงานสรุปยอดขาย ระบบนี้ประกอบไปด้วยระบบนี้ประกอบไปด้วย การจัดการเมนู การรับออเดอร์ การออกใบเสร็จ การจัดการของแจก แกลเลอรีรูปภาพ และการจัดการผู้ใช้งาน ทางคณะผู้จัดทำมีวิธีการพัฒนาระบบตามวงจรการพัฒนาระบบโดยใช้โดยใช้ฐานข้อมูลเชิงสัมพันธ์ (Relational Database) รองรับความสัมพันธ์ทั้งแบบ One-to-One, One-to-Many และ Many-to-Many พร้อมพัฒนาโดยยึดหลักการ SOLID เพื่อทำให้ซอร์สโค้ดเข้าใจง่าย ยืดหยุ่น และบำรุงรักษาได้ง่ายขึ้น โดยระบบรองรับการเชื่อมต่อผ่าน REST API และมีส่วน Frontend (Next.js) เพื่อให้ใช้งานได้ง่ายและสะดวก ช่วยลดความผิดพลาด เพิ่มความรวดเร็ว ซึ่งส่งผลต่อความสำเร็จและการเติบโตของธุรกิจคาเฟ่ในระยะยาว

กิตติกรรมประกาศ

รายงานเล่มนี้สำเร็จลุล่วงได้ด้วยความกรุณาจากรองศาสตราจารย์ปัญญาพล หอระตะ อาจารย์ที่ปรึกษารายงาน ที่ได้ให้ข้อเสนอแนะ แนวคิด วิธีปฏิบัติ ตลอดจนการแก้ไขข้อบกพร่องต่างๆ มาโดยตลอด ทำให้รายงานเล่มนี้เสร็จสมบูรณ์ คณะผู้จัดทำจึงขอขอบพระคุณเป็นอย่างสูง ขอขอบคุณไปยังผู้สนับสนุนทุกท่าน ตลอดจนผู้ที่ให้ความร่วมมือในการทดสอบและเก็บรวบร่วมข้อมูลในงานวิจัย สุดท้ายนี้ขอบคุณคณะผู้จัดทำทุกท่านที่ให้ความร่วมมือ จนรายงานงานนี้สำเร็จลุล่วงไปด้วยดี

คณะผู้จัดทำ

สารบัญ

	หน้า
บทคัดย่อ	ก
กิตติกรรมประกาศ	ข
สารบัญ	ค
สารบัญภาพ	ง
สารบัญตาราง	จ
บทที่ 1 บทนำ	1
1.1 ความเป็นมาและความสำคัญ	1
1.2 วัตถุประสงค์	2
1.3 ขอบเขตของโครงงาน	2
1.4 ประโยชน์ที่คาดว่าจะได้รับ	2
บทที่ 2 วิธีดำเนินการ	3
2.1 ขั้นตอนการดำเนินการ	3
2.2 เครื่องมือและเทคโนโลยีที่ใช้	6
2.3 การวิเคราะห์และออกแบบระบบ	9
บทที่ 3 ผลการดำเนินการ	25
3.1 ผลการพัฒนาระบบ	25
บทที่ 4 สรุปผล	35
4.1 สรุปผล	35
4.2 ปัญหาและข้อจำกัด	35
4.3 ข้อเสนอแนะ	35
เอกสารอ้างอิง	36

สารบัญภาพ

	หน้า
ภาพที่ 1 Wireframe ของหน้า Login	4
ภาพที่ 2 Wireframe ของหน้า MenuOrder	4
ภาพที่ 3 Wireframe ของหน้า History	5
ภาพที่ 4 Wireframe ของหน้า Member	5
ภาพที่ 5 Use Case Diagram	16
ภาพที่ 6 Sequence Diagram ของการเข้าสู่ระบบ	17
ภาพที่ 7 Sequence Diagram ของการจัดการข้อมูลเมนู	17
ภาพที่ 8 Sequence Diagram ของการเพิ่มข้อมูลเมนู	18
ภาพที่ 9 Sequence Diagram ของกระบวนการขาย	18
ภาพที่ 10 Sequence Diagram ของการจัดการข้อมูลพนักงาน	19
ภาพที่ 11 Sequence Diagram ของการเพิ่มข้อมูลพนักงาน	19
ภาพที่ 12 Sequence Diagram ของการดูรายงานยอดขาย	20
ภาพที่ 13 Sequence Diagram ทั้งหมด	21
ภาพที่ 14 Domain Model	22
ภาพที่ 15 Class Diagram	23
ภาพที่ 16 Activity Diagram	24
ภาพที่ 17 หน้าเข้าสู่ระบบ (Login)	25
ภาพที่ 18 หน้า MenuOrder	26
ภาพที่ 19 หน้า Add New Menu	27
ภาพที่ 20 หน้า Add Menu	28
ภาพที่ 21 หน้าใบเสร็จการขาย (Receipt Page)	29
ภาพที่ 22 หน้า History	30
ภาพที่ 23 หน้า Member List	31
ภาพที่ 24 หน้า Add New Member	32
ภาพที่ 25 หน้า Edit Member	33
ภาพที่ 26 หน้า Delete Member	34

สารบัญตาราง

หน้า

ตารางที่ 1 Use Case ของการเข้าสู่ระบบ 9
ตารางที่ 2 Use Case ของการจัดการข้อมูลเมนู 10
ตารางที่ 3 Use Case ของการเพิ่มข้อมูลเมนู 11
ตารางที่ 4 Use Case ของกระบวนการขาย 12
ตารางที่ 5 Use Case ของการจัดการข้อมูลพนักงาน 13
ตารางที่ 6 Use Case ของการเพิ่มข้อมูลพนักงาน 14
ตารางที่ 7 Use Case ของการดูรายงานยอดขาย 15

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญ

ในยุคปัจจุบัน เทคโนโลยีได้เข้ามามีบทบาทสำคัญต่อการดำเนินชีวิตประจำวันในหลายด้าน โดยเฉพาะในภาคธุรกิจที่มีการแข่งขันสูง การนำเทคโนโลยีสารสนเทศเข้ามาประยุกต์ใช้ในการบริหารจัดการจึงเป็นปัจจัยสำคัญที่ช่วยเพิ่มประสิทธิภาพและความสะดวกในการดำเนินงาน หนึ่งในธุรกิจที่ได้รับอิทธิพลจากเทคโนโลยีอย่างมากคือธุรกิจร้านคาเฟ่ ซึ่งมีการเติบโตและการแข่งขันเพิ่มขึ้นอย่างต่อเนื่อง การบริหารจัดการร้านคาเฟ่ให้มีความเป็นระบบและมีประสิทธิภาพจึงเป็นสิ่งจำเป็น เพื่อช่วยลดข้อผิดพลาดในการทำงานและเพิ่มความพึงพอใจให้แก่ลูกค้า

อย่างไรก็ตาม การจัดการร้านคาเฟ่ในรูปแบบดั้งเดิมยังคงประสบปัญหาในหลายด้าน พนักงานส่วนใหญ่ต้องจดออเดอร์ด้วยมือ ซึ่งอาจทำให้เกิดข้อผิดพลาด เช่น ลืมเมนู คำนวณราคาผิด หรือบันทึกข้อมูลไม่ครบถ้วน นอกจากนี้ การตรวจสอบยอดขายและประวัติการสั่งซื้อในแต่ละครั้งต้องใช้เวลานาน เพราะไม่มีระบบที่สามารถแสดงข้อมูลแบบเรียลไทม์ ทำให้ไม่สามารถติดตามผลการขายได้อย่างทันท่วงที อีกทั้งเมื่อมีเมนูที่ขายหมด พนักงานมักไม่ทราบทันที ส่งผลให้เกิดความสับสนในการรับออเดอร์จากลูกค้า รวมถึงการคำนวณยอดขายรายวันหรือรายสัปดาห์ที่ต้องทำด้วยตนเอง ซึ่งอาจเกิดความล่าช้าและความคลาดเคลื่อนของข้อมูลได้

ปัญหาเหล่านี้ส่งผลให้ประสิทธิภาพในการดำเนินงานของร้านลดลง เพิ่มภาระให้แก่พนักงาน และอาจกระทบต่อความพึงพอใจของลูกค้าโดยตรง ดังนั้น การพัฒนาระบบจัดการร้านคาเฟ่ (Café Management System) จึงมีความสำคัญอย่างยิ่ง เพื่อช่วยให้การบริหารจัดการภายในร้านเป็นระบบระเบียบมากขึ้น ระบบถูกออกแบบมาให้สามารถจัดการรายการสินค้า คำสั่งซื้อ และการชำระเงินได้อย่างมีประสิทธิภาพ พร้อมทั้งสามารถบันทึกข้อมูลเจ้าของร้าน พนักงาน และลูกค้า รวมถึงสร้างรายงานสรุปยอดขายได้อย่างสะดวกและรวดเร็ว ระบบดังกล่าวจะช่วยลดความผิดพลาดในการทำงาน เพิ่มความสะดวก รวดเร็ว และสร้างประสบการณ์ที่ดีให้แก่ทั้งผู้ให้บริการและผู้ใช้บริการ ซึ่งจะส่งผลต่อความสำเร็จและการเติบโตของธุรกิจคาเฟ่ในระยะยาว

1.2 วัตถุประสงค์

1.2.1 เพื่อพัฒนาระบบบริหารจัดการร้านคาเฟ่ให้มีความเป็นระบบและมีประสิทธิภาพมากขึ้น

1.2.2 เพื่อให้สามารถจัดการรายการสินค้าได้อย่างสะดวก โดยสามารถเพิ่ม ลบ และแก้ไขเมนูได้ตามต้องการ

1.2.3 เพื่ออำนวยความสะดวกในการรับคำสั่งซื้อจากลูกค้า ลดข้อผิดพลาดจากการจดบันทึกด้วยมือ

1.2.4 เพื่อจัดการกระบวนการชำระเงินและออกใบเสร็จให้เป็นระบบและตรวจสอบได้

1.2.5 เพื่อให้สามารถตรวจสอบยอดขายและสร้างรายงานสรุปยอดขายได้อย่างรวดเร็วและแม่นยำ

1.2.6 เพื่อช่วยให้เจ้าของร้านสามารถติดตามผลการดำเนินงานของร้านได้แบบเรียลไทม์

1.3 ขอบเขตของโครงงาน

1.3.1 ผู้ใช้สามารถ login เข้าสู่ระบบ เพื่อดูแลและดำเนินการต่าง ๆ ได้

1.3.2 ผู้ใช้สามารถเพิ่ม/ลบ/แก้ไข ข้อมูลรายการเมนูต่าง ๆ ได้

1.3.3 ผู้ใช้สามารถยืนยันรายการคำสั่งซื้อของลูกค้าได้

1.3.4 ผู้ใช้สามารถรวมรายการคำสั่งซื้อของลูกค้าได้

1.3.5 ผู้ใช้สามารถพิมพ์ใบเสร็จให้ลูกค้าได้

1.3.6 ผู้ใช้สามารถดูข้อมูลรายการคำสั่งซื้อทั้งหมดและสรุปยอดขายประจำวันได้

1.3.7 ผู้ใช้สามารถดูรายงานยอดขายทั้งหมดจากระบบได้

1.3.8 ผู้ใช้สามารถเพิ่ม/ลบ/แก้ไข ข้อมูลสมาชิกในระบบได้

1.3.9 ผู้ใช้สามารถ logout ออกจากระบบได้

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1.4.1 ช่วยให้การบริหารจัดการร้านคาเฟ่มีความเป็นระบบระเบียบและมีประสิทธิภาพมากยิ่งขึ้น

1.4.2 ลดความผิดพลาดในการรับออเดอร์ การคำนวณราคา และการบันทึกข้อมูล

1.4.3 เพิ่มความสะดวกและความรวดเร็วในการจัดการคำสั่งซื้อ การชำระเงิน และการออกใบเสร็จ

1.4.4 ช่วยให้เจ้าของร้านสามารถตรวจสอบยอดขาย รายการสั่งซื้อ และข้อมูลลูกค้าได้แบบเรียลไทม์

1.4.5 ช่วยลดภาระงานของพนักงาน ทำให้สามารถให้บริการลูกค้าได้อย่างมีประสิทธิภาพมากขึ้น

บทที่ 2
วิธีดำเนินการ

2.1 ขั้นตอนการดำเนินการ

การดำเนินการพัฒนาโครงงานระบบจัดการร้านคาเฟ่ (Café Management System) แบ่งออกเป็นขั้นตอนดังนี้

2.1.1 ศึกษาและวิเคราะห์ความต้องการของระบบ

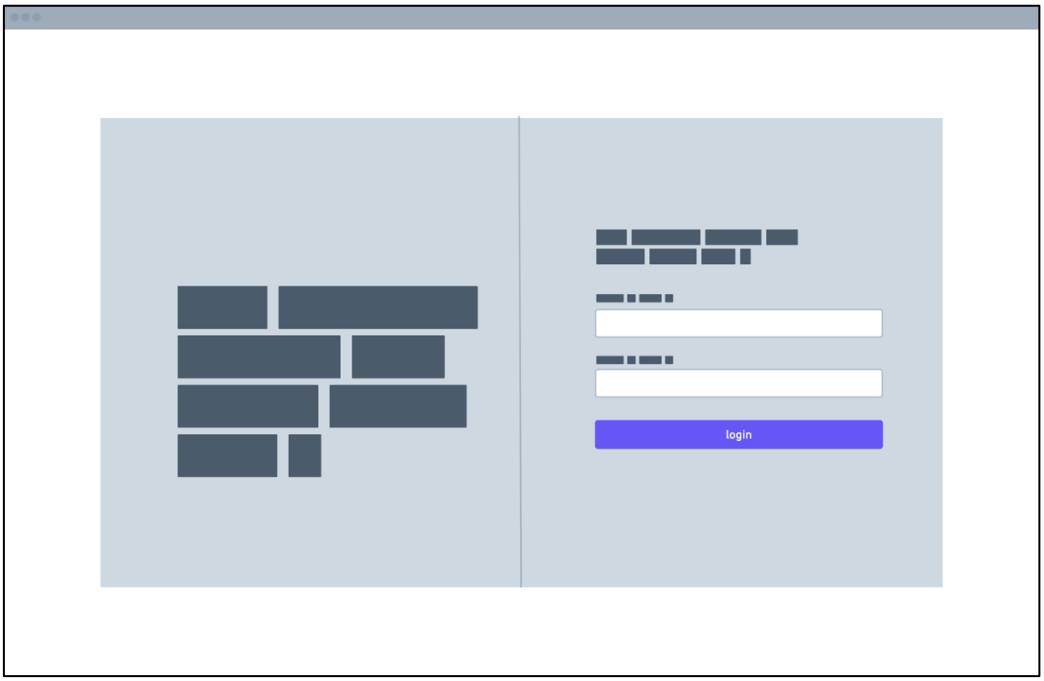
คณะผู้จัดทำได้ศึกษาข้อมูลพื้นฐานของการบริหารจัดการร้านคาเฟ่จริง รวมถึงกระบวนการทำงานภายในร้าน ได้แก่ การรับออเดอร์ การชำระเงิน การจัดการเมนู และการจัดการข้อมูลพนักงาน จากนั้นรวบรวมความต้องการของผู้ใช้งาน เพื่อระบุปัญหาและสิ่งที่ระบบต้องตอบโจทย์ ได้แก่ ลดความผิดพลาดในการรับออเดอร์ การบันทึกข้อมูลการขาย และการสรุปยอดขายรายวัน

