ระบบจองและจัดการร้านเมืองเลยไดร์ฟกอล์ฟ

Muang Loei Drive Golf Management System

ณฐนนท์ คงสุข
นฤสรณ์ มนตรี

โครงงานฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตร์บัณฑิต
สาขาวิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์และเทคโนโลยี
มหาวิทยาลัยราชภัฏเลย ปีการศึกษา 2569

ลิขสิทธิ์เป็นของมหาวิทยาลัยราชภัฏเลย

ระบบจองและจัดการร้านเมืองเลยไดร์ฟกอล์ฟ
Muang Loei Drive Golf Management System

ณฐนนท์ คงสุข
นฤสรณ์ มนตรี

โครงงานฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตร์บัณฑิต
สาขาวิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์และเทคโนโลยี
มหาวิทยาลัยราชภัฏเลย ปีการศึกษา 2569
ลิขสิทธิ์เป็นของมหาวิทยาลัยราชภัฏเลย

หัวข้อโครงงาน : ระบบจองและจัดการร้านเมืองเลยไดร์ฟกอล์ฟ

: Muang Loei Drive Golf Management System

ผู้จัดทำโครงงาน : นายณฐนนท์ คงสุข รหัส 6640248105 หมู่เรียน ว.6604

นายนฤสรณ์ มนตรี รหัส 6640248112 หมู่เรียน ว.6604

สาขาวิชา : วิทยาการคอมพิวเตอร์

อาจารย์ที่ปรึกษา : อาจารย์ตันติกร โนนศรี

สถาบันการศึกษา : มหาวิทยาลัยราชภัฏเลย

ปีการศึกษา : 2569

คณะกรรมการสอบโครงงาน

ลงชื่อ..... ประธานกรรมการ

(ดร.นรุวรรณ อยู่สำราญ)

ลงชื่อ..... กรรมการ

(อาจารย์ตันติกร โนนศรี)

ลงชื่อ..... กรรมการ

(ผู้ช่วยศาสตราจารย์ทอแสง พิมพ์เบ้าธรรม)

คณะกรรมการสอบรายวิชาโครงงานทางวิทยาการคอมพิวเตอร์ ได้ตรวจสอบและรับรองว่าโครงงานนี้เป็นไปตามมาตรฐานของหลักสูตรวิทยาศาสตรบัณฑิต สาขาวิชาวิทยากาคอมพิวเตอร์ คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยราชภัฏเลย

ลงชื่อ.....

(อาจารย์ดุลชาติ ศิริวัลลภ)

ประธานคณะกรรมการผู้รับผิดชอบหลักสูตรวิทยาศาสตรบัณฑิต

สาขาวิชาวิทยาการคอมพิวเตอร์

วันที่.....เดือน.....พ.ศ. .....

กิตติกรรมประกาศ

การศึกษาโครงงานทางวิทยาการคอมพิวเตอร์เรื่องระบบจองและจัดการร้านเมืองเลยไดร์ฟกอล์ฟครั้งนี้สำเร็จได้ด้วยดี โดยผู้จัดทำได้รับความอนุเคราะห์อย่างสูงจากท่านอาจารย์ที่ปรึกษาโครงงาน อาจารย์ตันติกร โนนศรี อาจารย์ประจำรายวิชาโครงงานทางวิทยาการคอมพิวเตอร์ และคณาจารย์ทุกท่านที่ให้ข้อเสนอแนะ เพิ่มเติมทำให้โครงงานฉบับนี้ให้มีความถูกต้องสมบูรณ์ยิ่งขึ้น คณะผู้จัดทำจึงขอกราบขอบพระคุณอย่าง สูงไว้ ณ โอกาสนี้

ขอขอบคุณผู้บริหาร เจ้าหน้าที่ และพนักงานของร้านเมืองเลยไดร์ฟกอล์ฟ ที่ให้ความร่วมมือในการให้ข้อมูลและแนะนำที่เป็นประโยชน์ต่อการทำโครงงานในครั้งนี้ และให้ความช่วยเหลือในการติดต่อประสานงาน ซึ่งช่วยให้คณะผู้จัดทำสามารถนำข้อมูลมาวิเคราะห์ออกแบบ และพัฒนาระบบได้อย่างมีประสิทธิภาพ

ขอกราบขอบพระคุณบิดา มารดาผู้ที่อยู่เบื้องหลังและให้กำลังใจแก่คณะผู้จัดทำด้วยดีเสมอมา คุณค่าและประโยชน์อันเกิดจากโครงงานในครั้งนี้คณะผู้จัดทำขอมอบบูชาพระคุณบิดามารดา ตลอดจนผู้มีพระคุณทุกท่านที่มีส่วนร่วมในการช่วยเหลือด้วยความรักและเมตตาในการศึกษาค้นคว้า ด้วยดีตลอดมา

ณฐนนท์ คงสุข
นฤสรณ์ มนตรี

หัวข้อโครงงาน : ระบบจองและจัดการร้านเมืองเลยไดร์ฟกอล์ฟ
: Muang Loei Drive Golf Management System
ผู้จัดทำโครงงาน : นายณฐนนท์ คงสุข รหัส 6640248105 หมู่เรียน ว.6604
นายนฤสรณ์ มนตรี รหัส 6640248112 หมู่เรียน ว.6604
สาขาวิชา : วิทยาการคอมพิวเตอร์
อาจารย์ที่ปรึกษา : อาจารย์ตันติกร โนนศรี
สถาบันการศึกษา : มหาวิทยาลัยราชภัฏเลย
ปีการศึกษา : 2569

บทคัดย่อ

โครงงานนี้มีวัตถุประสงค์เพื่อพัฒนา ระบบจองและจัดการร้านเมืองเลยไดร์ฟกอล์ฟ เพื่อปรับปรุงกระบวนการให้บริการจากรูปแบบเดิมที่ใช้การจองผ่านโทรศัพท์หรือการบันทึกด้วยกระดาษ ซึ่งมักเกิดปัญหาข้อมูลสูญหาย การจองซ้ำซ้อน และความไม่สะดวกในการตรวจสอบสถานะการใช้งานอุปกรณ์ ระบบที่พัฒนาขึ้นช่วยให้ลูกค้าสามารถจองวันและเวลา ระบุจำนวนผู้ใช้บริการ ความต้องการยืมไม้กอล์ฟ ระดับพื้นฐานการเล่น และชื่อผู้จองหลักได้อย่างสะดวกผ่านเว็บไซต์ ระบบยังช่วยให้พนักงานสามารถตรวจสอบการจอง ยืนยันลูกค้าที่มาใช้บริการจริง และบันทึกการใช้งานอุปกรณ์ได้อย่างเป็นระบบ ขณะเดียวกันผู้ดูแลสามารถจัดการข้อมูลลูกค้า ตารางเวลา และรายงานสถิติการใช้งานได้ในระบบเดียว รวมถึงสามารถติดตามสถานะและประวัติการบำรุงรักษาอุปกรณ์ได้อย่างแม่นยำ การพัฒนาระบบใช้โมเดลการพัฒนาแบบน้ำตก (Waterfall Model) โดยออกแบบด้วยแผนภาพบริบท (Context Diagram) แผนภาพการไหลของข้อมูล (DFD) และแบบจำลองความสัมพันธ์ของข้อมูล (ER Diagram) ระบบต้นแบบพัฒนาด้วยเทคโนโลยีเว็บสมัยใหม่ โดยใช้ React สำหรับส่วนติดต่อผู้ใช้ (Frontend) และ Node.js + Express สำหรับส่วนเซิร์ฟเวอร์ (Backend) เชื่อมต่อ ฐานข้อมูล Firebase Firestore พร้อมใช้ Firebase Authentication สำหรับยืนยันตัวตนผู้ใช้ ใช้ Firebase Hosting สำหรับเผยแพร่ระบบบนเครือข่ายอินเทอร์เน็ต และใช้ Firebase CLI เพื่อสั่งงานและจัดการส่วนต่างๆได้อย่างรวดเร็ว ส่วนการออกแบบส่วนติดต่อผู้ใช้ใช้ Figma และการแสดงผลสถิติข้อมูลใช้ Chart.js ในการสร้างกราฟและแผนภูมิประกอบรายงาน ผลลัพธ์คาดว่าจะช่วยเพิ่มความสะดวกและความถูกต้องในการให้บริการ ลดข้อผิดพลาดในการจัดการข้อมูล และยกระดับประสิทธิภาพการบริหารของร้านเมืองเลยไดร์ฟกอล์ฟให้มีความทันสมัยและมีมาตรฐานมากยิ่ง

สารบัญ

เรื่อง	หน้า
หน้าอนุมัติ	ก
กิตติกรรมประกาศ	ข
บทคัดย่อ	ค
สารบัญ	ง
สารบัญภาพประกอบ	ฉ
สารบัญตาราง	ซ
บทที่ 1 บทนำ	1
1.1 หลักการและเหตุผล	1
1.2 วัตถุประสงค์	1
1.3 ขอบเขตของโครงการ	2
1.4 อุปกรณ์เครื่องมือที่ใช้ในการพัฒนา	5
1.5 ระยะเวลาและแผนดำเนินงาน	6
1.6 ประโยชน์ที่คาดว่าจะได้รับ	6
1.7 คำนิยามศัพท์เฉพาะ	7
บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง	8
2.1 ร้านเมืองเลยไดร์ฟกอล์ฟ	8
2.2 แบบจำลองน้ำตก	10
2.3 ระบบฐานข้อมูลและเทคโนโลยี NoSQL	11
2.4 Firebase Framework	13
2.5 Responsive Web Design	15
2.6 ภาษา CSS	16
2.7 Tailwind Framework	17
2.8 Figma	18
2.9 ภาษา JavaScript	19
2.10 Node.js Framework	20
2.11 Express.js Framework	21
2.12 React.js Framework	22
2.13 Chart.js Library	23

สารบัญ(ต่อ)

เรื่อง	หน้า
2.14 โปรแกรม Visual Studio Code	23
2.15 งานวิจัยที่เกี่ยวข้อง	24
บทที่ 3 วิธีการออกแบบระบบและดำเนินการศึกษา	26
3.1 ความเป็นมาของระบบเดิม	26
3.2 กระบวนการวิเคราะห์ระบบงานใหม่	29
3.3 การออกแบบหน้ายูสเซอร์อินเตอร์เฟซ (user interface design)	48
บรรณานุกรม	64
ประวัติคณะผู้จัดทำโครงงานวิทยาการคอมพิวเตอร์	67

สารบัญภาพประกอบ

ภาพประกอบที่	หน้า
2-1 ร้านเมืองเลยไดร์ฟกอล์ฟ	9
2-2 โมเดลการพัฒนาแบบน้ำตก (Waterfall Model)	11
2-3 ฐานข้อมูล NoSQL: เสริมศักยภาพการจัดการข้อมูลยุคใหม่	12
2-4 ประโยชน์ของการใช้ Firebase ในการพัฒนาเว็บไซต์และแอปพลิเคชัน	14
2-5 ภาพแสดงหน้าจอแสดงผลแบบ Responsive	16
2-6 โลโก CSS	17
2-7 ตัวอย่างการกำหนดรูปแบบของ Tailwind	18
2-8 เว็บไซต์ Figma	19
2-9 ตัวอย่างภาษา JavaScript	20
2-10 โลโก Node.js	21
2-11 โลโก Express.js	22
2-12 ตัวอย่างกราฟ Chart.js	23
2-13 การพัฒนาด้วย VS Code	24
3-1 Flowchart ระบบงานเดิม	27
3-2 Flowchart กระบวนการชำระเงิน	28
3-3 Context Diagram	29
3-4 Data Flow Diagram Level 0 ระบบจองและจัดการร้านเมืองเลยไดร์ฟกอล์ฟ	30
3-7 Data Flow Diagram Level 1 : Process 1 จัดการข้อมูลผู้ใช้งาน	33
3-9 Data Flow Diagram Level 1 : Process 2 จัดการไม้กอล์ฟ	35
3-10 Data Flow Diagram Level 1 : Process 3 ตั้งค่าค่าบริการและแต้ม	35
3-12 Data Flow Diagram Level 1 : Process 4 จัดการวันปิดร้าน	36
3-13 Data Flow Diagram Level 1 : Process 5 สถานะเลนซ้อม	37
3-14 Data Flow Diagram Level 1 : Process 6 การจัดการและยืนยันการจอง	37
3-16 Data Flow Diagram Level 1 : Process 7 การจัดการชำระเงิน	39
3-18 Data Flow Diagram Level 1 : Process 8 คะแนนและความคิดเห็น	40
3-19 Data Flow Diagram Level 1 : Process 9 ประวัติการจอง	41
3-20 Data Flow Diagram Level 1 : Process 10 รายงาน	42
3-21 แสดงภาพ Entity Relationship Diagram ระบบจองและจัดการร้านเมืองเลยไดร์ฟกอล์ฟ	43

สารบัญภาพประกอบ(ต่อ)

ภาพประกอบที่	หน้า
3-22 หน้าสมัครสมาชิกสำหรับลูกค้า	48
3-23 หน้าเข้าสู่ระบบผู้ใช้งาน	49
3-24 หน้ากู้คืนรหัสผ่านผู้ใช้งาน	49
3-25 หน้าตั้งค่ารหัสผ่านใหม่	50
3-26 หน้าแดชบอร์ดเจ้าของร้านและพนักงาน	51
3-27 หน้าจัดการไม้กอล์ฟ	51
3-28 หน้าจัดการผู้ใช้งาน	52
3-29 หน้าตั้งค่าราคาค่าบริการและตั้งค่าแต้มสะสม	53
3-30 หน้าตั้งวันปิดร้านล่วงหน้า	53
3-31 หน้าแก้ขสถานะเลนซ้อมและจัดการข้อมูลการจอง	54
3-32 หน้ารายการรายได้	55
3-33 หน้าชำระเงิน	55
3-34 หน้าใบเสร็จ	56
3-35 หน้าคะแนนและความคิดเห็น	57
3-36 หน้าประวัติการจอง	58
3-37 หน้าแดชบอร์ดลูกค้า	59
3-38 หน้าเพิ่มการจอง	60
3-39 หน้ารายละเอียดการจอง	61
3-40 หน้าเลือกไม้กอล์ฟ	62
3-41 หน้ารายการจองของลูกค้า	63
3-42 หน้าการให้คะแนนและความคิดเห็น	63

สารบัญตาราง

ตารางที่	หน้า
1-1 ระยะเวลาและแผนการดำเนินงาน	6
3-1 Users (ข้อมูลผู้ใช้งาน)	44
3-2 Golf Clubs (ข้อมูลไม้กอล์ฟ)	44
3-3 Service Settings (ข้อมูลการตั้งค่าค่าบริการ)	45
3-4 Point Settings (ข้อมูลการตั้งค่าแต้มสะสม)	45
3-5 Shop Closures (ข้อมูลวันปิดร้าน)	46
3-6 Lanes (ข้อมูลเลนซ้อม)	46
3-7 Bookings (ข้อมูลการจอง)	46
3-8 Payments (ข้อมูลการชำระเงิน)	47
3-9 Reviews (ข้อมูลคะแนนและความคิดเห็น)	47

บทที่ 1
บทนำ

1.1 หลักการและเหตุผล

ปัจจุบันเทคโนโลยีสารสนเทศมีบทบาทสำคัญในการเพิ่มประสิทธิภาพและลดข้อผิดพลาดในการดำเนินงานทางธุรกิจ ร้านเมืองเลยไดร์ฟกอล์ฟ ซึ่งเป็นสถานที่ให้บริการฝึกซ้อมกอล์ฟ ประสบปัญหาจากการบริหารจัดการรูปแบบเดิมที่ใช้วิธีรับจองผ่านโทรศัพท์และจดบันทึกลงสมุด ส่งผลให้เกิดความผิดพลาดในการปฏิบัติงาน เช่น การจองเวลาซ้ำซ้อน การจัดเตรียมอุปกรณ์ไม่เพียงพอ และการสูญหายของข้อมูล ผู้ศึกษาจึงเล็งเห็นความจำเป็นในการนำเทคโนโลยีเว็บแอปพลิเคชันเข้ามาช่วยบริหารจัดการเพื่อยกระดับคุณภาพการให้บริการ

ในการพัฒนาระบบ ผู้ศึกษาเลือกใช้เทคโนโลยีที่ทันสมัยและมีประสิทธิภาพ โดยส่วนแสดงผล (Frontend) พัฒนาด้วย React.js ร่วมกับ Tailwind CSS เพื่อให้ได้ส่วนติดต่อผู้ใช้ที่สวยงามและตอบสนองไว พร้อมใช้ Chart.js สำหรับแสดงผลกราฟสถิติ ในส่วนการประมวลผล (Backend) เลือกใช้ Node.js และ Express.js เชื่อมต่อกับฐานข้อมูล Firebase Firestore ซึ่งเป็นฐานข้อมูลแบบ NoSQL ที่รองรับการทำงานแบบ Real-time และใช้ Firebase Authentication ในการจัดการระบบยืนยันตัวตนที่มีความปลอดภัยสูง

ระบบที่พัฒนาขึ้นจะช่วยอำนวยความสะดวกให้แก่ผู้ใช้งานทุกฝ่าย ได้แก่ ลูกค้า สามารถทำรายการจอง ระบุจำนวนผู้เล่น และความต้องการยืมอุปกรณ์ได้ด้วยตนเอง พนักงาน สามารถตรวจสอบตารางการจองและยืนยันลูกค้าได้อย่างถูกต้อง รวดเร็ว ส่วน ผู้ดูแลระบบ สามารถบริหารจัดการข้อมูลร้าน ดูรายงานสถิติรายได้ภายในระบบเดียว ช่วยลดภาระงานซ้ำซ้อนและเพิ่มความน่าเชื่อถือให้กับทางร้านอย่างยั่งยืน

1.2 วัตถุประสงค์

1.2.1 เพื่อศึกษาและออกแบบระบบจองและจัดการร้านเมืองเลยไดร์ฟกอล์ฟ

1.2.2 เพื่อพัฒนาระบบจองและจัดการร้านเมืองเลยไดร์ฟกอล์ฟ

1.2.3 เพื่อประเมินความพึงพอใจของผู้ใช้งานต่อระบบจองและจัดการร้านเมืองเลยไดร์ฟกอล์ฟ

1.3 ขอบเขตของโครงการ

1.3.1 **ผู้ดูแลระบบ (เจ้าของร้าน)**

- การเข้าสู่ระบบ

1.3.1.1 สามารถเข้าสู่ระบบได้

1.3.1.2 สามารถกดลืมรหัสผ่านและรีเซ็ตได้ โดยระบบจะส่งลิงก์ผ่านทางอีเมล

- ตั้งค่าระบบร้าน

1.3.1.3 สามารถเพิ่ม แก้ไข และยกเลิกข้อมูลพนักงานได้

1.3.1.4 สามารถแก้ไข ลบ และยกเลิกข้อมูลลูกค้าได้

1.3.1.5 สามารถเพิ่ม แก้ไข และยกเลิกข้อมูลไม้กอล์ฟได้

1.3.1.6 สามารถเพิ่ม แก้ไข และยกเลิกค่าบริการภายในระบบได้

1.3.1.7 สามารถแก้ไข และยกเลิกเกณฑ์การแลกแต้มสะสมได้

1.3.1.8 สามารถแก้ไขสถานะเลนซ้อมได้

1.3.1.9 สามารถตั้งวันที่ปิดร้านล่วงหน้าได้

- การจอง

1.3.1.10 สามารถเพิ่ม แก้ไข และลบข้อมูลการจองได้

1.3.1.11 สามารถตรวจสอบสถานะเลนซ้อมและจำนวนการจองได้

1.3.1.12 สามารถตรวจสอบรายการจองและรายละเอียดการจองผ่านระบบได้ ได้แก่ วันเวลาที่จอง หมายเลขเลนซ้อมที่ใช้บริการ ข้อมูลลูกค้า ชื่อ เบอร์โทร จำนวนผู้เข้าใช้บริการ ความต้องการเช่าไม้ และความต้องการผู้สอน

1.3.1.13 สามารถทำการยืนยันการจองให้ลูกค้าได้ โดยระบบจะเปลี่ยนสถานะเลนซ้อมเป็น “กำลังใช้งาน”

1.3.1.14 สามารถตรวจสอบประวัติการจองทั้งหมดได้

- การเงินและรายงาน

1.3.1.15 สามารถบันทึกหรือยกเลิกรายการรายได้ได้

1.3.1.16 สามารถรับชำระค่าบริการได้ด้วยเงินสด เงินโอน หรือรวมทั้งสองได้

1.3.1.17 สามารถดูจำนวนการจอง จำนวนสถานะเลนซ้อมที่กำลังใช้งานและรายได้ในระบบได้

- คะแนนและความคิดเห็น

1.3.1.18 สามารถตรวจสอบคะแนนและความคิดเห็นได้

1.3.1.19 สามารถลบคะแนนและความคิดเห็นได้ เพื่อใช้วิเคราะห์คุณภาพบริการและกรองความคิดเห็นสแปม

1.3.2 **พนักงาน**

- การเข้าสู่ระบบ

1.3.2.1 สามารถเข้าสู่ระบบได้

1.3.1.2 สามารถกดลืมรหัสผ่านและรีเซ็ตได้ โดยระบบจะส่งลิงก์ผ่านทางอีเมล

- การจอง

1.3.2.3 สามารถเพิ่ม ลบ และแก้ไขข้อมูลการจองได้

1.3.2.4 สามารถตรวจสอบสถานะเลนซ้อมและจำนวนการจองได้

1.3.2.5 สามารถตรวจสอบรายการจองและรายละเอียดการจองผ่านระบบได้ ได้แก่ วันเวลาที่จอง หมายเลขเลนซ้อมที่ใช้บริการ ข้อมูลลูกค้า ชื่อ เบอร์โทร จำนวนผู้เข้าใช้บริการ ความต้องการเช่าไม้ และความต้องการผู้สอน

1.3.2.6 สามารถทำการยืนยันให้ลูกค้าได้ โดยระบบจะเปลี่ยนสถานะเลนซ้อมเป็น “กำลังใช้งาน”

- การชำระเงินและรายงาน

1.3.2.7 สามารถบันทึกหรือยกเลิกรายการรายได้ได้

1.3.2.8 สามารถรับชำระค่าบริการได้ด้วยเงินสด เงินโอน หรือรวมทั้งสองได้

1.3.2.9 สามารถดูจำนวนการจอง จำนวนสถานะเลนซ้อมที่กำลังใช้งานและรายได้ในระบบได้

- คะแนนและความคิดเห็น

1.3.2.10 สามารถตรวจสอบคะแนนและความคิดเห็นได้

1.3.3 **ลูกค้า**

- บัญชีผู้ใช้งาน

1.3.3.1 สามารถสมัครสมาชิกและเข้าสู่ระบบได้ โดยใช้การสมัครผ่านอีเมล หรือผ่านบัญชี Google/Facebook

1.3.3.2 สามารถกดลืมรหัสผ่านและรีเซ็ตได้ โดยระบบจะส่งลิงก์ผ่านทางอีเมล

- การจอง

1.3.3.3 สามารถทำการจองเลนซ้อมได้ โดยเลือกวันที่และเวลา โดยระบบจะแสดงสถานะเลนซ้อมว่า "ว่าง" "กำลังใช้งาน" "มีการจองแล้ว" หรือ"ปิดปรับปรุง"

1.3.3.4 สามารถระบุรายละเอียดการจอง

1.3.3.5 สามารถยกเลิกหรือแก้ไขการจองได้ ภายในเวลาที่ร้านกำหนด

- การชำระเงินและรายงาน

1.3.3.6 สามารถเลือกช่องทางการชำระเงินที่สะดวกได้ และเลือกใช้คะแนนสะสมแลกเป็นส่วนลดค่าบริการได้

1.3.3.7 สามารถรับใบเสร็จเพื่อตรวจสอบข้อมูลรายละเอียดการจอง

1.3.3.8 สามารถตรวจสอบแต้มสะสมปัจจุบันของตนเอง

1.3.3.9 สามารถตรวจสอบประวัติการจองย้อนหลังของตนเองได้

- คะแนนและความคิดเห็น

1.3.3.10 สามารถให้คะแนนและความคิดเห็นได้เฉพาะ "การใช้บริการที่เสร็จสิ้นแล้ว" โดยระบบจะผูกกับประวัติการใช้บริการครั้งนั้น

1.3.3.11 สามารถแก้ไขคะแนนและความคิดเห็นได้เฉพาะ "การใช้บริการที่เสร็จสิ้นแล้ว" โดยระบบจะผูกความคิดเห็นกับประวัติการใช้บริการครั้งนั้น

1.4 อุปกรณ์เครื่องมือที่ใช้ในการพัฒนา

1.4.1 ฮาร์ดแวร์

1.4.1.1 คอมพิวเตอร์โน๊ตบุ๊ค Acer AMD Ryzen 5 5600H with Radeon Graphics 3.30 GHz RAM 16.0 GB

1.4.2 ซอฟต์แวร์

1.4.2.1 โปรแกรม Visual Studio Code

1.4.2.2 Figma

1.4.3 ภาษาคอมพิวเตอร์ที่ใช้

1.4.3.1 ภาษา JavaScript

1.4.3.2 ภาษา CSS

1.4.4 เฟรมเวิร์ก

1.4.4.1 Tailwind CSS

1.4.4.2 Node.js

1.4.4.3 Express

1.4.4.4 React.js

1.4.4.5 Firebase Firestore

1.4.4.6 Firebase Authentication

1.4.4.7 Firebase CLI

1.4.4.8 Firebase Hosting

1.4.5 ไลบรารี

1.4.5.1 Chart.js

1.5 ระยะเวลาและแผนดำเนินงาน

ตารางที่ 1-1 ระยะเวลาและแผนการดำเนินงาน

ขั้นตอนการดำเนินงาน	ระยะเวลาการดำเนินการ											
	พ.ศ.2568		พ.ศ. 2569									
	พ.ย.	ธ.ค.	ม.ค.	ก.พ.	มี.ค.	เม.ย	พ.ค.	มิ.ย.	ก.ค.	ส.ค.	ก.ย.	ต.ค.
1. เสนอหัวข้อโครงงาน												
2. ค้นหาและศึกษาเอกสารงานวิจัยที่เกี่ยวข้อง												
3. ค้นหาและศึกษาบทความที่เกี่ยวข้อง												
4. วิเคราะห์ระบบ												
5. ออกแบบระบบ												
6. พัฒนาระบบ												
7. ทดสอบระบบ												
8. แก้ไขปรับปรุงระบบ												
9. สอบจบโครงงานทางวิทยาการคอมพิวเตอร์												
10. เขียนบทความ												

1.6 ประโยชน์ที่คาดว่าจะได้รับ

1.6.1 **ผู้ดูแลระบบ (เจ้าของร้าน)**

1.6.1.1 สามารถตรวจสอบข้อมูลการจองทั้งหมดได้

1.6.1.2 สามารถแก้ไขราคาค่าบริการและเกณฑ์แต้มสะสมได้

1.6.2 **พนักงานบริการ**

1.6.2.1 ช่วยลดขั้นตอนการจดบันทึกด้วยมือและป้องกันข้อผิดพลาดในการจัดการอุปกรณ์

1.6.3 **ลูกค้า**

1.6.3.1 สามารถได้รับแต้มสะสมจากการชำระค่าบริการและนำแต้มไปแลกรับส่วนลดราคาค่าบริการผ่านระบบได้