2.1.2 วิเคราะห์ระบบและออกแบบระบบ

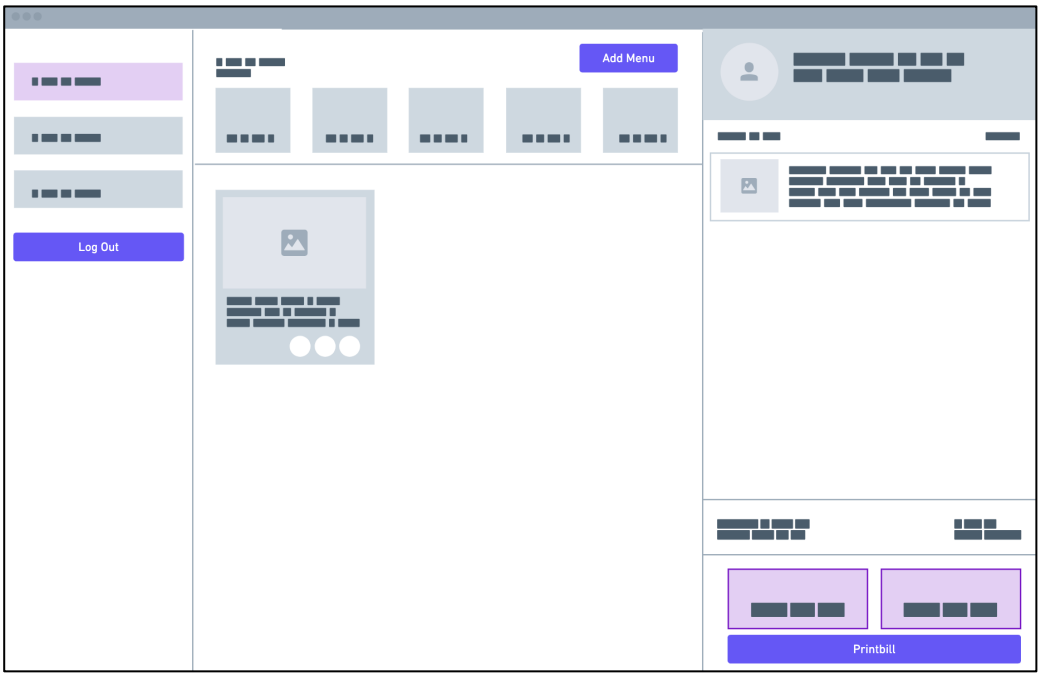
เมื่อได้ข้อมูลความต้องการแล้ว จึงดำเนินการออกแบบฐานข้อมูลเชิงสัมพันธ์ (Relational Database) ที่รองรับความสัมพันธ์แบบ One-to-One, One-to-Many และ Many-to-Many เพื่อให้ระบบสามารถจัดเก็บข้อมูลได้อย่างมีประสิทธิภาพ และออกแบบระบบโดยใช้หลักการเชิงวัตถุ (Object-Oriented Design) และใช้เครื่องมือออกแบบ ได้แก่ Use Case Diagram เพื่อแสดงความสัมพันธ์ระหว่างผู้ใช้และระบบ, Sequence Diagram เพื่อแสดงลำดับการทำงานของแต่ละกระบวนการ, Class Diagram เพื่อแสดงโครงสร้างและความสัมพันธ์ของคลาสข้อมูล และ Activity Diagram เพื่ออธิบายขั้นตอนการทำงานของแต่ละฟังก์ชัน

2.1.3 ออกแบบส่วนติดต่อผู้ใช้ (User Interface Design)

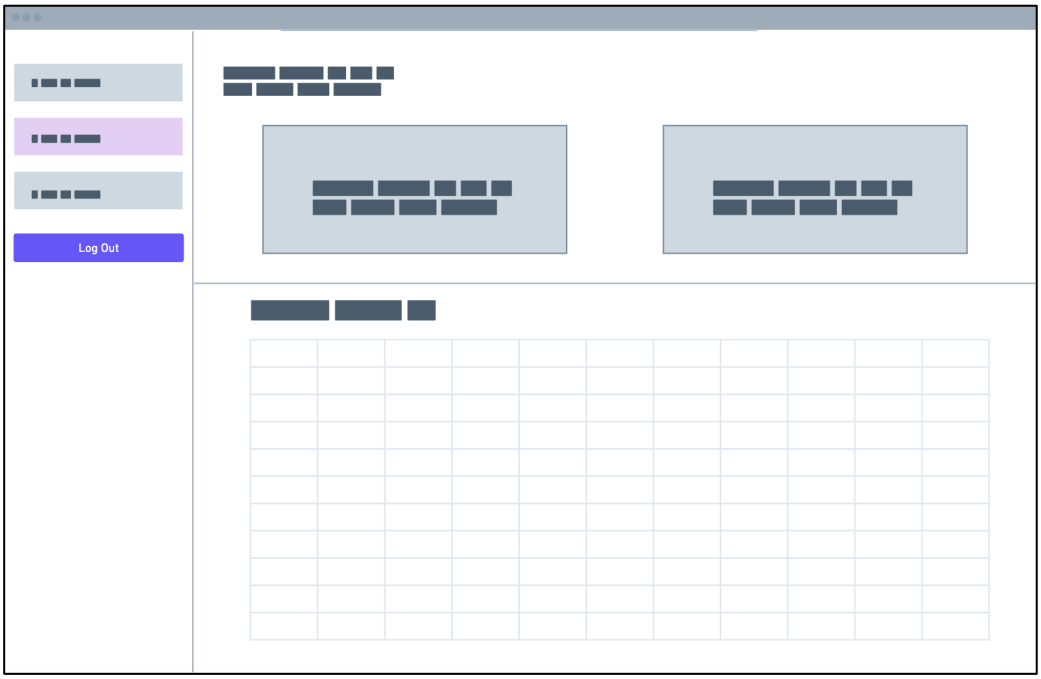
ได้ดำเนินการออกแบบหน้าจอการใช้งาน (UI) ให้มีความสวยงาม เข้าใจง่าย และเหมาะสมกับการใช้งานจริงในร้านคาเฟ่ โดยใช้ Whimsical ในการร่างแบบต้นฉบับ (Prototype)



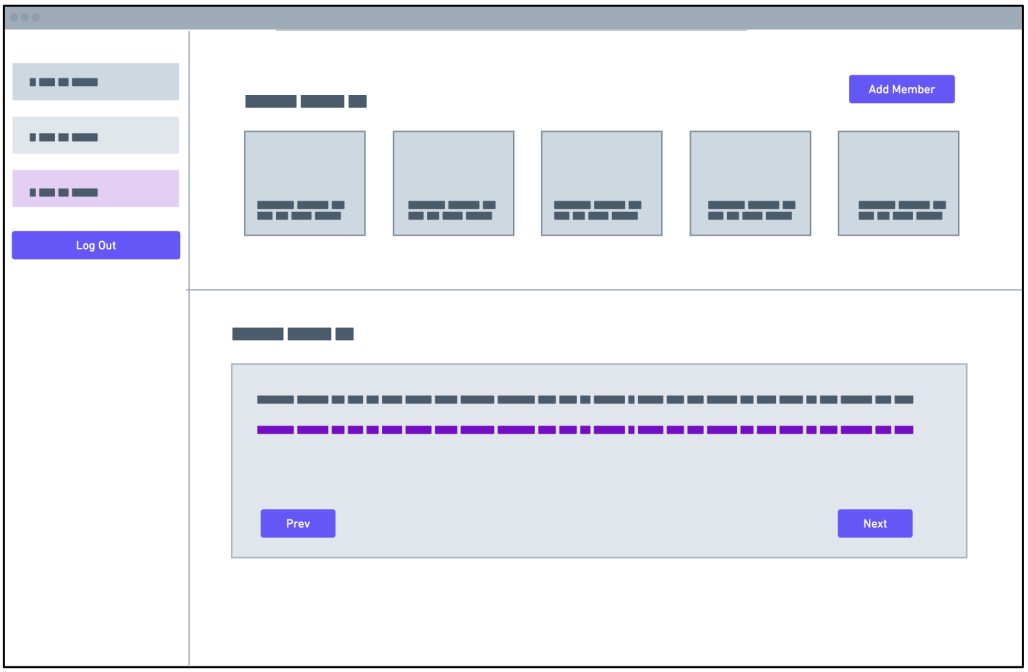
ภาพที่ 1 Wireframe ของหน้า Login



ภาพที่ 2 Wireframe ของหน้า MenuOrder



ภาพที่ 3 Wireframe ของหน้า History



ภาพที่ 4 Wireframe ของหน้า Member

2.1.4 การพัฒนาระบบ

เริ่มการพัฒนาระบบจริงตามแบบที่ออกแบบไว้ โดยแบ่งเป็น 3 ส่วนหลัก ได้แก่

(1) Frontend : ใช้ Next.js และ Tailwind CSS ในการพัฒนาอินเทอร์เฟซผู้ใช้ เพื่อให้เว็บตอบสนองรวดเร็วและใช้งานง่าย

(2) Backend : ใช้ Java Spring Boot Framework ในการพัฒนาระบบเบื้องหลังของเว็บไซต์

(3) Database : ใช้ SQLite ในการจัดเก็บข้อมูล ได้แก่ เมนูสินค้า รายการขาย ใบเสร็จ และข้อมูลพนักงาน

2.1.5 การทดสอบระบบ

ได้ทำการทดสอบระบบในแต่ละส่วน และทดสอบการทำงานรวมกัน โดยใช้ Postman สำหรับทดสอบ REST API ของ Backend เพื่อให้มั่นใจว่าระบบสามารถรับ–ส่งข้อมูลได้ถูกต้อง และไม่มีข้อผิดพลาดในการประมวลผล

2.1.6 การปรับปรุงและเผยแพร่ระบบ

เมื่อผ่านการทดสอบแล้ว จึงนำระบบขึ้นสู่เซิร์ฟเวอร์โดยใช้ Hetzner Cloud เป็นพื้นที่โฮสต์ และใช้ Cloudflare เพื่อช่วยด้านความปลอดภัยและเพิ่มความเร็วในการเข้าถึงเว็บไซต์

2.2 เครื่องมือและเทคโนโลยีที่ใช้

2.2.1 เว็บไซต์ (Frontend)

2.2.1.1 Next.js

Next.js เป็น React framework ที่จัดการด้านการสร้าง user interface (UI) สำหรับแอปพลิเคชันเว็บที่มีการเรนเดอร์ด้านเซิร์ฟเวอร์ (server-side rendering) ได้ รวมถึงการสร้างไซต์ที่มี static generation ที่ได้รับการปรับปรุงสำหรับประสิทธิภาพและความเร็ว มีจุดเด่นในการทำให้เว็บแอปพลิเคชันที่สร้างด้วย React สามารถทำการ SEO ได้ดียิ่งขึ้น และช่วยให้การพัฒนามีประสิทธิภาพกว่า

คุณสมบัติของ Next.js มีดังนี้

(1) Server-side Rendering (SSR) : การ Render HTML จากฝั่ง Server

(2) Static Site Generation (SSG) : การ Render HTML ออกมาตั้งแต่ตอน Build

(3) Automatic Code Splitting : การทำ Code Splitting โดยอัตโนมัติ ซึ่งโค้ดที่ไม่จำเป็นจะไม่ถูกโหลดพร้อมกันทั้งหมดเมื่อเข้าเว็บไซต์ แต่จะถูกโหลดเมื่อจำเป็นจริงๆ ช่วยให้เว็บไซต์ทำงานได้เร็วขึ้น

(4) API Routes : ฟีเจอร์ที่ช่วยให้นักพัฒนาสามารถสร้าง API endpoint ได้โดยตรงภายในโปรเจกต์ Next.js ของตัวเอง

(5) Built-in CSS and Sass Support : Next.js ได้ติดตั้ง Sass มาให้เราใช้งานได้ทันที และยังสามารถเลือก Tailwind CSS เข้ามาใช้งานใน Project ได้ตั้งแต่ขั้นตอนแรกของการติดตั้ง ช่วยให้การพัฒนาเว็บไซต์ง่ายและสะดวกขึ้น

2.2.1.2 Tailwind CSS

Tailwind CSS คือ CSS Utility Framework ที่ช่วยให้นักพัฒนาสร้าง UI ที่สำคัญได้ด้วยตัวเองอย่างรวดเร็ว และยังสามารถปรับแต่งในรายละเอียดปลีกย่อยได้ง่าย เนื่องจากมาพร้อมกับ class สำเร็จรูปสุดอเนกประสงค์ที่ใช้งานได้ทันทีในกรณีที่ต้องการเปลี่ยน UI หลักของเฟรมเวิร์ก เช่น สี ขนาด การจัดวาง หรือปุ่มต่างๆ

2.2.2 ระบบเบื้องหลังของเว็บไซต์ (Backend)

(1) Java Spring Boot

Spring Boot คือ Framework ใน Spring อันหนึ่ง สามารถช่วยทำให้สร้าง Web application หรือ Web service ได้ง่ายขึ้น เพราะ Spring Boot มี Auto Configuration ซึ่งช่วยลดความยุ่งยากในการกำหนดค่าต่างๆ และสามารถใช้งานได้ทันที เนื่องจาก Spring Boot มี Java Web Server ที่ built-in มาให้แล้ว คือ Tomcat ทำให้ง่ายต่อการใช้งาน โดยมี Port default คือ 8080 ซึ่งสามารถแก้ไขเปลี่ยน Port ได้ที่ไฟล์ application.properties