1.7 คำนิยามศัพท์เฉพาะ

1.7.1 ระบบจองและจัดการร้านเมืองเลยไดร์ฟกอล์ฟ หมายถึง ระบบสารสนเทศที่พัฒนาขึ้นมาเพื่อใช้บริหารจัดการร้านเมืองเลยไดร์ฟกอล์ฟ โดยผู้ใช้งานที่เป็นผู้ดูแลระบบและพนักงานร้านสามารถตรวจสอบตารางการจองของลูกค้าที่เป็นสมาชิกได้อย่างสะดวกรวดเร็ว และลดข้อผิดพลาดในการจองและจัดการร้าน ส่วนลูกค้าที่เป็นสมาชิกสามารถตรวจสอบตารางเวลาของเลนซ้อมได้ และทำการจองผ่านเว็บไซต์ได้อย่างสะดวก รวดเร็ว และเป็นระบบ

1.7.2 เลนซ้อม หมายถึง พื้นที่หรือช่องซ้อมสำหรับการฝึกตีลูกกอล์ฟ ซึ่งแต่ละเลนซ้อมจะมีหมายเลขกำกับเพื่อใช้ในการจองและจัดตารางการใช้งาน โดยระบบจะบันทึกสถานะของแต่ละเลนซ้อมว่า “ว่าง”, “กำลังใช้งาน”,”ปิดใช้งาน” เพื่อป้องกันการจองซ้ำและช่วยให้บริหารจัดการพื้นที่ได้อย่างเหมาะสม

1.7.3 เปิดเลนซ้อม หมายถึง ขั้นตอนที่พนักงานบริการทำการสร้างรายการใช้บริการในระบบให้กับลูกค้าที่มาใช้บริการ ณ หน้าร้าน กรณีที่ลูกค้าท่านนั้นไม่ได้จองเข้ามาในระบบ โดยพนักงานจะต้องระบุข้อมูลที่จำเป็นเข้าสู่ระบบ เช่น เลือกเลนซ้อม วันที่และเวลาเริ่มใช้บริการ จำนวนผู้เข้าใช้บริการ ความต้องการยืมไม้กอล์ฟ ประเภทของไม้ที่ยืม เมื่อทำการเปิดเลนซ้อมเรียบร้อยแล้ว ระบบจะทำการเปลี่ยนสถานะเลนซ้อมเป็น “กำลังใช้งาน” โดยอัตโนมัติ เพื่อป้องกันการจองซ้ำในช่วงเวลาดังกล่าว

บทที่ 2
ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

ในการพัฒนาระบบจองและจัดการร้านเมืองเลยไดร์ฟกอล์ฟ ผู้พัฒนาได้ดำเนินการศึกษาทฤษฎีและงานวิจัยที่เกี่ยวข้องดังนี้

2.1 ร้านเมืองเลยไดร์ฟกอล์ฟ

2.2 แบบจำลองน้ำตก

2.3 ระบบฐานข้อมูลและเทคโนโลยี NoSQL

2.4 Firebase Framework

2.5 Responsive web Design

2.6 ภาษา CSS

2.7 Tailwind CSS Framework

2.8 Figma

2.9 ภาษา JavaScript

2.10 Node.js Framework

2.11 Express.js Framework

2.12 React.js Framework

2.13 Chart.js Library

2.14 โปรแกรม Visual Studio Code

2.15 งานวิจัยที่เกี่ยวข้อง

2.1 ร้านเมืองเลยไดร์ฟกอล์ฟ

ร้านเมืองเลยไดร์ฟกอล์ฟเป็นธุรกิจให้บริการสถานที่ฝึกซ้อมกีฬากอล์ฟ ตั้งอยู่ในจังหวัดเลย สถานที่ตั้งร้านเมืองเลยไดร์ฟกอล์ฟตั้งอยู่ที่ บ.ฟากนา, 180, ตำบล นาอาน อำเภอเมืองเลย จังหวัดเลย เปิดให้บริการแก่ประชาชนทั่วไป นักกีฬากอล์ฟ และผู้ที่สนใจฝึกซ้อมพื้นฐานการเล่นกอล์ฟ ร้านมีเลนซ้อมสำหรับลูกค้าใช้งานจำนวน 15 เลน เพื่อรองรับผู้ใช้บริการได้ในเวลาเดียวกัน ร้านให้บริการฝึกซ้อมกอล์ฟพร้อมอุปกรณ์เสริม ได้แก่ การให้ยืมไม้กอล์ฟ และการจำหน่ายลูกกอล์ฟสำหรับฝึกซ้อม โดยร้านมีการคิดค่าบริการเป็น ถาดละ 30 บาท จำนวน 30 ลูกต่อถาด และมีบริการให้ยืมไม้กอล์ฟ คิดค่าบริการรอบละ 20 บาทต่อไม้ สามารถใช้งานได้ตลอดทั้งวัน หากอุปกรณ์เกิดการชำรุดหรือเสียหายจากการใช้งาน จะมีการเรียกเก็บ ค่าปรับเป็นจำนวนเงิน 100 บาทต่อไม้

ในปัจจุบัน ร้านดำเนินกิจการโดย เจ้าของร้าน 1 คน และพนักงานบริการ 1 คน เปิดให้บริการทุกวัน ตั้งแต่เวลา 08.00 น. – 19.00 น. ร้านเมืองเลยไดร์ฟกอล์ฟเป็นสถานที่ที่ได้รับความนิยมจากนักกีฬากอล์ฟและประชาชนในจังหวัดเลย เนื่องจากเป็นแหล่งฝึกซ้อมที่สะดวกและมีอุปกรณ์รองรับอย่างครบถ้วน

รูปแบบการให้บริการในปัจจุบัน ลูกค้าที่ต้องการเข้าใช้บริการจะทำการ จองเลนผ่านทางโทรศัพท์ โดยพนักงานจะเป็นผู้รับสายและบันทึกรายการจองลงในสมุดบันทึกการจองแบบเอกสาร โดยมีการสอบถามชื่อลูกค้าที่เป็นผู้จอง จำนวนผู้เข้ามาใช้บริการ ความต้องการยืมไม้กอล์ฟ และแจ้งรายละเอียดค่าใช้จ่าย ได้แก่ ค่าถาดลูกกอล์ฟ ค่ายืมไม้กอล์ฟและค่าปรับกรณีลูกค้าเกิดทำอุปกรณ์ชำรุดเสียหาย (ณัฐดนัย แสนสีละ, การสื่อสารส่วนบุคคล, 2568)

ภาพประกอบที่ 2-1 ร้านเมืองเลยไดร์ฟกอล์ฟ

2.2 แบบจำลองน้ำตก

แบบจำลองน้ำตก (Waterfall Model) รูปแบบน้ำตกเป็นแนวทางเชิงเส้นและต่อเนื่อง ที่ได้รับความนิยมในวิศวกรรมซอฟต์แวร์และการพัฒนาผลิตภัณฑ์ แบบจำลองนี้กำหนดจุดสิ้นสุดหรือเป้าหมายที่ชัดเจนสำหรับแต่ละขั้นตอนของกระบวนการพัฒนา จุดสิ้นสุดหรือเป้าหมายเหล่านี้ไม่สามารถกลับมาทบทวนได้หลังจากเสร็จสิ้นแล้ว

แบบจำลองน้ำตกยังคงถูกนำมาใช้ว่าเป็นวิธีการแรกในการพัฒนาซอฟต์แวร์ นอกจากนี้แบบจำลองนี้ยังถูกใช้โดยทั่วไปในฐานะวิธีการจัดการโครงการระดับสูงสำหรับโครงการที่ซับซ้อนและมีหลายแง่มุม

เมื่อใช้กับโปรเจ็กต์ที่ซับซ้อน เช่น การพัฒนาซอฟต์แวร์ วิธีการพัฒนาแบบ Waterfall จะมี 7 ขั้นตอนดังนี้

2.2.1 การกำหนดปัญหา (Requirements) และแนวทางปฏิบัติของโครงการที่อาจเกิดขึ้นจะถูกวิเคราะห์และจัดทำเป็นเอกสารข้อกำหนดอย่างเป็นทางการ ขั้นตอนแรกของการพัฒนานี้จะกำหนดและวางแผนโครงการโดยไม่ระบุกระบวนการเฉพาะเจาะจง นอกจากนี้ยังกำหนดขอบเขตของโครงการสมาชิกในทีม ผู้มีส่วนได้ส่วนเสีย กระบวนการรวบรวมข้อกำหนด การใช้เครื่องมือช่วยเหลือ เช่น เทมเพลตและไดอะแกรมเวิร์กโฟลว์ และแผนงานโดยรวมของโครงการ

2.2.2 การวิเคราะห์ (Analysis) วิเคราะห์ข้อมูลจำเพาะของระบบเพื่อสร้างแบบจำลองผลิตภัณฑ์และตรรกะทางธุรกิจเพื่อเป็นแนวทางในการผลิต นอกจากนี้ยังต้องมีการตรวจสอบทรัพยากรทางการเงินและทางเทคนิคเพื่อประเมินความเป็นไปได้อีกด้วย

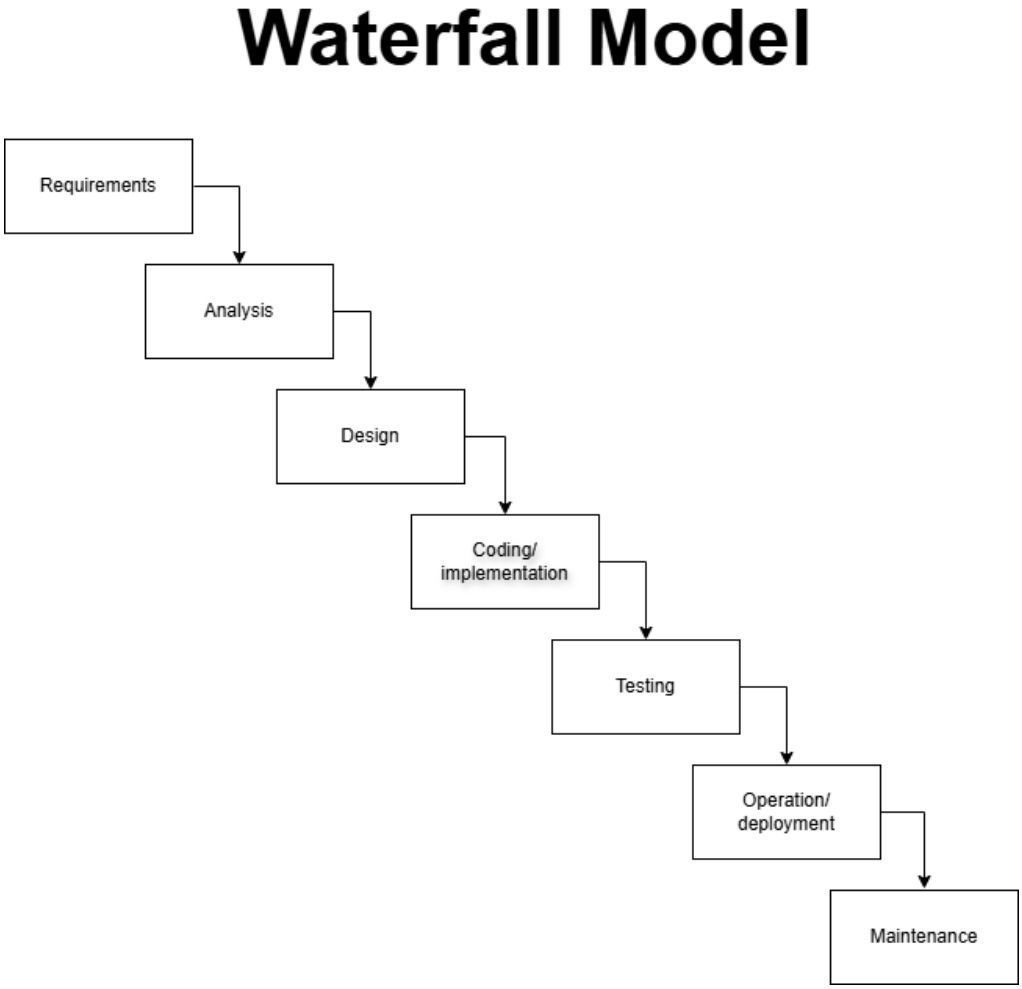
2.2.3 การออกแบบ (Design) การออกแบบเอกสารข้อกำหนดการออกแบบถูกสร้างขึ้นเพื่อสรุปข้อกำหนดทางเทคนิคในระหว่างขั้นตอนการออกแบบ ซึ่งรวมถึงภาษาโปรแกรม ฮาร์ดแวร์ แหล่งข้อมูล สถาปัตยกรรม และบริการต่างๆ

2.2.4 การเขียนโค้ดและการนำไปใช้งาน (Coding and implementation) ซอร์สโค้ดได้รับการพัฒนาโดยใช้แบบจำลอง ตรรกะ และข้อกำหนดเฉพาะที่กำหนดไว้ในขั้นตอนก่อนหน้า โดยทั่วไประบบจะถูกเขียนโค้ดเป็นส่วนประกอบย่อยๆ หรือหน่วยย่อยๆ ก่อนที่จะนำมาประกอบเข้าด้วยกัน

2.2.5 การทดสอบ (Testing) คือการที่ฝ่ายประกันคุณภาพและการทดสอบระบบ ระบุปัญหาที่ต้องแก้ไข การทำเช่นนี้จะบังคับให้ต้องทำซ้ำขั้นตอนการเขียนโค้ดเพื่อแก้ไขจุดบกพร่อง หากระบบผ่านกระบวนการบูรณาการและการทดสอบ กระบวนการ Waterfall จะยังคงดำเนินต่อไป

2.2.6 การดำเนินงานและการปรับใช้ (Operation and deployment) ผลิตภัณฑ์หรือแอปพลิเคชันนี้ถือว่าทำงานได้เต็มประสิทธิภาพและปรับใช้ในสภาพแวดล้อมจริง

2.2.7 การบำรุงรักษา (Maintenance) การบำรุงรักษาเชิงแก้ไข เชิงปรับตัว และเชิงปรับปรุงจะดำเนินการอย่างต่อเนื่องเพื่อปรับปรุง อัปเดต และพัฒนาผลิตภัณฑ์และฟังก์ชันการทำงาน ซึ่งอาจรวมถึงการเผยแพร่แพตช์อัปเดตและเวอร์ชันใหม่ (TechTarget, 2567)



ภาพประกอบที่ 2-2 โมเดลการพัฒนาแบบน้ำตก (Waterfall Model)

2.3 ระบบฐานข้อมูลและเทคโนโลยี NoSQL

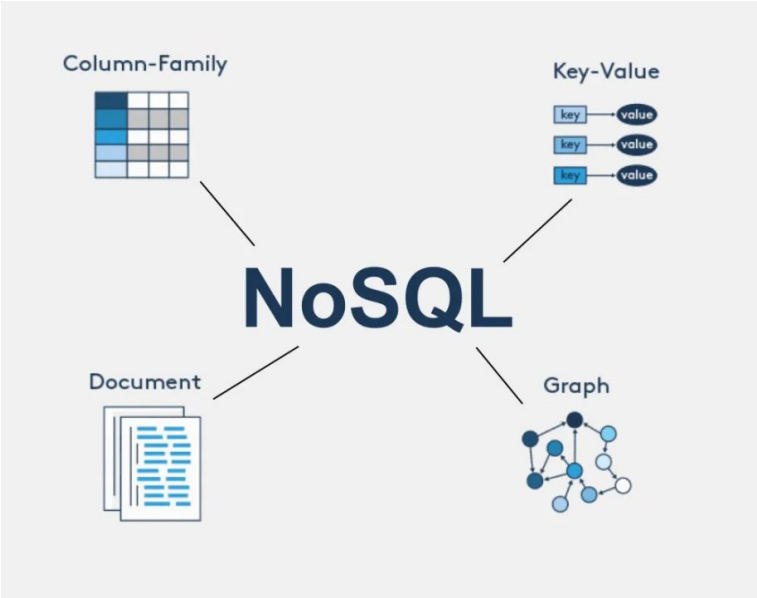
2.3.1 ระบบฐานข้อมูล (Database Systems)

ปกติในระบบแฟ้มข้อมูล ข้อมูลจะถูกจัดเก็บแบบกระจายไปตามหน่วยงานหรือแผนกต่างๆ ซึ่งแต่ละแผนกต่างก็มีกระบวนการจัดเก็บแฟ้มข้อมูลเป็นของตนเอง โดยหากมีผู้หนึ่งผู้ใดในแผนกได้มีการเข้าถึงข้อมูลเพื่อใช้งานในขณะนั้น บุคคลอื่นในแผนกจะไม่สามารถเข้าถึงข้อมูลชุดเดียวกันในช่วงเวลาดังกล่าวได้ แต่สำหรับแนวคิดของฐานข้อมูลจะตรงกันข้ามกับระบบแฟ้มข้อมูลโดยสิ้นเชิง กล่าวคือ ฐานข้อมูลคือศูนย์รวมของข้อมูลต่าง ๆ ที่มีความสัมพันธ์กัน (Relationship) โดยจะมีกระบวนการจัดหมวดหมู่ข้อมูลอย่างมีระเบียบแบบแผน ก่อให้เกิดฐานข้อมูลที่เป็นแหล่งรวมของข้อมูลจากแผนกต่าง ๆ ซึ่งถูกจัดเก็บไว้อย่างมีระบบภายในฐานข้อมูลชุดเดียว โดยผู้ใช้งานแต่ละแผนกสามารถเข้าถึงข้อมูลส่วนกลางนี้เพื่อนำไปประมวลผลร่วมกันได้ และการที่มีศูนย์กลางข้อมูลเพียงแหล่งเดียว รวมถึงความสามารถในการเข้าถึงข้อมูล

เพื่อใช้งานร่วมกันได้จะช่วยแก้ปัญหาความซ้ำซ้อนของข้อมูล และที่สำคัญ ข้อมูลในฐานข้อมูลจะไม่ผูกติดกับโปรแกรม กล่าวคือ จะมีความอิสระในข้อมูล (Program-Data Independence) และด้วยแนวคิดของฐานข้อมูลนี้เอง จึงทำให้สามารถแก้ไขปัญหาต่างๆ ที่เกิดขึ้นจากการประมวลผลด้วยวิธีแฟ้มข้อมูลได้เป็นอย่างดี แต่อย่างไรก็ตาม แนวคิดของระบบฐานข้อมูลนี้ผู้ใช้จำเป็นต้องเรียนรู้ถึงกระบวนการจัดการและต้องพึ่งพาผู้เชี่ยวชาญ หรือผู้ที่มีประสบการณ์ด้านฐานข้อมูลเป็นอย่างดี (โอภาส เอี่ยมสิริวงศ์, 2551)

2.3.2 เทคโนโลยี NoSQL

คือฐานข้อมูลที่ไม่ใช้ภาษา SQL สำหรับจัดการกับข้อมูลในฐานข้อมูล ฐานข้อมูลแบบ NOSQL ไม่ได้เก็บข้อมูลในแบบตาราง แต่จะเก็บข้อมูลในรูปแบบอื่นๆ เช่น เก็บในรูปของเอกสาร (Document), เก็บในรูปของกราฟ (Graph stores), เก็บในรูปของออบเจ็กต์ (Key-value stores) หรือเก็บในรูปของคอลัมน์ (Wide-column stores) ฯลฯ ข้อมูลที่อยู่ในฐานข้อมูลไม่จำเป็นต้องมีโครงสร้างที่แน่นอนสามารถปรับเปลี่ยนโครงสร้างได้ในอนาคต ฐานข้อมูลแบบ NOSQL ซึ่งนิยมใช้งานในปัจจุบัน ได้แก่ MongoDB, Redis, HBase, Cassandra, Neo4J และ Firebase (จีราวุธ วารินทร์ , 2566)



ภาพประกอบที่ 2-3 ฐานข้อมูล NoSQL: เสริมศักยภาพการจัดการข้อมูลยุคใหม่
ที่มา : ดร. ศรีนิวาส จาการ์ลาปูดี (2566)

2.4 Firebase Framework

Firebase คือ Platform ที่รวบรวมเครื่องมือต่าง ๆ สำหรับการจัดการในส่วนของหลังบ้าน หรือ ฝั่งเซิร์ฟเวอร์ ซึ่งทำให้สามารถสร้างแอปพลิเคชันบนเว็บและมือถือได้อย่างมีประสิทธิภาพ และยังลดเวลาและค่าใช้จ่ายของการทำฝั่งเซิร์ฟเวอร์หรือการวิเคราะห์ข้อมูลให้อีกด้วย โดยมีทั้งเครื่องมือที่ฟรี และเครื่องมือที่มีค่าใช้จ่าย Firebase มีบริการหลายอย่างมาก ๆ จะขอยกตัวอย่างบางส่วน โดยแบ่งหัวข้อของ Firebase ดังนี้ (เจษฎา แสงขาว, 2561)

2.4.1 Cloud Firestore

คือหนึ่งในบริการของ Firebase ช่วยจัดการเกี่ยวกับ Database โดยมีการเก็บโครงสร้างข้อมูลแบบ Document Database ซึ่งเป็นส่วนหนึ่งของ NoSQL Database ช่วยให้การออกแบบฐานข้อมูลมีความสะดวกสบาย และยังรองรับการจัดเก็บข้อมูลที่มีหลากหลายรูปแบบมันทำให้เราทำงานสะดวกขึ้นด้วย ตัวอย่างข้อดีของ Firestore เช่น มีเขียนโค้ดน้อยลง มีการกรองข้อมูล (Filter) การจำกัดข้อมูล (Limits) การแบ่งหน้าข้อมูล (Paginate) การจัดเรียงข้อมูล (Sorting) การทำดัชนี (Index) ช่วยให้ค้นหาข้อมูลได้รวดเร็วขึ้น รองรับการขยายตัวของข้อมูลที่มากขึ้น (Scale) ดึงข้อมูลแบบ Realtime เมื่อมีการ Update ข้อมูล (โมเซอร์เปอร์, 2564)

2.4.2 Firebase Authentication

การพัฒนาแอพพลิเคชันที่มีระบบสมาชิก เราสามารถใช้บริการ Firebase Authentication เพื่อจัดการกับการลงทะเบียนสมาชิกใหม่ การเข้าสู่ระบบ และ การออกจากระบบ Firebase Authentication เป็นบริการที่ใช้สำหรับ ตรวจสอบสิทธิ์ การเข้าใช้งาน และยืนยันตัวตนของผู้ใช้ ได้แก่ การลงทะเบียนผู้ใช้ การเข้าสู่ระบบ และ การออกจากระบบ เป็นต้น (จีราวุธ วารินทร์ , 2565)

ข้อดีของการใช้บริการ Firebase Authentication มีดังนี้

2.4.2.1 เราสามารถมีระบบสมาชิก โดยไม่ต้องเขียนโค้ดใดๆ ในฝั่งเซิร์ฟเวอร์เลย

2.4.2.2 มีความปลอดภัยสูง เพราะดูแลและจัดการโดย google

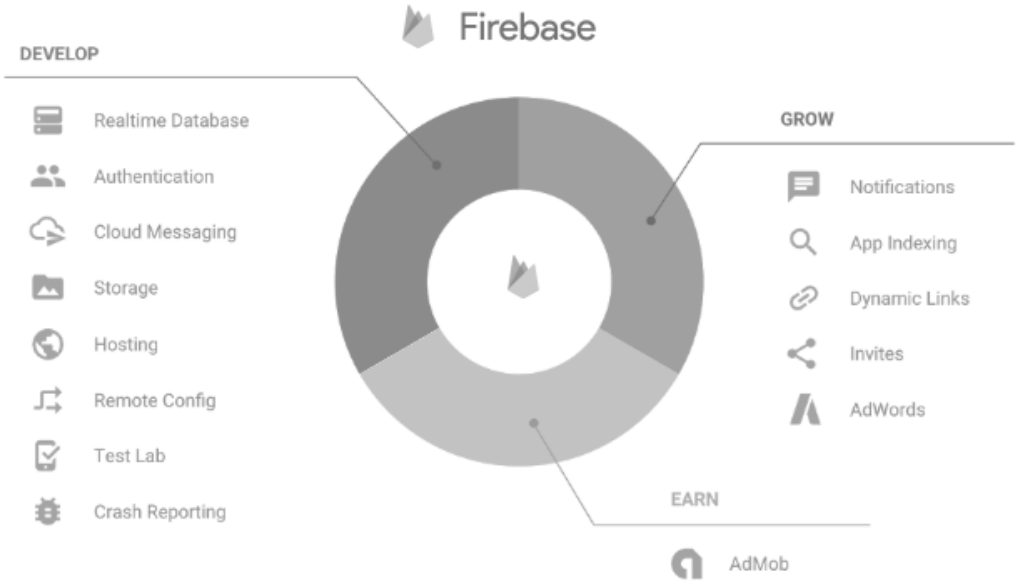
2.4.2.3 รองรับระบบล็อกอินได้หลายแบบ เช่น สามารถล็อกอินด้วยอีเมล/รหัสผ่านหรือล็อกอินผ่านโซเชียลเน็ตเวิร์ค ได้แก่ Google, Facebook ฯลฯ

2.4.3 Firebase Hosting

หากเราต้องการนำแอพพลิเคชันไปออนไลน์บนอินเทอร์เน็ต เราสามารถนำไปฝากไว้กับผู้ใช้บริการเว็บโฮสติ้งต่างๆ ซึ่งที่นิยมใช้งานก็มีอยู่หลายราย เช่น Hostinger,Godaddy,Hostneverdie, z.com ฯลฯ นอกจากนั้นเรายังสามารถฝากเว็บแอพพลิเคชันไว้กับ Firebase ผ่านทางบริการที่มีชื่อว่า Firebase Hosting (จีราวุธ วารินทร์ , 2565)

2.4.4 Firebase CLI

Firebase CLI เป็นเครื่องมือที่สั่งงานด้วยวิธีพิมพ์คำสั่ง ผ่านทางหน้าต่าง Command Line (หรือผ่านทางหน้าจอ Terminal) โดยเรียกวิธีสั่งงานแบบนี้ว่า Command Line Interface หรือ CLI ข้อดีของการสั่งงานแบบนี้คือ สามารถทำงานได้อย่างรวดเร็ว และสามารถกำหนดออบชันได้หลายรูปแบบ แต่ผู้ใช้งานจะต้องกรอกคำสั่งต่างๆ ด้วยการพิมพ์คำสั่งด้วยตนเอง ซึ่งบางคำสั่งอาจซับซ้อนและจดจำยาก เช่น เมื่อต้องการนำแอปพลิเคชันไปออนไลน์บน Firebase Hosting ก็สามารถสั่งงานผ่าน Firebase CLI หรือถ้าต้องการลบข้อมูลออกจาก Cloud Storage ก็สามารถสั่งผ่าน Firebase CLI เป็นต้น (จีราวุธ วารินทร์ , 2565)



ภาพประกอบที่ 2-4 ประโยชน์ของการใช้ Firebase ในการพัฒนาเว็บไซต์และแอปพลิเคชัน
ที่มา : ปาร์ธ ปาเทล (2559)

2.5 Responsive Web Design

การออกแบบเว็บไซต์ให้ตอบสนองได้ทุกอุปกรณ์ หรือ Responsive Web Design คือรูปแบบการสร้างเว็บไซต์ด้วยภาษา HTML และใช้ระบบปฏิบัติการ CSS เป็นส่วนใหญ่

คอนเซปต์สำคัญของการทำ Responsive Web Design นั้น คือ'การสร้างเว็บไซต์ให้สามารถรองรับอุปกรณ์ได้ทุกรูปแบบ ทุกขนาดหน้าจอ และทุกความคมชัด' ในอดีต คอมพิวเตอร์คือช่องทางเดียวที่สามารถเปิดใช้งานเว็บไซต์ได้ แต่เมื่อเวลาผ่านไปและนวัตรกรรมต่าง ๆ เกิดการพัฒนาขึ้น การมาถึงของสมาร์ทโฟน และโทรศัพท์มือถือที่สามารถใช้เชื่อมต่ออินเทอร์เน็ตได้นั้น ถูกใช้งานเพื่อท่องโลกออนไลน์ควบคู่ไปกับคอมพิวเตอร์ และมีอัตราการใช้งานมากขึ้นเรื่อย ๆ จนใกล้เคียงกับปริมาณผู้ใช้คอมพิวเตอร์ทำให้เว็บไซต์ในปัจจุบัน และสื่อโซเชียลมีเดียต่าง ๆ ต้องทำการออกแบบเว็บไซต์ให้มีโครงสร้างที่รองรับกับการแสดงผลบนจอโทรศัพท์มือถือที่ดีอีกด้วย หรือที่เรียกว่ากันว่า Mobile Site นั่นเอง

2.5.1 ข้อดีที่ทำให้ Responsive Web Design ต่างจาก Mobile Site

2.5.1.1 Responsive Web Design นั้น ใช้ระบบปฏิบัติการเว็บไซต์เพียงตัวเดียว เพราะฉะนั้นในทุก ๆ การอัปเดตเว็บไซต์ สามารถเสร็จสิ้นได้ด้วยการอัปเดตเพียงแค่ครั้งเดียว ต่างจากรูปแบบ Mobile Site ที่มีระบบปฏิบัติการแยกกันชัดเจน

2.5.1.2 Responsive Web Design นั้น เนื่องจากเป็นการออกแบบเว็บไซต์ให้แสดงผลลัพธ์ได้บนทุก ๆ หน้าจอผ่านระบบปฏิบัติการตัวเดียว ทำให้ขั้นตอนที่น้อยกว่ากว่า Mobile Site ที่จะต้องทำแยกกันชัดเจน ส่งผลให้มีค่าใช้จ่ายที่น้อยกว่า

2.5.1.3 Responsive Web Design ใช้ระบบปฏิบัติการเพียงตัวเดียว จึงมีการออกแบบและพัฒนาระบบได้ในระยะเวลาอันรวดเร็วกว่า Mobile Site

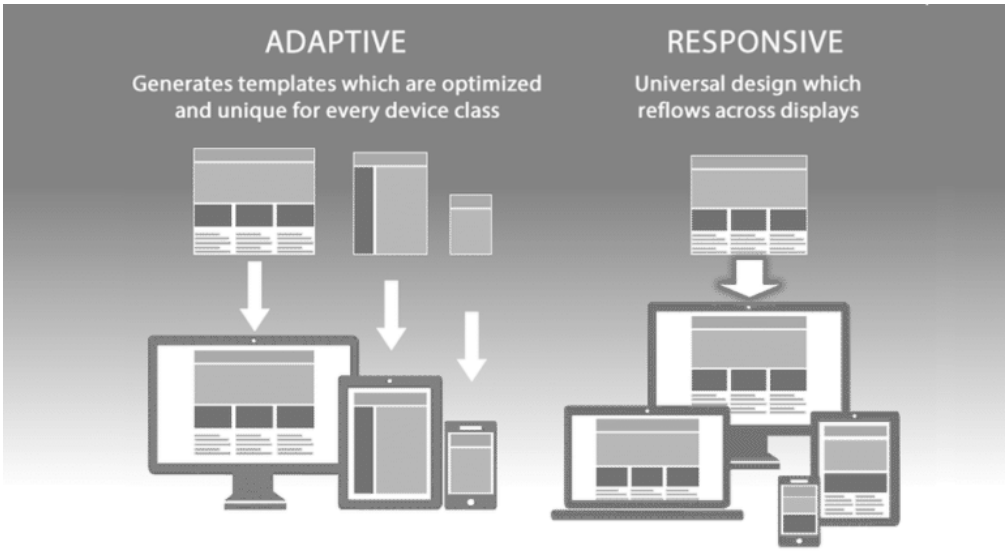
2.5.1.4 Responsive Web Design นั้น สามารถทำได้แม้ว่าจะเป็นองค์กรขนาดเล็ก เนื่องจากดูแลเว็บไซต์ผ่านระบบปฏิบัติการเพียงตัวเดียว

2.5.2 ข้อด้อยของ Responsive Web Design ในปัจจุบัน

2.5.2.1 Responsive Web Design มีการใช้ไฟล์ภาพ หรือองค์ประกอบต่าง ๆ ภายในเว็บไซต์ที่มีขนาดไฟล์ใหญ่กว่าในรูปแบบของ Mobile Site แม้ว่าจะสามารถแสดงผลลัพธ์บนจอโทรศัพท์มือถือได้ทุกรูปแบบ แต่ก็ต้องแลกมากับความเร็วเว็บไซต์ที่ช้ากว่า Mobile Site

2.5.2.2 Responsive Web Design นั้น ไม่สามารถใช้เอฟเฟกต์ โมชัน หรือกราฟิกที่มีขนาดใหญ่ และหวือหวาเป็นจำนวนมากในเว็บไซต์ได้ด้วยข้อจำกัดของการแสดงผลลัพธ์บนหน้าจอโทรศัพท์มือถือที่มีอยู่อย่างจำกัด

2.5.2.3 กระบวนการเขียนโค้ดสำหรับ Responsibe Web Design นั้น จำเป็นต้องใช้ความชำนาญในการเขียนโค้ด และจะต้องหลีกเลี่ยงความผิดพลาดให้ได้มากที่สุด เพราะหากช่องทางใดช่องทางหนึ่งเกิดความเสียหาย นั่นแปลว่าเว็บไซต์ทุกช่องทางจะไม่สามารถแสดงผลได้ (1001click, 2565)



ภาพประกอบที่ 2-5 ภาพแสดงหน้าจอแสดงผลแบบ Responsive
ที่มา : มัตเตโอ ดูโอ (2567)

2.6 ภาษา CSS

CSS ย่อมาจากคำว่า “Cascading Style Sheets” ซึ่งแปลเป็นไทยได้ว่า “สไตล์ชีตแบบซ้อนทับ” คือ ภาษาที่ใช้กำหนดรูปแบบการแสดงผลของ HTML โดยใช้หลักการของการกำหนดกฎเกณฑ์หรือ “กฎของ CSS” (CSS Rules) เพื่อควบคุมลักษณะการแสดงผลขององค์ประกอบต่างๆ ในหน้าเว็บ เช่น ขนาดตัวอักษร สีพื้นหลัง ขอบของกล่อง และการจัดวางองค์ประกอบ

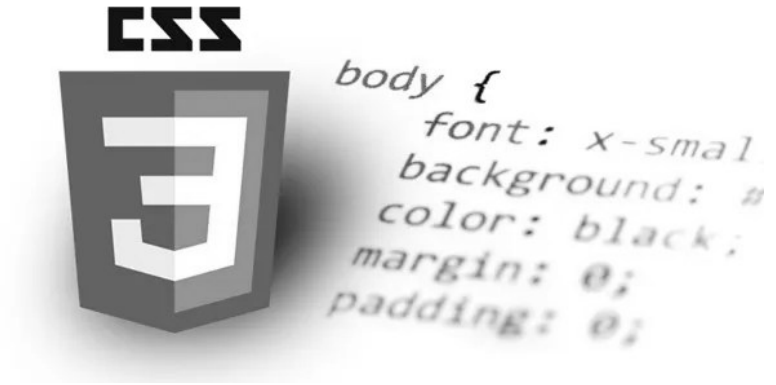
2.6.1 ประโยชน์ของ CSS

2.6.1.1 การแยกเนื้อหาและการออกแบบ คือการแยกโครงสร้างเนื้อหา (HTML) ออกจากการออกแบบ (CSS) ซึ่งทำให้การปรับแต่งรูปแบบของเว็บไซต์ทำได้ง่ายขึ้น และสามารถนำสไตล์ชีตไปใช้กับหลายๆ หน้าได้โดยไม่ต้องแก้ไขโค้ด HTML หลายครั้ง

2.6.1.2 ความสะดวกในการบำรุงรักษา CSS ช่วยให้การบำรุงรักษาเว็บไซต์ทำได้ง่ายขึ้น หากต้องการปรับเปลี่ยนรูปแบบของเว็บไซต์ เพียงแค่แก้ไขไฟล์ CSS ไฟล์เดียว การเปลี่ยนแปลงจะส่งผลกับทุกหน้าเว็บที่ใช้ไฟล์นั้นทันที

2.6.1.3 เพิ่มประสิทธิภาพในการโหลดหน้าเว็บ CSS ช่วยลดขนาดของโค้ด HTML โดยการแยกการจัดรูปแบบออกมาเป็นไฟล์แยก ทำให้หน้าเว็บมีขนาดเล็กลงและโหลดได้เร็วขึ้น

2.6.1.4 การออกแบบที่ยืดหยุ่นและหลากหลาย CSS สามารถปรับเปลี่ยนรูปแบบการแสดงผลของเว็บไซต์ได้ตามต้องการ เช่น การแสดงผลที่แตกต่างกันระหว่างหน้าจอคอมพิวเตอร์และอุปกรณ์มือถือ (Visperhost, 2567)



ภาพประกอบที่ 2-6 โลโก CSS
ที่มา : ซาฮิล ดาห์วัน (2561)

2.7 Tailwind Framework

Tailwind CSS คือ CSS Utility Framework ที่ช่วยให้นักพัฒนาสร้าง UI ที่สำคัญได้ด้วยตัวเองอย่างรวดเร็ว และยังสามารถปรับแต่งในรายละเอียดย่อยต่างๆได้ง่าย เนื่องจากมาพร้อมกับ class สำเร็จรูปอเนกประสงค์ที่ใช้งานได้ทันที

2.7.1 ข้อดีของ Tailwind CSS

2.7.1.1 ปรับแต่งลูกเล่นได้หลากหลาย

โดยปกติแล้ว CSS Framework จะมาพร้อมกับโค้ด HTML และ JavaScript ด้วย จึงทำให้การปรับแต่งแต่ละอย่างอาจไม่สะดวกมากนักในบางครั้ง แต่ Tailwind CSS เป็น utility ที่คุณสามารถนำไปใส่กับอะไรก็ได้ แถมยังสามารถปรับแต่งดีไซน์ ธีม ระยะห่าง สี และลูกเล่นอื่นๆ ได้สะดวกและเข้าใจง่ายอีกด้วย

2.7.1.2 ลดความยุ่งยากในการตั้งชื่อ Class

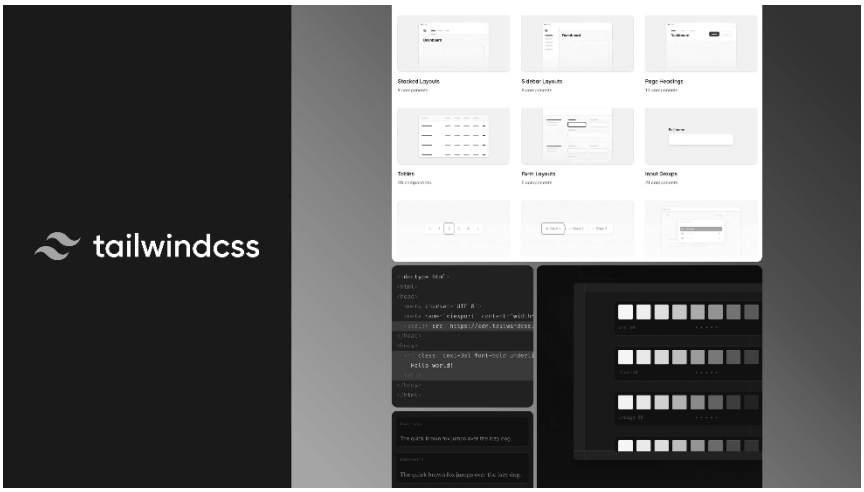
ปัญหาหลักของการเขียน CSS แบบเดิมคือต้องคอยคิดชื่อ Class ใหม่ๆ ให้สื่อความหมาย ซึ่งเสียเวลาและจัดการยาก แต่ Tailwind CSS ช่วยแก้ปัญหานี้โดยการให้เราเรียกใช้ชื่อ Class สำเร็จรูปที่มีมาตรฐานได้เลย (เช่น p-4, text-center, flex) ทำให้โค้ดเป็นระเบียบ เข้าใจง่าย และนักพัฒนาไม่ต้องปวดหัวกับการตั้งชื่อ Class ซ้ำซ้อนอีกต่อไป

2.7.1.3 การเพิ่มประสิทธิภาพ Purge CSS

Tailwind CSS ทำงานร่วมกับฟีเจอร์ที่เรียกว่า Purge CSS ซึ่งทำหน้าที่เหมือน "ตัวสแกนขยะ" โดยจะตรวจสอบไฟล์ HTML ทั้งหมดว่าเราใช้ Class ตัวไหนบ้าง และจะ "ลบ Class ที่ไม่ได้ใช้งานทิ้งไป" ก่อนนำเว็บไซต์ขึ้นใช้งานจริง ทำให้ไฟล์ CSS สุดท้ายมีขนาดเล็กมาก ส่งผลให้เว็บไซต์โหลดเร็วและมีประสิทธิภาพสูง

2.7.1.4 ทำเว็บไซต์ Responsive Design ได้ง่ายขึ้น

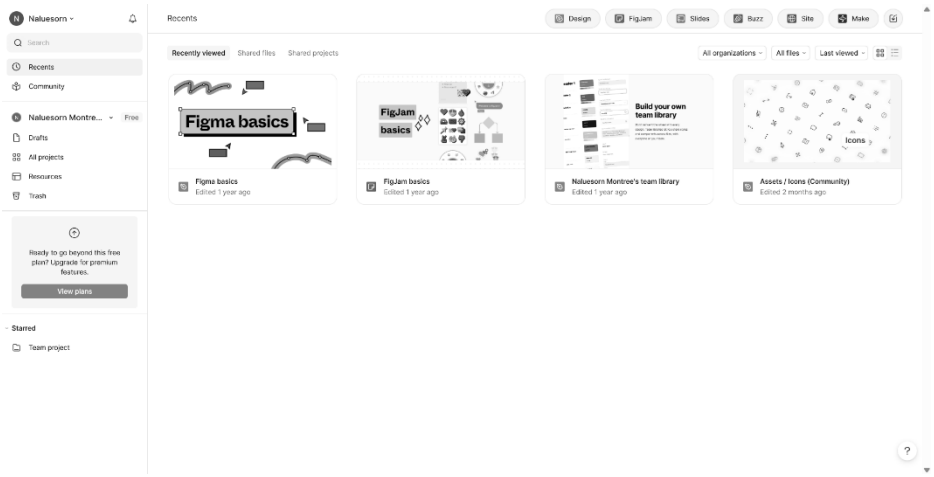
การกำหนดค่าเริ่มต้นของ Tailwind CSS จะมุ่งเน้นที่โทรศัพท์มือถือเป็นอันดับแรก นั่นจึงทำให้ utility class ทั้งหมดใน Tailwind สามารถปรับเปลี่ยนเงื่อนไขในจุดสั่งหยุดที่แตกต่างกันได้ง่าย (ชรินทร เรืองไลคราม, 2566)



ภาพประกอบที่ 2-7 ตัวอย่างการกำหนดรูปแบบของ Tailwind
ที่มา : ดาวิด คีลบาซา (2566)

2.8 Figma

Figma เป็นเครื่องมือออกแบบอินเทอร์เฟซแบบทำงานร่วมกัน ซึ่งความสามารถในการทำงานร่วมกัน กลายเป็นจุดเด่นที่ทำให้ Figma ได้รับความนิยมในแวดวงนักออกแบบ UX/UI เพราะในโลกของการทำงานจริงเหล่านักออกแบบหลายครั้งต้องทำงานร่วมกับนักออกแบบคนอื่นๆ ไปจนถึงผู้ที่เกี่ยวข้อง นักออกแบบจึงไม่ได้ใช้ Figma สำหรับแค่การจัดวางเลย์เอาต์อินเทอร์เฟซ แต่ยังใช้สำหรับการสร้างแบบจำลอง (Prototype) และพรีเซ้นท์งานเพื่อให้เกิดความเข้าใจที่ตรงกัน และยังสามารถแชร์ Design System เพื่อให้สามารถหยิบยืมงานกันได้ และทำให้งานออกแบบในภาพรวมมีความสอดคล้องกัน (True Digital Academy, 2566)



ภาพประกอบที่ 2-8 เว็บไซต์ Figma

2.9 ภาษา JavaScript

JavaScript เป็นภาษาคอมพิวเตอร์ ที่ถูกออกแบบมาให้ทำงานบนเบราว์เซอร์ จุดประสงค์เพื่อทำให้เว็บเพจสามารถโต้ตอบกับผู้ใช้ได้ หรือมีความสามารถบางอย่างเพิ่มขึ้น เช่น สามารถใช้ JavaScript เพื่อขยายรูปภาพเมื่อผู้ใช้เลื่อนเมาส์ไปที่รูปภาพ, ใช้ JavaScript เพื่อสร้างแอนิเมชันเลื่อนข้อความจากซ้ายไปขวา, ใช้ JavaScript เพื่อนำเนื้อหาใหม่มาแทนที่เนื้อหาเดิมโดยไม่ต้องโหลดเว็บเพจใหม่ เป็นต้น

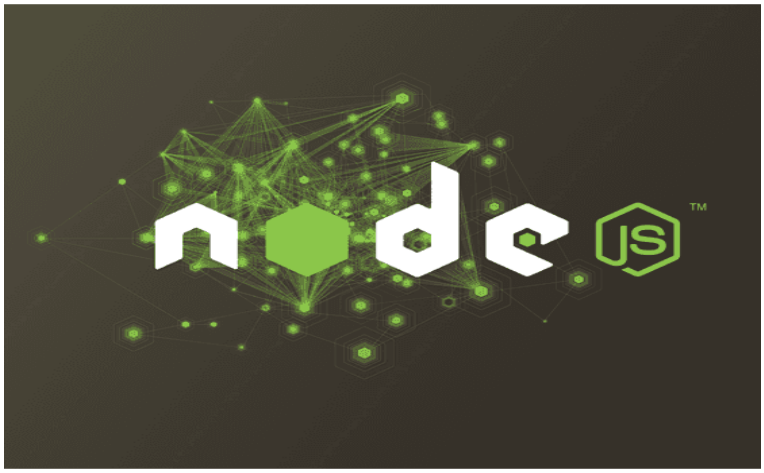
JavaScript เป็นภาษาคอมพิวเตอร์ที่ออกแบบมาสำหรับใช้กับเว็บเพจ ดังนั้น โค้ด JavaScript จึงถูกรันเพื่อใช้งานบนเบราว์เซอร์เป็นหลัก นอกจากนั้น JavaScript ยังสามารถใช้กับสภาพแวดล้อมอื่น ๆ ที่ไม่ใช่เบราว์เซอร์ได้อีกด้วย เช่น ใช้ JavaScript กับ Node.js และ Apache CouchDB เป็นต้น (จีราวุธ วารินทร์ , 2566)

```
1   import React, { useState, useEffect } from 'react';
2   import { auth, db } from './firebase'; // เพิ่ม db เข้ามาด้วย
3   import { onAuthStateChanged, signOut } from "firebase/auth";
4   import { doc, getDoc } from "firebase/firestore"; // สำหรับดึงข้อมูล Role
5   import Auth from './components/Auth';
6   import Checkout from './components/Checkout';
7   import OwnerDashboard from './components/OwnerDashboard'; // นำเข้า Dashboard ของเจ้าของ
8
9   function App() {
10    const [user, setUser] = useState(null);
11    const [role, setRole] = useState(null); // เพิ่ม State สำหรับเก็บสิทธิ์ (owner, staff, customer)
12    const [loading, setLoading] = useState(true);
13
14    useEffect(() => {
15      const unsubscribe = onAuthStateChanged(auth, async (currentUser) => {
16        if (currentUser) {
17          setUser(currentUser);
18          // --- ส่วนสำคัญ: ดึงสิทธิ์จาก Firestore ---
19          const userDoc = await getDoc(doc(db, "users", currentUser.uid));
20          if (userDoc.exists()) {
21            setRole(userDoc.data().role);
22          }
23        } else {
24          setUser(null);
25          setRole(null);
26        }
27        setLoading(false);
28      });
29      return () => unsubscribe();
30    }, []);
```

ภาพประกอบที่ 2-9 ตัวอย่างภาษา JavaScript

2.10 Node.js Framework

Node.js เป็นการกำหนดสภาพแวดล้อมเพื่อให้สามารถรัน JavaScript จากคอมพิวเตอร์โดยไม่ต้องใช้เบราว์เซอร์ จุดเด่นของ Node.js ช่วยให้นักพัฒนาสามารถใช้ JavaScript สร้างแอปพลิเคชันในฝั่งเซิร์ฟเวอร์, สามารถอ่านเขียนไฟล์ในคอมพิวเตอร์ ตลอดจนสามารถควบคุมการรับส่งข้อมูลผ่านเครือข่ายได้ และ Node.js ยังสามารถทำงานข้ามแพลตฟอร์มได้ เช่น สามารถรันบนวินโดวส์, แมคโอเอส หรือลินุกซ์ ภายใน Node.js จะประกอบด้วย JavaScript Engine V8 ของ Google Chrome ใช้แปลชุดคำสั่ง JavaScriptและมีโค้ดคำสั่งภาษา C++ เพื่อกำหนดสภาพแวดล้อมต่าง ๆ ที่จำเป็น เช่น กำหนดไลบรารี สำหรับรันชุดคำสั่งที่ใช้เวลาในการประมวลผล หรือใช้เพื่อติดต่อกับระบบปฏิบัติการ เนื่องจาก Node.js ถูกออกแบบให้ใช้จำนวนหน่วยประมวลผลน้อย (Thread), รองรับชุดคำสั่งในแบบที่ต้องใช้เวลาในการประมวลผล (Asynchronous) และสามารถทำงานหลายอย่างได้พร้อมกันได้ต่อเนื่องโดยไม่ต้องรอให้งานแรกเสร็จก่อน ด้วยเหตุผลดังกล่าว แอปพลิเคชันที่สร้างจาก Nodejs จึงสามารถทำงานได้รวดเร็วและ มีประสิทธิภาพมาก ปัจจุบัน Node.js ถูกนำมาใช้งานอย่างกว้างขวาง โดยเฉพาะการสร้างเน็ตเวิร์คแอปพลิเคชัน (จีราวุธ วารินทร์ , 2566)



ภาพประกอบที่ 2-10 โลโก Node.js
ที่มา : เอมิลี่ ทานากะ-เดลกาโด (2554)

2.11 Express.js Framework

Express คือ เฟรมเวิร์คของ Node.js ใช้สำหรับสร้างเว็บแอปพลิเคชัน แม้ว่า Express จะมีลักษณะเป็นเฟรมเวิร์คที่วางกรอบการใช้งานเอาไว้ แต่ Express เป็นเฟรมเวิร์คที่ใช้งานง่าย นักพัฒนาสามารถใช้ JavaScript สร้างเว็บเซิร์ฟเวอร์ และจัดการกับ Request และ Response รูปแบบต่าง ๆ ได้อย่างรวดเร็วโดยใช้โค้ดเพียงไม่กี่บรรทัด พื้นฐานการทำงานของ Express สามารถสรุปได้ดังนี้

2.11.1 เบราว์เซอร์ส่ง HTTP Request ไปยังเซิร์ฟเวอร์ เพื่อร้องขอข้อมูลบางอย่างจากเซิร์ฟเวอร์

2.11.2 เซิร์ฟเวอร์ส่งต่อ HTTP Request ไปให้กับ Express App

2.11.3 Express App เรียกใช้งานฟังก์ชันที่เรียกว่า Middleware เพื่อเพิ่มหรือปรับแต่งคุณสมบัติให้กับ Request หรือ Response ให้มีคุณสมบัติตามต้องการ

2.11.4 นักพัฒนาสามารถกำหนด Middleware ได้หลายตัว โดยจะเรียกใช้งาน Middleware ตามลำดับที่ได้กำหนดไว้ Middleware ตัวสุดท้ายที่ถูกเรียกใช้งาน จะเป็นผู้กำหนดรูปแบบของผลลัพธ์ (เช่น เว็บเพจหรือข้อมูลลักษณะต่าง ๆ) แล้วส่งต่อไปให้กับเซิร์ฟเวอร์

2.11.5 เซิร์ฟเวอร์ส่ง HTTP Response กลับไปให้กับเบราว์เซอร์ (จีราวุธ วารินทร์ , 2566)

express

ภาพประกอบที่ 2-11 โลโก Express.js
ที่มา : ทีเจ โฮโลเวย์ชุก (2553)

2.12 React.js Framework

React เป็นเฟรมเวิร์ก JavaScript ที่พัฒนาโดย Facebook เพื่อช่วยสร้าง User Interface (UI) ที่มีประสิทธิภาพ โดย React มุ่งเน้นการสร้าง Component ซึ่งเป็นส่วนประกอบของ UI ที่สามารถใช้ซ้ำได้ และแต่ละ Component สามารถเก็บสถานะ (state) และเมทอด (methods) ต่างๆ เพื่อการจัดการกับข้อมูลและการแสดงผล

React เหมาะสำหรับการพัฒนาแอปพลิเคชันเว็บที่มีขอบเขตใหญ่หรือซับซ้อน โดยเฉพาะอย่างยิ่งเมื่อต้องการในการจัดการข้อมูลแบบ Real-time หรือการสร้าง UI ที่ตอบสนองต่อเหตุการณ์ต่าง ๆ อย่างรวดเร็ว เช่น เว็บแอปพลิเคชันสำหรับการซื้อขาย, เกม, แพลตฟอร์มการค้าออนไลน์ เป็นต้น

ข้อดี-ข้อเสียของ React

2.12.1 ข้อดี

2.12.1.1 ประสิทธิภาพสูง: React มี Virtual DOM ที่ช่วยลดการทำงานกับ DOM จริง ทำให้มีประสิทธิภาพการแสดงผล UI ที่ดี

2.12.1.2 Component-based: เป็นการสร้าง UI ที่มีความยืดหยุ่น และสามารถใช้ซ้ำได้

2.12.1.3 การจัดการสถานะของ Component: React ช่วยให้การจัดการสถานะของ Component เป็นไปอย่างมีประสิทธิภาพ