2.2.3 ฐานข้อมูล (Database)

(1) SQLite

SQLite เป็นไลบรารี่ที่ถูกสร้างโดยใช้ภาษา C เพื่อเป็นตัวจัดการฐานข้อมูล (database engine) สำหรับฐานข้อมูลเชิงสัมพันธ์ (relational database) ที่มีตัวจัดการฐานข้อมูลแบบทรานแซคชั่นในตัวเอง (transactional) มีขนาดเล็ก ทำงานได้เร็ว ทำงานได้ด้วยตนเอง เชื่อถือได้สูง ไม่ต้องมีเซิร์ฟเวอร์ ไม่ต้องตั้งค่าใดๆก่อนใช้งาน และมีความสามารถในการจัดการฐานข้อมูลด้วย SQL อย่างครบถ้วน

2.2.4 เครื่องมือ (Tools)

(1) Visual Studio Code (VSCode)

Visual Studio Code เป็นโปรแกรมแก้ไขซอร์สโค้ดที่พัฒนาโดยไมโครซอฟท์ สำหรับ Windows, Linux และ macOS มีการสนับสนุนสำหรับการดีบัก การควบคุม Git ในตัวและ GitHub การเน้นไวยากรณ์ การเติมโค้ดอัจฉริยะ ตัวอย่าง และ code refactoring สามารถปรับแต่งได้หลายอย่าง ให้ผู้ใช้สามารถเปลี่ยนธีม แป้นพิมพ์ลัด การตั้งค่า และติดตั้งส่วนขยายที่เพิ่มฟังก์ชันการทำงานเพิ่มเติม ซอร์สโค้ดนั้นฟรีและโอเพนซอร์ส และเผยแพร่ภายใต้สิทธิ์การใช้งาน MIT ไบนารีที่คอมไพล์แล้วเป็นฟรีแวร์และฟรีสำหรับการใช้ส่วนตัวหรือเพื่อการค้า

(2) Postman

Postman เป็นเครื่องมือทดสอบที่ช่วยให้นักพัฒนาแอปพลิเคชัน สามารถสร้าง ทดสอบ และแก้ไข API ได้ ซึ่งอินเทอร์เฟซผู้ใช้ของเครื่องมือทดสอบ API นี้ช่วยให้ developer และ QA สามารถส่ง HTTP request และติดตามการตอบสนองได้ง่าย

(3) Git

Git คือ VCS (Version Control System) ที่ช่วยให้จัดการเวอร์ชันของโค้ดได้ง่าย และสะดวก มีประสิทธิภาพ สามารถย้อนกลับไปสู่เวอร์ชันของโค้ดก่อนหน้าได้ ช่วยให้การทำงานร่วมกับ developers หรือ programmers ในทีมได้สะดวกและมีประสิทธิภาพ

(4) GitHub Desktop

GitHub Desktop เป็นแอปพลิเคชันโอเพนซอร์สที่ช่วยให้ใช้คำสั่ง Git ผ่านอินเทอร์เฟซแบบกราฟิก (GUI) แทนการใช้บรรทัดคำสั่ง (CLI) เพื่อจัดการโปรเจกต์ของเราบน GitHub Docs ช่วยให้การโคลน (clone) สร้างสาขา (branch) และส่งการเปลี่ยนแปลง (commit & push) ทำได้ง่ายขึ้น

2.2.5 Deployment / Hosting

(1) Hetzner

Hetzner เป็นบริษัทผู้ให้บริการ เว็บโฮสติ้ง (Web Hosting) และ โครงสร้างพื้นฐานคลาวด์ (Cloud Infrastructure Provider) จากประเทศเยอรมนี

(2) Cloudflare

Cloudflare คือ ผู้ให้บริการระบบ Network เครือข่ายข้อมูลที่ใหญ่ที่สุดในโลก และให้ทุกคนที่มีเว็บสามารถใช้งานได้ฟรี ช่วยเพิ่มความเร็วในการเปิดหน้าเว็บไซต์ และเป็นตัวช่วยด้านความปลอดภัยอีกชั้นหนึ่ง เพื่อให้ไม่ให้ Server ถูกแฮค หรือถูกจู่โจมแบบตรงๆ โดยใช้เครือข่าย network ทั่วโลกเป็นตัวกลางในการทำงาน

2.3 การวิเคราะห์และออกแบบระบบ

2.3.1 Use Case

(1) การเข้าสู่ระบบ

ตารางที่ 1 Use Case ของการเข้าสู่ระบบ

หัวข้อ	รายละเอียด
Use Case Name	เข้าสู่ระบบ
Primary Actor	Staff
Stakeholders and Interests	Staff ต้องการเข้าสู่ระบบจัดการร้านคาเฟ่เพื่อดูแลและดำเนินการต่าง ๆ ได้
Preconditions	Staff ต้องทราบ Username และ Password ที่ถูกต้องสำหรับการเข้าสู่ระบบ
Postconditions	ระบบอนุญาตให้เข้าสู่หน้า WebApp สำหรับจัดการคาเฟ่ ได้สำเร็จ และสามารถใช้งานฐานข้อมูลได้
Main Success Scenario	1. Staff เปิดหน้า เข้าสู่ระบบ (Login Page) ใน WebApp 2. กรอก Username และ Password ลงในแบบฟอร์ม 3. ระบบตรวจสอบความถูกต้องของข้อมูลจากฐานข้อมูล 4. หากข้อมูลถูกต้อง ระบบเข้าสู่หน้าหลักของระบบจัดการคาเฟ่
Exception Conditions	หาก Username หรือ Password ไม่ถูกต้อง ระบบแจ้งข้อผิดพลาด (“ชื่อผู้ใช้หรือรหัสผ่านไม่ถูกต้อง”) และให้ Staff กรอกใหม่อีกครั้ง

(2) การจัดการข้อมูลเมนู

ตารางที่ 2 Use Case ของการจัดการข้อมูลเมนู

หัวข้อ	รายละเอียด
Use Case Name	จัดการข้อมูลเมนู
Primary Actor	Staff
Stakeholders and Interests	Staff ต้องการจัดการข้อมูลเมนูให้ถูกต้อง เช่น การแก้ไขข้อมูลหรือลบเมนูที่ไม่ต้องการ
Preconditions	Staff ต้องเข้าสู่ระบบจัดการเรียบร้อยแล้ว
Postconditions	ระบบบันทึกการจัดการเมนูสำเร็จ (แก้ไขหรือลบ) และอัปเดตข้อมูลในฐานข้อมูลเรียบร้อย
Main Success Scenario	1. Staff เปิดหน้า Menu Order 2. เลือกเมนูที่ต้องการจัดการ 2.1 หากต้องการแก้ไขข้อมูลเมนู กดปุ่ม ดินสอ(Edit) 2.1.1 กรอกข้อมูลที่ต้องการแก้ไขให้ถูกต้อง 2.1.2 กดปุ่ม ยืนยัน (Save) เพื่อบันทึกการแก้ไข 2.1.3 ระบบบันทึกข้อมูลการแก้ไขลงในฐานข้อมูล 2.2 หากต้องการลบเมนู กดปุ่ม ถังขยะ (Delete) 2.2.1 ระบบลบข้อมูลเมนูนั้นออกจากฐานข้อมูล 3. ระบบแสดงข้อความแจ้งว่า “ทำรายการสำเร็จ”
Exception Conditions	หากกรอกข้อมูลในการแก้ไขไม่ครบถ้วนหรือไม่ถูกต้อง ระบบแจ้งข้อผิดพลาด และให้ Staff แก้ไขข้อมูลก่อนยืนยันอีกครั้ง

(3) การเพิ่มข้อมูลเมนู

ตารางที่ 3 Use Case ของการเพิ่มข้อมูลเมนู

หัวข้อ	รายละเอียด
Use Case Name	เพิ่มข้อมูลเมนู (Add New Menu)
Primary Actor	Staff
Stakeholders and Interests	Staff ต้องการเพิ่มข้อมูลเมนู
Preconditions	Staff ต้องเข้าสู่ระบบจัดการเรียบร้อยแล้ว
Postconditions	เพิ่มข้อมูลเมนูที่ต้องการสำเร็จ และบันทึกข้อมูลลงในฐานข้อมูลเรียบร้อย
Main Success Scenario	1. Staff เปิดหน้า Menu Order 2. เลือกเมนู Add New Menu 3. กรอกข้อมูลเมนูใหม่ให้ครบถ้วน (เช่น ชื่อเมนู, ราคา, หมวดหมู่, รูปภาพ) 4. Staff กดยืนยันการเพิ่มเมนู 5. ระบบตรวจสอบความถูกต้องของข้อมูล 6. ระบบบันทึกข้อมูลเมนูลงในฐานข้อมูล 7. ระบบแสดงข้อความแจ้งว่า “เพิ่มเมนูสำเร็จ”
Exception Conditions	หากกรอกข้อมูลไม่ครบถ้วนหรือไม่ถูกต้อง ระบบแจ้งข้อผิดพลาด และให้ Staff แก้ไขข้อมูลก่อนบันทึกอีกครั้ง

(4) กระบวนการขาย

ตารางที่ 4 Use Case ของกระบวนการขาย

หัวข้อ	รายละเอียด
Use Case Name	กระบวนการขาย (Process Sale)
Primary Actor	Staff
Stakeholders and Interests	1. Staff ต้องการทำรายการรับชำระได้ถูกต้อง ไม่มีข้อผิดพลาด 2. ลูกค้า ต้องการซื้อสินค้าและได้รับบริการที่รวดเร็ว
Preconditions	Staff ต้อง Login เข้าระบบของร้านก่อนเสมอถึงใช้งานระบบได้
Postconditions	ระบบบันทึกรายการขายลงฐานข้อมูลได้สำเร็จ สามารถพิมพ์ใบเสร็จและรับชำระได้ครบถ้วน
Main Success Scenario	1. Staff เปิดหน้า Menu Order 2. Staff เพิ่มเมนูตามที่ลูกค้าสั่ง 3. Staff เลือกรูปแบบ Payment ที่ลูกค้าใช้ในการชำระ 4. ลูกค้าชำระเงินตามจำนวนที่ระบบคำนวณไว้ 5. Staff กด Print Bill เพื่อพิมพ์ใบเสร็จให้ลูกค้า 6. ระบบบันทึกข้อมูลการขายลงในฐานข้อมูล
Exception Conditions	1. หาก Staff เลือกเมนูผิดพลาด ทำการลบรายการและเพิ่มใหม่ตามที่ลูกค้าสั่ง 2. หาก Staff เลือก Payment ผิด ทำการเลือกใหม่ให้ถูกต้องก่อนยืนยันการชำระ

(5) การจัดการข้อมูลพนักงาน

ตารางที่ 5 Use Case ของการจัดการข้อมูลพนักงาน

หัวข้อ	รายละเอียด
Use Case Name	จัดการข้อมูลพนักงาน
Primary Actor	Staff
Stakeholders and Interests	Staff ต้องการจัดการข้อมูลพนักงาน เช่น การแก้ไขหรือลบข้อมูล เพื่อให้ข้อมูลพนักงานถูกต้องและเป็นปัจจุบัน
Preconditions	Staff ต้องเข้าสู่ระบบจัดการเรียบร้อยแล้ว
Postconditions	ระบบบันทึกการจัดการข้อมูลพนักงาน(แก้ไขหรือลบ) และอัปเดตข้อมูลในฐานข้อมูลเรียบร้อย
Main Success Scenario	1. Staff เปิดหน้า Member 2. เลือกฟังก์ชันที่ต้องการจัดการ 2.1 หากต้องการแก้ไขข้อมูลพนักงาน กดปุ่ม Edit 2.1.1 กรอกข้อมูลที่ต้องการแก้ไขให้ถูกต้อง 2.1.2 กดปุ่ม ยืนยัน (Save) เพื่อบันทึกการแก้ไข 2.1.3 ระบบบันทึกข้อมูลการแก้ไขลงในฐานข้อมูล 2.2 หากต้องการลบพนักงาน กดปุ่ม Delete 2.2.1 ระบบลบข้อมูลเมนูนั้นออกจากฐานข้อมูล 3. ระบบแสดงข้อความแจ้งว่า “ทำรายการสำเร็จ”
Exception Conditions	หากกรอกข้อมูลในการแก้ไขไม่ครบถ้วนหรือไม่ถูกต้อง ระบบแจ้งข้อผิดพลาด และให้ Staff แก้ไขข้อมูลก่อนยืนยันอีกครั้ง