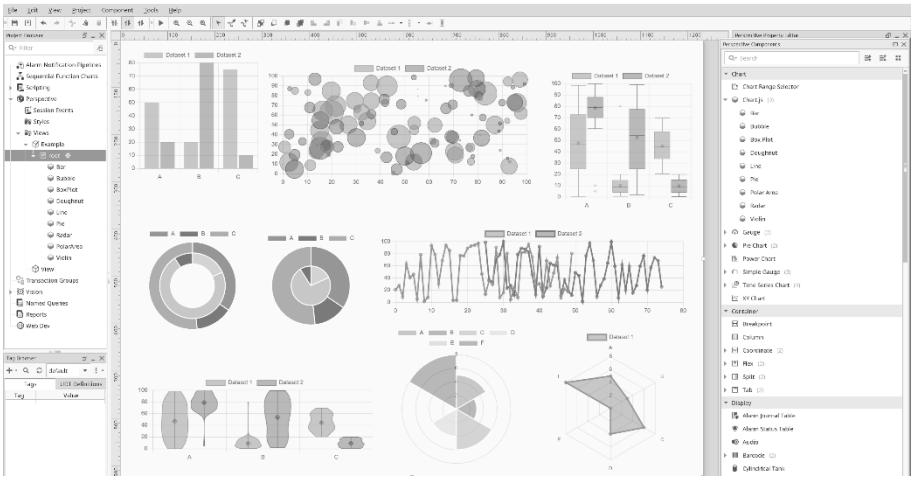
2.12.2 ข้อเสีย

2.12.2.1 การเรียนรู้และความซับซ้อน: React อาจมีความซับซ้อนสำหรับผู้เริ่มต้น โดยเฉพาะผู้ที่ไม่เคยใช้ Component-based architecture มาก่อน

2.12.2.2 สถานะ (state) ที่ซับซ้อน: การจัดการสถานะที่ซับซ้อนอาจทำให้โค้ดซับซ้อนขึ้น (ประพิมพ์ภัทร นิ่มประเสริฐ, 2567)

2.13 Chart.js Library

Chart.js เป็น JavaScript library ที่ช่วยให้นักพัฒนาสามารถสร้างกราฟและแผนภูมิในเว็บไซต์ได้อย่างง่ายดาย โดยไม่จำเป็นต้องมีความรู้ด้านการเขียนโค้ดที่ซับซ้อน นอกจากนี้ยังมีคุณสมบัติที่ยืดหยุ่นและปรับแต่งได้ตามความต้องการของผู้ใช้งาน Chart.js รอบรับการสร้างกราฟหลากหลายประเภท เช่น กราฟแท่ง กราฟเส้น กราฟวงกลม กราฟเรดาร์เป็นต้น อีกทั้ง Chart.js ยังมีขนาดเล็กและง่ายต่อการใช้งาน ทำให้เป็นตัวเลือกที่ดีสำหรับการสร้างแดชบอร์ดที่มีประสิทธิภาพ (อิซารัป เอ็กซินโคล, 2567)

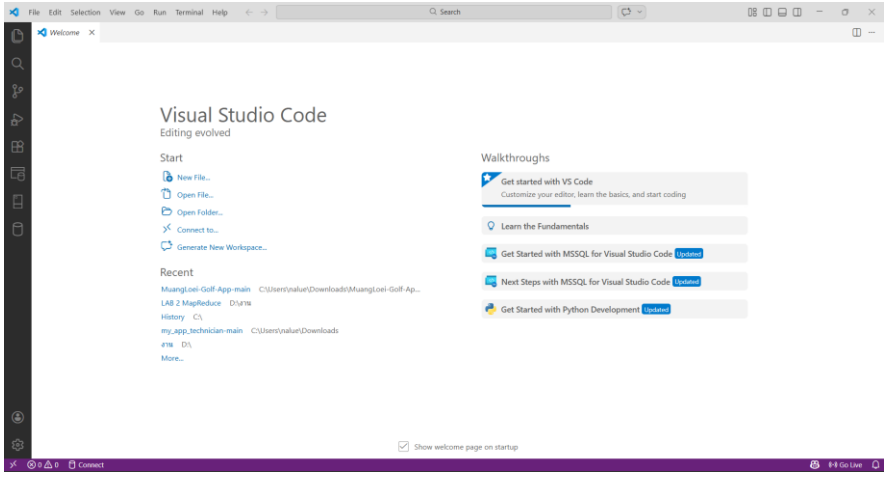


ภาพประกอบที่ 2-12 ตัวอย่างกราฟ Chart.js
ที่มา : เบน มัสสัน (2568)

2.14 โปรแกรม Visual Studio Code

Visual Studio Code หรือ VS Code คือ โปรแกรม Code Editor ของ Microsoft ที่ใช้สำหรับเขียน แก้ไขและตรวจสอบความผิดปกติของโค้ด คุณสมบัติของโปรแกรมมีลักษณะเบา มีประสิทธิภาพสูง มีส่วนขยาย (Extension) เพื่อเพิ่มขีดความสามารถและสามารถเขียนโค้ดได้หลากหลายภาษา

VS Code สนับสนุนการเขียนโค้ดแทบทุกภาษาคอมพิวเตอร์ เช่น JavaScript, TypeScript, CSS และ HTML และยังมีส่วนขยาย (Extension) ภายในพื้นที่ขายของ VS Code Marketplace ให้เราเลือกใช้ได้อีกมากมาย (Webdodee, 2567)



ภาพประกอบที่ 2-13 การพัฒนาด้วย VS Code

2.15 งานวิจัยที่เกี่ยวข้อง

ปิติ โง้วธนสุวรรณ และคณะ (2568) ได้ศึกษาและพัฒนาโปรแกรมบริหารจัดการจองห้องแบบบูรณาการ มหาวิทยาลัยขอนแก่น งานวิจัยนี้มีวัตถุประสงค์หลักเพื่อใช้ในการบริหารจัดการจองห้องประชุม การเรียน การสอน สัมมนา และอบรมผ่านระบบอินเทอร์เน็ตมหาวิทยาลัยขอนแก่น มีการเชื่อมต่อกับระบบต่าง ๆ ที่เกี่ยวข้องผ่าน API Web Service การพัฒนาโปรแกรมใช้ภาษา JavaScript ทั้งในส่วนติดต่อกับผู้ใช้งาน (user interface) และในส่วนของ APIWeb Service ที่ติดต่อกับฐานข้อมูล การพัฒนาส่วนติดต่อกับผู้ใช้งานเป็นแบบ Single Page Application (SPA) ทำให้การใช้งานระหว่างผู้ใช้งานกับระบบมีความราบรื่นน่าใช้งานไม่รู้สึกว่าต้องรอการโหลดจากเว็บไซต์ซึ่งผู้วิจัยได้พัฒนาโปรแกรมแบ่งการทำงานออกเป็น 2 ส่วน (1) Front End เป็นส่วนติดต่อกับผู้ใช้งานผู้วิจัยพัฒนาโดยใช้ React Library (2) Back End เป็นส่วนติดต่อกับฐานข้อมูลผู้วิจัยพัฒนาโดยใช้ Node.js การทำงานในส่วนต่างๆ ของระบบสื่อสารกันผ่าน API โดยรับส่งข้อมูลในรูปแบบ JSON นอกจากนี้เมื่อมีการจองห้องใช้งานโปรแกรมจะทำการตรวจสอบสิทธิ์การเข้าใช้งาน ในส่วนของบุคลากรภายในโปรแกรมจะตรวจสอบผ่านระบบ KKU SSO ของมหาวิทยาลัย แต่ถ้าเป็นบุคคลภายนอกโปรแกรมจะตรวจสอบผ่านระบบ Social Login ต่าง ๆ ที่ให้บริการ เมื่อมีการขอใช้บริการห้องหรืออนุมัติการใช้บริการห้อง โปรแกรมจะทำการส่งอีเมลให้กับผู้ที่เกี่ยวข้องทั้งหมด และมีการส่งข้อมูลผ่าน API ให้กับระบบการเงินการคลังของมหาวิทยาลัยขอนแก่น (KKUFMIS) และ KKU PAYMENT HUB โปรแกรมสามารถจัดเก็บและสืบค้นข้อมูลการจองห้องได้อย่างถูกต้อง รวดเร็ว และมีประสิทธิภาพ จากการประเมินความพึงพอใจของผู้ใช้งานโปรแกรมที่พัฒนาขึ้น พบว่า

ผลการประเมินความพึงพอใจของผู้ใช้งานโปรแกรมบริหารจัดการจองห้องแบบบูรณาการ มหาวิทยาลัยขอนแก่น อยู่ในระดับมาก (มีค่าเฉลี่ยรวม 3.99 ค่า S.D. เท่ากับ 0.83)

ศุภนิดา รุ่งคูหา และ บัญชา วัฒนะ (2568) ได้ศึกษาและพัฒนาระบบการจัดการการเบิกจ่ายวัตถุดิบสำหรับการผลิต กรณีศึกษา โรงงานผลิตชิ้นส่วนยานยนต์ งานวิจัยนี้มีวัตถุประสงค์เพื่อพัฒนาระบบการจัดการและตรวจสอบสถานะการเบิกจ่ายวัตถุดิบสำหรับการผลิตกรณีศึกษาโรงงานผลิตชิ้นส่วนยานยนต์โดยมุ่งเน้นการลดขั้นตอนที่ซ้ำซ้อนเพิ่มความสะดวกในการตรวจสอบข้อมูลและรองรับการติดตามสถานะการเบิกจ่ายวัตถุดิบได้ตามเวลาจริงระบบที่พัฒนาขึ้นได้รับการออกแบบตามทฤษฎีวงจรการพัฒนาระบบ (SDLC) ในรูปแบบเว็บแอปพลิเคชันด้วย Node.js, Express.js, React.js และฐานข้อมูล PostgreSQL ซึ่งรองรับการประมวลผลตามเวลาจริง มีความปลอดภัยและรองรับการขยายขนาดได้สูงผลการทดสอบพบว่าระบบสามารถลดขั้นตอนการทำงานเดิมลงจาก 10 ขั้นตอนเหลือ 5 ขั้นตอนและผลการประเมินความพึงพอใจพบว่า ด้านความสะดวกในการใช้งานระบบและด้านการออกแบบระบบมีค่าเฉลี่ย 3.94 และ 4.11 ส่วนด้านประสิทธิภาพของระบบและด้านประโยชน์ของระบบต่อการปฏิบัติงานมีค่าเฉลี่ยเท่ากับ 4.78 และ 4.72 โดยรวมความพึงพอใจของระบบอยู่ในระดับมาก มีค่าเฉลี่ย 4.39 จึงสรุปได้ว่าระบบที่พัฒนาขึ้นสามารถช่วยเพิ่มประสิทธิภาพในการดำเนินงาน ซึ่งนอกจากจะช่วยให้ผู้ใช้งานสามารถตรวจสอบข้อมูลได้อย่างสะดวกแล้ว ยังสามารถติดตามสถานะการเบิกจ่ายวัตถุดิบได้อย่างมีประสิทธิภาพและแม่นยำยิ่งขึ้นตรงตามความต้องการของผู้ใช้งาน

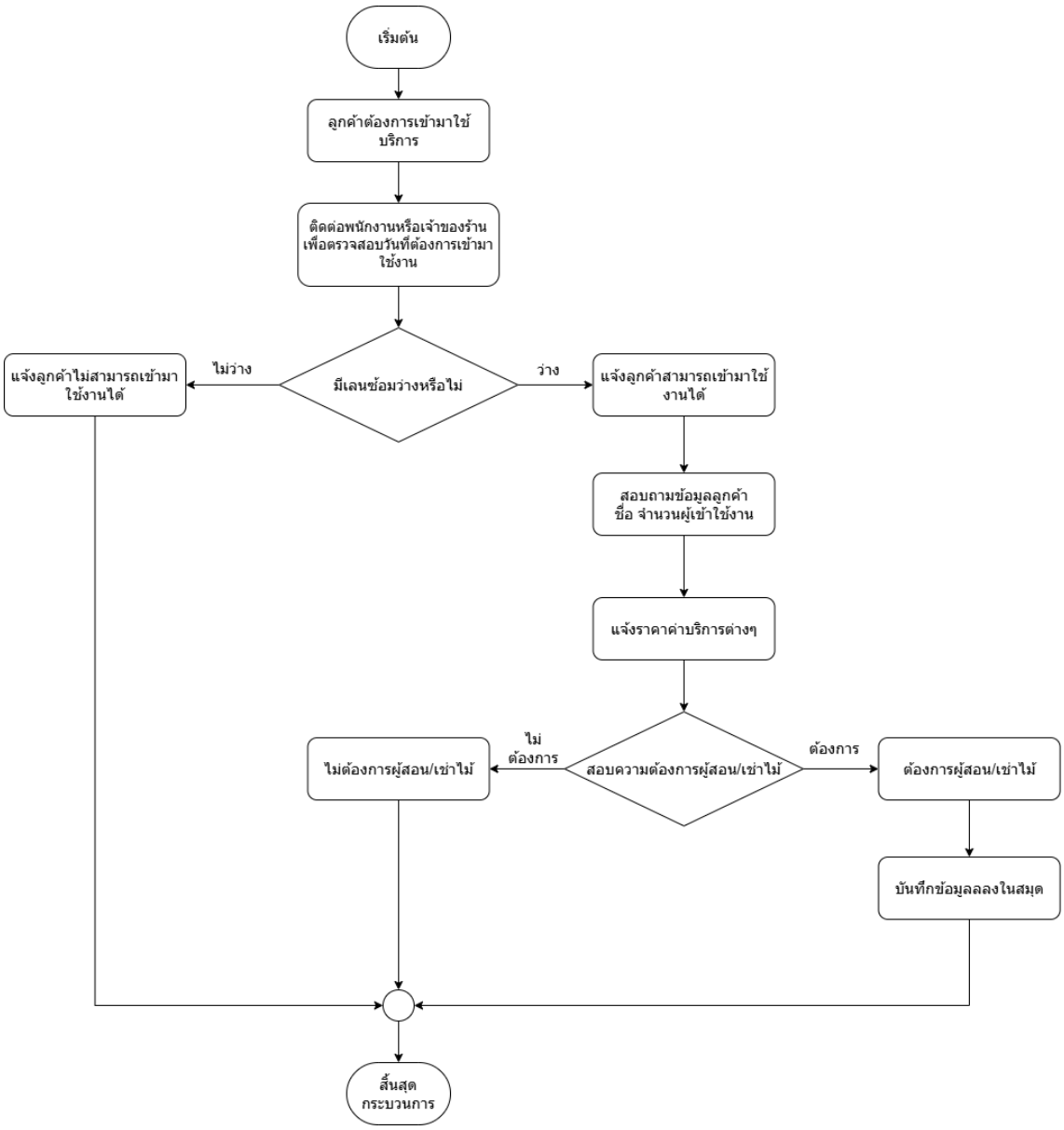
ทักษิณ น้อยบุดดี และ รัฐกรณ์ ศรีจันทร์ (2567) ได้ศึกษาและพัฒนาระบบจองสนามกีฬาออนไลน์ กรณีศึกษาคณะวิทยาการสารสนเทศ มหาวิทยาลัยมหาสารคาม งานวิจัยนี้มีวัตถุประสงค์เพื่อเพิ่มประสิทธิภาพในการบริหารจัดการสนามและอุปกรณ์กีฬาผ่านระบบเครือข่ายอินเทอร์เน็ต การพัฒนาโปรแกรมใช้ภาษา JavaScript เป็นหลัก ทั้งในส่วนติดต่อกับผู้ใช้งาน (User Interface) และส่วนประมวลผลที่ติดต่อกับฐานข้อมูล พัฒนาในรูปแบบเว็บแอปพลิเคชันโดยแบ่งการทำงานออกเป็น 2 ส่วน คือ (1) Front End เป็นส่วนติดต่อกับผู้ใช้งานพัฒนาโดยใช้ React Library (2) Back End เป็นส่วนประมวลผลพัฒนาโดยใช้ Node.js และ Express.js ร่วมกับฐานข้อมูล MongoDB ซึ่งเป็นฐานข้อมูลประเภท NoSQL เพื่อรองรับการจัดเก็บข้อมูลที่มีความยืดหยุ่นสูงและรองรับการทำงานแบบเรียลไทม์ (Real-time) ระบบประกอบด้วยฟังก์ชันหลัก ได้แก่ การจองสนามกีฬา การเช่าอุปกรณ์กีฬา และการสรุปรายงานผลการดำเนินงาน ผู้วิจัยได้ทำการทดสอบการทำงานของระบบด้วยเทคนิคการทดสอบกล่องดำ (Black Box Testing) ครอบคลุมทั้งหมด 24 กรณีทดสอบ เช่น ระบบสมัครสมาชิก การจองสนาม และการชำระเงิน พบว่าระบบสามารถทำงานได้ถูกต้องแม่นยำตามเงื่อนไขที่กำหนดร้อยละ 100

บทที่ 3
วิธีการออกแบบระบบและดำเนินการศึกษา

3.1 ความเป็นมาของระบบเดิม

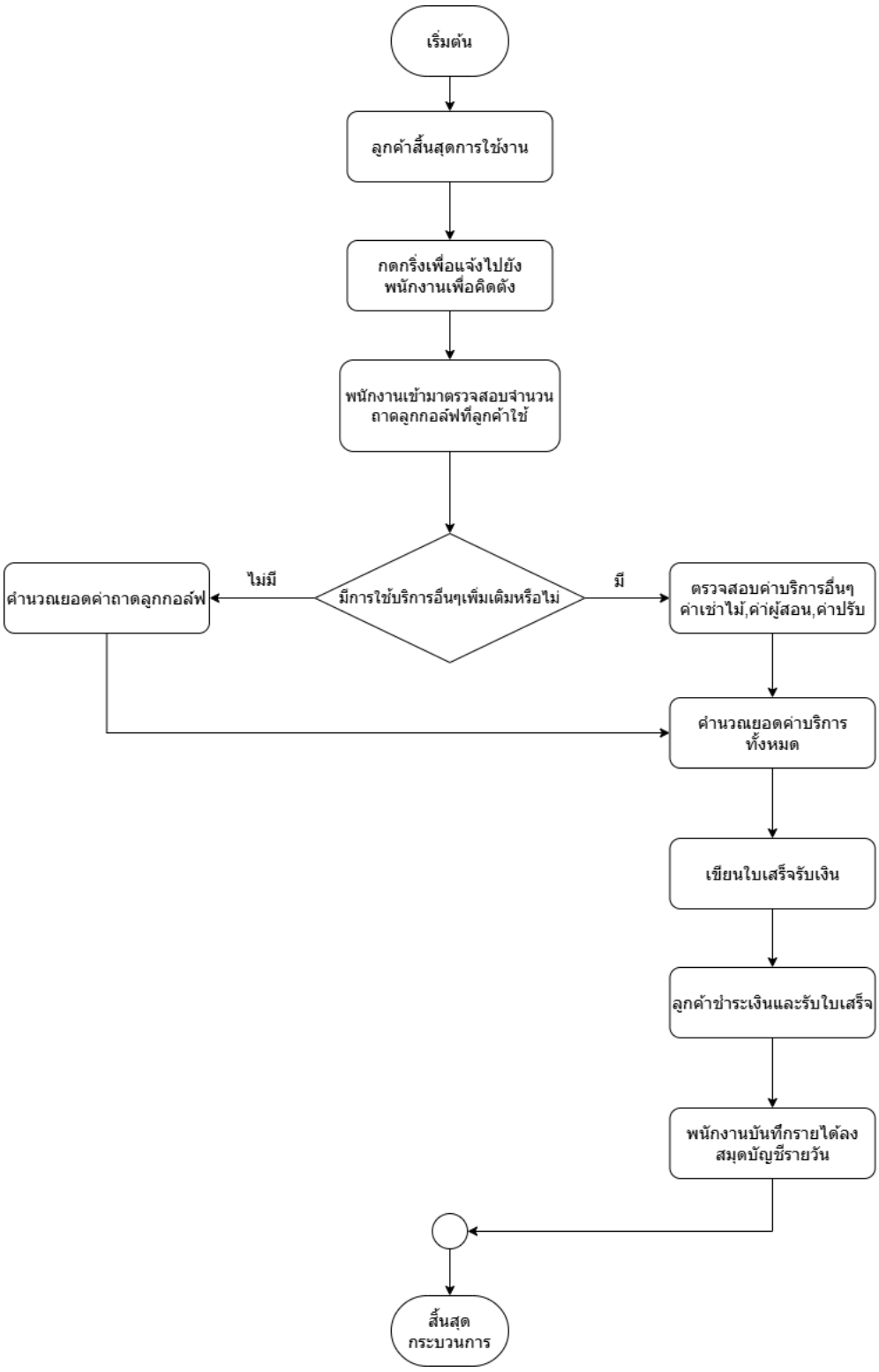
ร้านเมืองเลยไดร์ฟกอล์ฟเป็นธุรกิจเปิดให้บริการสถานที่ฝึกซ้อมกีฬากอล์ฟ เปิดให้บริการแก่ประชาชนทั่วไป นักกีฬากอล์ฟ และผู้ที่สนใจฝึกซ้อมพื้นฐานการเล่นกอล์ฟ ร้านมีเลนซ้อมให้บริการแก่ลูกค้าใช้งานจำนวน 15 เลน เพื่อสามารถรองรับผู้ใช้บริการได้ในเวลาเดียวกัน ร้านให้บริการฝึกซ้อมกอล์ฟพร้อมอุปกรณ์เสริม ได้แก่ การเช่าไม้กอล์ฟ และการบริการถาดลูกกอล์ฟสำหรับฝึกซ้อม ร้านดำเนินกิจการโดย เจ้าของร้าน 1 คน และพนักงานบริการ 1 คน เปิดให้บริการทุกวัน ร้านเมืองเลยไดร์ฟกอล์ฟเป็นสถานที่ที่ได้รับความนิยมจากนักกีฬากอล์ฟและประชาชนในจังหวัด รูปแบบการให้บริการในปัจจุบัน หากลูกค้าที่ต้องการเข้ามาใช้บริการจะทำการ จองเลนผ่านทางโทรศัพท์ โดยพนักงานจะเป็นผู้รับสายและบันทึกรายการจองลงในสมุดบันทึก โดยมีการสอบถามข้อมูลรายละเอียดผู้จองดังนี้ 1) ชื่อลูกค้าที่เป็นผู้จอง 2) จำนวนผู้เข้ามาใช้บริการ 3) ความต้องการผู้สอน 4) ความต้องการยืมไม้กอล์ฟ จากนั้นเมื่อลูกค้าเข้ามาจองจะทำการเปิดเลนซ้อมให้ลูกค้า โดยการจัดเตรียมถาดลูกกอล์ฟและไม้กอล์ฟที่ลูกค้าต้องการไปยังเลนที่กำหนด พร้อมทั้งอธิบายขั้นตอนการใช้บริการและกฎระเบียบต่างๆ ให้แก่ลูกค้าทราบก่อนเริ่มฝึกซ้อม ในส่วนของกระบวนการชำระเงิน เมื่อลูกค้าสิ้นสุดการใช้งาน ลูกค้าก็จะกดกริ่งเพื่อแจ้งพนักงานให้เข้ามาคำนวณค่าบริการทั้งหมด ซึ่งประกอบด้วย ค่าถาดลูกกอล์ฟ ถาดละ 30 บาท 30 ลูกต่อถาด ค่าเช่าไม้กอล์ฟ 60 บาท ค่าผู้สอน 100 บาท และค่าปรับกรณีเกิดอุปกรณ์ชำรุดเสียหาย 100 บาท โดยจัดทำใบเสร็จรับเงินแบบเขียนมือให้กับลูกค้า จากนั้นพนักงานจะรวบรวมยอดรายได้สรุปใส่สมุดบัญชีรายวันเพื่อให้เจ้าของร้านตรวจสอบในภายหลัง จากการดำเนินงานดังกล่าว พบว่าปัญหาสำคัญคือความล่าช้าและความผิดพลาดในการจัดการข้อมูล เนื่องจากต้องจดบันทึกลงสมุดด้วยมือทั้งหมด ทำให้เสี่ยงต่อการจองเลนซ้ำซ้อนและไม่สามารถตรวจสอบสถานะเลนว่างได้ในทันที อีกทั้งการคำนวณค่าบริการรวมถึงการออกใบเสร็จแบบเขียนมือยังเสี่ยงต่อความคลาดเคลื่อนของยอดเงิน นอกจากนี้เจ้าของร้านยังไม่สามารถตรวจสอบรายได้หรือสถิติการใช้งานได้แบบทันเวลา เนื่องจากต้องรอพนักงานสรุปยอดบัญชีรายวันเมื่อปิดร้านเพียงอย่างเดียว

3.1.1 Flowchart ระบบงานเดิม



ภาพประกอบที่ 3-1 Flowchart ระบบงานเดิม

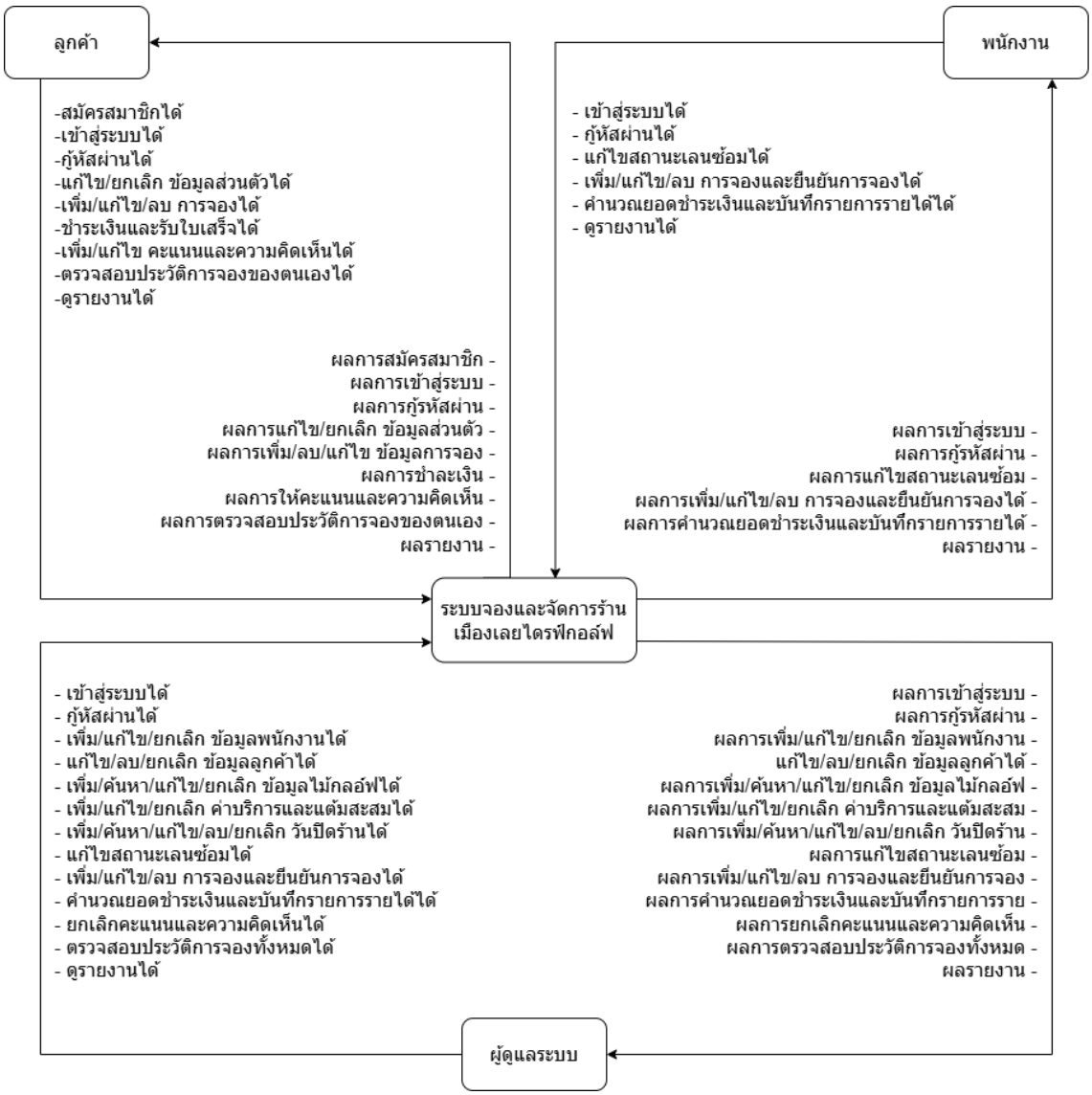
3.1.2 Flowchart กระบวนการชำระเงิน



ภาพประกอบที่ 3-2 Flowchart กระบวนการชำระเงิน

3.2 กระบวนการวิเคราะห์ระบบงานใหม่

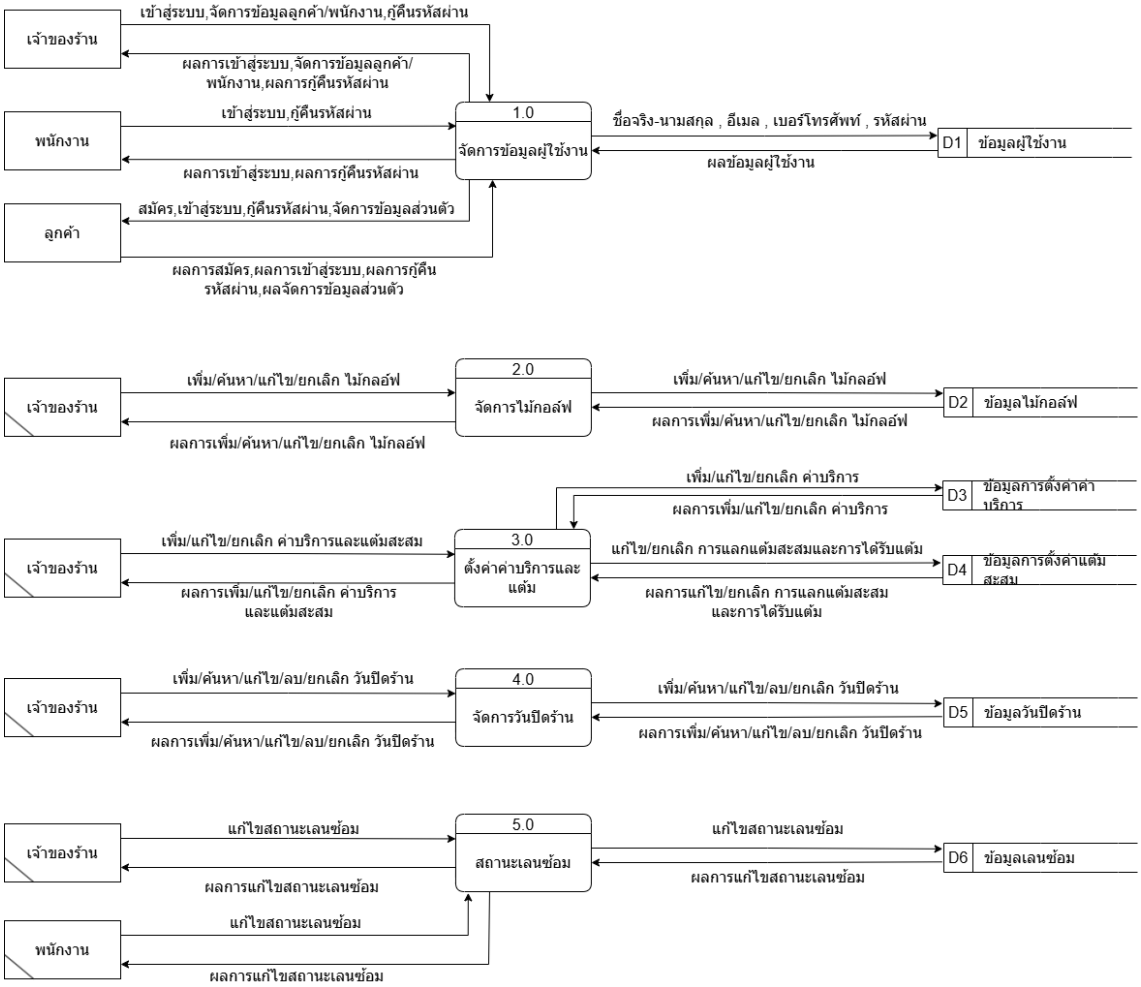
3.2.1 Context Diagram



ภาพประกอบที่ 3-3 Context Diagram

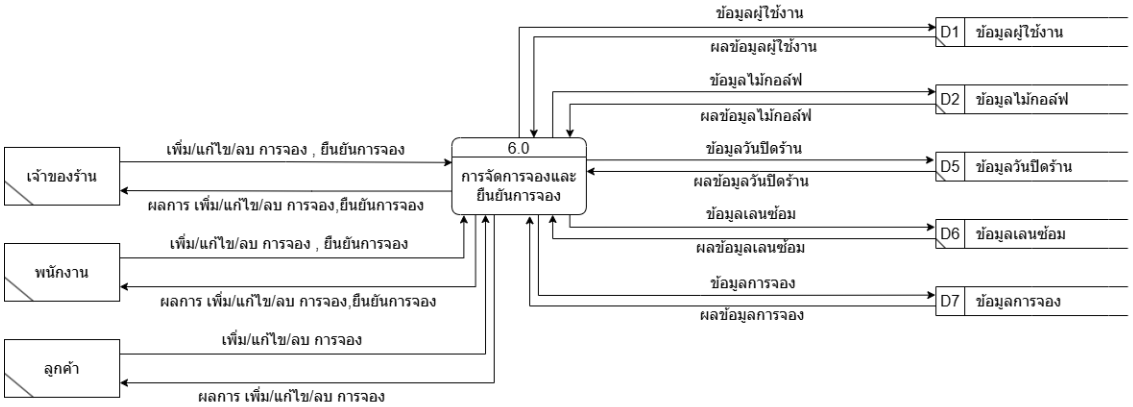
3.2.2 **Data Flow Diagram Level 0 ระบบจองและจัดการร้านเมืองเลยไดร์ฟกอล์ฟ**

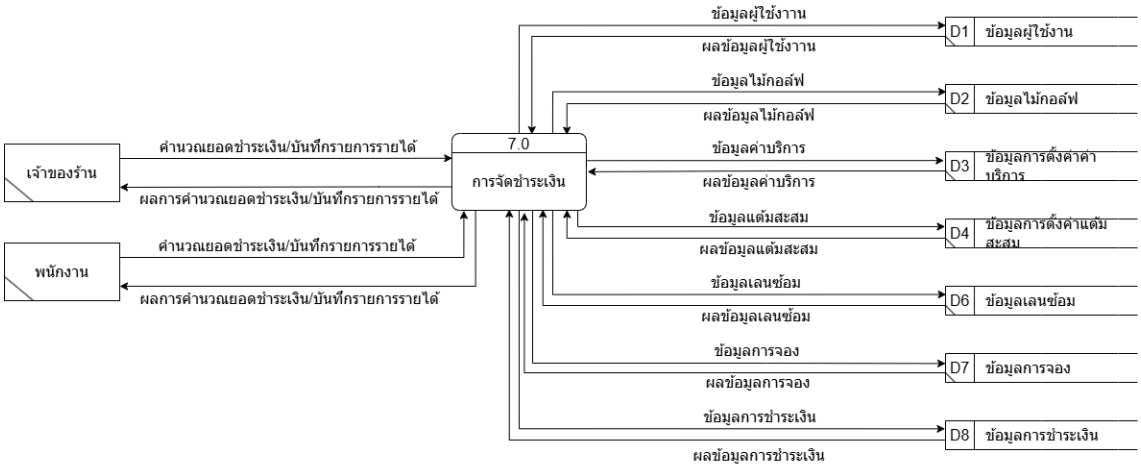
3.2.2.1 Data Flow Diagram Level 0 Process 1-5 จัดการข้อมูลผู้ใช้งาน จัดการไม้กอล์ฟ ตั้งค่าบริการและแต้ม จัดการวันปิดร้าน และสถานะเลนซ้อม



ภาพประกอบที่ 3-4 Data Flow Diagram Level 0 ระบบจองและจัดการร้านเมืองเลยไดร์ฟกอล์ฟ

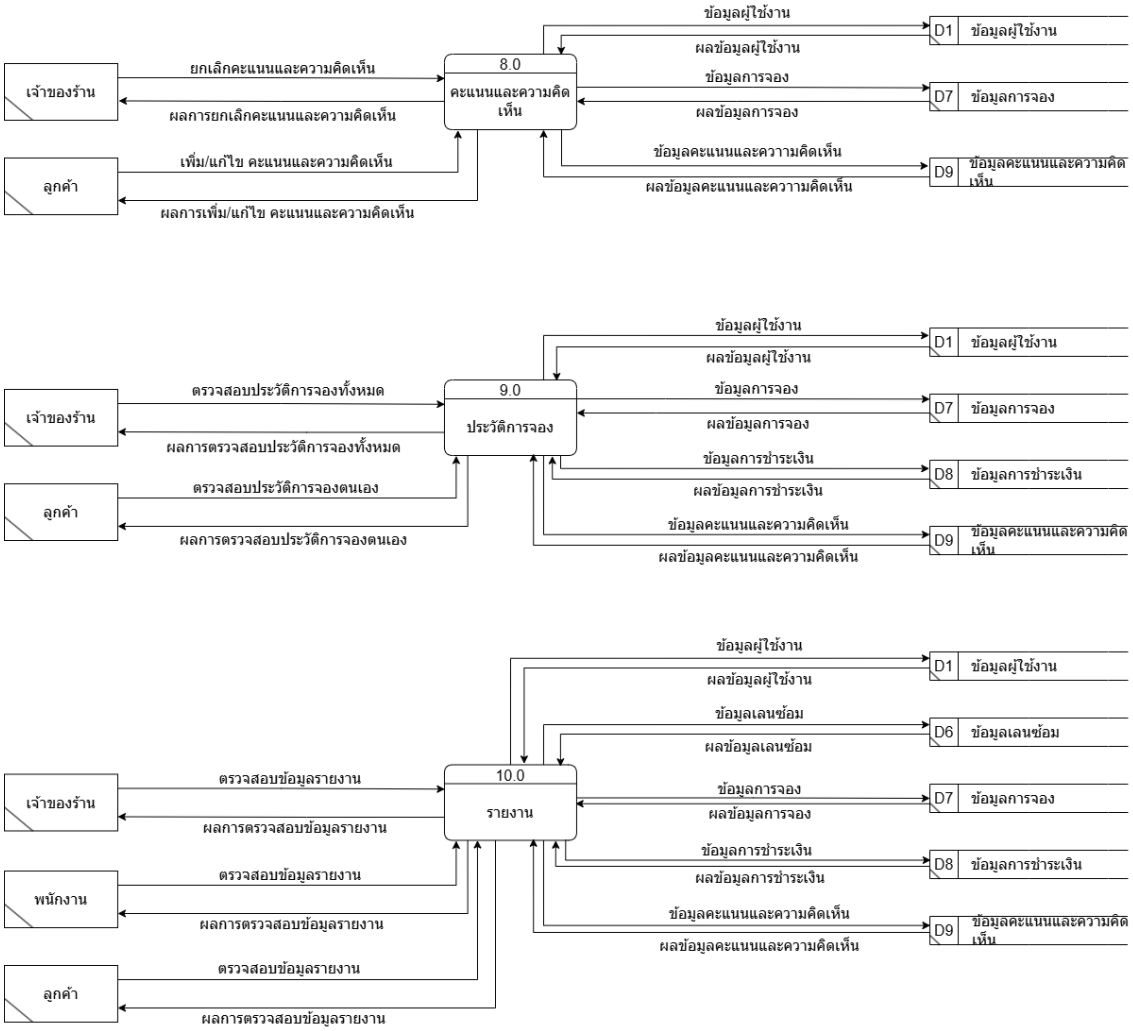
3.2.2.2 Data Flow Diagram Level 0 Process 6-7 จัดการการจองและยืนยันการจองและการจัดการชำระเงิน





ภาพประกอบที่ 3-5 Data Flow Diagram Level 0 ระบบจองและจัดการร้านเมืองเลยไดร์ฟกอล์ฟ (ต่อ)

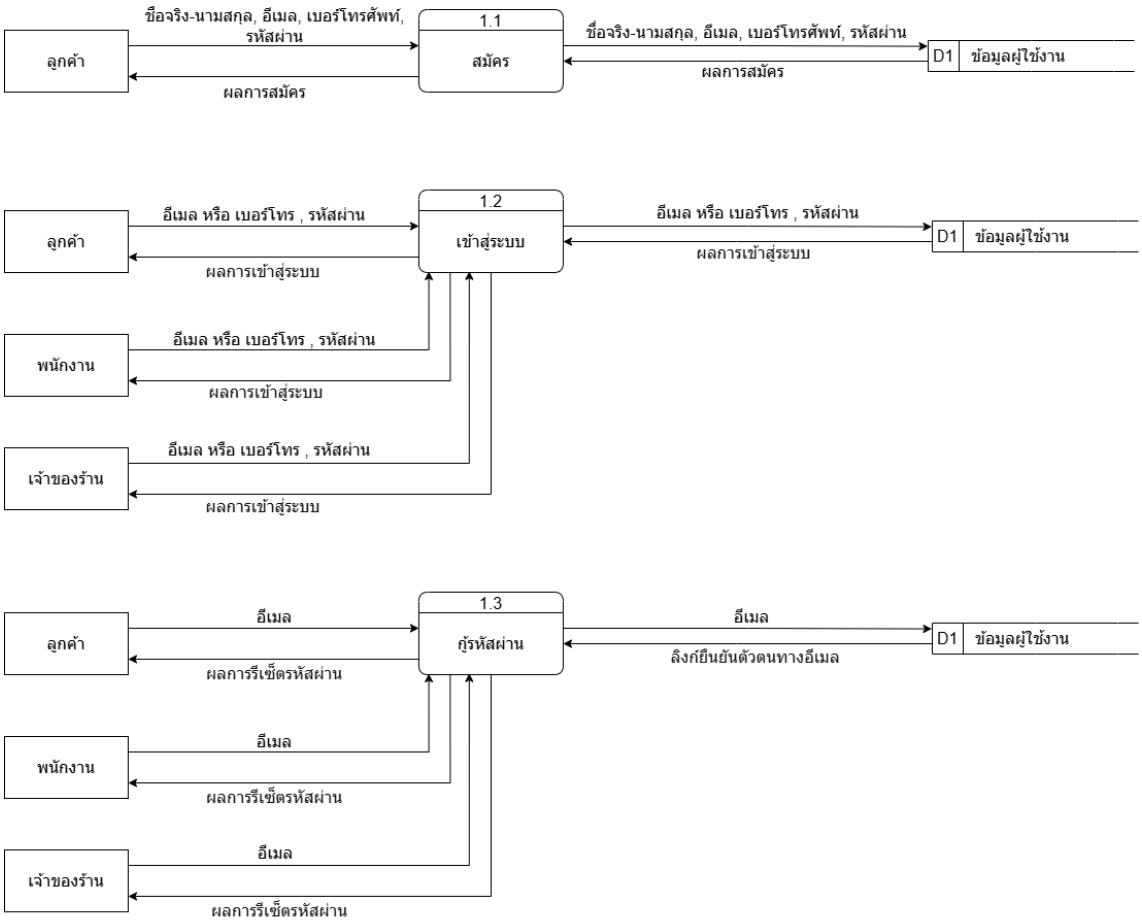
3.2.2.3 Data Flow Diagram Level 0 Process 8-10 คะแนนและความคิดเห็น ประวัติการจอง และรายงาน



ภาพประกอบที่ 3-6 Data Flow Diagram Level 0 ระบบจองและจัดการร้านเมืองเลยไดร์ฟกอล์ฟ (ต่อ)

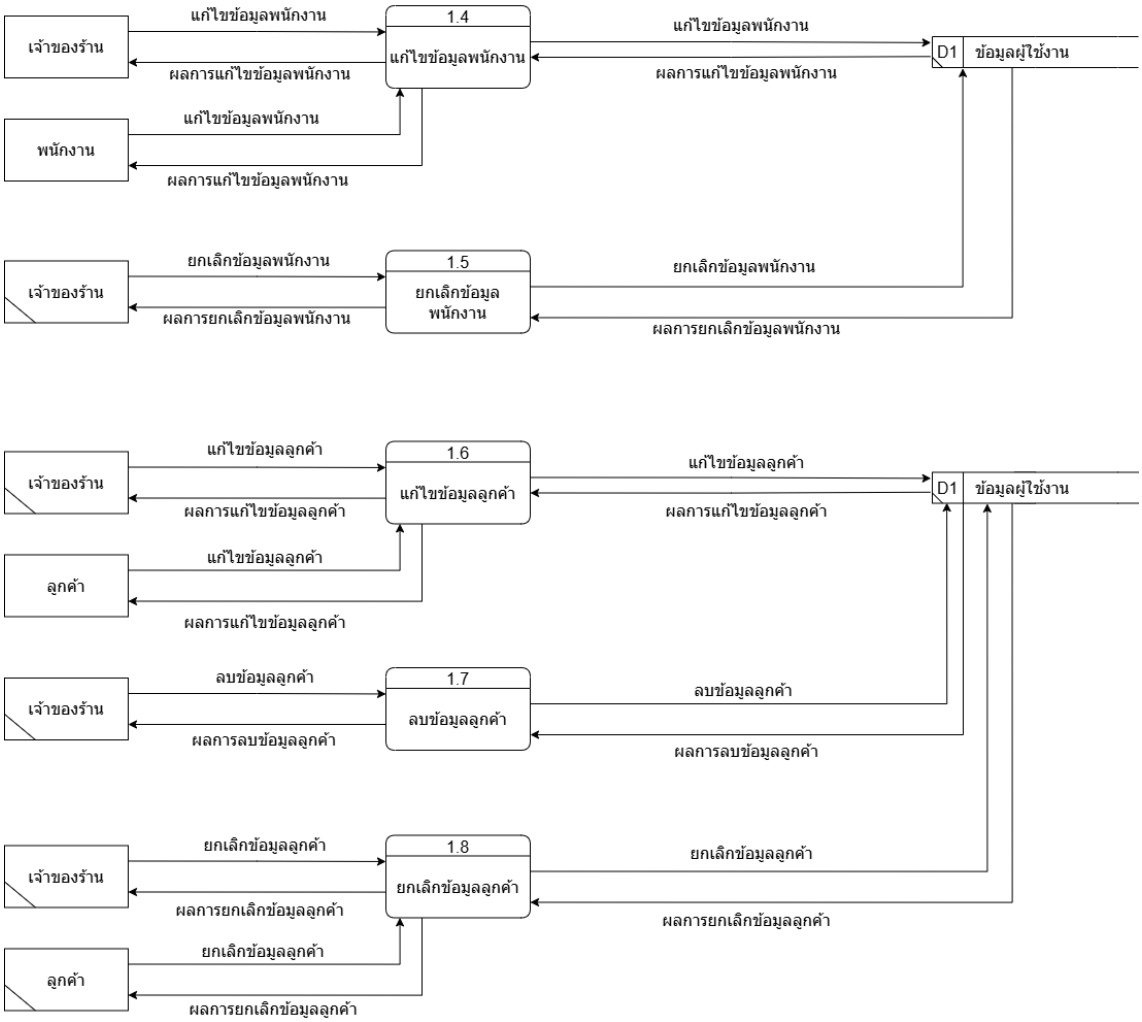
3.2.3 **Data Flow Diagram Level 1**

3.2.3.1 Data Flow Diagram Level 1 : Process 1.1- 1.3 สมัคร เข้าสู่ระบบ และกู้รหัสผ่าน



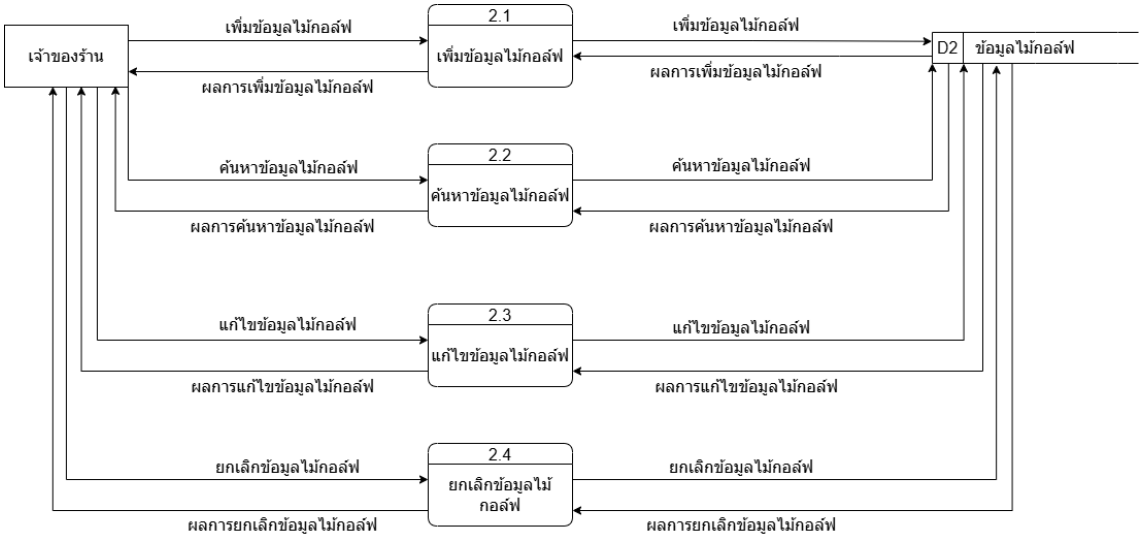
ภาพประกอบที่ 3-7 Data Flow Diagram Level 1 : Process 1 จัดการข้อมูลผู้ใช้งาน

3.2.3.2 Data Flow Diagram Level 1 : Process 1.4 – 1.8 แก้ไขข้อมูลพนักงาน ยกเลิกข้อมูลพนักงาน แก้ไขข้อมูลลูกค้า ลบข้อมูลลูกค้า และยกเลิกข้อมูลลูกค้า



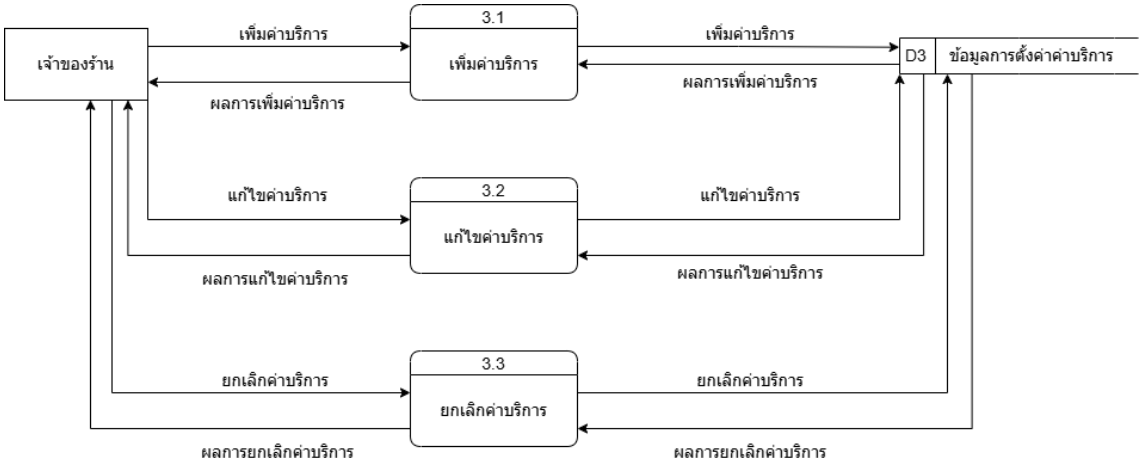
ภาพประกอบที่ 3-8 Data Flow Diagram Level 1 : Process 1 จัดการข้อมูลผู้ใช้งาน (ต่อ)

3.2.3.3 Data Flow Diagram Level 1 : Process 2.1 - 2.4 เพิ่มข้อมูลไม้กอล์ฟ ค้นหาข้อมูลไม้กอล์ฟ แก้ไขข้อมูลไม้กอล์ฟ ยกเลิกข้อมูลไม้กอล์ฟ



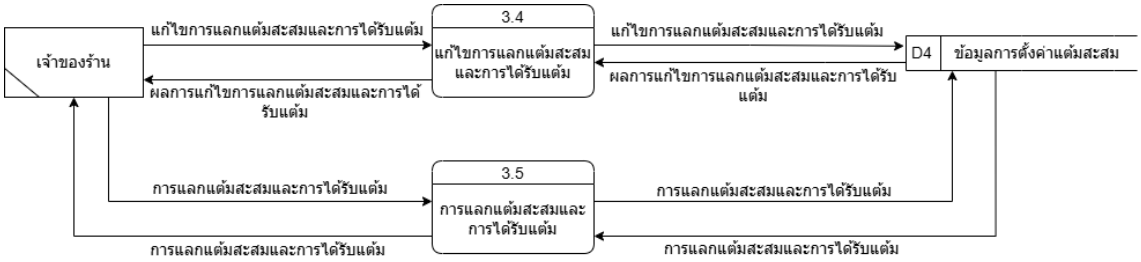
ภาพประกอบที่ 3-9 Data Flow Diagram Level 1 : Process 2 จัดการไม้กอล์ฟ