(6) การเพิ่มข้อมูลพนักงาน

ตารางที่ 6 Use Case ของการเพิ่มข้อมูลพนักงาน

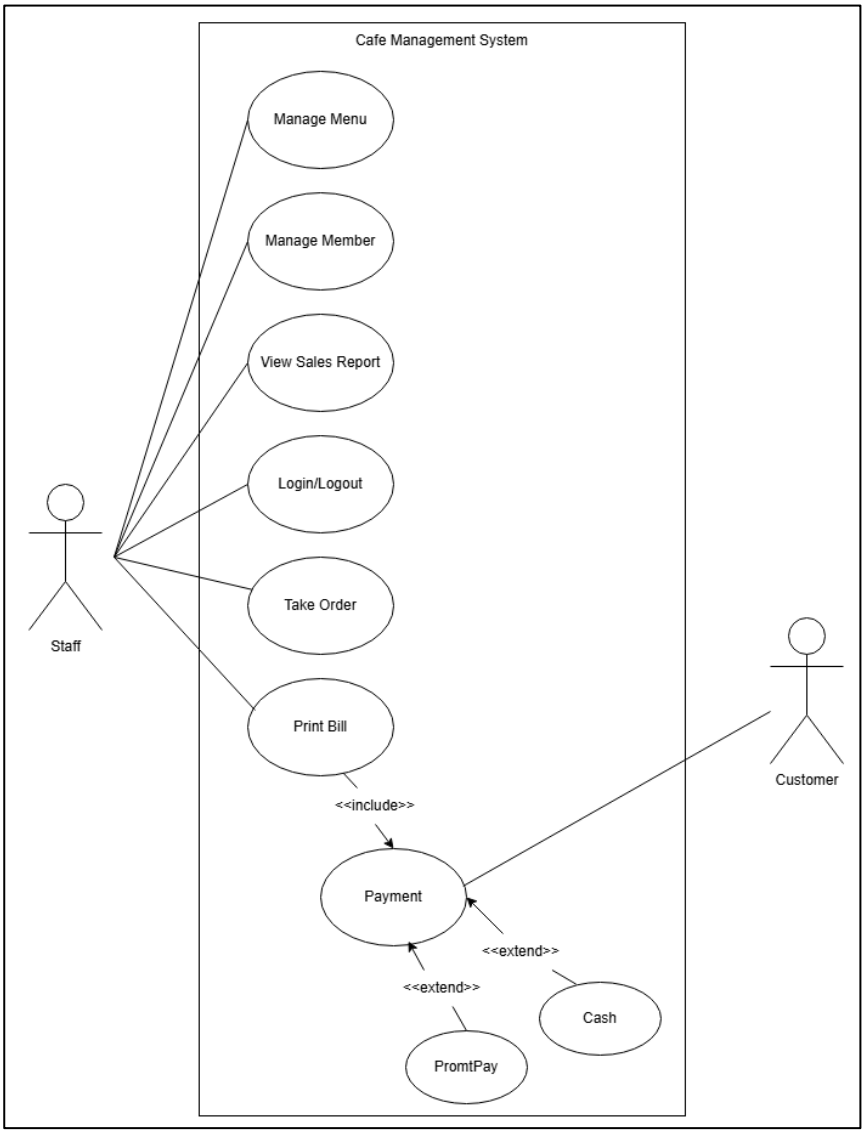
หัวข้อ	รายละเอียด
Use Case Name	เพิ่มข้อมูลพนักงาน (Add Member)
Primary Actor	Staff
Stakeholders and Interests	Staff ต้องการเพิ่มข้อมูลพนักงานใหม่เข้าสู่ระบบ เพื่อให้ข้อมูลบุคลากรเป็นปัจจุบันและสามารถจัดการได้
Preconditions	Staff ต้องเข้าสู่ระบบจัดการเรียบร้อยแล้ว
Postconditions	เพิ่มข้อมูลพนักงานที่ต้องการสำเร็จ และบันทึกข้อมูลลงในฐานข้อมูลเรียบร้อย
Main Success Scenario	1. Staff เปิดหน้า Member 2. เลือก Add New Member 3. กรอกข้อมูลพนักงานใหม่ให้ครบถ้วน (เช่น ชื่อ, สถานะพนักงาน) 4. Staff กดยืนยันการเพิ่มข้อมูลพนักงาน 5. ระบบตรวจสอบความถูกต้องของข้อมูล 6. ระบบบันทึกข้อมูลพนักงานลงในฐานข้อมูล 7. ระบบแสดงข้อความแจ้งว่า “เพิ่มพนักงานสำเร็จ
Exception Conditions	หากกรอกข้อมูลไม่ครบถ้วนหรือไม่ถูกต้อง ระบบแจ้งข้อผิดพลาด และให้ Staff แก้ไขข้อมูลก่อนบันทึกอีกครั้ง

(7) การดูรายงานยอดขาย

ตารางที่ 7 Use Case ของการดูรายงานยอดขาย

หัวข้อ	รายละเอียด
Use Case Name	ดูรายงานยอดขาย
Primary Actor	Staff
Stakeholders and Interests	Staff ต้องการดูรายงานยอดขายของร้าน เช่น ยอดขายรวม ยอดขายรายเมนู หรือยอดขายรายบิล เพื่อใช้ในการวิเคราะห์และตรวจสอบข้อมูลการขาย
Preconditions	Staff ต้องเข้าสู่ระบบจัดการเรียบร้อยแล้ว
Postconditions	Staff สามารถดูรายงานยอดขายทั้งหมดจากระบบได้ เช่น ยอดรวม ยอดขายรายเมนู และยอดขายรายบิล
Main Success Scenario	1. Staff เปิดหน้า History (รายงานยอดขาย) 2. เลือกรายการที่ต้องการดู เช่น ยอดขายรวม หรือ ยอดขายรายเมนู 2.1 หากเลือกเมนูแต่ละรายการ ระบบแสดงยอดขายของเมนูนั้น 3. ระบบแสดงรายการบิลทั้งหมดพร้อมยอดขายในแต่ละบิล 4. Staff ตรวจสอบข้อมูลยอดขายได้จากหน้าระบบ
Exception Conditions	1. หากไม่มีข้อมูลยอดขาย ระบบแสดงข้อความ “ไม่พบข้อมูลยอดขาย” 2. หากระบบเชื่อมต่อฐานข้อมูลไม่ได้ ระบบแจ้งเตือน “ไม่สามารถดึงข้อมูลจากฐานข้อมูลได้ กรุณาลองใหม่อีกครั้ง”

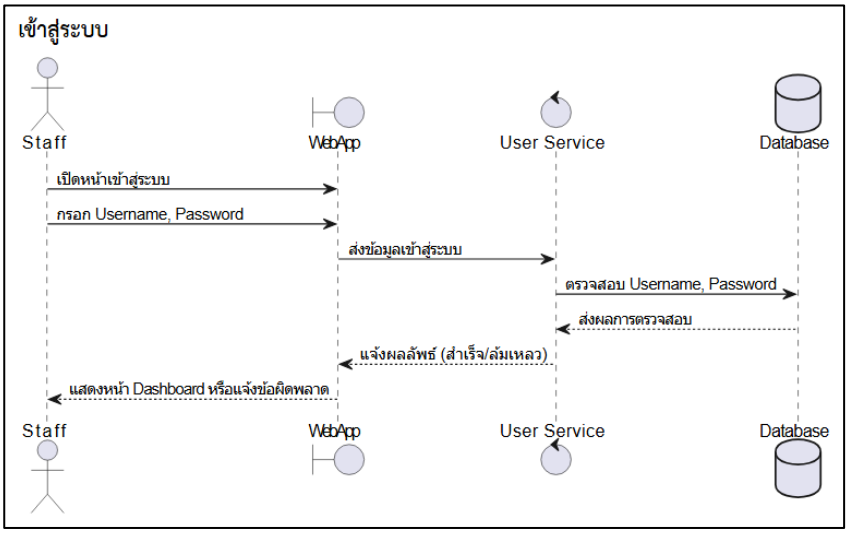
2.3.2 Use Case Diagram



ภาพที่ 5 Use Case Diagram

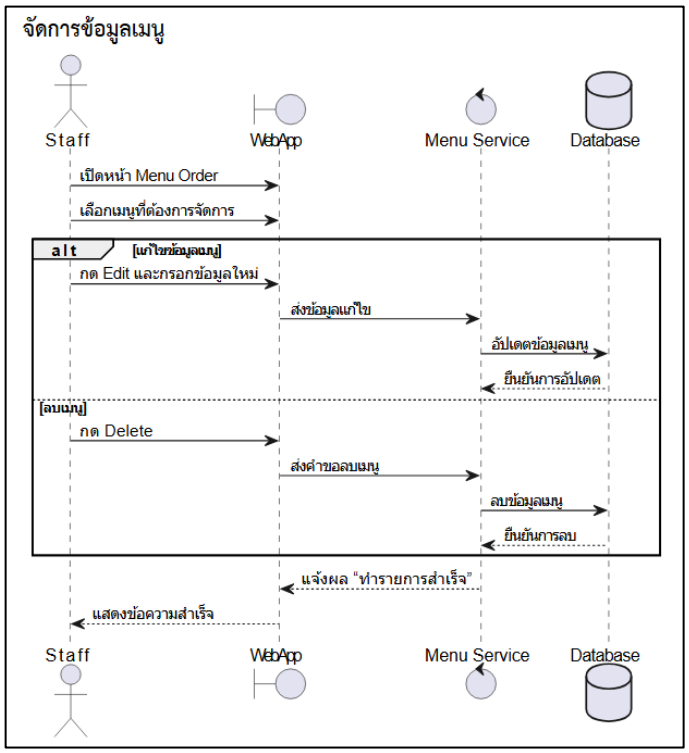
2.3.3 Sequence Diagram

(1) การเข้าสู่ระบบ



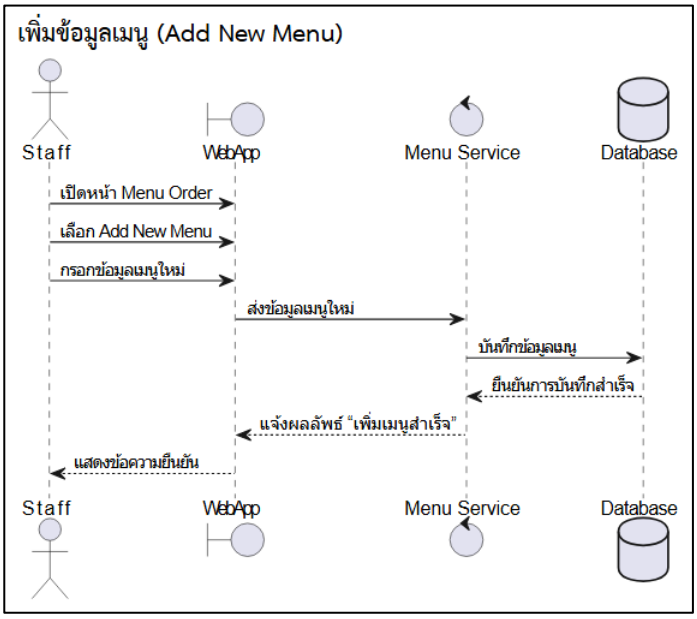
ภาพที่ 6 Sequence Diagram ของการเข้าสู่ระบบ

(2) การจัดการข้อมูลเมนู



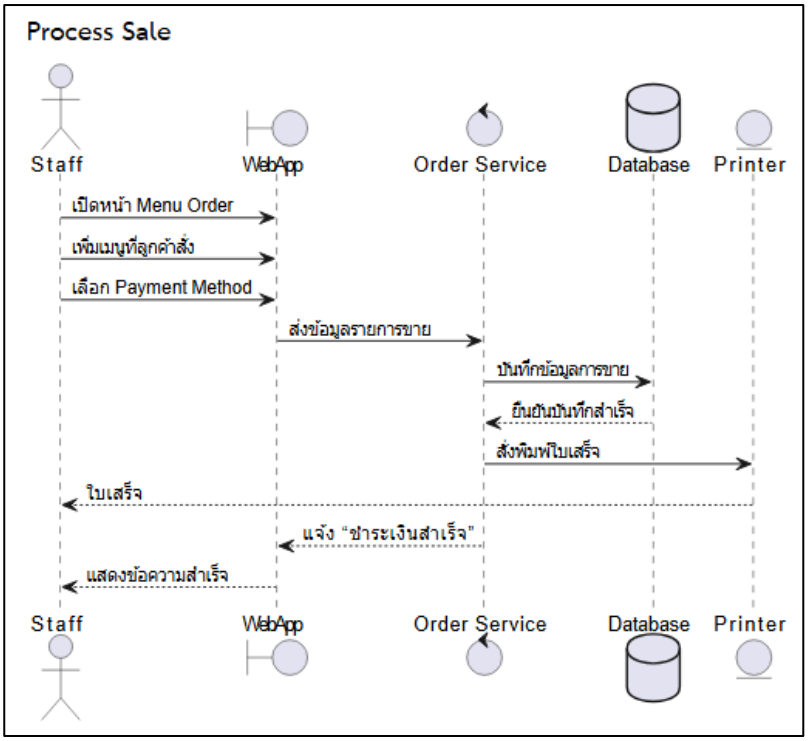
ภาพที่ 7 Sequence Diagram ของการจัดการข้อมูลเมนู

(3) การเพิ่มข้อมูลเมนู



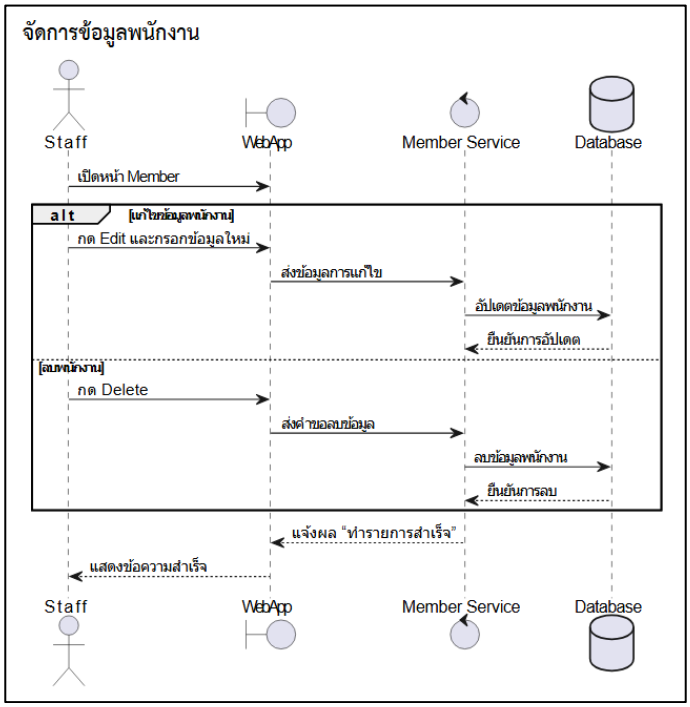
ภาพที่ 8 Sequence Diagram ของการเพิ่มข้อมูลเมนู

(4) กระบวนการขาย



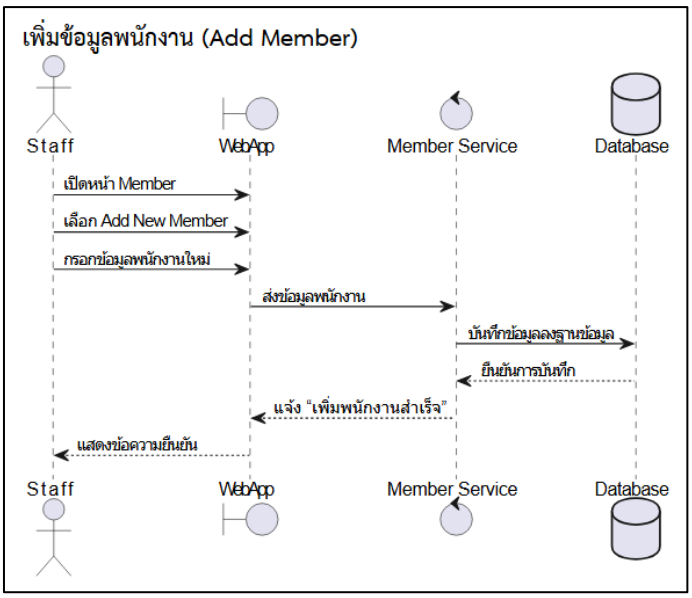
ภาพที่ 9 Sequence Diagram ของกระบวนการขาย