3.2.3.4 Data Flow Diagram Level 1 : Process 3.1 – 3.3 เพิ่มค่าบริการ แก้ไขค่าบริการ และยกเลิกค่าบริการ



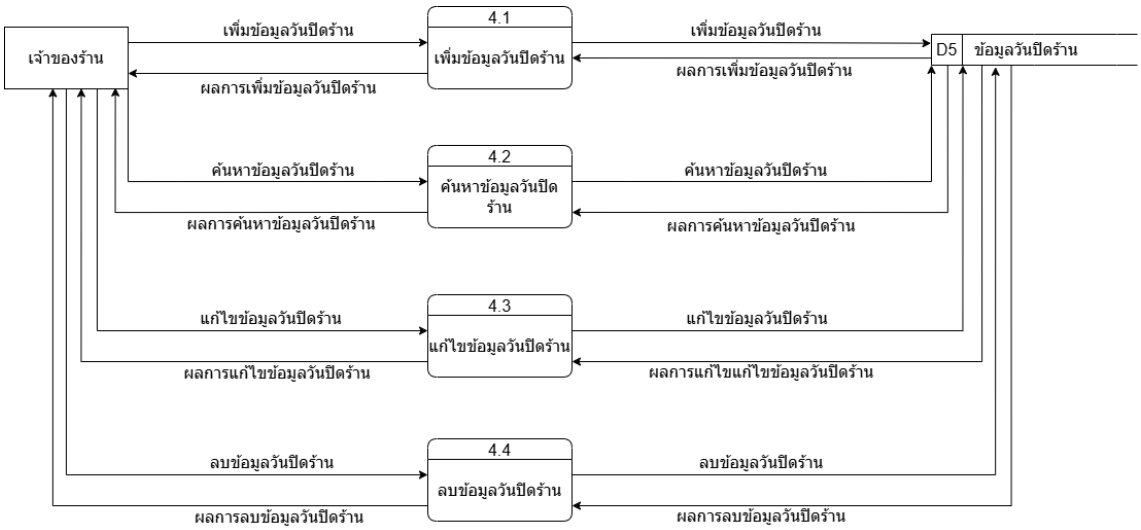
ภาพประกอบที่ 3-10 Data Flow Diagram Level 1 : Process 3 ตั้งค่าค่าบริการและแต้ม

3.2.3.5 Data Flow Diagram Level 1 : Process 3.4 – 3.5 แก้ไขการแลกแต้มสะสมและการได้รับแต้ม และยกเลิกการแลกแต้มสะสมและการได้รับแต้ม



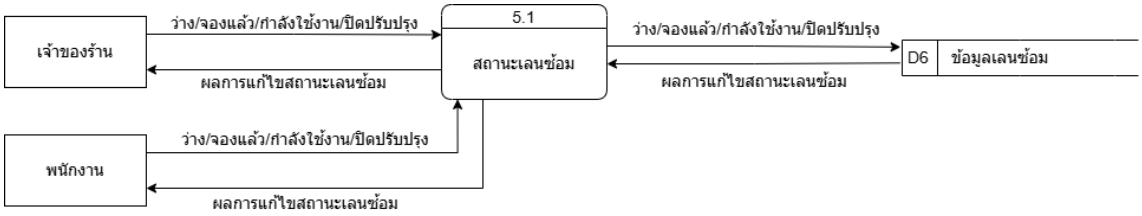
ภาพประกอบที่ 3-11 Data Flow Diagram Level 1 : Process 3 ตั้งค่าค่าบริการและแต้ม (ต่อ)

3.2.3.6 Data Flow Diagram Level 1 : Process 4.1 – 4.4 เพิ่มข้อมูลวันปิดร้าน ค้นหาข้อมูลวันปิดร้าน แก้ไขข้อมูลวันปิดร้าน และลบข้อมูลวันปิดร้าน



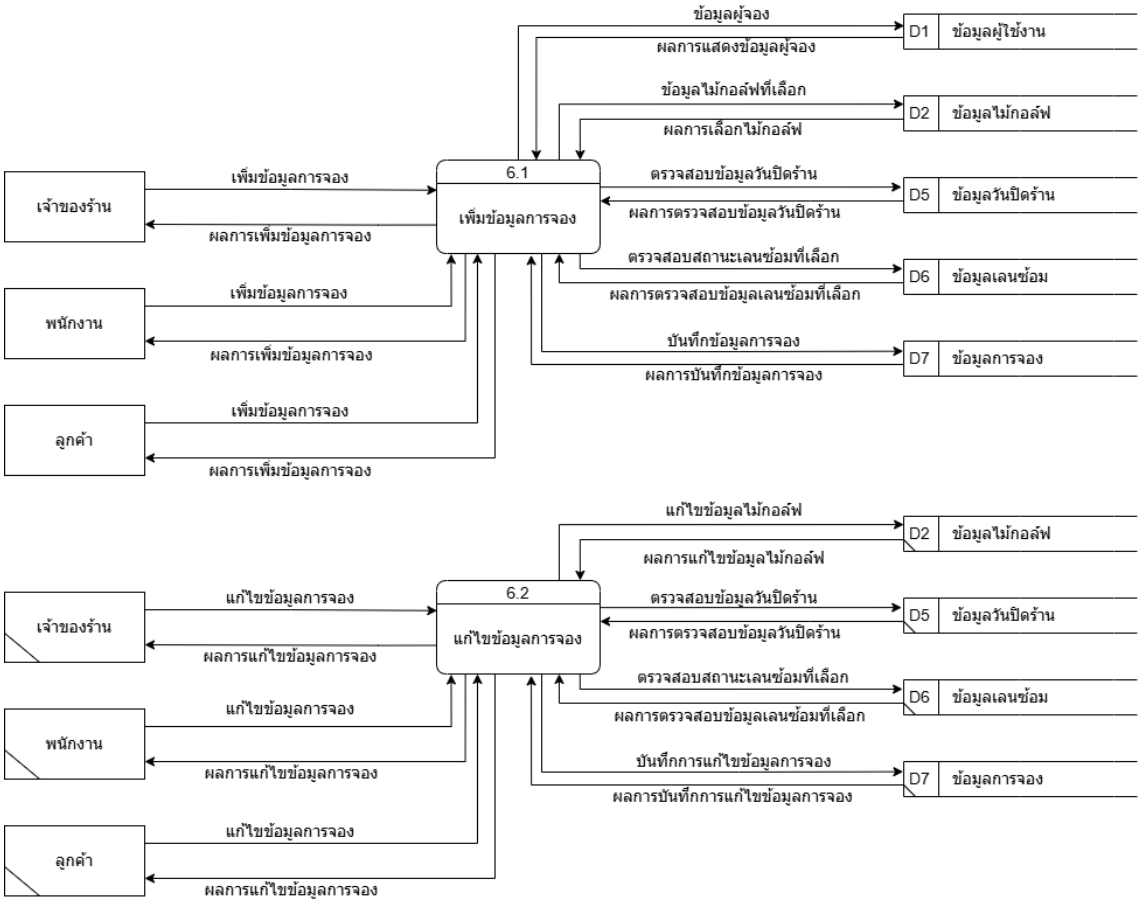
ภาพประกอบที่ 3-12 Data Flow Diagram Level 1 : Process 4 จัดการวันปิดร้าน

3.2.3.7 Data Flow Diagram Level 1 : Process 5.1 สถานะเลนซ้อม



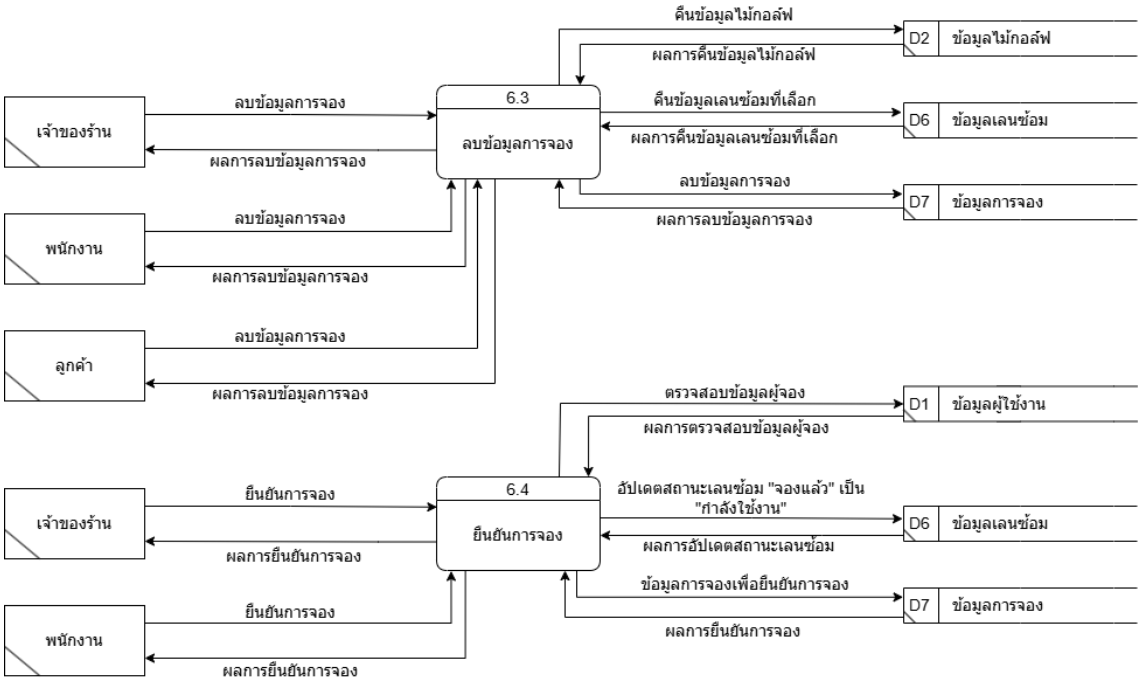
ภาพประกอบที่ 3-13 Data Flow Diagram Level 1 : Process 5 สถานะเลนซ้อม

3.2.3.8 Data Flow Diagram Level 1 : Process 6.1 – 6.2 เพิ่มข้อมูลการจอง และแก้ไขข้อมูลการจอง



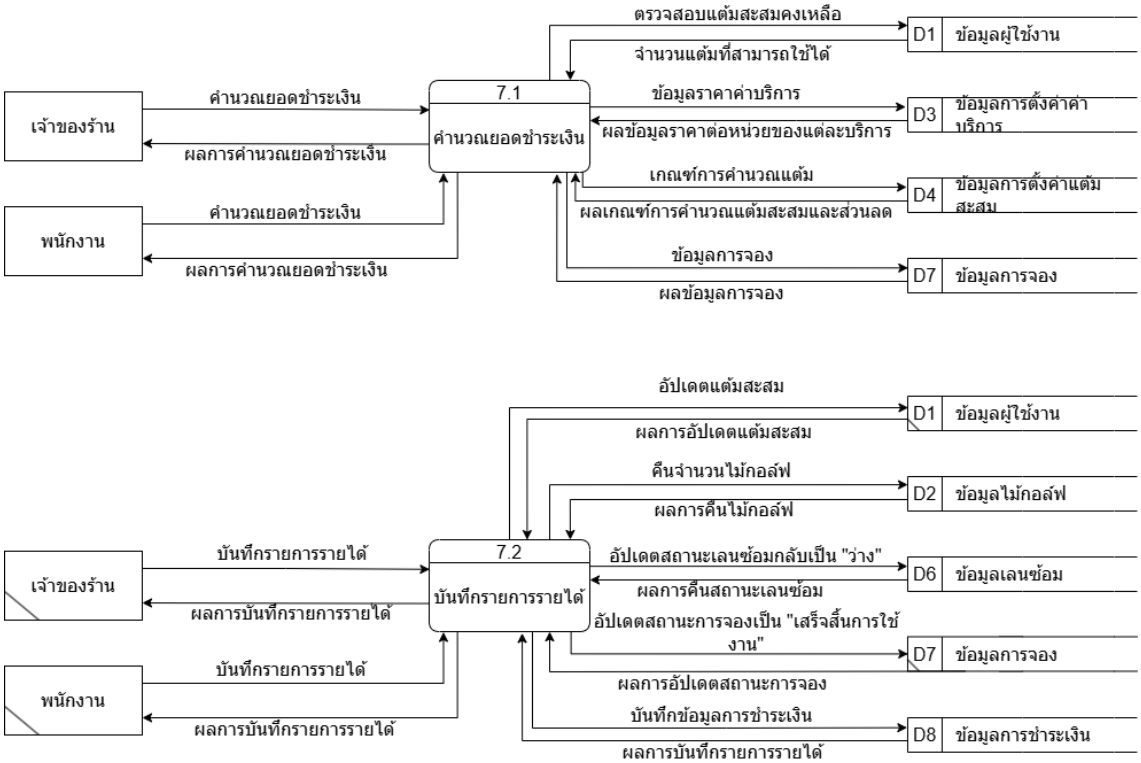
ภาพประกอบที่ 3-14 Data Flow Diagram Level 1 : Process 6 การจัดการและยืนยันการจอง

3.2.3.9 Data Flow Diagram Level 1 : Process 6.3 – 6.4 ลบข้อมูลการจองและยืนยันการจอง



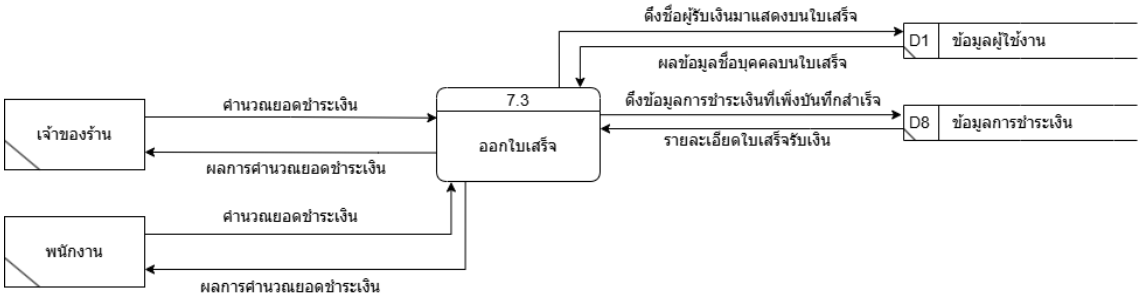
ภาพประกอบที่ 3-15 Data Flow Diagram Level 1 : Process 6 การจัดการและยืนยันการจอง (ต่อ)

3.2.3.10 Data Flow Diagram Level 1 : Process 7.1 – 7.2 คำนวณยอดชำระเงินและบันทึกรายการรายได้



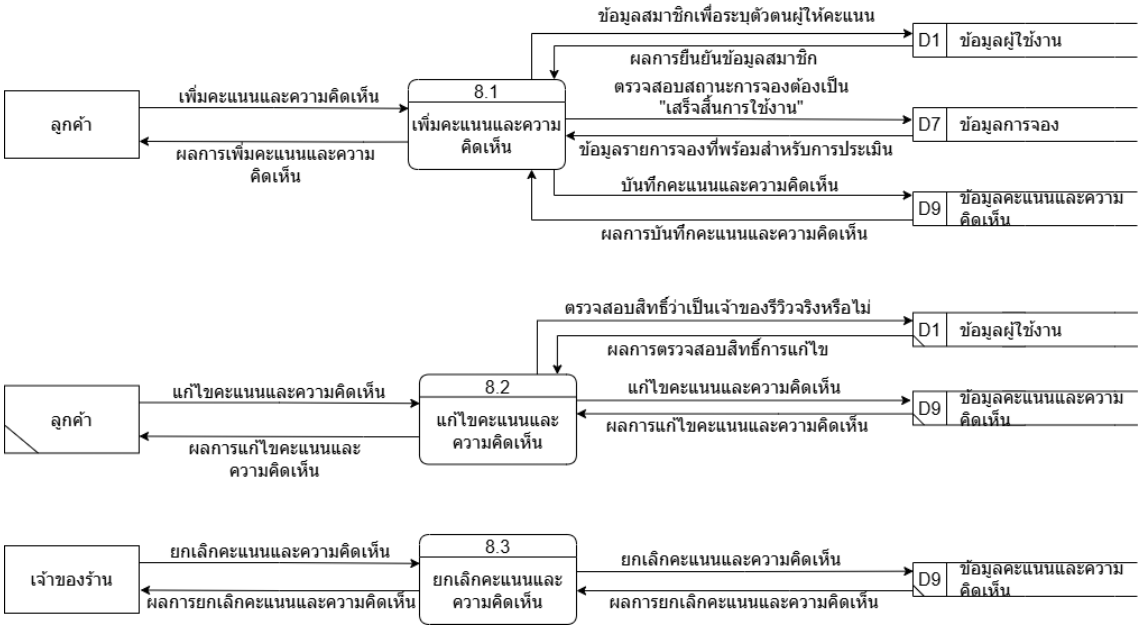
ภาพประกอบที่ 3-16 Data Flow Diagram Level 1 : Process 7 การจัดการชำระเงิน

3.2.3.11 Data Flow Diagram Level 1 : Process 7.3 ออกใบเสร็จ



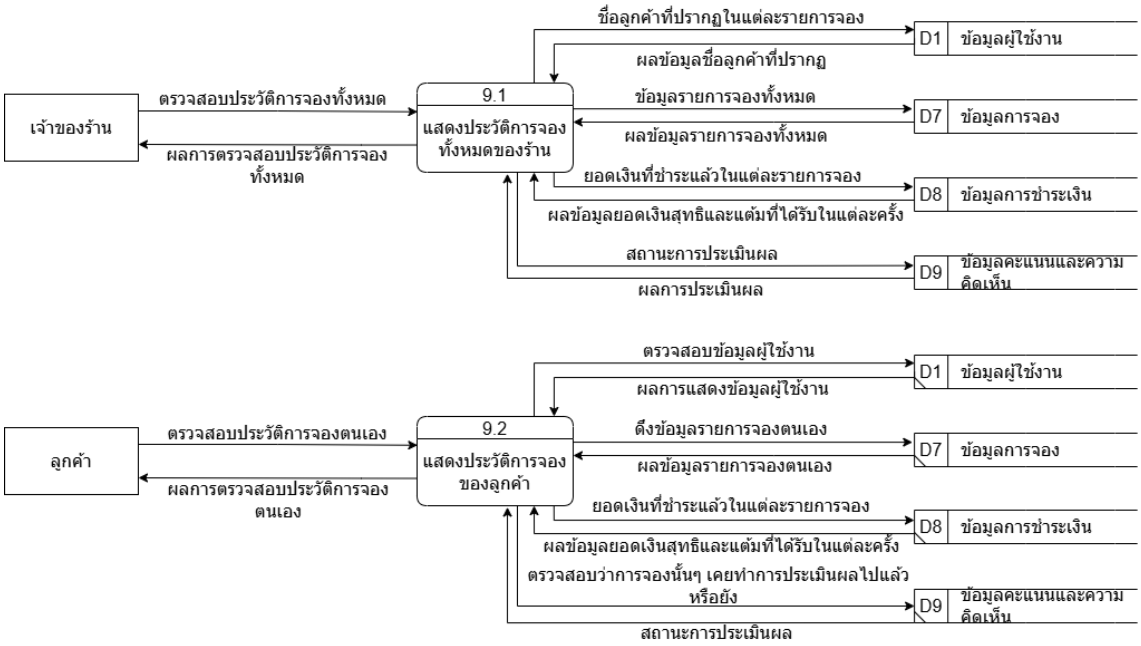
ภาพประกอบที่ 3-17 Data Flow Diagram Level 1 : Process 7 การจัดการชำระเงิน (ต่อ)

3.2.3.12 Data Flow Diagram Level 1 : Process 8.1 – 8.3 เพิ่มคะแนนและความคิดเห็น แก้ไขคะแนนและความคิดเห็น และยกเลิกคะแนนและความคิดเห็น



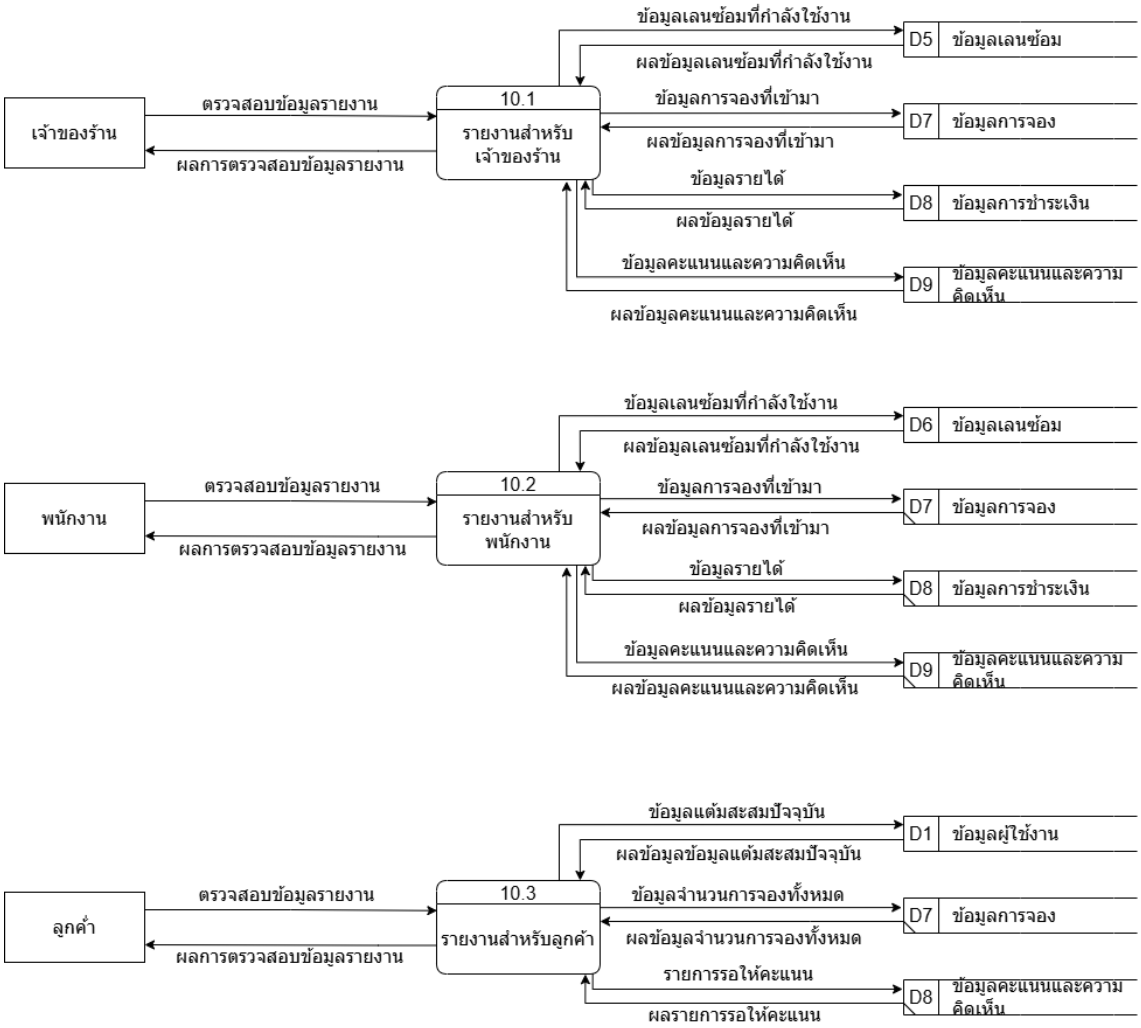
ภาพประกอบที่ 3-18 Data Flow Diagram Level 1 : Process 8 คะแนนและความคิดเห็น

3.2.3.13 Data Flow Diagram Level 1 : Process 9.1 – 9.2 แสดงประวัติการจองทั้งหมดของร้าน และแสดงประวัติการจองของลูกค้า



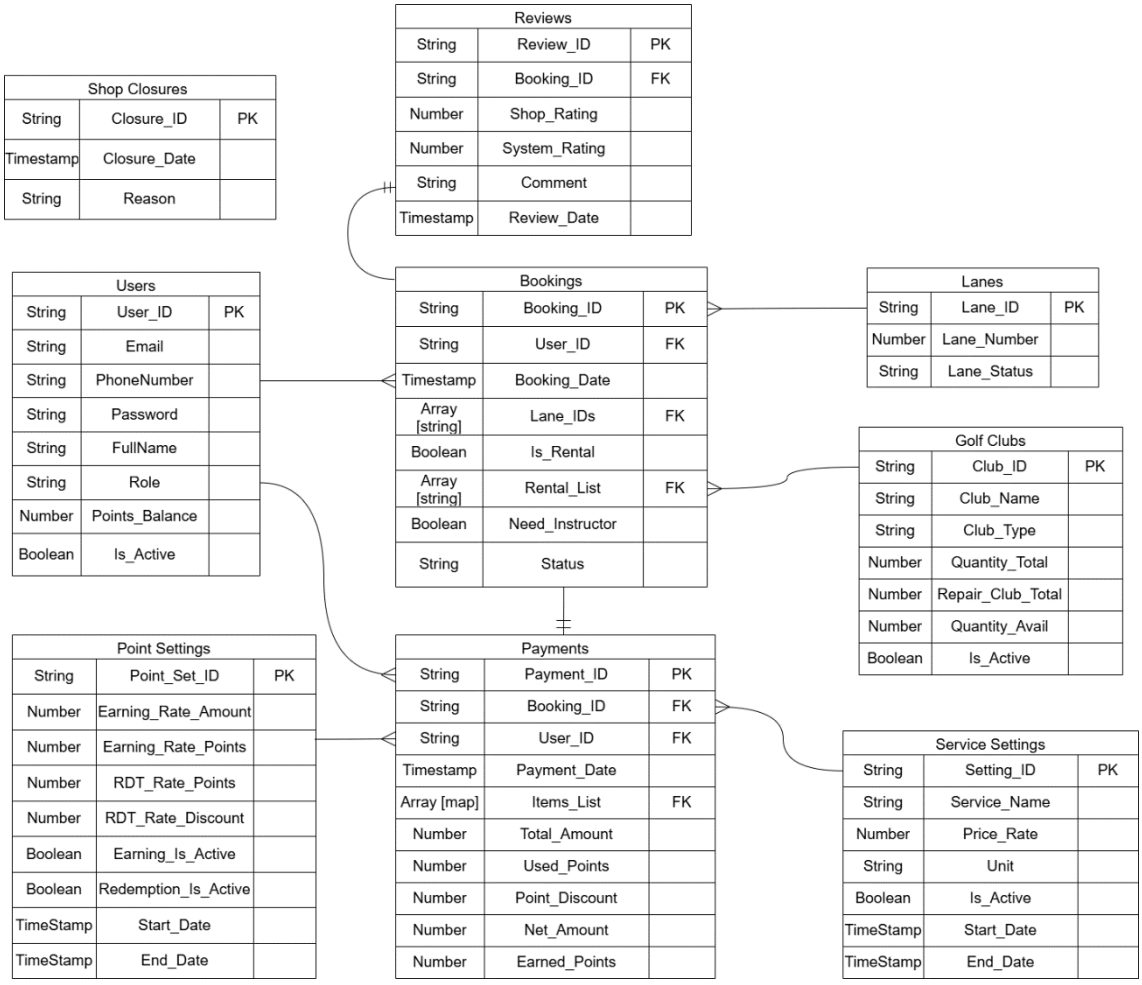
ภาพประกอบที่ 3-19 Data Flow Diagram Level 1 : Process 9 ประวัติการจอง

3.2.3.14 Data Flow Diagram Level 1 : Process 10.1 – 10.3 รายงานสำหรับเจ้าของร้าน รายงานสำหรับพนักงาน และรายงานสำหรับลูกค้า



ภาพประกอบที่ 3-20 Data Flow Diagram Level 1 : Process 10 รายงาน

3.2.4 **Entity Relationship Diagram**



ภาพประกอบที่ 3-21 แสดงภาพ Entity Relationship Diagram ระบบจองและจัดการร้านเมืองเลย ไดร์ฟกอล์ฟ

3.2.5 **พจนานุกรมข้อมูล (Data Dictionary)**

ตารางที่ 3-1 Users (ข้อมูลผู้ใช้งาน)

No	FieldName	Description	Data Type	Key	Reference
1	User_ID	รหัสผู้ใช้งาน	String	PK	
2	Email	อีเมล	String		
3	PhoneNumber	เบอร์โทรศัพท์	String		
4	Password	รหัสผ่าน	String		
5	FullName	ชื่อ-นามสกุล	String		
6	Role	ประเภทผู้ใช้ (เจ้าของร้าน/พนักงาน/สมาชิก/ลูกค้าวอคอิน)	String		
7	Points_Balance	แต้มสะสมคงเหลือ	Number		
8	Is_Active	สถานะการใช้งาน (ใช้งาน/ยกเลิกใช้งาน)	Boolean		

ตารางที่ 3-2 Golf Clubs (ข้อมูลไม้กอล์ฟ)

No	FieldName	Description	Data Type	Key	Reference
1	Club_ID	รหัสไม้กอล์ฟ	String	PK	
2	Club_Name	ชื่อไม้กอล์ฟ	String		
3	Club_Type	ประเภทของไม้	String		
4	Quantity_Total	จำนวนทั้งหมดที่มี	Number		
5	Repair_Club_Total	จำนวนทั้งหมดที่ชำรุด	Number		
6	Quantity_Avail	จำนวนที่ว่างพร้อมให้เช่า	Number		
7	Is_Active	สถานะการใช้งาน (ใช้งาน/ยกเลิกใช้งาน)	Boolean		

ตารางที่ 3-3 Service Settings (ข้อมูลการตั้งค่าค่าบริการ)

No	FieldName	Description	Data Type	Key	Reference
1	Service_ID	รหัสรายการบริการ	String	PK	
2	Service_Name	ชื่อบริการ	String		
3	Price_Rate	ราคาต่อหน่วย	Number		
4	Unit	หน่วยของค่าบริการ	String		
5	Is_Active	สถานะการใช้งาน (ใช้งาน/ยกเลิกใช้งาน)	Boolean		
6	Start_Date	วันที่เริ่มใช้ราคานี้	TimeStamp		
7	End_Date	วันที่สิ้นสุดการใช้ราคานี้	TimeStamp		

ตารางที่ 3-4 Point Settings (ข้อมูลการตั้งค่าแต้มสะสม)

No	FieldName	Description	Data Type	Key	Reference
1	Point_Set_ID	รหัสการตั้งค่า	String	PK	
2	Earning_Rate_Amount	ยอดจ่ายสะสม	Number		
3	Earning_Rate_Points	แต้มที่ได้รับ	Number		
4	RDT_Rate_Points	แต้มที่ใช้แลก	Number		
5	RDT_Rate_Discount	ส่วนลดที่ได้รับ	Number		
6	Earning_Is_Active	สถานะการใช้งานรับแต้ม (ใช้งาน/ยกเลิกใช้งาน)	Boolean		
7	Redemption_Is_Active	สถานะการใช้งานแลกแต้ม (ใช้งาน/ยกเลิกใช้งาน)	Boolean		
8	Start_Date	วันที่เริ่มใช้เกณฑ์แต้มสะสมนี้	TimeStamp		
9	End_Date	วันที่สิ้นสุดการใช้เกณฑ์แต้มสะสมนี้	TimeStamp		

ตารางที่ 3-5 Shop Closures (ข้อมูลวันปิดร้าน)

No	FieldName	Description	Data Type	Key	Reference
1	Closure_ID	รหัสวันปิดร้าน	String	PK	
2	Closure_Date	วันปิดร้าน	Timestamp		
3	Reason	เหตุผลในการปิด	String		

ตารางที่ 3-6 Lanes (ข้อมูลเลนซ้อม)

No	FieldName	Description	Data Type	Key	Reference
1	Lane_ID	รหัสเลนซ้อม	String	PK	
2	Lane_Number	หมายเลขของเลน	Number		
3	Lane_Status	สถานะเลน (ว่าง/จองแล้ว/กำลังใช้งาน/ปิดปรับปรุง)	String		

ตารางที่ 3-7 Bookings (ข้อมูลการจอง)

No	FieldName	Description	Data Type	Key	Reference
1	Booking_ID	รหัสการจอง	String	PK	
2	User_ID	รหัสผู้ใช้งาน	String	FK	Users(User_ID)
3	Booking_Date	วันเวลาที่จองใช้บริการ	Timestamp		
4	Lane_IDs	รายการรหัสเลนที่เลือก	Array [string]	FK	Lanes(Lane_ID)
5	Is_Rental	ต้องการเช่าไม้กอล์ฟ (ต้องการ/ไม่ต้องการ)	Boolean		
6	Rental_List	รายการไม้ที่เช่า	Array [string]	FK	Golf Clubs(Club_ID)
7	Need_Instructor	ต้องการผู้สอน (ต้องการ/ไม่ต้องการ)	Boolean		
8	Status	สถานะการจอง (จองแล้ว/กำลังใช้งาน/ยกเลิกการจอง/เสร็จสิ้นการใช้งาน)	String		

ตารางที่ 3-8 Payments (ข้อมูลการชำระเงิน)

No	FieldName	Description	Data Type	Key	Reference
1	Payment_ID	รหัสการชำระเงิน	String	PK	
2	Booking_ID	รหัสการจอง	String	FK	Bookings(Booking_ID)
3	User_ID	รหัสพนักงานที่คิดเงิน	String	FK	Users (User_ID)
4	Payment_Date	วันเวลาที่จ่ายเงิน	Timestamp		
5	Items_List	รายการบริการย่อย (รหัสบริการ, จำนวน, ราคาต่อหน่วย, ยอดรวมย่อย)	Array [map]	FK	Service_Settings (Service_ID)
6	Total_Amount	ยอดรวมราคาสินค้าก่อนหักส่วนลด	Number		
7	Used_Points	แต้มที่ลูกค้าใช้เป็นส่วนลด	Number		
8	Point_Discount	ส่วนลดที่ได้จากการใช้แต้ม	Number		
9	Net_Amount	ยอดเงินสุทธิที่ลูกค้าชำระจริง	Number		
10	Earned_Points	แต้มได้รับจากการชำระเงิน	Number		

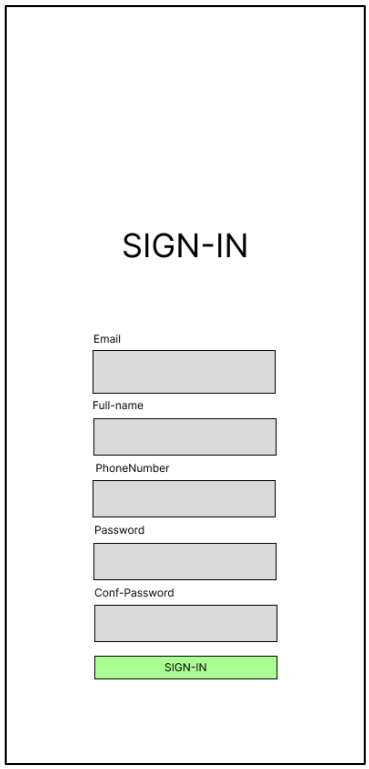
ตารางที่ 3-9 Reviews (ข้อมูลคะแนนและความคิดเห็น)

No	FieldName	Description	Data Type	Key	Reference
1	Review_ID	รหัสรายการประเมิน	String	PK	
2	Booking_ID	รหัสการจองที่ถูกประเมิน	String	FK	Bookings(Booking_ID)
3	Shop_Rating	คะแนนความพึงพอใจร้าน	Number		
4	System_Rating	คะแนนความพึงพอใจระบบ	Number		
5	Comment	ความคิดเห็นเพิ่มเติม	String		
6	Review_Date	วันที่ลูกค้าประเมิน	Timestamp		

3.3 การออกแบบหน้ายูสเซอร์อินเตอร์เฟซ (user interface design)

3.3.1 หน้าสมัครสมาชิกสำหรับลูกค้า

มีการจัดวางช่องสำหรับกรอกข้อมูลพื้นฐาน ได้แก่ อีเมล (Email), ชื่อ-นามสกุล (Full-name), เบอร์โทรศัพท์ (PhoneNumber) และการกำหนดรหัสผ่านพร้อมยืนยัน (Password/Confirm password) ไว้ในแนวตั้งกึ่งกลางหน้าจอ เพื่อให้ผู้ใช้งานใหม่สามารถสมัครสมาชิกเข้าสู่ระบบได้อย่างสะดวก โดยมีปุ่ม SIGN-IN ด้านล่างสำหรับยืนยันการลงทะเบียนข้อมูลเข้าสู่ฐานข้อมูล ดังภาพประกอบที่ 3-22



ภาพประกอบที่ 3-22 หน้าสมัครสมาชิกสำหรับลูกค้า

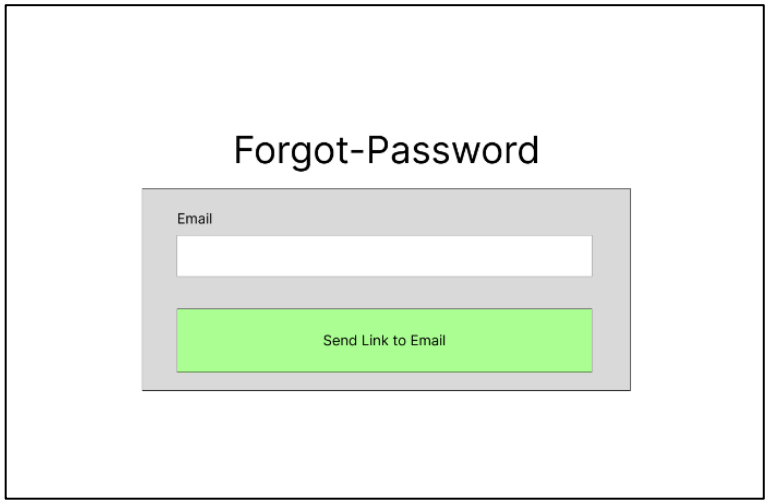
3.3.2 หน้าเข้าสู่ระบบผู้ใช้งาน

มีการจัดวางช่องกรอกชื่อผู้ใช้ (Email or PhoneNumber) และรหัสผ่าน (Password) ไว้บริเวณกึ่งกลางหน้าจอ การออกแบบเน้นความเรียบง่ายและชัดเจน เพื่อให้ผู้ใช้งานทั้งลูกค้าพนักงานและเจ้าของร้านสามารถเข้าสู่ระบบเพื่อจัดการการจองและบริการต่าง ๆ ได้อย่างสะดวกและรวดเร็ว นอกจากนี้ระบบยังรองรับการเข้าใช้งานผ่าน Google/Facebook เพื่ออำนวยความสะดวก และมีฟังก์ชันรองรับการสมัครสมาชิกใหม่ (Sign-in) รวมถึงการกู้คืนรหัสผ่าน (Forgot-Password) ในหน้าจอเดียวกัน ดังภาพประกอบที่ 3-23

ภาพประกอบที่ 3-23 หน้าเข้าสู่ระบบผู้ใช้งาน

3.3.3 หน้ากู้คืนรหัสผ่านผู้ใช้งาน

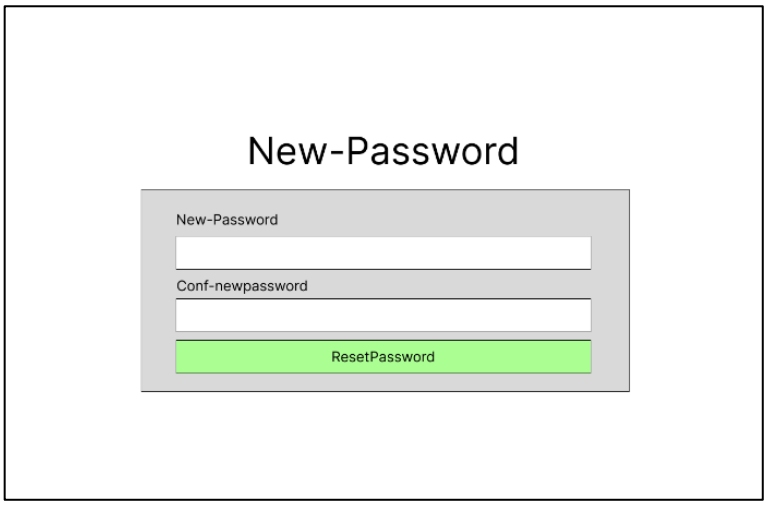
มีการจัดวางช่องสำหรับกรอกอีเมล (Email) ไว้บริเวณกึ่งกลางหน้าจอเพื่อให้ผู้ใช้งานที่ลืมรหัสผ่านสามารถระบุที่อยู่อีเมลที่ลงทะเบียนไว้ในระบบได้อย่างชัดเจน การออกแบบเน้นความเรียบง่ายเพื่อให้ผู้ใช้งานสามารถดำเนินการกู้คืนสิทธิ์การเข้าถึงได้อย่างรวดเร็ว โดยมีปุ่ม Send Link to Email สำหรับยืนยันการส่งลิงก์ตั้งค่ารหัสผ่านใหม่ไปยังอีเมลของผู้ใช้โดยตรง ดังภาพประกอบที่ 3-24



ภาพประกอบที่ 3-24 หน้ากู้คืนรหัสผ่านผู้ใช้งาน

3.3.4 หน้าตั้งค่ารหัสผ่านใหม่

มีการจัดวางช่องสำหรับกรอกรหัสผ่านใหม่ (New-Password) และช่องยืนยันรหัสผ่านใหม่ (Confirm -newpassword) ไว้บริเวณกึ่งกลางหน้าจอ เพื่อให้ผู้ใช้งานที่ขอรีเซ็ตรหัสผ่านสามารถกำหนดรหัสผ่านใหม่เข้าสู่ระบบได้อย่างปลอดภัย โดยมีปุ่ม ResetPassword สำหรับยืนยันการเปลี่ยนแปลงข้อมูล ดังภาพประกอบที่ 3-25



ภาพประกอบที่ 3-25 หน้าตั้งค่ารหัสผ่านใหม่

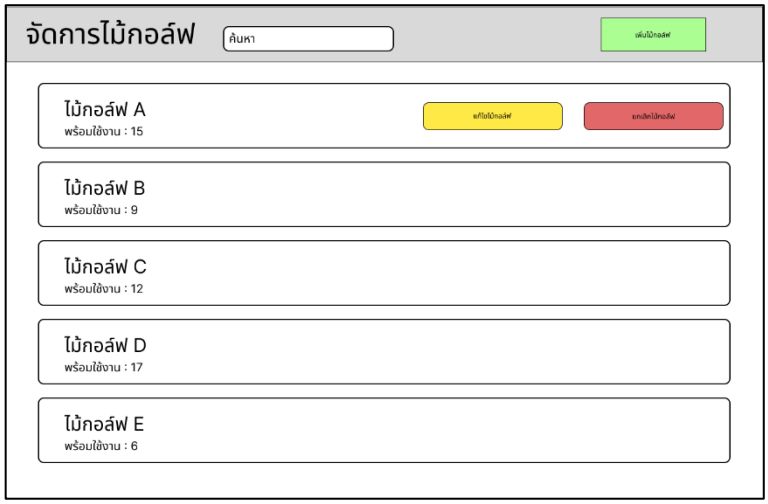
3.3.5 หน้าแดชบอร์ดเจ้าของร้านและพนักงาน

เป็นหน้าสรุปภาพรวมการดำเนินงานของสนามกอล์ฟในรูปแบบการ์ดข้อมูลที่เข้าใจง่าย ประกอบด้วยจำนวนการจองวันนี้, จำนวนเลนที่กำลังใช้งาน, รายได้ประจำวัน และคะแนนรีวิวเฉลี่ย นอกจากนี้ยังแสดงรายการ การจองล่าสุด และ ความคิดเห็นล่าสุด จากลูกค้า เพื่อให้เจ้าของร้านและพนักงานติดตามสถานะและคำแนะนำได้ทันทีในหน้าเดียว ดังภาพประกอบที่ 3-26

ภาพประกอบที่ 3-26 หน้าแดชบอร์ดเจ้าของร้านและพนักงาน

3.3.6 หน้าจัดการไม้กอล์ฟ

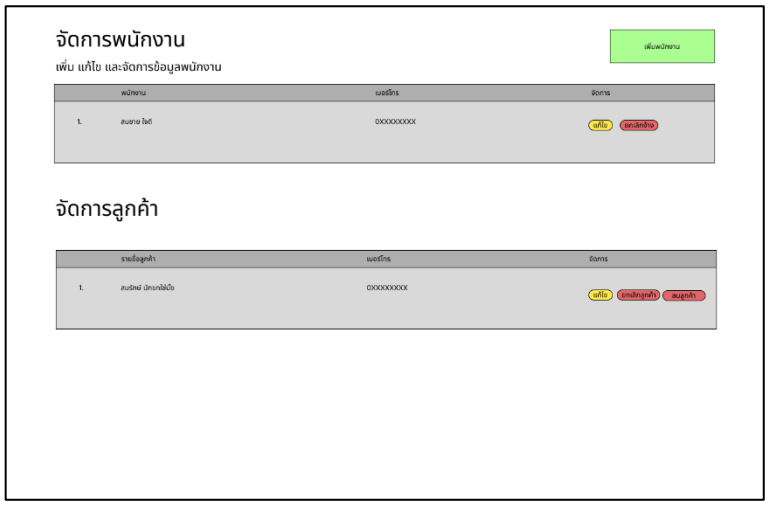
แสดงรายการไม้กอล์ฟทั้งหมดที่มีให้บริการพร้อมระบุจำนวนคงเหลือที่ พร้อมใช้งานในแต่ละประเภท มีฟังก์ชันการค้นหา และปุ่มเพิ่มรายการไม้กอล์ฟใหม่ (เพิ่มไม้กอล์ฟ) รวมถึงปุ่มสำหรับแก้ไขข้อมูลหรือยกเลิกรายการไม้กอล์ฟนั้นๆ เพื่อให้การบริหารจัดการสต็อกอุปกรณ์เป็นไปอย่างมีประสิทธิภาพ ดังภาพประกอบที่ 3-27



ภาพประกอบที่ 3-27 หน้าจัดการไม้กอล์ฟ

3.3.7 หน้าจัดการผู้ใช้งาน

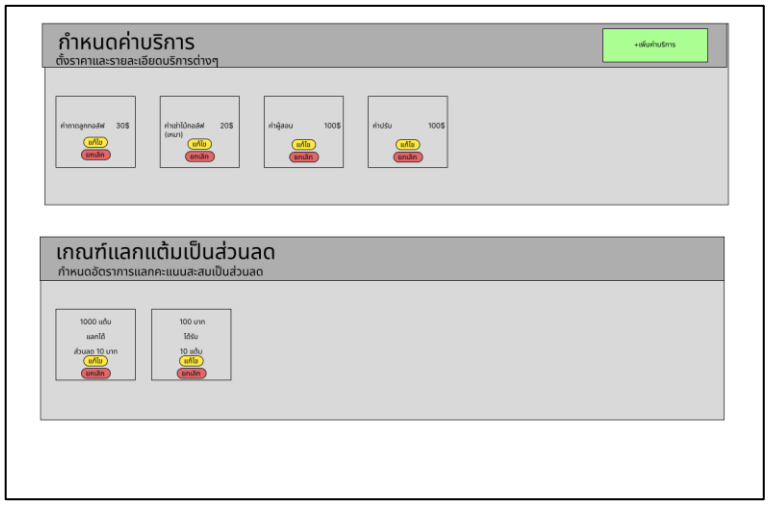
เป็นส่วนสำหรับการบริหารจัดการฐานข้อมูลบุคคลในระบบ โดยแบ่งออกเป็นส่วนของ จัดการพนักงาน และ จัดการลูกค้า อย่างชัดเจน แสดงรายละเอียดรหัส รายชื่อ และเบอร์โทรศัพท์ พร้อมปุ่มจัดการข้อมูล (แก้ไข/ยกเลิก) และปุ่มลบข้อมูล เพื่อให้เจ้าของร้านสามารถควบคุมสิทธิ์ และเข้าถึงข้อมูลผู้ใช้งานได้โดยตรง ดังภาพประกอบที่ 3-28



ภาพประกอบที่ 3-28 หน้าจัดการผู้ใช้งาน

3.3.8 หน้าตั้งค่าราคาค่าบริการและตั้งค่าแต้มสะสม

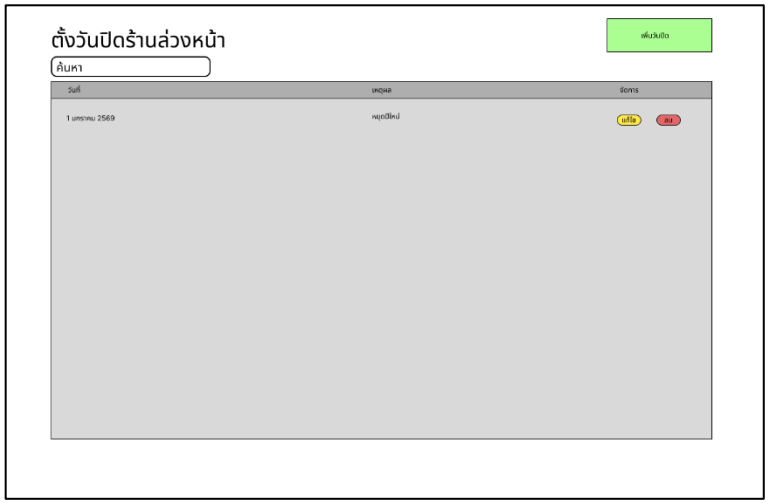
แสดงการตั้งค่าหลักของระบบใน 2 ส่วน คือ กำหนดค่าบริการ สำหรับตั้งราคาถาดลูกกอล์ฟ ค่าเช่าไม้ และค่าผู้สอน และส่วนของ เกณฑ์แลกแต้มเป็นส่วนลด เพื่อกำหนดอัตราการสะสมคะแนนและการแลกคะแนนเป็นเงินส่วนลด ช่วยให้เจ้าของร้านสามารถปรับเปลี่ยนราคาและโปรโมชั่นให้เหมาะสมกับสถานการณ์ได้ทันที ดังภาพประกอบที่ 3-29



ภาพประกอบที่ 3-29 หน้าตั้งค่าราคาค่าบริการและตั้งค่าแต้มสะสม

3.3.9 หน้าตั้งวันปิดร้านล่วงหน้า

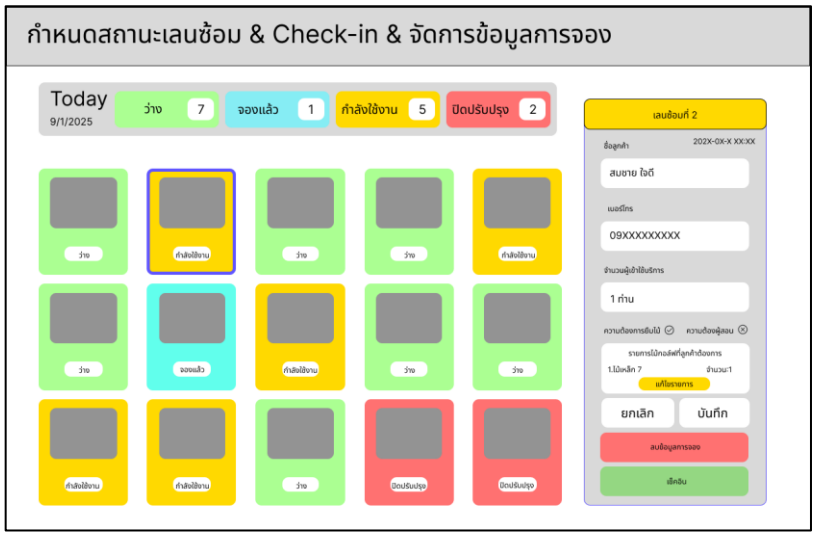
หน้าจอสำหรับจัดการวันหยุดหรือวันปิดปรับปรุงสนาม โดยแสดงรายการวันที่ปิดร้านพร้อมระบุเหตุผลประกอบ เช่น หยุดปีใหม่ มีระบบการค้นหาวันที่และปุ่มสำหรับเพิ่ม แก้ไข หรือลบวันปิดร้าน เพื่อป้องกันไม่ให้ลูกค้าทำรายการจองล่วงหน้าในวันที่สนามไม่เปิดให้บริการ ดังภาพประกอบที่ 3-30



ภาพประกอบที่ 3-30 หน้าตั้งวันปิดร้านล่วงหน้า

3.3.10 หน้าแก้ขสถานะเลนซ้อมและจัดการข้อมูลการจอง

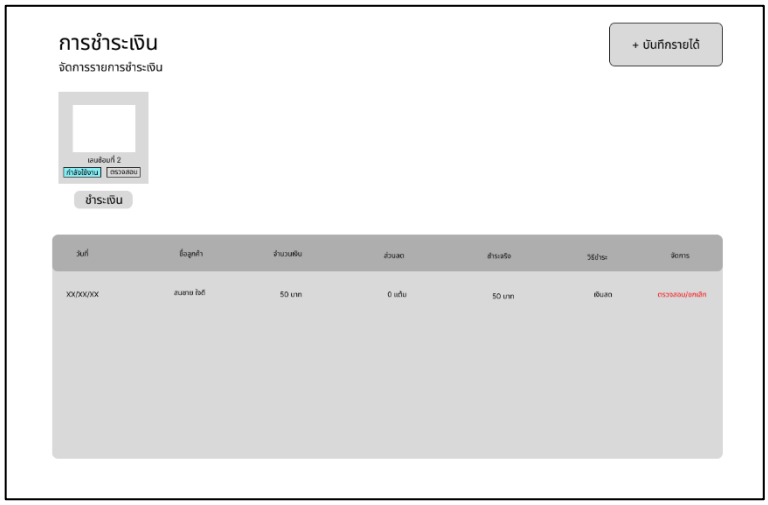
แสดงแผนผังเลนซ้อมทั้งหมดในรูปแบบสีที่จำแนกตามสถานะ (ว่าง, จองแล้ว, กำลังใช้งาน, ปิดปรับปรุง) เมื่อคลิกเลือกเลน ระบบจะแสดงแถบข้อมูลด้านข้างเพื่อระบุชื่อลูกค้าและรายละเอียดการจอง พร้อมปุ่มดำเนินการต่างๆ เช่น ยกเลิก, บันทึก หรือปุ่ม ยืนยัน สีเขียวเพื่อยืนยันการเข้าใช้งานจริงของลูกค้า ดังภาพประกอบที่ 3-31