(5) การจัดการข้อมูลพนักงาน



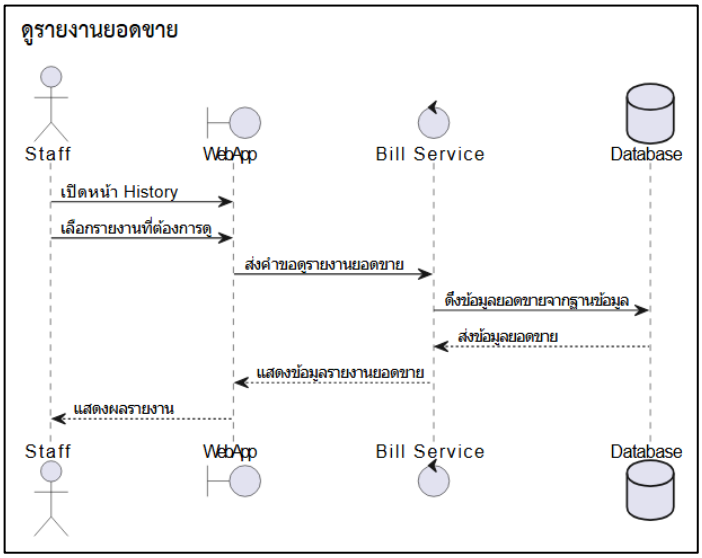
ภาพที่ 10 Sequence Diagram ของการจัดการข้อมูลพนักงาน

(6) การเพิ่มข้อมูลพนักงาน



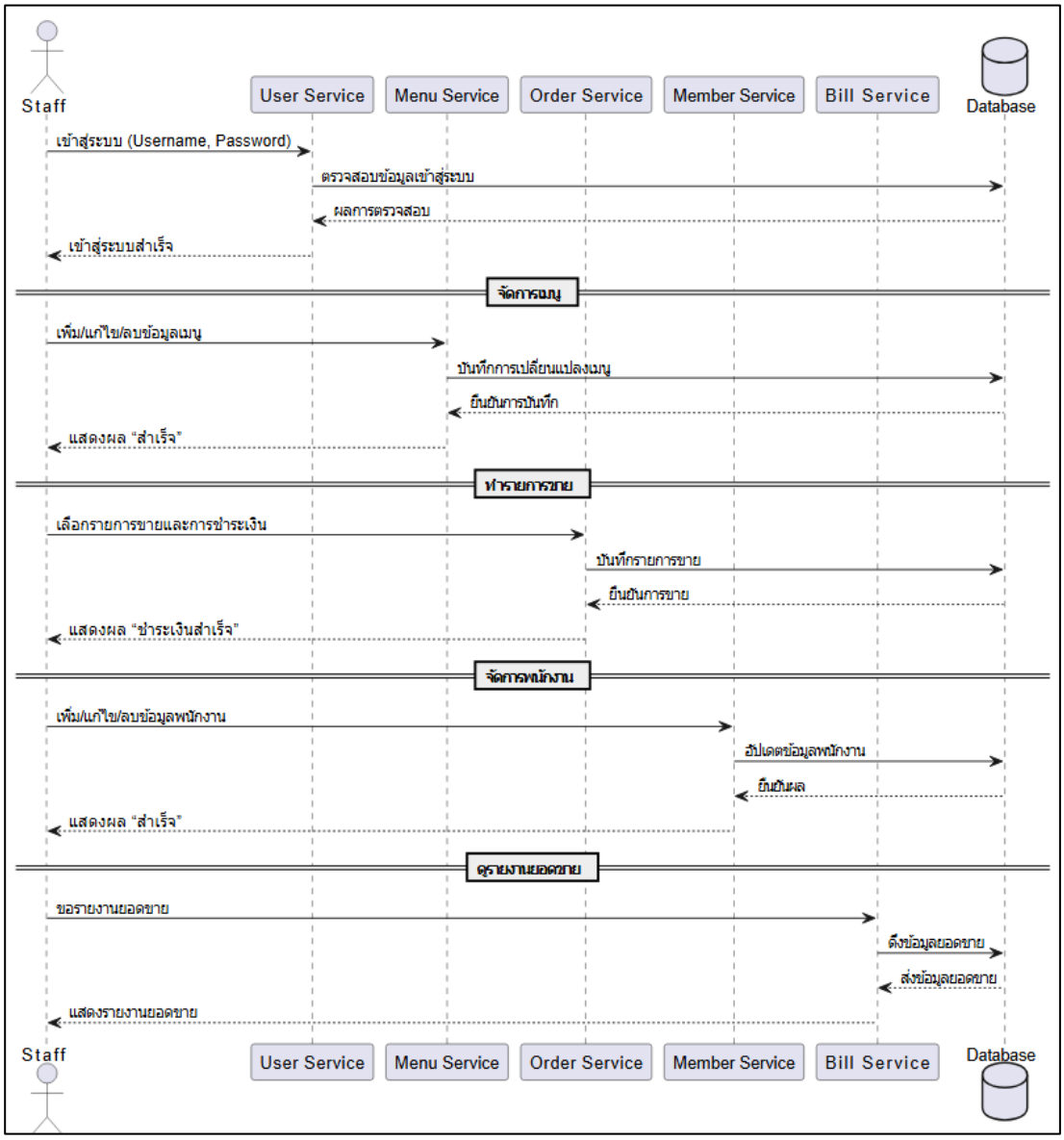
ภาพที่ 11 Sequence Diagram ของการเพิ่มข้อมูลพนักงาน

(7) การดูรายงานยอดขาย



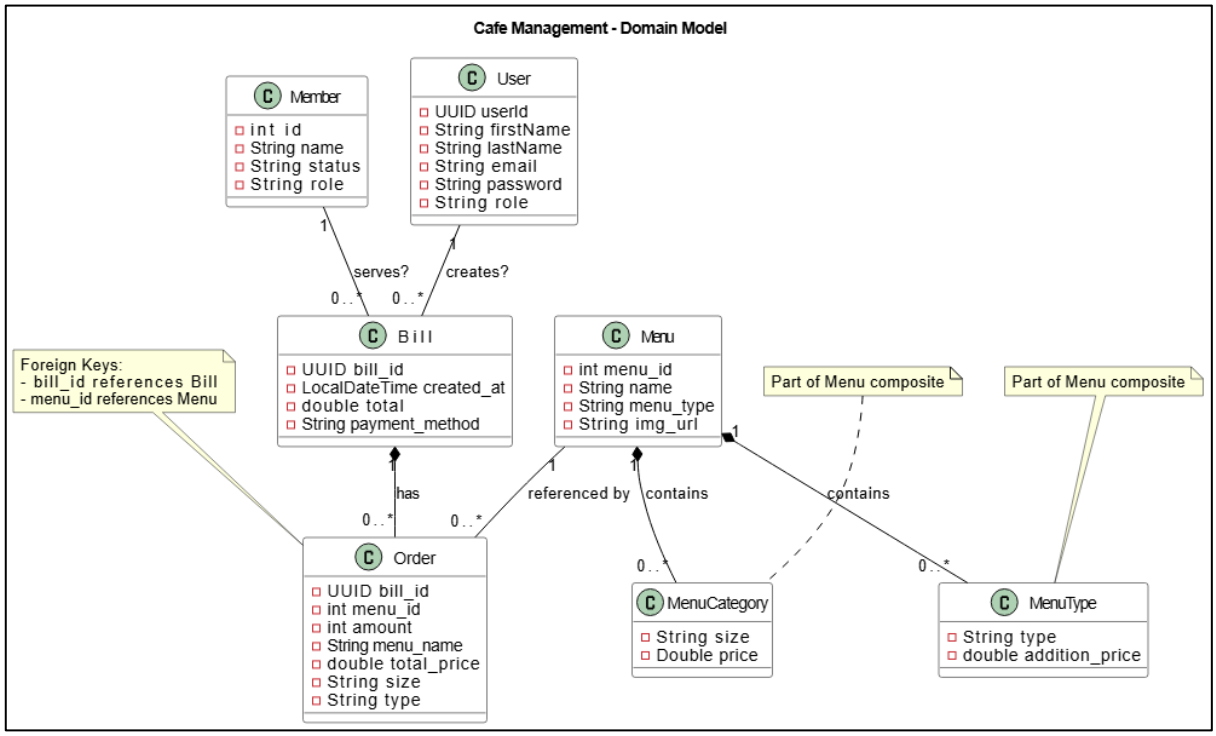
ภาพที่ 12 Sequence Diagram ของการดูรายงานยอดขาย

(8) Sequence Diagram ทั้งหมด



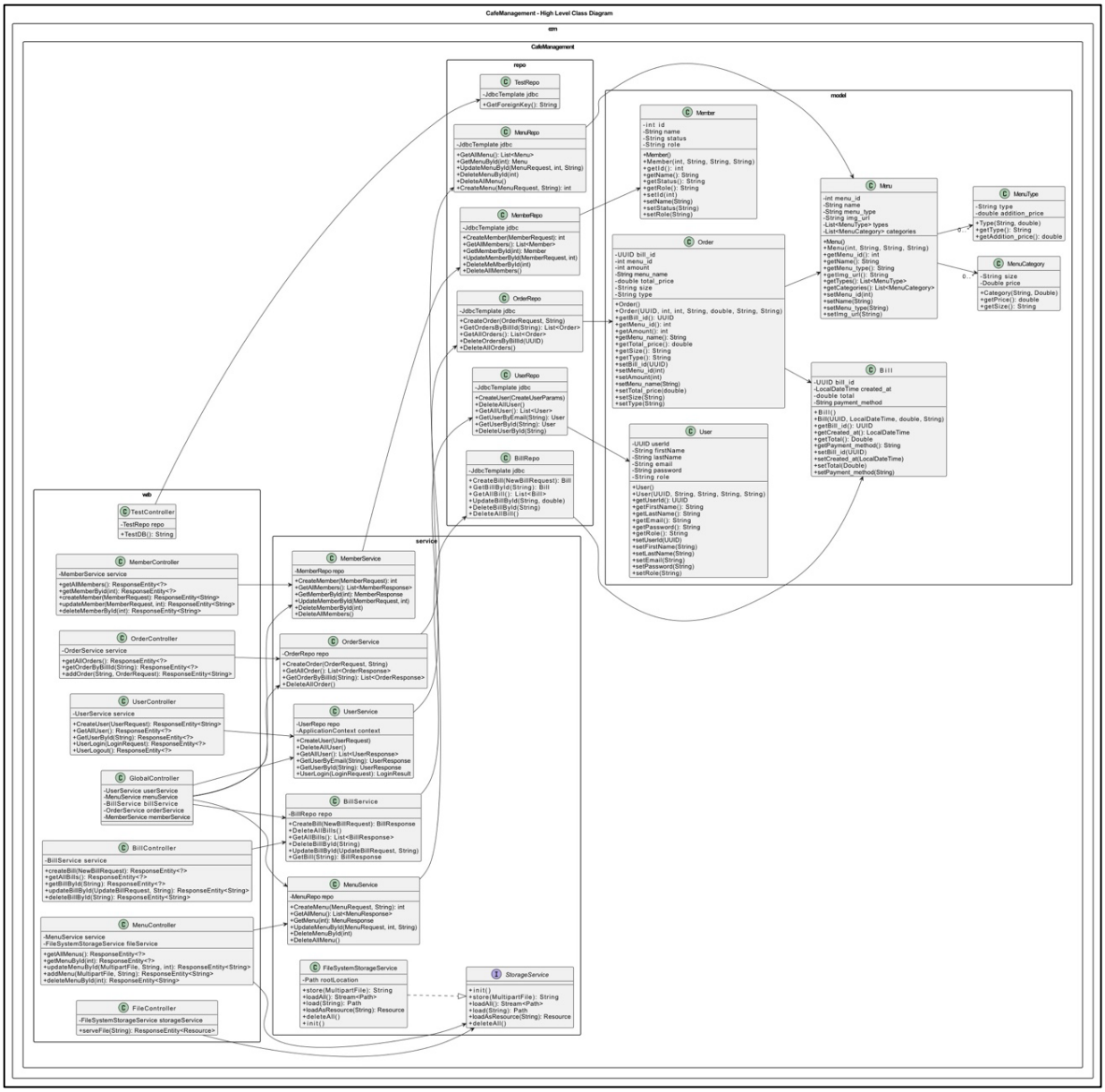
ภาพที่ 13 Sequence Diagram ทั้งหมด

2.3.4 Domain Model



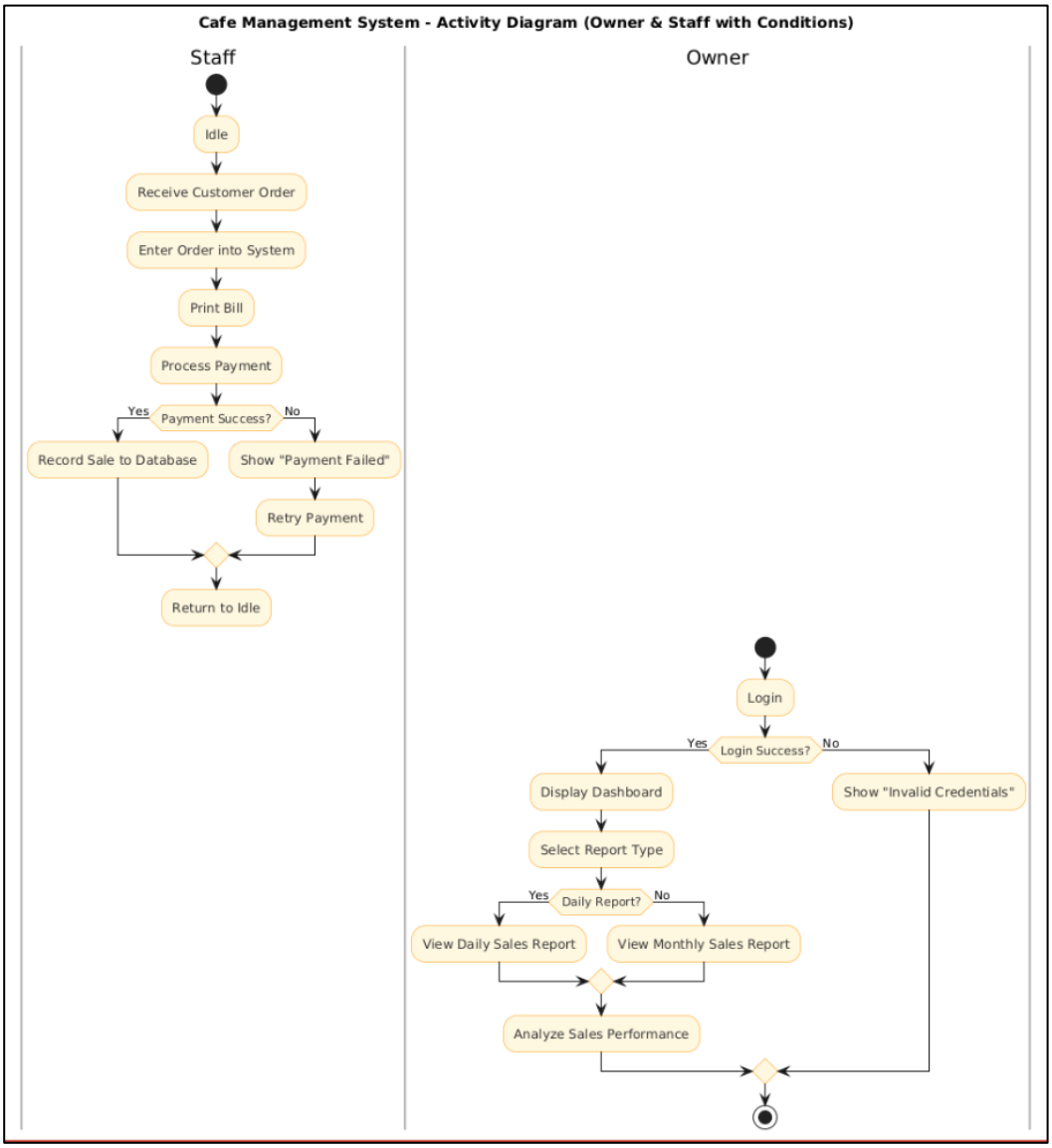
ภาพที่ 14 Domain Model

2.3.5 Class Diagram



ภาพที่ 15 Class Diagram

2.3.6 Activity Diagram



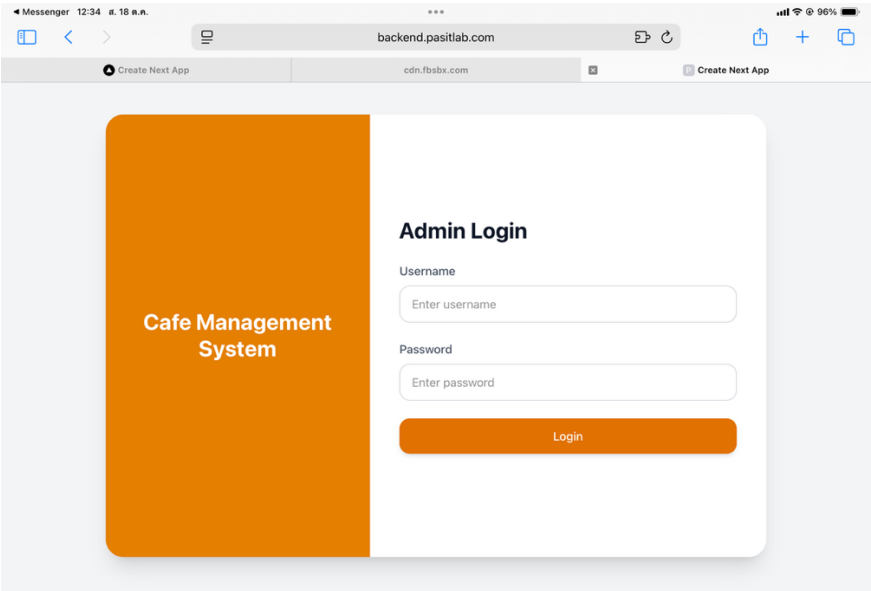
ภาพที่ 16 Activity Diagram

บทที่ 3
ผลการดำเนินการ

3.1 ผลการพัฒนาระบบ

จากการดำเนินการวิเคราะห์และออกแบบระบบเว็บไซต์การจัดการร้านคาเฟ่ ได้พัฒนาเป็นระบบที่สมบูรณ์ โดยหน้าหลักของระบบประกอบด้วย Login, MenuOrder, History, Member และ Log Out

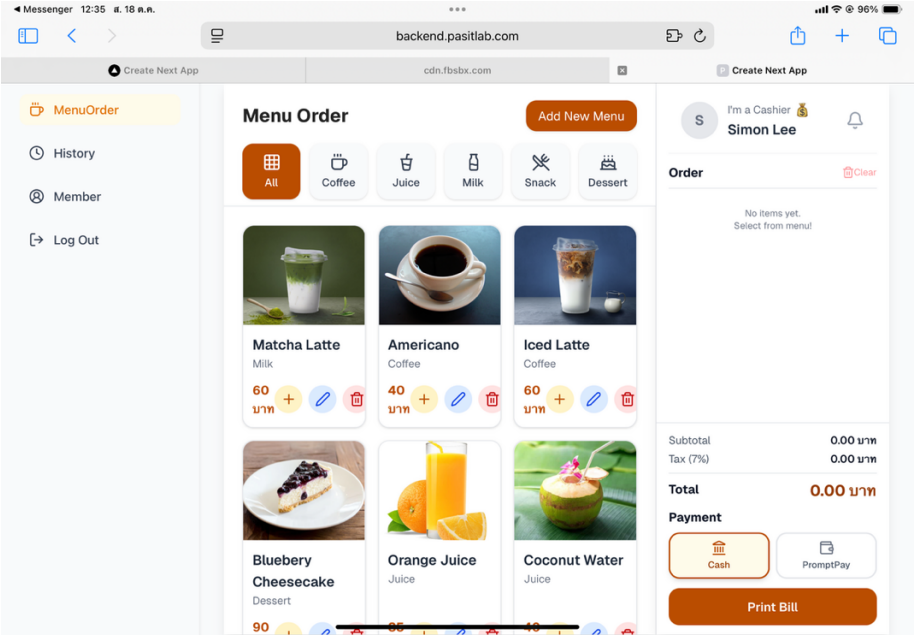
3.1.1 การเข้าสู่ระบบ (Login)



ภาพที่ 17 หน้าเข้าสู่ระบบ (Login)

จากภาพที่ 17 Admin จะต้องเข้าสู่ระบบเว็บไซต์ก่อน โดยจะต้องกรอกอีเมลผู้ใช้งาน (Username) และกรอกรหัสผ่าน (Password) เพื่อเข้าสู่ระบบเว็บไซต์

3.1.2 หน้าแสดงเมนูทั้งหมดเพื่อรับออเดอร์ (MenuOrder)



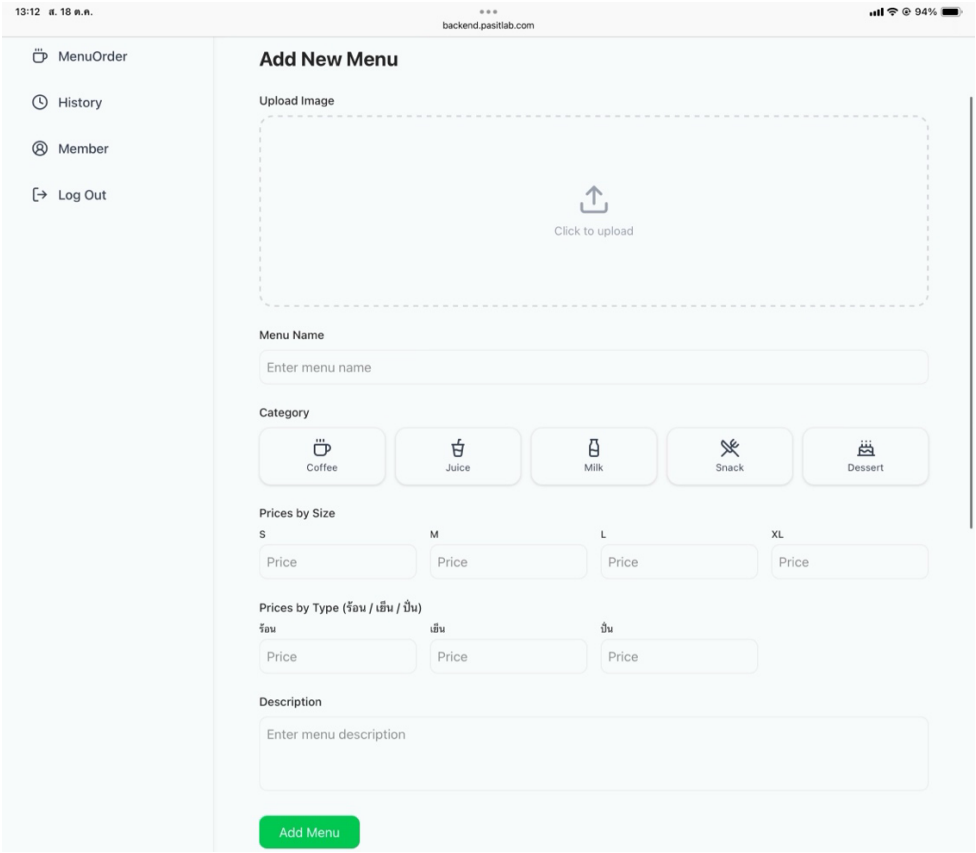
ภาพที่ 18 หน้า MenuOrder

จากภาพที่ 18 เป็นหน้า MenuOrder ซึ่งใช้สำหรับแสดงรายการเมนูทั้งหมด โดยมีส่วนประกอบของหน้าจอ ได้แก่

(1) ปุ่ม “Add New Menu” สำหรับเพิ่มเมนูใหม่เข้าระบบ

(2) แถบกรองเมนู (Filter Bar) ได้แก่ All, Coffee, Juice, Milk, Snack, Dessert ใช้กรองประเภทของสินค้า ซึ่งแต่ละเมนูมีปุ่ม + เพื่อเพิ่มเข้าในรายการสั่งซื้อ ปุ่มรูปดินสอ สำหรับแก้ไขเมนู และปุ่มรูปถังขยะ สำหรับลบเมนู

(3) ในส่วนขวามือ แสดงชื่อผู้ใช้งานปัจจุบัน (เช่น “Simon Lee” เป็นแคชเชียร์) โดยพื้นที่ Order จะแสดงรายการสินค้าที่ลูกค้าสั่ง และแสดงยอดรวม (Subtotal, Tax 7%, และ Total) ในส่วน Payment มีตัวเลือกการชำระเงิน ได้แก่ Cash และ PromptPay และปุ่ม Print Bill สำหรับพิมพ์ใบเสร็จรับเงิน



ภาพที่ 19 หน้า Add New Menu

จากภาพที่ 19 เป็นหน้า Add New Menu สำหรับผู้ดูแลร้านหรือแคชเชียร์ เพื่อเพิ่มเมนูใหม่เข้าสู่ฐานข้อมูล โดยข้อมูลที่บันทึกจะถูกนำไปแสดงในหน้า Menu Order เพื่อให้ผู้ใช้เลือกสั่งได้ในภายหลัง ในส่วนของหน้านี้ ประกอบด้วย

(1) Upload Image : เป็นกล่องอัปโหลดรูปภาพเมนู ผู้ใช้สามารถคลิก “Click to upload” เพื่อเลือกรูปภาพจากเครื่อง

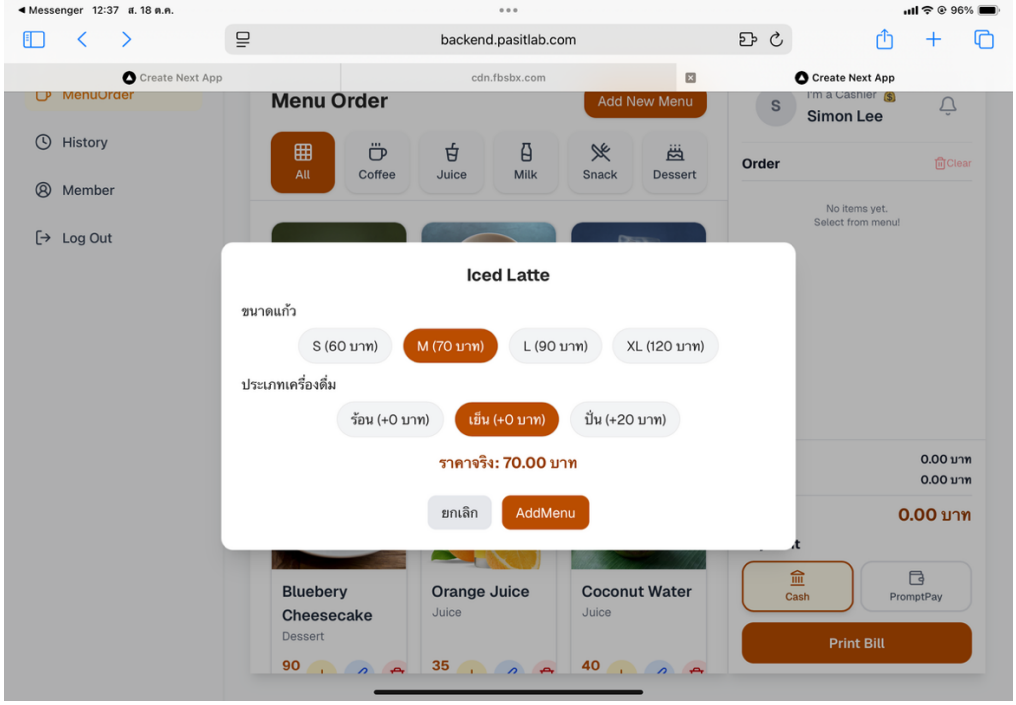
(2) Menu Name :เป็นกล่องข้อความสำหรับกรอกชื่อเมนู เช่น “Matcha Latte” หรือ “Blueberry Cheesecake”

(3) Category : เป็นปุ่มเลือกหมวดของเมนู ได้แก่ Coffee, Juice, Milk, Snack และ Dessert

(4) Prices by Size : เป็นการกำหนดราคาตามขนาดแก้ว ได้แก่ S (เล็ก), M (กลาง), L (ใหญ่), XL (ใหญ่พิเศษ)

(5) Prices by Type : เป็นการกำหนดราคาตามลักษณะของเครื่องดื่ม ได้แก่ ร้อน (Hot), เย็น (Iced) และ ปั่น (Blended)

(6) Description : เป็นกล่องข้อความสำหรับกรอกคำอธิบายรายละเอียดของเมนู โดยในส่วนนี้ จะใส่หรือไม่ใส่ก็ได้



ภาพที่ 20 หน้า Add Menu

จากภาพที่ 20 เป็นหน้าต่างป๊อปอัพ (Popup Window) ในระบบ MenuOrder ซึ่งปรากฏขึ้นเมื่อผู้ใช้เลือกเมนูเครื่องจากหน้าหลัก เพื่อกำหนดขนาดและประเภทของเครื่องดื่มก่อนเพิ่มลงในรายการสั่งซื้อ ในส่วนของหน้านี้ ประกอบด้วย

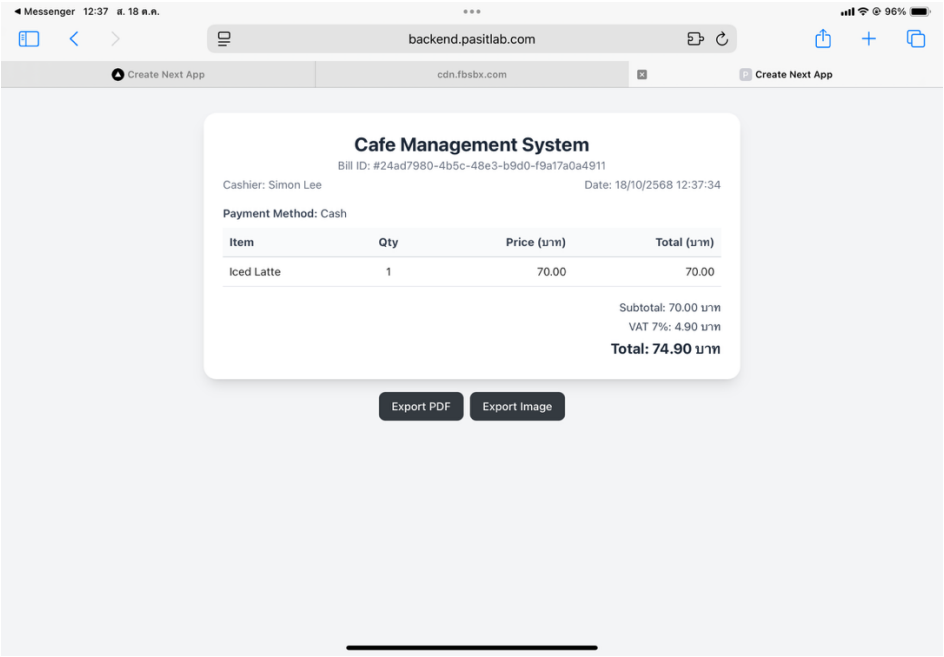
(1) ชื่อเมนู (Menu Title) : แสดงชื่อเมนูที่ผู้ใช้เลือก เช่น Iced Latte

(2) ขนาดแก้ว : ให้ผู้ใช้เลือกขนาดแก้วของเครื่องดื่ม โดยแต่ละขนาดมีราคากำกับไว้ เมื่อเลือกขนาด ระบบจะอัปเดตราคาพื้นฐานตามที่กำหนดในฐานข้อมูล

(3) ประเภทเครื่องดื่ม : ใช้เลือกประเภทของเครื่องดื่ม ได้แก่ ร้อน, เย็น หรือปั่น โดยระบบจะเพิ่มหรือลดราคาตามประเภทที่เลือก เช่น “ปั่น” เพิ่ม 20 บาทจากราคาพื้นฐาน

(4) ราคาจริง : แสดงผลราคารวมสุดท้ายแบบเรียลไทม์หลังจากเลือกขนาดและประเภท

(5) ปุ่มคำสั่งด้านล่าง : ยกเลิก สำหรับปิดหน้าต่างโดยไม่เพิ่มเมนูลงในรายการสั่งซื้อ และ Add Menu สำหรับเพิ่มสินค้าที่เลือกเข้าสู่รายการสั่งซื้อทางด้านขวาของหน้าจอ



ภาพที่ 21 หน้าใบเสร็จการขาย (Receipt Page)

จากภาพที่ 21 เป็นหน้าใบเสร็จการขายที่สรุปรายการสั่งซื้อหลังจากลูกค้าชำระเงินเรียบร้อยแล้ว ในส่วนของหน้านี้ ประกอบด้วย

(1) หัวข้อหลัก (Header Section) : ประกอบด้วย ชื่อระบบ, Bill ID (รหัสประจำใบเสร็จ), Cashier (ชื่อพนักงานที่ทำรายการ), Date/Time (วันที่และเวลาที่ออกบิล)

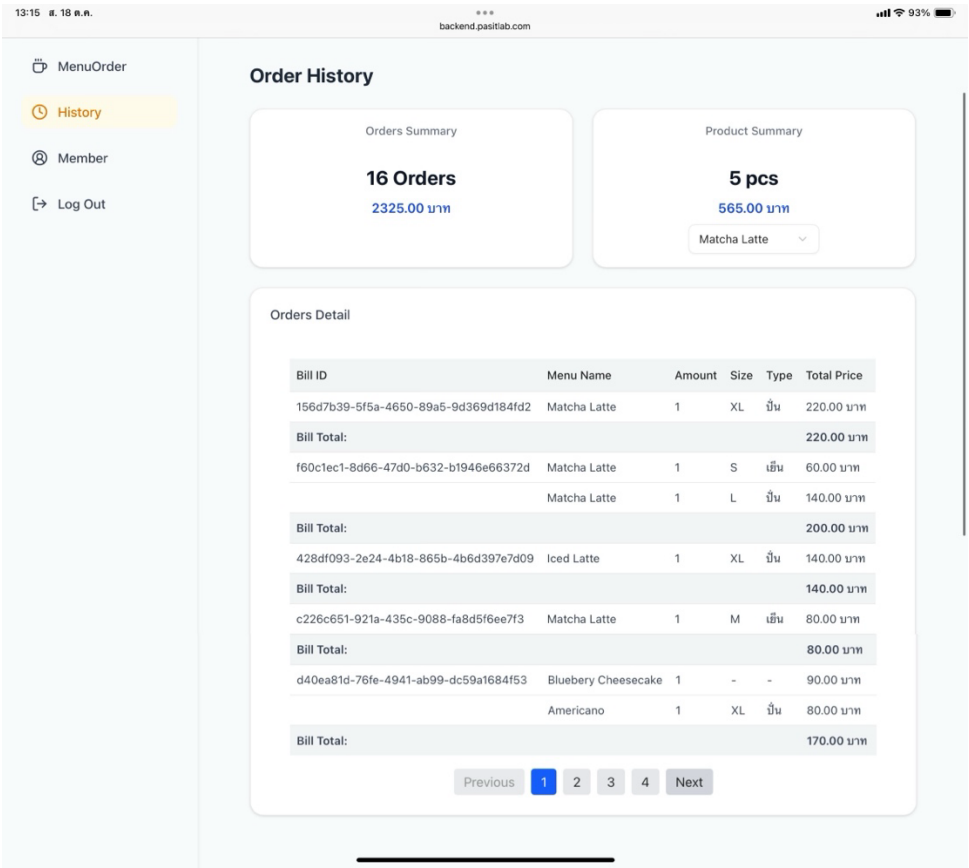
(2) Payment Details : แสดงรายละเอียดการชำระเงิน เช่น Cash (เงินสด)

(3) ตารางแสดงรายการสินค้า : ประกอบด้วย Item (ชื่อสินค้า), Qty (จำนวนสินค้าที่สั่งซื้อ), Price (ราคาของสินค้านั้นๆ) และ Total (ราคารวมสุทธิ)

(4) สรุปราคา : ประกอบด้วย Subtotal (ราคารวมก่อนภาษี), VAT 7% (ภาษีมูลค่าเพิ่ม 7%) และ Total (ราคารวมสุทธิ)

(5) ปุ่มคำสั่งด้านล่าง : Export PDF สำหรับดาวน์โหลดใบเสร็จในรูปแบบไฟล์ PDF และ Export Image สำหรับบันทึกใบเสร็จเป็นรูปภาพ เช่น PNG หรือ JPG

3.1.3 หน้าแสดงประวัติการสั่งซื้อที่ผ่านมา (History)



ภาพที่ 22 หน้า History

จากภาพที่ 22 เป็นหน้าที่แสดงข้อมูลประวัติการสั่งซื้อทั้งหมดที่เกิดขึ้นในร้าน เพื่อให้พนักงานหรือผู้ดูแลสามารถตรวจสอบยอดขายย้อนหลังได้อย่างสะดวก ในส่วนของหน้านี้ ประกอบด้วย

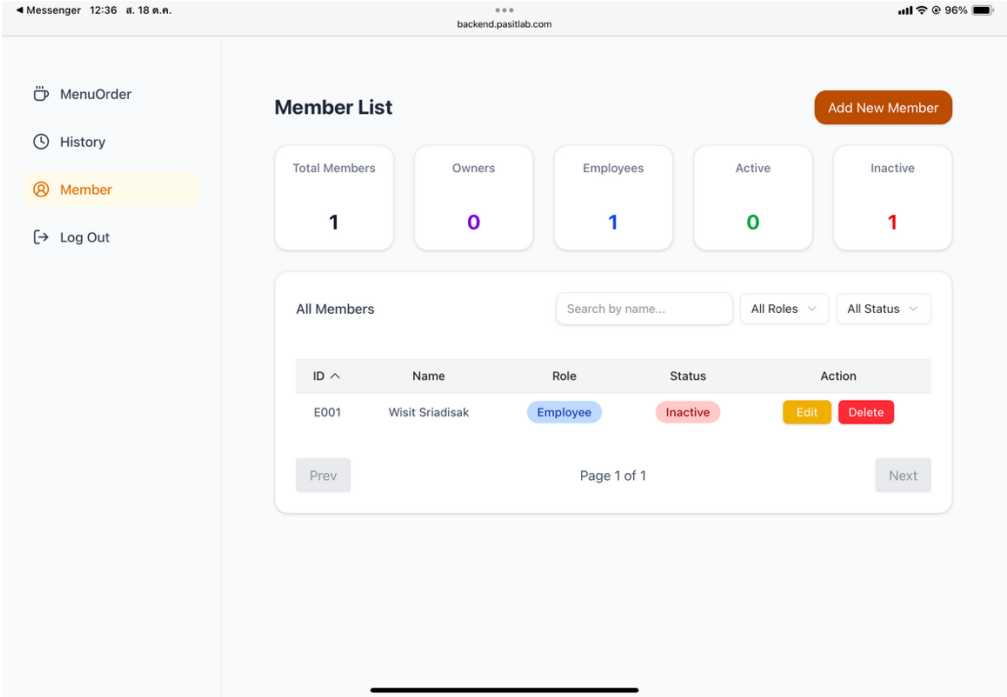
(1) Orders Summary : แสดงจำนวนออเดอร์ทั้งหมด และยอดรวมรายได้ เช่น มีทั้งหมด 16 Orders ยอดรวม 2325.00 บาท

(2) Product Summary : แสดงยอดรวมสินค้าขายดี (Top Product) เช่น จำนวน 5 pcs เมนู Matcha Latte ยอดรวม 565.00 บาท

(3) Orders Detail : ตารางข้อมูลสรุปรายการสั่งซื้อทั้งหมด โดยแต่ละบิลจะมีรายละเอียด ได้แก่ Bill ID, Menu Name, Amount, Size, Type, Total Price โดยด้านล่างของแต่ละบิลมี Bill Total สำหรับแสดงยอดรวมของบิลนั้น

(4) ด้านล่างสุดของตารางมีการแบ่งหน้า (Pagination) เพื่อเลื่อนไปดูบิลถัดไป ได้แก่ ปุ่ม Previous, 1, 2, 3, 4, Next

3.1.4 หน้าจัดการข้อมูลสมาชิก (Member)



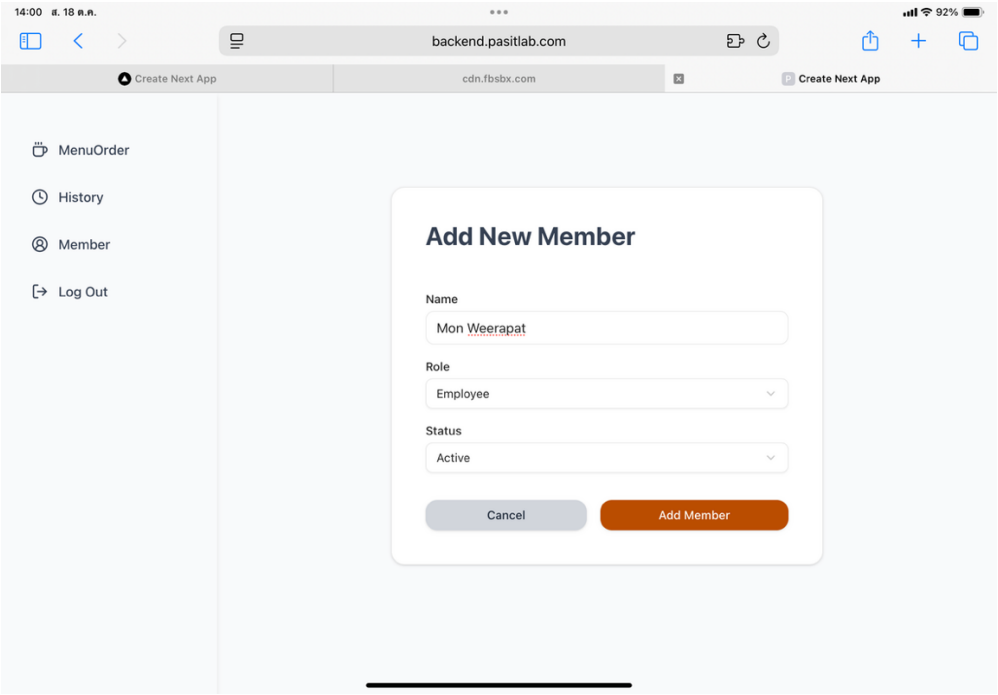
ภาพที่ 23 หน้า Member List

จากภาพที่ 23 เป็นหน้าที่ใช้สำหรับจัดการข้อมูลผู้ใช้งานภายในร้าน ได้แก่ เจ้าของร้าน หรือพนักงาน โดยในส่วนของด้านบนมีช่องค้นหา (Search by name) เพื่อค้นหาชื่อสมาชิกระบบ และตัวกรอง (All Roles, All Status) เพื่อค้นหาสมาชิกตามเงื่อนไข และด้านล่างมีปุ่ม Prev / Next สำหรับเปลี่ยนหน้าหากมีข้อมูลหลายรายการ ในส่วนของหน้านี้ ประกอบด้วย

(1) ปุ่ม Add New Member : สำหรับเพิ่มสมาชิกใหม่เข้าสู่ระบบ

(2) แผงสรุปข้อมูลสมาชิก : แสดงจำนวนสมาชิกในแต่ละหมวดหมู่ ประกอบด้วย Total Members (จำนวนสมาชิกทั้งหมดในระบบ), Owners (เจ้าของร้าน), Employees (พนักงาน), Active (สมาชิกที่ใช้งานอยู่) และ Inactive (สมาชิกที่ไม่ได้ใช้งานแล้ว)

(3) ตารางรายชื่อสมาชิก : แสดงข้อมูลของสมาชิกทั้งหมดในระบบ ประกอบด้วย ID (รหัสสมาชิก), Name (ชื่อพนักงาน), Role (บทบาทในระบบ), Status (สถานะการใช้งาน) และ Action (ปุ่มคำสั่ง ได้แก่ Edit (แก้ไข) และ Delete (ลบข้อมูล))



ภาพที่ 24 หน้า Add New Member

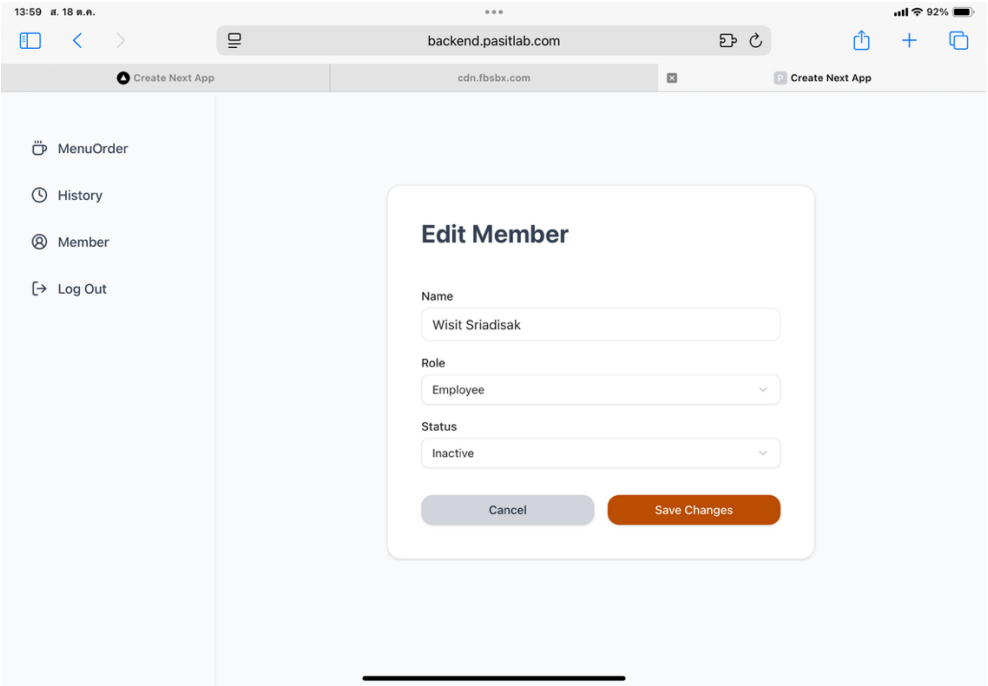
จากภาพที่ 24 เป็นหน้าสำหรับเพิ่มข้อมูลสมาชิกใหม่ (ได้แก่ พนักงาน หรือเจ้าของร้าน) เข้าสู่ระบบบริหารจัดการร้าน ในส่วนของหน้านี้ ประกอบด้วย

(1) Name : ช่องกรอกชื่อสมาชิก

(2) Role : เลือกบทบาทของสมาชิก ได้แก่ Employee (พนักงาน) หรือ Owner (เจ้าของร้าน)

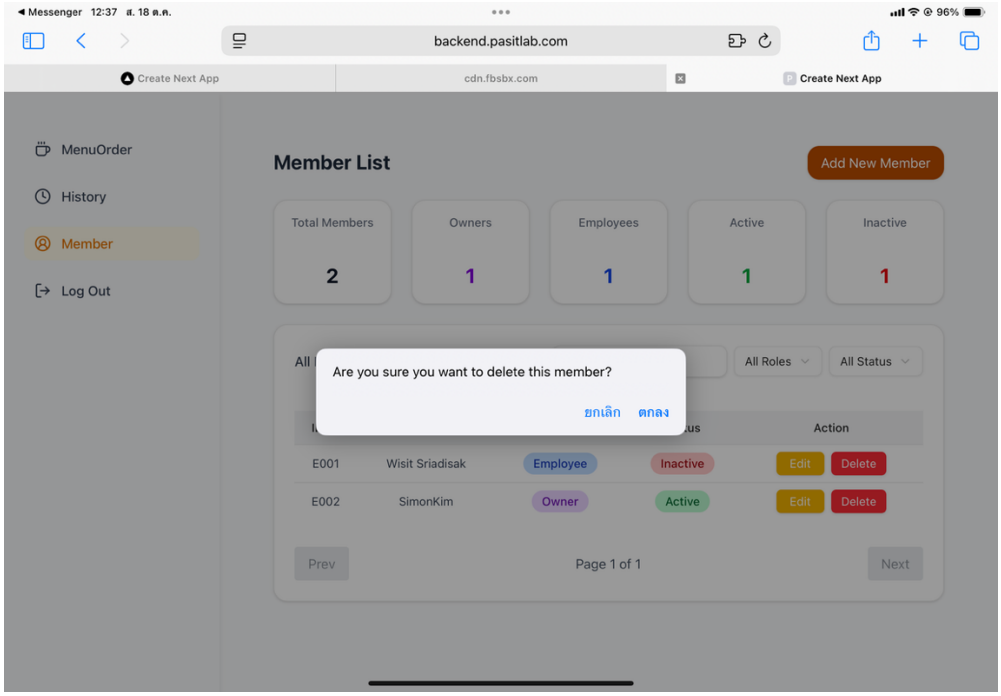
(3) Status : เลือกสถานะการใช้งาน ได้แก่ Active (สมาชิกที่ใช้งานอยู่) หรือ Inactive (สมาชิกที่ไม่ได้ใช้งานแล้ว)

(4) ปุ่มคำสั่งด้านล่าง : Cancel สำหรับยกเลิกการเพิ่มข้อมูลและกลับไปหน้ารายชื่อสมาชิก และ Add Member สำหรับยืนยันการบันทึกข้อมูลสมาชิกใหม่เข้าสู่ระบบ



ภาพที่ 25 หน้า Edit Member

จากภาพที่ 25 เป็นหน้าสำหรับแก้ไขข้อมูลสมาชิกในระบบ เมื่อผู้ใช้กดปุ่ม Edit (ปุ่มสีส้ม) ระบบจะแสดงหน้านี้ เพื่อให้ผู้ใช้แก้ไขชื่อของสมาชิก, แก้ไขบทบาทของสมาชิก และแก้ไขสถานะการใช้งาน และมีปุ่มคำสั่งด้านล่าง ได้แก่ Cancel สำหรับยกเลิกการแก้ไขข้อมูลและกลับไปหน้ารายชื่อสมาชิก และ Save Changes สำหรับยืนยันการบันทึกการแก้ไขข้อมูล



ภาพที่ 26 หน้า Delete Member

จากภาพที่ 26 เป็นหน้าสำหรับลบข้อมูลสมาชิกออกจากระบบ เมื่อผู้ใช้กดปุ่ม Delete (ปุ่มสีแดง) ระบบจะแสดงกล่องข้อความเพื่อยืนยันการลบสมาชิก และมีปุ่มด้านล่างให้เลือก ได้แก่ ยกเลิก (ยกเลิกการลบ) และ ตกลง (ยืนยันการลบข้อมูลสมาชิกออกจากระบบ)

บทที่ 4
สรุปผล

4.1 สรุปผล

การจัดทำการพัฒนาระบบการจัดการร้านคาเฟ่ในรูปแบบเว็บไซต์ พบว่าระบบสามารถนำมาใช้งานได้จริง ได้แก่ ระบบการจัดการคำสั่งซื้อ ระบบสรุปยอดขายประจำวัน ระบบจัดการใบเสร็จ ระบบจัดการข้อมูลพนักงานภายในร้าน การเพิ่มพนักงาน การเพิ่ม/แก้ไขเมนู หลังจากทดลองใช้งานระบบ ส่งผลให้การบริการมีประสิทธิภาพมากขึ้น เนื่องจากลดความผิดพลาดในการรับคำสั่งซื้อของสินค้า นอกจากนี้ยังสามารถจัดเก็บรายการสั่งซื้อของลูกค้าได้เป็นระบบยิ่งขึ้น

โดยการสร้างเว็บไซต์พัฒนาระบบการจัดการร้านคาเฟ่ ได้พัฒนาจาก Java Spring Boot ในการควบคุมระบบ ใช้ Next.js และ Tailwind CSS ในการสร้างและออกแบบการแสดงผลบนเว็บไซต์ เนื้อหาบนเว็บไซต์การพัฒนาระบบการจัดการร้านคาเฟ่จะมีส่วนของเจ้าของร้าน โดยจะต้องเข้าสู่ระบบเพื่อเข้าใช้งานเว็บไซต์

คณะผู้วิจัยได้พัฒนาระบบตามแผนการวิเคราะห์ระบบงานที่วางไว้ ผลที่ได้คือสามารถพัฒนาระบบได้สำเร็จตามวัตถุประสงค์ที่ตั้งไว้ อันเนื่องมาจากความประสบผลสำเร็จทางด้านการวิเคราะห์และออกแบบระบบที่สามารถทำได้เป็นอย่างดี

4.2 ปัญหาและข้อจำกัด

ในการใช้งานในช่วงแรกผู้ใช้งานอาจจะสับสนเนื่องจากไม่มีคู่มือในการใช้งานระบบใหม่ผู้ใช้อาจจะต้องใช้เวลาในการปรับตัวกับระบบใหม่ที่ทำงานบนเว็บไซต์

4.3 ข้อเสนอแนะ

จากการพัฒนาระบบการจัดการร้านคาเฟ่ หากมีการศึกษาและพัฒนาเพิ่มเติมเกี่ยวกับระบบการควบคุมวัตถุดิบ และหากมีการพัฒนาปรับปรุงระบบอย่างต่อเนื่อง คาดว่าจะเป็นประโยชน์ต่อผู้ประกอบการมากขึ้น

เอกสารอ้างอิง

EXPERT-PROGRAMMING-TUTOR. (2556). **Next JS คืออะไร ใช้งานอย่างไร ตัวอย่างการใช้งาน บอกข้อดีข้อเสีย**, วันที่สืบค้น 15 ตุลาคม 2568, จาก https://expert-programming-tutor.com/tutorial/article/KE000865_What_is_Next_JS_How_to_use_Example_of_use_Discuss_the_advantages_and_disadvantages.php

Phanupong permpimol. (2564). **Next.js คืออะไร ช่วยให้เราทำงานได้ง่ายขึ้นยังไงบ้าง.** วันที่สืบค้น 15 ตุลาคม 2568, จาก https://codinggun.com/next-js/

Charintorn Ruanglaikram. (2566). **Tailwind CSS เฟรมเวิร์กที่ช่วยให้ Dev ทำงานง่ายขึ้น.** วันที่สืบค้น 15 ตุลาคม 2568, จาก https://morphos.is/th/blog/tailwind-css-a-framework-that-makes-dev-work-easier

Teerawat Amornrattanakij. (2563). **Spring Boot มีไว้ทำอะไร?.** วันที่สืบค้น 15 ตุลาคม 2568, จาก https://medium.com/@Teerawat.amo/spring-boot-มีไว้ทำอะไร-c1d84a7796d7

Marupat. (2565). **Database : SQLite : ฐานข้อมูลเชิงสัมพันธ์แบบฝังตัวที่มีน้ำหนักเบา.** วันที่สืบค้น 15 ตุลาคม 2568, จาก https://marupatnote.home.blog/2022/09/27/what-is-sqlite/

วิกิพีเดีย. (ม.ป.ป.). (2568). **วิชวลสตูดิโอโค้ด.** วันที่สืบค้น 15 ตุลาคม 2568, จาก https://th.wikipedia.org/wiki/วิชวลสตูดิโอโค้ด

Seven Peaks. (2565). **Postman: เครื่องมือทดสอบ.** วันที่สืบค้น 15 ตุลาคม 2568, จาก https://sevenpeakssoftware.com/th/blog/postman-testing-tools-recommended/

DevHub. (2567). **ใช้งาน Git & GitHub สำหรับผู้เริ่มต้น ในปี 2024.** วันที่สืบค้น 15 ตุลาคม 2568, จาก https://devhub.in.th/blog/git-github

GitHub. (2568). **เริ่มต้นใช้งาน GitHub Desktop.** วันที่สืบค้น 16 ตุลาคม 2568, จาก https://docs.github.com/en/desktop/overview/getting-started-with-github-desktop

PADVEE Web School. (2565). **Cloudflare คืออะไร มีประโยชน์อะไรบ้าง แนะนำวิธีใช้คลาวแฟร์ อย่างถูกต้อง.** วันที่สืบค้น 16 ตุลาคม 2568, จาก https://padveewebschool.com/cloudflare-is/