ภาพประกอบที่ 3-31 หน้าแก้ขสถานะเลนซ้อมและจัดการข้อมูลการจอง

3.3.11 หน้ารายการรายได้

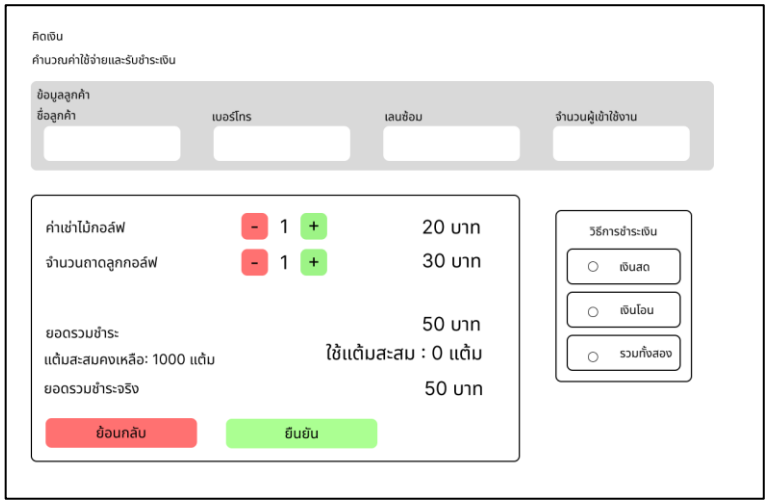
แสดงตารางสรุปรายการที่รอการชำระเงินและประวัติการรับเงินในแต่ละวัน ระบุชื่อลูกค้า จำนวนเงิน ส่วนลด และยอดชำระจริง โดยพนักงานสามารถกดปุ่ม ชำระเงิน จากรายการเลนที่ใช้งานเสร็จสิ้นเพื่อเข้าสู่ขั้นตอนการสรุปยอดค่าใช้จ่ายต่อไป ดังภาพประกอบที่ 3-32



ภาพประกอบที่ 3-32 หน้ารายการรายได้

3.3.12 หน้าชำระเงิน

เป็นหน้าจอสรุปยอดเงินสุดท้าย โดยระบบจะดึงข้อมูลค่าเช่าไม้และจำนวนถาดลูกกอล์ฟมาคำนวณอัตโนมัติ มีส่วนแสดงแต้มสะสมคงเหลือและช่องสำหรับระบุแต้มที่ต้องการใช้แลกเป็นส่วนลด พร้อมตัวเลือกวิธีการชำระเงิน (เงินสด, เงินโอน) เพื่อให้พนักงานกดยืนยันการรับชำระเงินและปิดรายการได้อย่างถูกต้องแม่นยำ ดังภาพประกอบที่ 3-33



ภาพประกอบที่ 3-33 หน้าชำระเงิน

3.3.13 หน้าใบเสร็จ

เป็นหน้าจอที่แสดงผลรายละเอียดการใช้บริการและยอดชำระเงินทั้งหมดของลูกค้า เพื่อให้การบันทึกรายการเสร็จสมบูรณ์ โดยมีรายละเอียดข้อมูล แสดงชื่อลูกค้า วันเวลาที่เข้าใช้งาน และเบอร์โทรศัพท์ติดต่ออย่างชัดเจน สรุปรายการบริการทั้งหมดที่เกิดขึ้น เช่น ค่าเช่าไม้กอล์ฟ และจำนวนถาดลูกกอล์ฟ พร้อมระบุราคาต่อหน่วยและราคารวมของแต่ละรายการ แสดงยอดรวมค่าบริการทั้งหมด การใช้แต้มสะสมแลกส่วนลด และยอดรวมที่ลูกค้าต้องชำระจริงหลังจากหักส่วนลดแล้ว มีปุ่ม เสร็จสิ้น สำหรับพนักงานใช้กดเพื่อยืนยันการรับชำระเงินและปิดรายการการจองนั้น ๆ มีปุ่มยกเลิกบันทึกรายการ สำหรับใช้ในกรณีที่พบข้อมูลผิดพลาดและต้องการแก้ไขข้อมูลก่อนทำการรับชำระเงิน ดังภาพประกอบที่ 3-34

ภาพประกอบที่ 3-34 หน้าใบเสร็จ

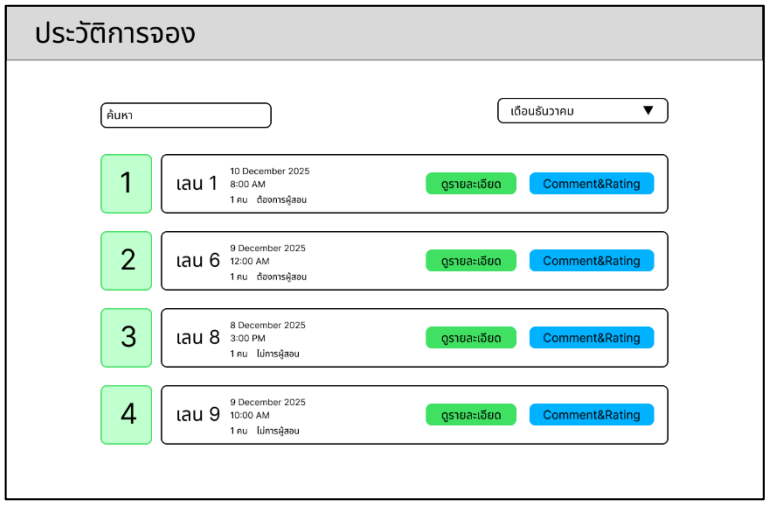
3.3.14 หน้าคะแนนและความคิดเห็น

หน้าจอสำหรับแสดงผลการประเมินจากผู้ใช้งาน โดยมีการแสดงค่าเฉลี่ยคะแนนร้านค้า คะแนนระบบ และจำนวนการรีวิวทั้งหมดในรูปแบบการ์ดด้านบน ส่วนด้านล่างจะแสดงรายการความคิดเห็นล่าสุดจากลูกค้า พร้อมวันที่และระดับคะแนนที่ได้รับ ช่วยให้เจ้าของร้านสามารถติดตามภาพรวมความพึงพอใจและจัดการข้อเสนอแนะผ่านปุ่ม ลบความคิดเห็น ได้หากพบข้อความที่ไม่เหมาะสม ดังภาพประกอบที่ 3-35

ภาพประกอบที่ 3-35 หน้าคะแนนและความคิดเห็น

3.3.15 หน้าประวัติการจอง

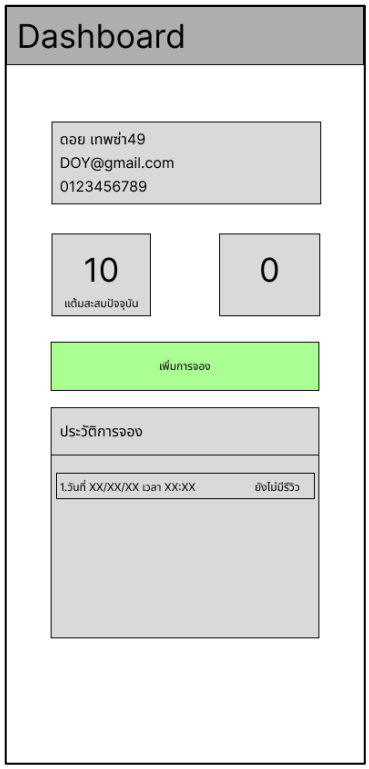
เป็นหน้าที่รวบรวมรายการจองย้อนหลังทั้งหมด โดยมีการแสดงลำดับ หมายเลขเลน วันเวลา และสถานะความต้องการผู้สอนในรูปแบบรายการ ผู้ใช้งานสามารถใช้ช่องค้นหาหรือเลือกดูตามเดือนที่ต้องการได้อย่างรวดเร็ว พร้อมปุ่ม ดูรายละเอียด เพื่อตรวจสอบข้อมูลการจองในอดีต และปุ่ม Comment&Rating สำหรับเข้าไปตรวจผลประเมินหลังจบการใช้บริการ ดังภาพประกอบที่ 3-36



ภาพประกอบที่ 3-36 หน้าประวัติการจอง

3.3.16 หน้าแดชบอร์ดลูกค้า

แสดงชื่อผู้ใช้งาน อีเมล และเบอร์โทรศัพท์ที่ใช้ในการลงทะเบียนเพื่อยืนยันตัวตน มีการแสดง แต้มสะสมปัจจุบัน ในรูปแบบการ์ดที่ชัดเจน เพื่อให้ลูกค้าทราบจำนวนคะแนนคงเหลือสำหรับนำไปใช้แลกเป็นส่วนลด มีปุ่ม เพิ่มการจอง เพื่อให้ลูกค้าสามารถเริ่มต้นทำรายการจองเลนซ้อมใหม่ได้ทันทีจากหน้าหลักของตนเอง และมีรายการประวัติการเข้าใช้งานล่าสุด พร้อมระบุวันที่ เวลา และสถานะการรีวิว เพื่อให้ลูกค้าสามารถตรวจสอบและกลับไปให้คะแนนความพึงพอใจย้อนหลังได้ ดังภาพประกอบที่ 3-37



ภาพประกอบที่ 3-37 หน้าแดชบอร์ดลูกค้า

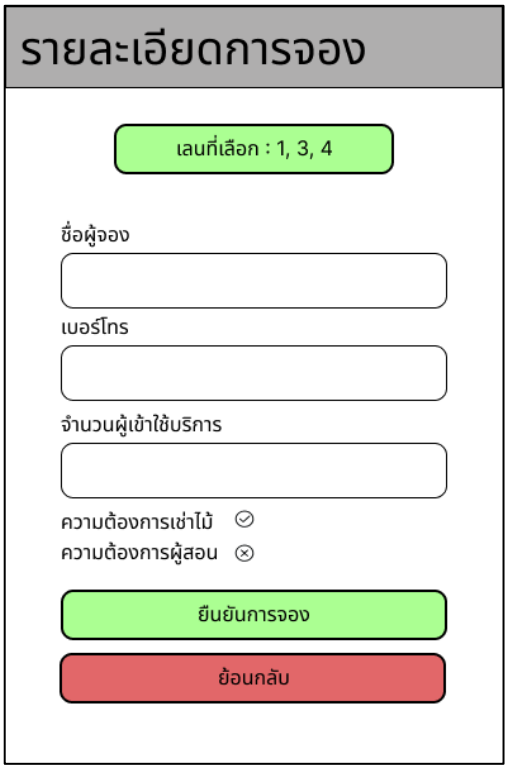
3.3.17 หน้าเพิ่มการจอง

มีการจัดวางส่วนเลือกวันและเวลาที่ต้องการเข้าใช้งานไว้ด้านบน พร้อมปุ่ม ตรวจสอบเลน เพื่ออัปเดตสถานะความว่างของสนามแบบ Real-time แผนผังเลนซ้อมจะแสดงสีตามสถานะที่แตกต่างกัน เช่น สีเขียว (ว่าง) และสีเหลือง/แดง (ไม่ว่าง) ช่วยให้ลูกค้าสามารถคลิกเลือกหมายเลขเลนที่ต้องการได้ด้วยตนเอง โดยมีสรุปหมายเลขเลนที่เลือกและปุ่ม ยืนยัน อยู่ด้านล่างสุดเพื่อเข้าสู่ขั้นตอนถัดไป ดังภาพประกอบที่ 3-38

ภาพประกอบที่ 3-38 หน้าเพิ่มการจอง

3.3.18 หน้ารายละเอียดการจอง

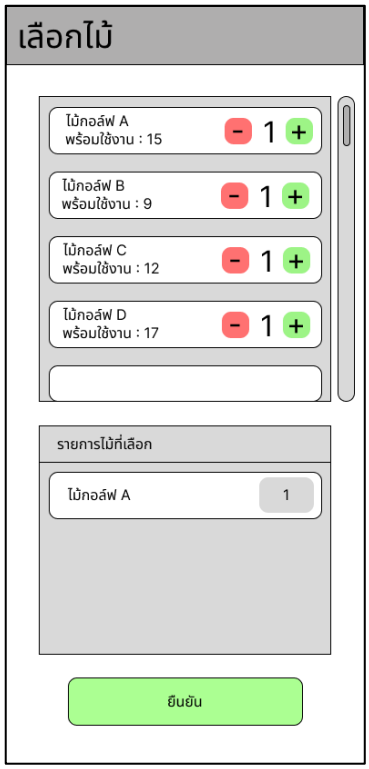
เป็นหน้าจอสำหรับตรวจสอบและระบุรายละเอียดการจองเพิ่มเติม โดยระบบจะทำการ ดึงข้อมูลชื่อผู้จองและเบอร์โทรศัพท์จากฐานข้อมูลสมาชิกมาแสดงผลให้โดยอัตโนมัติ เพื่อความรวดเร็วและลดความซ้ำซ้อนในการกรอกข้อมูล ลูกค้าเพียงแค่ระบุจำนวนผู้เข้าใช้บริการ และเลือกความต้องการพิเศษเพิ่มเติม เช่น ความต้องการเช่าไม้ หรือ ความต้องการผู้สอน พร้อมตรวจสอบหมายเลขเลนที่เลือกไว้ก่อนกดปุ่ม ยืนยันการจอง เพื่อบันทึกข้อมูลเข้าระบบ ดังภาพประกอบที่ 3-39



ภาพประกอบที่ 3-39 หน้ารายละเอียดการจอง

3.3.19 หน้าเลือกไม้กอล์ฟ

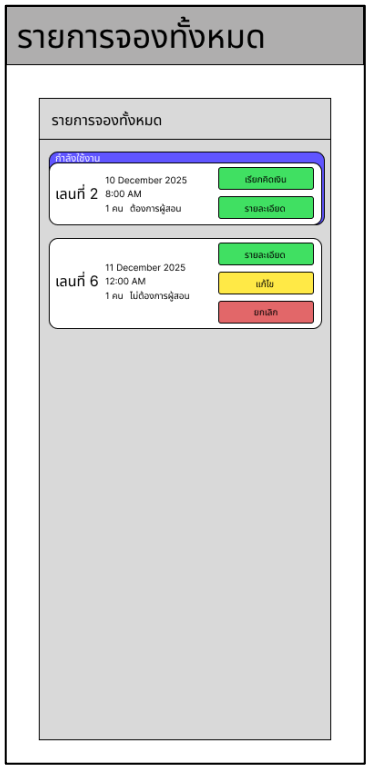
แสดงรายการประเภทไม้กอล์ฟที่มีให้บริการพร้อมระบุจำนวนคงเหลือที่ พร้อมใช้งาน ในขณะนั้น ลูกค้าสามารถกดปุ่มบวกหรือลบเพื่อระบุจำนวนไม้ที่ต้องการเช่าในแต่ละประเภทได้ตามต้องการ โดยระบบจะแสดงรายการไม้ที่เลือกไว้ในส่วนล่างของหน้าจอเพื่อให้ลูกค้าตรวจสอบความถูกต้องก่อนทำการบันทึกรายการ ดังภาพประกอบที่ 3-40



ภาพประกอบที่ 3-40 หน้าเลือกไม้กอล์ฟ

3.3.20 หน้ารายการจองของลูกค้า

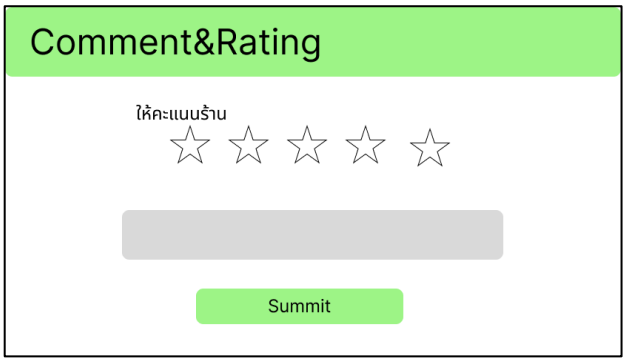
รวบรวมรายการจองของลูกค้าโดยแบ่งสถานะอย่างชัดเจน เช่น รายการที่ กำลังใช้งาน จะมีปุ่ม เรียกคิดเงิน เพื่อความรวดเร็ว ส่วนรายการจองอื่น ๆ จะแสดงรายละเอียดวันเวลา หมายเลขเลน และเงื่อนไขผู้สอน พร้อมตัวเลือกให้ลูกค้าสามารถกดดูรายละเอียด แก้ไข หรือยกเลิกการจองได้ด้วยตนเอง ดังภาพประกอบที่ 3-41



ภาพประกอบที่ 3-41 หน้ารายการจองของลูกค้า

3.3.21 หน้าการให้คะแนนและความคิดเห็น

เป็นหน้าจอที่ให้ลูกค้าสามารถประเมินความพึงพอใจหลังจากเสร็จสิ้นการใช้บริการ โดยมีรายละเอียดคือ ให้คะแนนระบบ ในรูปแบบสัญลักษณ์ดาว (1-5 ดาว) เพื่อให้ลูกค้าประเมินแยกตามส่วนงานได้ ส่วนความคิดเห็น (Comment) มีช่องรับข้อมูลแบบข้อความสำหรับให้ลูกค้าเขียนคำแนะนำหรือข้อเสนอแนะเพิ่มเติมได้ตามต้องการ และมีปุ่ม Summit ด้านล่างสำหรับยืนยันการส่งผลการประเมินเข้าสู่ระบบ ดังภาพประกอบที่ 3-42



ภาพประกอบที่ 3-42 หน้าการให้คะแนนและความคิดเห็น

บรรณานุกรม

จอช ฟรูห์ลิง. (2567, 1 มีนาคม). *What is JavaScript? The full-stack programming language*. Infoworld. https://www.infoworld.com/article/2263137/what-is-javascript-the-full-stack-programming-language.html

จีราวุธ วารินทร์. (2565). ติดตั้ง Firebase CLI. ใน พันจันทร์ ธนวัฒนเสถียร (บ.ก.), *ต่อยอดพัฒนาเว็บแอพพลิเคชันด้วย React + Firebase* (น. 220). บริษัท รีไวว่า จำกัด.

จีราวุธ วารินทร์. (2565). ทบทวนเกี่ยวกับ SQL และ NOSQL. ใน พันจันทร์ ธนวัฒนเสถียร (บ.ก.), *ต่อยอดพัฒนาเว็บแอพพลิเคชันด้วย React + Firebase* (น. 2). บริษัท รีไวว่า จำกัด.

จีราวุธ วารินทร์. (2565). บริการ Firebase Hosting. ใน พันจันทร์ ธนวัฒนเสถียร (บ.ก.), *ต่อยอดพัฒนาเว็บแอพพลิเคชันด้วย React + Firebase* (น. 219). บริษัท รีไวว่า จำกัด.

จีราวุธ วารินทร์. (2565). พื้นฐานเกี่ยวกับ Firebase Authentication. ใน พันจันทร์ ธนวัฒนเสถียร (บ.ก.), *ต่อยอดพัฒนาเว็บแอพพลิเคชันด้วย React + Firebase* (น. 125). บริษัท รีไวว่า จำกัด.

จีราวุธ วารินทร์. (2566). คำนำ. ใน จีราวุธ วารินทร์ (บ.ก.), *โมเดิร์น JavaScript เก่งได้ใน 30 วัน ฉบับ Node.js และ MongoDB* (น. 3). บริษัท ธิงค์ บียอนด์ บุ๊คส์ จำกัด.

จีราวุธ วารินทร์. (2566). รู้จักกับ Express. ใน จีราวุธ วารินทร์, *โมเดิร์น JavaScript เก่งได้ใน 30 วัน ฉบับ Node.js และ MongoDB* (น. 410). บริษัท ธิงค์ บียอนด์ บุ๊คส์ จำกัด.

จีราวุธ วารินทร์. (2566). สร้างแอปพลิเตชันในฝั่ง Backend ด้วย Node.js. ใน จีราวุธ วารินทร์ (บ.ก.), *โมเดิร์น JavaScript เก่งได้ใน 30 วัน ฉบับ Node.js และ MongoDB* (น. 384). บริษัท ธิงค์ บียอนด์ บุ๊คส์ จำกัด.

เจษฎา แสงขาว. (2561, 16 พฤษภาคม). *[Firebase] คืออะไร มาดูวิธีสร้าง Project และทำความรู้จักกับ Firebase*. Medium. https://medium.com/jed-ng/firebase-คืออะไร-มาดูวิธีสร้าง-project-และทำความรู้จักกับ-firebase-d48bfac67b14

ชรินทร เรืองไลคราม. (2566, 29 กันยายน). *Tailwind CSS เฟรมเวิร์กที่ช่วยให้ Dev ทำงานง่ายขึ้น*. Morpshosis. https://morphos.is/th/blog/tailwind-css-a-framework-that-makes-dev-work-easier

ซาฮิล ดาห์วัน. (2561, 13 พฤษภาคม). *What's new in CSS 3*. Medium. https://medium.com/beginners-guide-to-mobile-web-development/whats-new-in-css-3-dcd7fa6122e1

ดร. ศรีนิวาส จาการ์ลาปูดี. (2566, 29 มิถุนายน). *NoSQL Databases: Empowering Modern Data Management*. Linkedin. https://www.linkedin.com/pulse/nosql-databases-empowering-modern-data-management-jagarlapoodi

บรรณานุกรม(ต่อ)

ดาวิด คีลบาซา. (2566, 25 สิงหาคม). *How Does Tailwind Work? Understanding Tailwind CSS and Why Everyone is Using it*. Elpassion. https://www.elpassion.com/ blog/ tailwind-css-how-does-it-work

ทักษิณ น้อยบุดดี และ รัฐกรณ์ ศรีจันทร์. (2567). *ระบบจองสนามกีฬาออนไลน์ กรณีศึกษาคณะวิทยาการสารสนเทศ มหาวิทยาลัยมหาสารคาม.* คณะวิทยาการสารสนเทศ มหาวิทยาลัยมหาสารคาม.

ทีเจ โฮโลเวย์ชุก. (2553). *Express.js Framework.* expressjs. https://expressjs.com

เบน มัสสัน. (2568, 11 มิถุนายน). *Chart.js Library. Chart.js*. Mussonindustrial. https://docs.mussonindustrial.com/ignition/embr-charts/components/chart-js/

ประพิมพ์ภัทร นิ่มประเสริฐ. (2567, 13 พฤษภาคม). *ทำความรู้จักกับ React และการใช้งานเบื้องต้น*. Borntodev. https://www.borntodev.com/2024/05/13/ทำความรู้จักกับ-react/

ปาร์ธ ปาเทล. (2559, 29 สิงหาคม). *14 Key Benefits of Using Firebase for Web and App Development*. Cmarix. https://www.cmarix.com/blog/14-key-benefits-of-using-firebase-for-web-and-app-development/

ปิติ โง้วธนสุวรรณ และคณะ. (2568). *โปรแกรมบริหารจัดการจองห้องแบบบูรณาการ มหาวิทยาลัยขอนแก่น.* วิทยาลัยการคอมพิวเตอร์ มหาวิทยาลัยขอนแก่น.

มัตเตโอ ดูโอ. (2567, 1 สิงหาคม). *The Beginner's Guide to Responsive Web Design (Code Samples & Layout Examples)*. Kinsta. https://kinsta.com/blog/responsive-web-design/

โมเซอร์เปอร์. (2564, 2 ตุลาคม). *มาลองใช้ Cloud Firestore ด้วยกันนะ (สำหรับมือใหม่ — อ่านจบ Code ได้!!!)*. Medium. https://medium.com/foxbith/มาลองใช้-cloud-firestore-ด้วยกันนะ-สำหรับมือใหม่-อ่านจบ-code-ได้-e3cbd2cbeba8

ศุภนิดา รุ่งคูหา และ บัญชา วัฒนะ. (2568). *ระบบการจัดการการเบิกจ่ายวัตถุดิบสำหรับการผลิต กรณีศึกษา โรงงานผลิตชิ้นส่วนยานยนต์.* คณะวิศวกรรมศาสตร์ มหาวิทยาลัยมหาสารคาม.

อิซารัป เอ็กซินโคล. (2567, 24 ธันวาคม). *การสร้าง Dashboard โดยใช้ Chart.js ใน Next.js: จากโปรแกรมสำเร็จรูปสู่การพัฒนาเว็บไซต์*. BDI Big Data Institute. https://bdi.or.th/big-data-101/dashboard-by-chart-js

เอมิลี่ ทานากะ-เดลกาโด. (2554, 11 กรกฎาคม). *Evolving the Node.js Brand*. Nodejs. https://nodejs.org/en/blog/uncategorized/evolving-the-node-js-brand

โอภาส เอียมสิริวงศ์. (2551). ระบบฐานข้อมูล (Database System).*ระบบฐานข้อมูล Database System* (น. 35). บริษัท ซีเอ็ดยูเคชัน จำกัด.

บรรณานุกรม(ต่อ)

1001click. (2565, 7 เมษายน). *RESPONSIVE WEB DESIGN | ออกแบบเว็บอย่างไร ให้รองรับได้ทุก DEVICE*. https://www.1001click.com/blog/responsive-web-design

TechTarget. (2567, 15 พฤศจิกายน). *What is a Waterfall model? Definition and guide*. https://www-techtarget-com.translate.goog/searchsoftwarequality /definition/waterfall-model?_x_tr_sl=en&_x_tr_tl=th&_x_tr_hl=th&_x_tr_pto=tc

True Digital Academy. (2566, 26 มกราคม). *รู้จัก 'Figma' Tool ออกแบบ Web/App ที่ดีไซน์เนอร์ทั่วโลกเลือกใช้*. https://www.truedigitalacademy.com/blog/get-to-know-figma

Visperhost. (2567, 26 สิงหาคม). *CSS คืออะไร? ความหมายและประโยชน์ของ CSS ในการพัฒนาเว็บไซต์*. https://www.visperhost.net/what-is-css/

Webdodee. (2567, 2 กรกฎาคม). *Visual Studio Code หรือ VS Code คืออะไร ทำอะไรได้บ้าง*. https://webdodee.com/what-is-visual-studio-code-and-how-to-use/

ประวัติคณะผู้จัดทำโครงงานวิทยาการคอมพิวเตอร์

ชื่อ – สกุล	นายณฐนนท์ คงสุข
รหัสประจำตัว	6640248105
วันเกิด	วันที่ 1 เดือน ธันวาคม พ.ศ. 2547
ที่อยู่สามารถติดต่อได้	541 หมู่ 5 บ้านหนองผักกล้าม ตำบลเมือง อำเภอเมืองเลย จังหวัดเลย รหัสไปรษณีย์ 42000
เบอร์โทรศัพท์	0857397511
อีเมลล์	sb6640248105@lru.ac.th
ประวัติการศึกษา	มัธยมศึกษาปีที่ 6
ระดับมัธยมตอนต้น	โรงเรียนมหาไถ่ศึกษาเลย
ระดับอุดมศึกษา	วิทยาศาสตรบัณฑิต (วท.บ.) สาขาวิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยราชภัฏเลย

ชื่อ – สกุล	นายนฤสรณ์ มนตรี
รหัสประจำตัว	6640248112
วันเกิด	วันที่ 3 เดือน สิงหาคม พ.ศ. 2547
ที่อยู่สามารถติดต่อได้	222/2 หมู่ 1 บ้านธาตุ ตำบลธาตุ อำเภอเชียงคาน จังหวัดเลย รหัสไปรษณีย์ 42110
เบอร์โทรศัพท์	0981846461
อีเมลล์	sb6640248112@lru.ac.th
ประวัติการศึกษา	มัธยมศึกษาปีที่ 6
ระดับมัธยมตอนต้น	โรงเรียนมหาไถ่ศึกษาเลย
ระดับอุดมศึกษา	วิทยาศาสตรบัณฑิต (วท.บ.) สาขาวิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยราชภัฏเลย