

เว็บไซต์ร้านค้าออนไลน์

Online Store Website

นายนันทพงศ์ ปวงศ์คำมา รหัสนิสิต 67021860

รายงานเสนอเป็นส่วนหนึ่งของการศึกษา

รายวิชา 226297 ชื่อวิชา เทคโนโลยีการเขียนโปรแกรมเว็บไซต์

ภาคการศึกษา ปลาย ปีการศึกษา 2568

มหาวิทยาลัยพะเยา

บทที่1

บทนำ

ที่มาของปัญหา

ในปัจจุบันเทคโนโลยีสารสนเทศและระบบเครือข่ายอินเทอร์เน็ตได้เข้ามามีบทบาทสำคัญต่อการดำเนินชีวิตประจำวันของประชาชนอย่างกว้างขวาง ทั้งในด้านการติดต่อสื่อสาร การศึกษา การทำงาน ตลอดจนการซื้อขายสินค้าและบริการ โดยเฉพาะอย่างยิ่งการพาณิชย์อิเล็กทรอนิกส์ (Electronic Commerce: E-Commerce) ซึ่งเป็นรูปแบบการทำธุกรรมผ่านระบบออนไลน์ที่ช่วยอำนวยความสะดวกให้ผู้บริโภคสามารถเลือกซื้อสินค้า เปรียบเทียบราคา ตรวจสอบรายละเอียดสินค้า และทำการสั่งซื้อได้ตลอด 24 ชั่วโมง โดยไม่จำเป็นต้องเดินทางไปยังร้านค้าจริง ส่งผลให้พฤติกรรมของผู้บริโภคเปลี่ยนแปลงไปจากเดิมอย่างชัดเจน

แม้ว่าระบบบริการค้าออนไลน์จะได้รับความนิยมเพิ่มขึ้นอย่างต่อเนื่อง แต่ผู้ประกอบการรายย่อยหรือร้านค้าขนาดเล็กจำนวนมากยังคงประสบปัญหาในการบริหารจัดการข้อมูลสินค้าและคำสั่งซื้อเนื่องจากบางร้านยังคงใช้วิธีการจดบันทึกด้วยเอกสารหรือใช้โปรแกรมพื้นฐานที่ไม่ได้ออกแบบมาเพื่อรองรับการจัดการร้านค้าโดยเฉพาะ ส่งผลให้เกิดปัญหาหลายประการ เช่น การบันทึกข้อมูลผิดพลาด การคำนวณยอดเงินไม่ถูกต้อง การตรวจสอบจำนวนสินค้าในคลังไม่เป็นปัจจุบัน และการติดตาม

สถานะคำสั่งซื้อทำได้ยาก นอกจากนี้ยังขาดระบบจัดเก็บประวัติการทำธุรกรรมของลูกค้าอย่างเป็น

ระบบ ทำให้ไม่สามารถนำข้อมูลไปวิเคราะห์หรือพัฒนาธุรกิจต่อได้อย่างมีประสิทธิภาพ

อีกทั้งในด้านของผู้ใช้งานหรือผู้บริโภค ยังพบปัญหาจากเว็บไซต์บางแห่งที่ไม่มีระบบจัดการตรา

สินค้าอย่างเหมาะสม ไม่มีระบบแสดงยอดรวมที่ถูกต้องแบบเรียลไทม์ หรือไม่มีระบบจัดเก็บประวัติ

การสั่งซื้อ ทำให้ผู้ใช้ขาดความมั่นใจในการทำธุรกรรม นอกจากนี้ บางเว็บไซต์ยังไม่มีระบบจัดการ

บัญชีผู้ใช้ เช่น การสมัครสมาชิก การเข้าสู่ระบบ หรือการรักษาความปลอดภัยของรหัสผ่าน ส่งผลต่อ

ความน่าเชื่อถือของระบบโดยรวม

จากปัญหาดังกล่าว การพัฒนาระบบเว็บไซต์ร้านค้าออนไลน์ที่มีโครงสร้างฐานข้อมูลชัดเจน มีการ

จัดการสิทธิ์ผู้ใช้งานอย่างเหมาะสม และสามารถรองรับการทำงานทั้งในส่วนของผู้ดูแลระบบและ

ผู้ใช้งานทั่วไป จึงเป็นแนวทางสำคัญในการแก้ไขปัญหาที่เกิดขึ้น ระบบควรสามารถจัดการข้อมูลสินค้า

หมวดหมู่สินค้า การเพิ่มสินค้าเข้าสู่ตัวกรอง การคำนวณยอดรวมอัตโนมัติ การทำรายการสั่งซื้อ การ

บันทึกประวัติการซื้อ รวมถึงการจัดการยอดเงินคงเหลือของผู้ใช้ได้อย่างถูกต้องและมีประสิทธิภาพ

นอกจากนี้ การพัฒนาระบบในรูปแบบ Web Application ที่แบ่งการทำงานออกเป็นส่วน Frontend

และ Backend อย่างชัดเจน จะช่วยให้โครงสร้างระบบมีความเป็นระเบียบ สามารถพัฒนาและ

บำรุงรักษาได้ง่าย โดยใช้เทคโนโลยีที่เหมาะสม เช่น Node.js และ Express Framework สำหรับการ

พัฒนา Backend เพื่อจัดการเส้นทางการทำงาน (Routing) และเชื่อมต่อกับฐานข้อมูล MySQL ใน

การจัดเก็บข้อมูลอย่างเป็นระบบ ขณะที่ส่วน Frontend ใช้ HTML, CSS และ JavaScript ในการ

ออกแบบหน้าตาและประสบการณ์การใช้งานให้เหมาะสม รองรับทั้งคอมพิวเตอร์และอุปกรณ์พกพา

วัตถุประสงค์

1. เพื่อพัฒนาระบบเว็บไซต์ร้านค้าออนไลน์ในรูปแบบ Web Application ที่สามารถแสดงสินค้า แบ่งหมวดหมู่สินค้า และให้ผู้ใช้งานเลือกซื้อสินค้าได้อย่างสะดวก
2. เพื่อออกแบบและพัฒนาระบบฐานข้อมูลเชิงสัมพันธ์ (Relational Database) สำหรับจัดเก็บข้อมูลผู้ใช้งาน สินค้า หมวดหมู่สินค้า ตะกร้าสินค้า คำสั่งซื้อ และประวัติการทำธุรกรรมอย่างเป็นระบบ
3. เพื่อพัฒนาระบบสมัครสมาชิกและเข้าสู่ระบบ (Authentication System) ที่สามารถจัดการบัญชีผู้ใช้งานและรักษาความปลอดภัยของข้อมูล เช่น การเข้ารหัสผ่าน
4. เพื่อพัฒนาระบบทະกร้าสินค้า (Shopping Cart) ที่สามารถเพิ่ม ลด และลบจำนวนสินค้าพร้อมทั้งคำนวณยอดรวมอัตโนมัติได้อย่างถูกต้อง
5. เพื่อพัฒนาระบบสั่งซื้อสินค้า (Order Management System) ที่สามารถบันทึกข้อมูลคำสั่งซื้อ รายละเอียดสินค้าในแต่ละรายการ และแสดงประวัติการสั่งซื้อของผู้ใช้งานได้
6. เพื่อพัฒนาระบบจัดการยอดเงินคงเหลือและการเติมเงิน (Top-up System) สำหรับจำลองกระบวนการชำระเงินภายในระบบ
7. เพื่อศึกษาและประยุกต์ใช้เทคโนโลยีด้านการพัฒนาเว็บ ได้แก่ Node.js, Express Framework, MySQL รวมถึงการออกแบบส่วนติดต่อผู้ใช้งาน (Frontend) ด้วย HTML, CSS และ JavaScript

8. เพื่อฝึกทักษะการแบ่งส่วนการทำงานระหว่าง Frontend และ Backend และเข้าใจ

ขั้นตอนการเชื่อมต่อระบบเว็บแอปพลิเคชันกับฐานข้อมูล

9. เพื่อให้สามารถนำความรู้ด้าน Web Programming Technology มาประยุกต์ใช้ในการ

พัฒนาระบบสารสนเทศที่สามารถใช้งานได้จริง

ขอบเขต

1. ขอบเขตด้านการทำงานของระบบ (System Scope)

ระบบเว็บไซต์ร้านค้าออนไลน์ที่พัฒนาขึ้น มีฟังก์ชันการทำงานหลัก ดังนี้

1.1 ระบบสมัครสมาชิก (Register)

ผู้ใช้งานสามารถสมัครสมาชิกโดยกรอกชื่อผู้ใช้และรหัสผ่าน

ระบบจัดเก็บข้อมูลผู้ใช้งานลงฐานข้อมูล

รหัสผ่านถูกจัดเก็บในรูปแบบเข้ารหัส (Password Hash)

1.2 ระบบเข้าสู่ระบบ (Login / Logout)

ผู้ใช้งานสามารถเข้าสู่ระบบด้วยชื่อผู้ใช้และรหัสผ่าน

ระบบตรวจสอบความถูกต้องของข้อมูล

มีระบบ Session สำหรับควบคุมสถานการณ์เข้าสู่ระบบ

1.3 ระบบจัดการสินค้า (Product Management)

แสดงรายการสินค้า

แสดงรายละเอียดสินค้า

แบ่งหมวดหมู่สินค้า (Categories)

แสดงรูปภาพ ราคา และจำนวนคงเหลือของสินค้า

1.4 ระบบตะกร้าสินค้า (Shopping Cart)

เพิ่มสินค้าเข้าตะกร้า

แก้ไขจำนวนสินค้า

ลบสินค้าออกจากตะกร้า

คำนวณราคารวมอัตโนมัติ

1.5 ระบบสั่งซื้อสินค้า (Order System)

บันทึกข้อมูลคำสั่งซื้อ

บันทึกรายการสินค้าแต่ละชิ้นในคำสั่งซื้อ

แสดงประวัติการสั่งซื้อของผู้ใช้งาน

1.6 ระบบเติมเงิน (Top-up System)

ผู้ใช้งานสามารถเพิ่มยอดเงินเข้าสู่ระบบ

ระบบบันทึกประวัติการเติมเงิน

ใช้ยอดเงินภายในการสั่งซื้อสินค้า

2. ขอบเขตด้านฐานข้อมูล (Database Scope)

ระบบใช้ฐานข้อมูล MySQL ในการจัดเก็บข้อมูล โดยประกอบด้วยตารางหลัก ได้แก่

users, categories, products, cart_items, orders, order_items, topups

ระบบออกแบบฐานข้อมูลในรูปแบบ Relational Database พร้อมกำหนด Primary Key (PK) และ Foreign Key (FK) เพื่อควบคุมความสัมพันธ์ของข้อมูล

3. ขอบเขตด้านเทคโนโลยีที่ใช้ (Technology Scope)

3.1 Backend

[Node.js](#), Express Framework, MySQL, dotenv, express-session

3.2 Frontend

HTML, CSS, JavaScript, EJS Template Engine

4. ขอบเขตด้านผู้ใช้งาน (User Scope)

ระบบรองรับผู้ใช้งานทั่วไป (User) ซึ่งสามารถ

สมัครสมาชิก, เข้าสู่ระบบ, เลือกซื้อสินค้า, เพิ่มสินค้าในตะกร้า, สั่งซื้อสินค้า, เติมเงิน,
ตรวจสอบประวัติการสั่งซื้อ

5. ขอบเขตที่ไม่ครอบคลุม (Limitations)

ระบบยังไม่รองรับระบบชำระเงินผ่าน Payment Gateway จริง

ยังไม่มีระบบจัดการสต็อกสินค้า (Tracking System)

ยังไม่มีระบบจัดการหลังบ้าน (Admin Dashboard แบบสมบูรณ์)

ระบบพัฒนาเพื่อใช้ในการศึกษาและทดสอบเท่านั้น

แผนการทำโครงการ

ขั้นตอน	รายละเอียด	ระยะเวลา
วิเคราะห์ระบบ	ศึกษาความต้องการ	สัปดาห์ที่ 1
ออกแบบระบบ	ERD และ UI	สัปดาห์ที่ 2
พัฒนาโปรแกรม	Frontend/Backend	สัปดาห์ที่ 3-4
ทดสอบระบบ	ตรวจสอบการทำงาน	สัปดาห์ที่ 5
ปรับปรุงระบบ	แก้ไขข้อผิดพลาด	สัปดาห์ที่ 6
จัดทำรายงาน	เขียนรายงาน	สัปดาห์ที่ 7

บทที่2

เอกสารรายงานวิจัยที่เกี่ยวข้อง

ในการพัฒนาโครงการเว็บไซต์ร้านค้าออนไลน์ ผู้จัดทำได้ศึกษาเอกสาร แนวคิด ทฤษฎี และงานวิจัยที่เกี่ยวข้องกับระบบพาณิชย์อิเล็กทรอนิกส์ (E-Commerce) การพัฒนาเว็บแอปพลิเคชัน ระบบฐานข้อมูล และพฤติกรรมผู้บริโภคออนไลน์ เพื่อใช้เป็นแนวทางในการออกแบบและพัฒนาระบบที่มีประสิทธิภาพ โดยมีรายละเอียดดังต่อไปนี้

ในการพัฒนาโครงการเว็บไซต์ร้านค้าออนไลน์ ผู้จัดทำได้ศึกษาเอกสาร แนวคิด ทฤษฎี และงานวิจัยที่เกี่ยวข้องกับระบบพาณิชย์อิเล็กทรอนิกส์ (E-Commerce) การพัฒนาเว็บแอปพลิเคชัน ระบบฐานข้อมูล และพฤติกรรมผู้บริโภคออนไลน์ เพื่อใช้เป็นแนวทางในการออกแบบและพัฒนาระบบที่มีประสิทธิภาพ โดยมีรายละเอียดดังต่อไปนี้

1.1 แนวคิดเกี่ยวกับพาณิชย์อิเล็กทรอนิกส์ (E-Commerce)

พาณิชย์อิเล็กทรอนิกส์ (Electronic Commerce หรือ E-Commerce) หมายถึง การดำเนินกิจกรรมทางการค้า การซื้อขายสินค้าและบริการผ่านเครือข่าย อินเทอร์เน็ต โดยใช้ระบบคอมพิวเตอร์หรืออุปกรณ์อิเล็กทรอนิกส์เป็นสื่อกลางในการทำธุรกรรม ซึ่งครอบคลุมตั้งแต่การนำเสนอสินค้า การสั่งซื้อ การชำระเงิน การจัดส่งสินค้า ตลอดจนบริการหลังการขาย

ปัจจุบัน E-Commerce มีบทบาทสำคัญต่อระบบเศรษฐกิจอย่างมาก โดยเฉพาะในยุคดิจิทัลที่ผู้บริโภคสามารถเข้าถึงอินเทอร์เน็ตได้อย่างสะดวก การซื้อขายออนไลน์ช่วยลดต้นทุนหน้าร้าน ลดค่าใช้จ่ายด้านพนักงาน และขยายตลาดได้กว้างขึ้นทั้งในระดับประเทศและระดับสากล

รูปแบบของ E-Commerce แบ่งออกได้หลายประเภท เช่น

B2C (Business to Consumer) – ธุรกิจกับผู้บริโภค

B2B (Business to Business) – ธุรกิจกับธุรกิจ

C2C (Consumer to Consumer) – ผู้บริโภคกับผู้บริโภค

C2B (Consumer to Business) – ผู้บริโภคเสนอธุรกิจให้ธุรกิจ

สำหรับโครงงานเว็บไซต์ร้านค้าออนไลน์ จัดอยู่ในรูปแบบ B2C ซึ่งผู้ขายนำเสนอสินค้าให้กับลูกค้าทั่วไปผ่านระบบเว็บไซต์

1.2 แนวคิดเกี่ยวกับระบบเว็บแอปพลิเคชัน (Web Application)

เว็บแอปพลิเคชัน คือ โปรแกรมประยุกต์ที่ทำงานผ่านเว็บเบราว์เซอร์ โดยผู้ใช้งานไม่จำเป็นต้องติดตั้งโปรแกรมเพิ่มเติม ระบบจะประมวลผลผ่านเซิร์ฟเวอร์และแสดงผลผ่านหน้าเว็บ

โครงสร้างพื้นฐานของเว็บแอปพลิเคชันประกอบด้วย 3 ส่วนหลัก ได้แก่

Frontend – ส่วนที่ผู้ใช้มองเห็นและโต้ตอบ

Backend – ส่วนประมวลผลและจัดการข้อมูล

Database – ส่วนจัดเก็บข้อมูล

ในการพัฒนาเว็บไซต์ร้านค้าออนไลน์ ผู้จัดทำเลือกใช้ Node.js และ Express Framework สำหรับฝั่ง Backend และใช้ MySQL เป็นระบบฐานข้อมูล เนื่องจากเป็นเทคโนโลยีที่มีความเสถียร รองรับการจัดการข้อมูลจำนวนมาก และสามารถพัฒนาต่อยอดได้ในอนาคต

เว็บแอปพลิเคชันที่ต้องมีคุณสมบัติ ดังนี้

รองรับผู้ใช้งานหลายคนพร้อมกัน

มีระบบรักษาความปลอดภัย

จัดการข้อมูลได้อย่างเป็นระบบ

มีความรวดเร็วในการประมวลผล

ใช้งานง่าย (User-Friendly)

1.3 แนวคิดเกี่ยวกับระบบฐานข้อมูล (Database System)

ระบบฐานข้อมูลคือระบบที่ใช้ในการจัดเก็บข้อมูลอย่างเป็นโครงสร้าง สามารถเรียกได้ตาม แก้ไข เพิ่ม หรือลบข้อมูลได้อย่างมีประสิทธิภาพ โดยทั่วไปฐานข้อมูลจะประกอบด้วยตาราง (Table) ความสัมพันธ์ (Relationship) คีย์หลัก (Primary Key) และคีย์นอก (Foreign Key)

ในระบบบร้านค้าออนไลน์ ฐานข้อมูลมีบทบาทสำคัญ เช่น

เก็บข้อมูลสมาชิก

เก็บข้อมูลสินค้า

เก็บข้อมูลคำสั่งซื้อ

เก็บข้อมูลรายการสินค้าในตระกร้า

เก็บข้อมูลการเติมเงิน

การออกแบบฐานข้อมูลที่ต้องคำนึงถึงความสมัพนธ์ของข้อมูล เพื่อลดความซ้ำซ้อน และเพิ่มประสิทธิภาพในการค้นหา

งานวิจัยที่เกี่ยวข้อง

งานวิจัยที่ 1

สมชาย และคณะ (2564) ได้พัฒนาระบบบร้านค้าออนไลน์สำหรับผู้ประกอบการขนาดเล็ก โดยมีวัตถุประสงค์เพื่อสร้างระบบที่สามารถจัดการสินค้าและคำสั่งซื้อได้อย่างมีประสิทธิภาพ ผลการวิจัยพบว่า ระบบช่วยลดเวลาในการจัดการคำสั่งซื้อได้ร้อยละ 35 และเพิ่มความพึงพอใจของลูกค้าในระดับมาก

จากการวิจัยดังกล่าว พบร่วมกันว่า ระบบช่วยสำหรับผู้ประกอบการขนาดเล็ก ซึ่งเป็นแนวทางที่นำมาประยุกต์ใช้ในโครงงานนี้

งานวิจัยที่ 2

กิตติพงษ์ (2565) ศึกษาปัจจัยที่มีผลต่อการตัดสินใจซื้อสินค้าออนไลน์ของผู้บริโภค พบร่วมกันว่า ความน่าเชื่อถือของเว็บไซต์ ระบบชำระเงินที่ปลอดภัย และความชัดเจนของข้อมูลสินค้า มีผลต่อการตัดสินใจซื้อย่างมีนัยสำคัญ

โครงงานนี้จึงออกแบบระบบให้มีรายละเอียดสินค้าอย่างครบถ้วน และมีระบบยืนยันตัวตนผู้ใช้ก่อนทำธุกรรม

งานวิจัยที่ 3

ณัฐพล และคณะ (2563) ได้ทำการศึกษาเรื่อง “การพัฒนาระบบร้านค้าออนไลน์โดยใช้เทคโนโลยี Node.js และ MySQL” โดยมีวัตถุประสงค์เพื่อออกแบบและพัฒนาระบบขายสินค้าออนไลน์ที่สามารถจัดการข้อมูลสินค้า สมาชิก และคำสั่งซื้อได้แบบเรียลไทม์ งานวิจัยนี้ใช้แนวคิดการพัฒนาแบบ MVC (Model-View-Controller) เพื่อแยกส่วนการทำงานของระบบออกเป็นส่วน ๆ อย่างชัดเจน ได้แก่ ส่วนติดต่อผู้ใช้ ส่วนประมวลผล และส่วนฐานข้อมูล

ผลการทดลองใช้งานระบบทพบว่า ระบบสามารถรองรับผู้ใช้งานพร้อมกันได้มากกว่า 100 คนโดยไม่มีปัญหาการล่มของเซิร์ฟเวอร์ และสามารถลดขั้นตอนการจัดการคำสั่งซื้อแบบเดิมลงได้อย่างมีประสิทธิภาพ นอกจากนี้ยังพบว่าโครงสร้างฐานข้อมูลที่ออกแบบอย่างเหมาะสมช่วยลดความช้าของข้อมูลและเพิ่มความเร็วในการค้นหา

จากการวิจัยดังกล่าว ผู้จัดทำได้นำแนวคิดการแยกส่วนโครงสร้างระบบ และการออกแบบฐานข้อมูลที่สัมพันธ์กันอย่างชัดเจน มาประยุกต์ใช้ในโครงงานเว็บไซต์ร้านค้าออนไลน์นี้

งานวิจัยที่ 4

อรุณมา (2562) ได้ศึกษา “ปัจจัยที่มีผลต่อความพึงพอใจในการใช้งานเว็บไซต์พาณิชย์อิเล็กทรอนิกส์” โดยทำการเก็บข้อมูลจากผู้ใช้งานจำนวน 300 คน ผลการวิจัยพบว่า ปัจจัยด้านความสุขของหน้าเว็บ ความรวดเร็วในการโหลดข้อมูล และความง่ายในการค้นหาสินค้า มีผลต่อความพึงพอใจของผู้ใช้งานในระดับสูง

นอกจากนี้ งานวิจัยยังระบุว่า ผู้ใช้งานให้ความสำคัญกับระบบลงทะเบียนสินค้า และขั้นตอนการทำธุรกรรมที่ไม่ซับซ้อน หากกระบวนการสั่งซื้อยุ่งยากเกินไป ผู้ใช้งานมีแนวโน้มที่จะยกเลิกคำสั่งซื้อกลางคัน จากผลการศึกษานี้ โครงงานเว็บไซต์ร้านค้าออนไลน์จึงออกแบบระบบให้มีขั้นตอนการสั่งซื้อที่กระชับเข้าใจง่าย และมีโครงสร้างเมนูที่ชัดเจน

งานวิจัยที่ 5

ธีรภัทร และคณะ (2564) ได้พัฒนาระบบจัดการคำสั่งซื้อออนไลน์สำหรับร้านค้าปลีก โดยมีจุดมุ่งหมายเพื่อแก้ปัญหาการบันทึกข้อมูลคำสั่งซื้อแบบกระดาษ งานวิจัยนี้ใช้ฐานข้อมูลเชิงสัมพันธ์ (Relational Database) และออกแบบ ER Diagram ก่อนพัฒนาโปรแกรมจริง

ผลการวิจัยพบว่า ระบบช่วยลดความผิดพลาดในการบันทึกข้อมูลลงได้ร้อยละ 40 และสามารถตรวจสอบประวัติการสั่งซื้อย้อนหลังได้ทันที นอกจากนี้ยังสามารถสรุปรายงานยอดขายรายวัน รายเดือน และรายปีได้อัตโนมัติ

แนวคิดการจัดเก็บข้อมูลคำสั่งซื้อและรายการสินค้าในคำสั่งซื้อ (Order และ Order_Items) ถูกนำมาใช้ในการออกแบบฐานข้อมูลของโครงงานนี้

งานวิจัยที่ 6

ปวีณา (2565) ได้ศึกษาเรื่อง “การพัฒนาระบบสมาชิกและการยืนยันตัวตนในเว็บไซต์ พานิชย์อิเล็กทรอนิกส์” โดยเน้นการเข้ารหัสรหัสผ่าน (Password Hashing) และการจัดการ Session เพื่อเพิ่มความปลอดภัยของข้อมูลผู้ใช้งาน

ผลการศึกษาแสดงให้เห็นว่า การใช้ bcrypt ใน การเข้ารหัสรหัสผ่านช่วยลดความเสี่ยงจากการโจมตีแบบ Brute Force และการใช้ Session Management อย่างถูกต้องช่วยป้องกันการเข้าถึงข้อมูลโดยไม่ได้รับอนุญาต

โครงงานนี้จึงนำแนวคิดการเข้ารหัสรหัสผ่านและการจัดการ Session มาใช้ในระบบสมาชิก เพื่อเพิ่มความปลอดภัยให้กับข้อมูลผู้ใช้งาน

งานวิจัยที่ 7

ศุภชัย และคณะ (2563) ได้วิจัยเรื่อง “พัฒนาระบบซื้อสินค้าออนไลน์ของผู้บริโภคในยุคดิจิทัล” โดยพบว่า ผู้บริโภคให้ความสำคัญกับรีวิวสินค้า ความชัดเจนของรายละเอียดสินค้า และความน่าเชื่อถือของร้านค้า

ผลการวิจัยระบุว่า หากเว็บไซต์มีรูปภาพสินค้าและข้อมูลครบถ้วน จะช่วยเพิ่มโอกาสในการตัดสินใจซื้อได้มากขึ้น นอกจากนี้ ระบบแจ้งสถานะคำสั่งซื้อก็เป็นอีกปัจจัยสำคัญที่ช่วยสร้างความมั่นใจให้กับลูกค้า

ดังนั้น โครงการนี้จึงออกแบบให้มีรายละเอียดสินค้า เช่น ชื่อสินค้า ราคา คำอธิบาย และรูปภาพ เพื่อช่วยให้ผู้ใช้งานตัดสินใจได้ง่ายขึ้น

งานวิจัยที่ 8

จุฑามาศ (2564) ได้ศึกษา “การออกแบบประสบการณ์ผู้ใช้ (User Experience: UX) สำหรับเว็บไซต์ร้านค้าออนไลน์” โดยเน้นการออกแบบที่เรียบง่าย (Minimal Design) และการจัดวางองค์ประกอบที่เหมาะสม

ผลการศึกษาแสดงให้เห็นว่า เว็บไซต์ที่มีการจัดหมวดหมู่สินค้าอย่างชัดเจน และมีระบบค้นหาสินค้า จะช่วยให้ผู้ใช้งานสามารถเข้าถึงข้อมูลได้รวดเร็วขึ้น

โครงการนี้จึงมีระบบหมวดหมู่สินค้า (Categories) และหน้าแสดงรายการสินค้า (Products) แยกตามหมวดหมู่

งานวิจัยที่ 9

กานต์ธิดา และคณะ (2565) ได้พัฒนาระบบที่มีเงินเดือนในเว็บไซต์ร้านค้า โดยมีวัตถุประสงค์เพื่ออำนวยความสะดวกให้ผู้ใช้งานสามารถเติมเงินเข้าสู่ระบบก่อนทำการสั่งซื้อ

ผลการทดลองพบว่า ระบบเติมเงินช่วยลดปัญหาการชำระเงินล้มเหลว และช่วยให้ร้านค้าสามารถควบคุมยอดคงเหลือของลูกค้าได้อย่างแม่นยำ

โครงการนี้จึงมีตาราง Topups สำหรับบันทึกข้อมูลการเติมเงินของสมาชิก

งานวิจัยที่ 10

วิชาณุ (2566) ได้ศึกษา “การออกแบบฐานข้อมูลสำหรับระบบร้านค้าออนไลน์ขนาดเล็ก” โดยเน้นการทำ Normalization เพื่อลดความซ้ำซ้อนของข้อมูล

ผลการศึกษาแสดงให้เห็นว่า การแยกตาราง Users, Products, Orders และ Order_Items อย่างชัดเจน ช่วยให้ระบบสามารถขยายตัวในอนาคตได้ง่าย และลดปัญหาความไม่สอดคล้องของข้อมูล

โครงการนี้ได้นำแนวคิดการออกแบบฐานข้อมูลแบบ 3NF มาใช้ในการพัฒนา ER Diagram

งานวิจัยที่ 11

สุชาดา และคณะ (2564) ได้ศึกษาเรื่อง “การประยุกต์ใช้ระบบฐานข้อมูลเชิงสัมพันธ์ในการพัฒนาเว็บไซต์พาณิชย์อิเล็กทรอนิกส์” โดยมีวัตถุประสงค์เพื่อออกแบบฐานข้อมูลที่สามารถรองรับการจัดการข้อมูลจำนวนมากได้อย่างมีประสิทธิภาพ งานวิจัยนี้ได้เปรียบเทียบโครงสร้างฐานข้อมูลที่ไม่มีการทำ Normalization กับฐานข้อมูลที่ออกแบบตามหลัก Third Normal Form (3NF)

ผลการศึกษาพบว่า ฐานข้อมูลที่ผ่านกระบวนการ Normalization สามารถลดความซ้ำซ้อนของข้อมูลลงได้อย่างมีนัยสำคัญ และช่วยเพิ่มประสิทธิภาพในการค้นหาข้อมูลสินค้าและคำสั่งซื้อ อีกทั้งยังลดปัญหาความไม่สอดคล้องของข้อมูล (Data Inconsistency) เมื่อมีการแก้ไขหรืออัปเดตข้อมูล

จากแนวคิดดังกล่าว โครงการเว็บไซต์ร้านค้าออนไลน์นี้จึงออกแบบฐานข้อมูลให้แยกตารางอย่างชัดเจน ได้แก่ ตาราง Users, Categories, Products, Orders, Order_Items และ Cart_Items เพื่อให้ข้อมูลแต่ละประเภทถูกจัดเก็บอย่างเป็นระบบและสัมพันธ์กันอย่างถูกต้องผ่าน Primary Key และ Foreign Key

งานวิจัยที่ 12

พงศกร (2565) ได้ศึกษา “ระบบตະกร้าสินค้าและกระบวนการชำระเงินในเว็บไซต์ร้านค้าออนไลน์” โดยเน้นการออกแบบขั้นตอนการทำงานของระบบตั้งแต่การเพิ่มสินค้าในตະกร้า การปรับจำนวนสินค้า การคำนวณยอดรวม ไปจนถึงการยืนยันคำสั่งซื้อ

ผลการวิจัยพบว่า ระบบตະกร้าสินค้าที่มีการคำนวณยอดรวมแบบอัตโนมัติ และสามารถแก้ไขจำนวนสินค้าได้ทันที ช่วยลดข้อผิดพลาดของผู้ใช้งาน และเพิ่มความสะดวกในการสั่งซื้อ นอกจากนี้ยังพบว่า การแยกตาราง Cart_Items และ Order_Items อย่างชัดเจน ช่วยให้ระบบสามารถจัดเก็บข้อมูลก่อน และหลังการยืนยันคำสั่งซื้อได้อย่างมีประสิทธิภาพ

โครงการนี้จึงนำเสนอวิธีดังกล่าวมาปรับใช้ โดยออกแบบให้ตະกร้าสินค้าสามารถเลือกสินค้าแบบรายรายการ หรือเลือกทั้งหมด และระบบจะคำนวณยอดรวมเฉพาะสินค้าที่เลือกเท่านั้น เพื่อเพิ่มความถูกต้องของข้อมูล

งานวิจัยที่ 13

รัตนา และคณะ (2563) ได้ศึกษา “การพัฒนาระบบจัดการคำสั่งซื้อออนไลน์โดยใช้ Express Framework” โดยมุ่งเน้นการพัฒนา Backend ด้วย Node.js และ Express เพื่อรับรองการทำงานของ API และการเชื่อมต่อฐานข้อมูล MySQL

ผลการศึกษาแสดงให้เห็นว่า การใช้ Express Framework ช่วยลดความซับซ้อนของโค้ด และสามารถจัดการ Routing ได้อย่างเป็นระบบ ทำให้การพัฒนาระบบเป็นไปอย่างรวดเร็วและง่ายต่อ การบำรุงรักษาในอนาคต

จากการวิจัยนี้ โครงการเว็บไซต์ร้านค้าออนไลน์จึงใช้ Express Framework ในการจัดการเส้นทาง (Routes) เช่น /login, /register, /cart, /checkout และ /category เพื่อให้โครงสร้างระบบมีความเป็นระเบียบ และสามารถขยายฟังก์ชันเพิ่มเติมได้ในอนาคต

งานวิจัยที่ 14

ธัญญารัตน์ (2566) ได้ศึกษา “ความปลอดภัยของข้อมูลในระบบพาณิชย์อิเล็กทรอนิกส์” โดยวิเคราะห์ปัญหาการโจมตีทางไซเบอร์ที่พบบ่อย เช่น SQL Injection, Cross-Site Scripting (XSS) และ Session Hijacking

ผลการศึกษาพบว่า ระบบที่ไม่มีการตรวจสอบข้อมูลนำเข้า (Input Validation) มีความเสี่ยงสูงต่อการถูกโจมตี นอกจากนี้ การจัดการ Session อย่างไม่เหมาะสมอาจทำให้ข้อมูลผู้ใช้งานรั่วไหลได้ โครงการนี้จึงมีการตรวจสอบข้อมูลก่อนบันทึกลงฐานข้อมูล และใช้การเข้ารหัสผ่าน (Password Hashing) เพื่อเพิ่มความปลอดภัย รวมถึงกำหนด Session Secret ในไฟล์ .env เพื่อป้องกันการเข้าถึงโดยไม่ได้รับอนุญาต

งานวิจัยที่ 15

กิตติภพ และคณะ (2565) ได้ศึกษา “การประเมินประสิทธิภาพของเว็บไซต์ร้านค้าออนไลน์ขนาดเล็ก” โดยทำการทดสอบความเร็วในการโหลดหน้าเว็บ และความสามารถในการรองรับผู้ใช้งานหลายคนพร้อมกัน

ผลการทดลองพบว่า เว็บไซต์ที่มีการจัดโครงสร้างไฟล์อย่างเหมาะสม และใช้การแยกไฟล์ CSS, JavaScript และ Template อย่างชัดเจน สามารถลดเวลาในการโหลดหน้าเว็บได้อย่างมีประสิทธิภาพ

โครงการนี้จึงแยกโครงสร้างไฟล์ออกเป็นโฟลเดอร์ เช่น public/css, public/js และ views เพื่อให้การจัดการโค้ดเป็นระเบียบ และช่วยเพิ่มประสิทธิภาพในการทำงานของระบบ การวิเคราะห์เชิงเปรียบเทียบงานวิจัยที่เกี่ยวข้อง

เมื่อพิจารณางานวิจัยทั้งหมด พบร่วมกันนี้มีการพัฒนาเว็บไซต์ร้านค้าออนไลน์ในปัจจุบันให้ความสำคัญกับ 4 ประเด็นหลัก ได้แก่

1. โครงสร้างฐานข้อมูลที่มีความสมมัติฐานชัดเจน
2. ความปลอดภัยของระบบสมาชิก
3. ความสามารถในการใช้งานของผู้ใช้
4. ประสิทธิภาพและความรวดเร็วของระบบ

โครงการเว็บไซต์ร้านค้าออนไลน์นี้จึงได้บูรณาการแนวคิดจากงานวิจัยหลายฉบับเข้าด้วยกัน ทั้งด้านการออกแบบ ER Diagram การพัฒนาด้วย Node.js และ Express การใช้ MySQL เป็นฐานข้อมูล และการจัดการ Session เพื่อเพิ่มความปลอดภัย

กรอบแนวคิดในการพัฒนาโครงการ (Conceptual Framework)

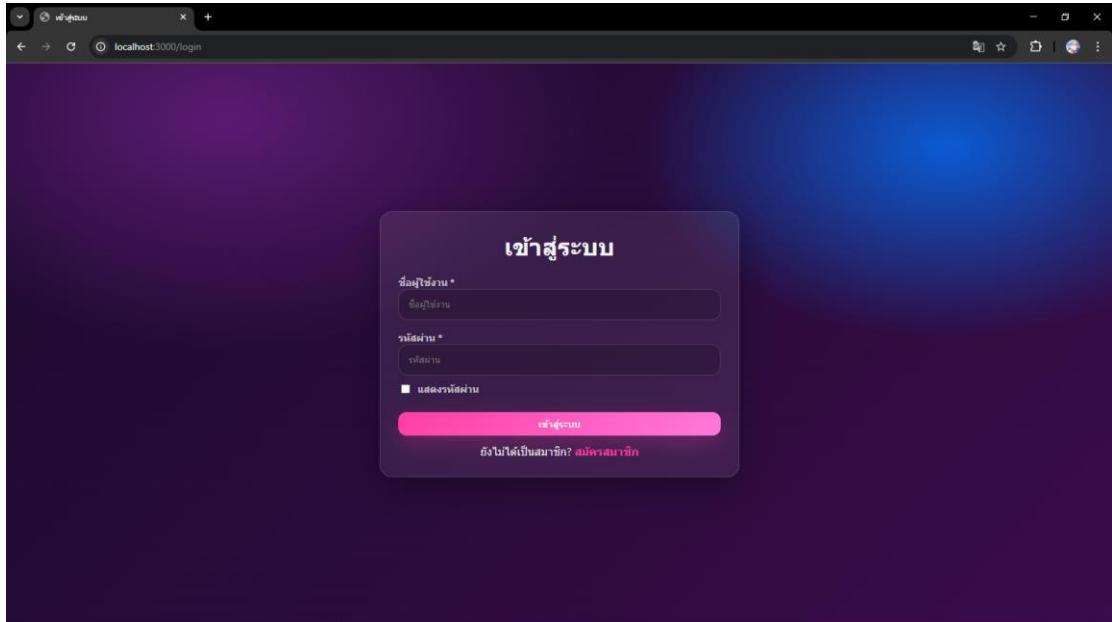
จากการศึกษางานวิจัยที่เกี่ยวข้อง สามารถสรุปกรอบแนวคิดในการพัฒนาโครงการได้ดังนี้

1. ผู้ใช้งาน (User) สมัครสมาชิกและเข้าสู่ระบบ
2. ผู้ใช้งานเลือกสินค้าและเพิ่มลงในตะกร้า
3. ระบบบันทึกข้อมูลลงในตาราง Cart_Items
4. เมื่อยืนยันการสั่งซื้อ ระบบจะสร้างข้อมูลในตาราง Orders และ Order_Items
5. ระบบสามารถบันทึกประวัติการสั่งซื้อ และตรวจสอบยอดเงินคงเหลือ

โครงสร้างดังกล่าวสะท้อนความสมมัติของตารางในฐานข้อมูล และแสดงให้เห็นถึงลำดับขั้นตอนการทำงานของระบบอย่างชัดเจน

Mockup

1. หน้าเข้าสู่ระบบ (Login Page)



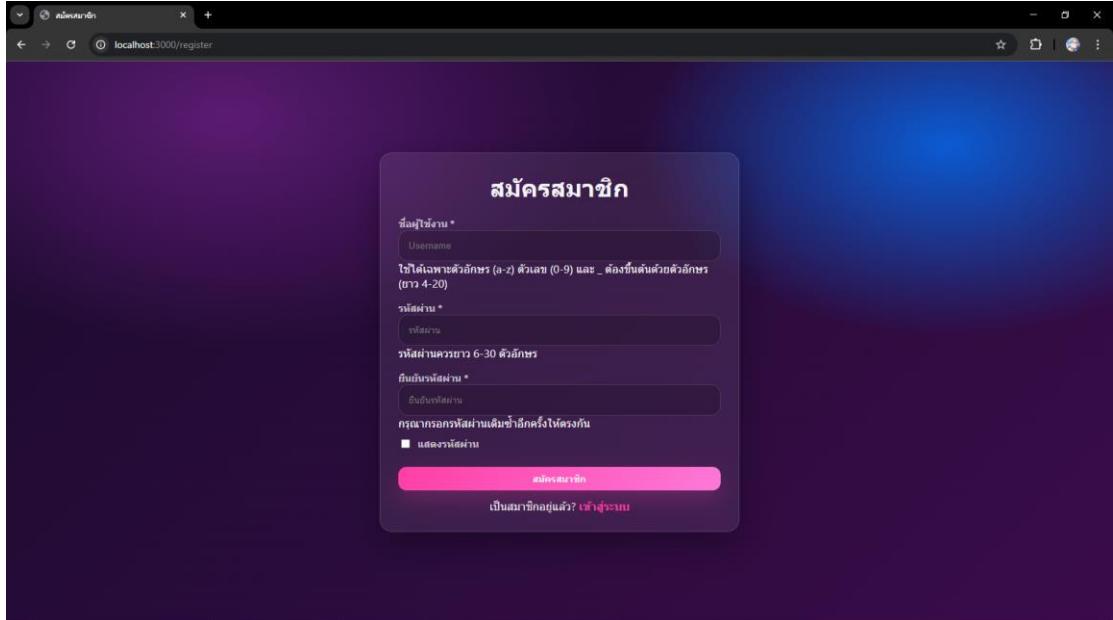
คำอธิบาย:

หน้าเข้าสู่ระบบเป็นหน้าหลักสำหรับการยืนยันตัวตนของผู้ใช้งานก่อนเข้าใช้งานระบบ โดยผู้ใช้ต้องกรอกชื่อผู้ใช้งาน (Username) และรหัสผ่าน (Password) ให้ถูกต้อง ระบบจะทำการตรวจสอบ

ข้อมูลกับฐานข้อมูลในตาราง Users หากข้อมูลถูกต้องจะอนุญาตให้เข้าสู่หน้าหลักของระบบ

ภายในหน้านี้มีฟังก์ชัน “แสดงรหัสผ่าน” เพื่อช่วยให้ผู้ใช้งานสามารถตรวจสอบความถูกต้องของรหัสผ่านก่อนกดเข้าสู่ระบบ ลดความผิดพลาดในการกรอกข้อมูล

2.หน้าสมัครสมาชิก (Register Page)



คำอธิบาย:

หน้าสมัครสมาชิกใช้สำหรับลงทะเบียนผู้ใช้งานใหม่เข้าสู่ระบบ โดยผู้ใช้งานต้องกรอกข้อมูลผู้ใช้งาน

รหัสผ่าน และยืนยันรหัสผ่าน

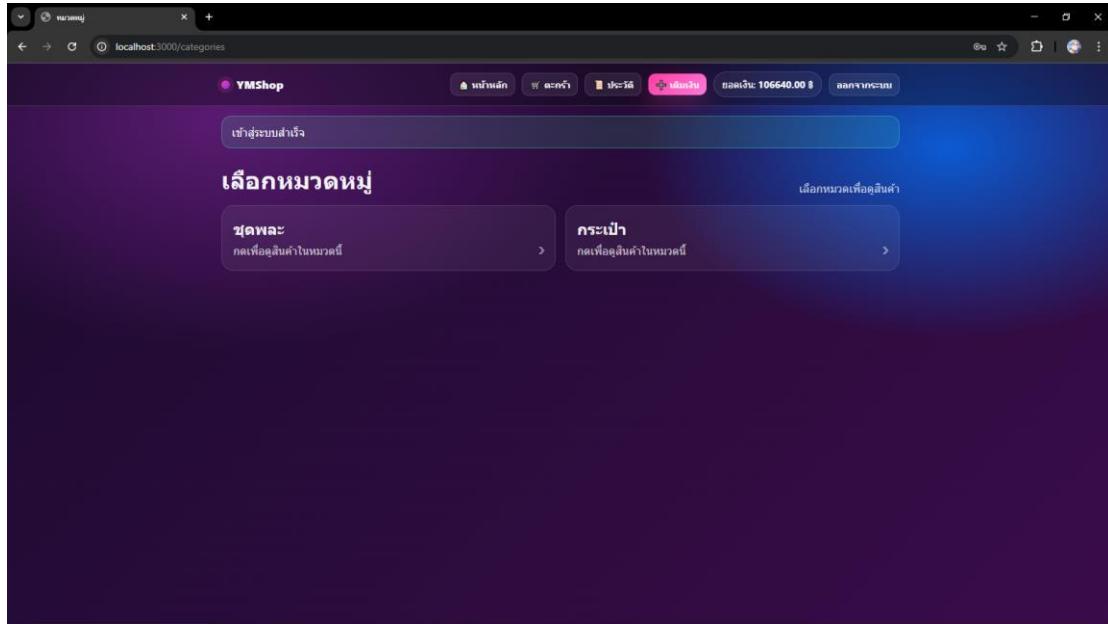
ระบบมีการกำหนดเงื่อนไขของข้อมูลผู้ใช้งาน เช่น ต้องมีต้นตัวอักษร และสามารถใช้ตัวอักษร

ภาษาอังกฤษและตัวเลขเท่านั้น รวมถึงกำหนดความยาวขั้นต่ำและสูงสุด เพื่อป้องกันข้อมูลที่ไม่

ถูกต้อง

รหัสผ่านจะถูกเข้ารหัส (Hash) ก่อนจัดเก็บลงฐานข้อมูล เพื่อเพิ่มความปลอดภัยของระบบ

3.หน้าหมวดหมู่สินค้า (Categories Page)



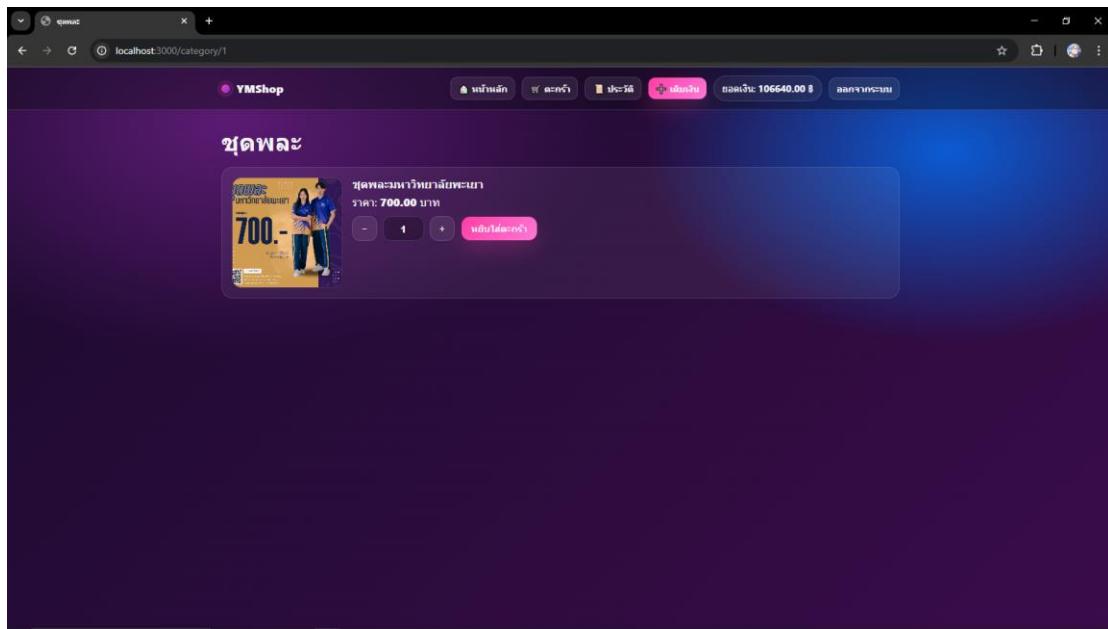
คำอธิบาย:

หน้าหมวดหมู่สินค้าแสดงรายการประเภทของสินค้า เช่น ชุดพละ กระเบื้า และสินค้าอื่น ๆ ที่มีใน

ระบบ ผู้ใช้งานสามารถคลิกเลือกหมวดหมู่เพื่อดูสินค้าในหมวดนั้น

ระบบจะดึงข้อมูลจากตาราง Categories มาแสดงผลแบบ Grid Layout เพื่อให้ง่ายต่อการเลือกใช้งาน

4.หน้ารายการสินค้า (Products Page)

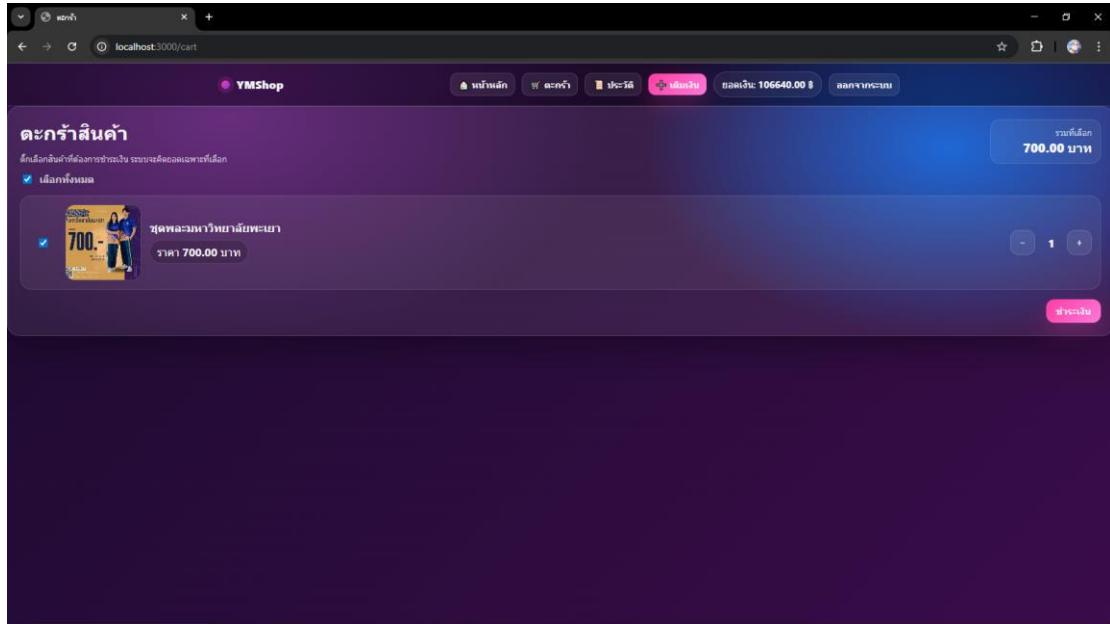


คำอธิบาย:

หน้านี้แสดงรายการสินค้าตามหมวดหมู่ที่ผู้ใช้งานเลือก โดยแต่ละรายการจะแสดงรูปภาพสินค้า ชื่อสินค้า และราคาสินค้า

ผู้ใช้งานสามารถกดเพิ่มสินค้าเข้าสู่ตะกร้าได้

5.หน้าตะกร้าสินค้า (Cart Page)



คำอธิบาย:

หน้าตะกร้าสินค้าแสดงรายการสินค้าที่ผู้ใช้งานเลือกไว้ โดยแสดงชื่อสินค้า ราคา จำนวนและราคารวม

ผู้ใช้งานสามารถ:

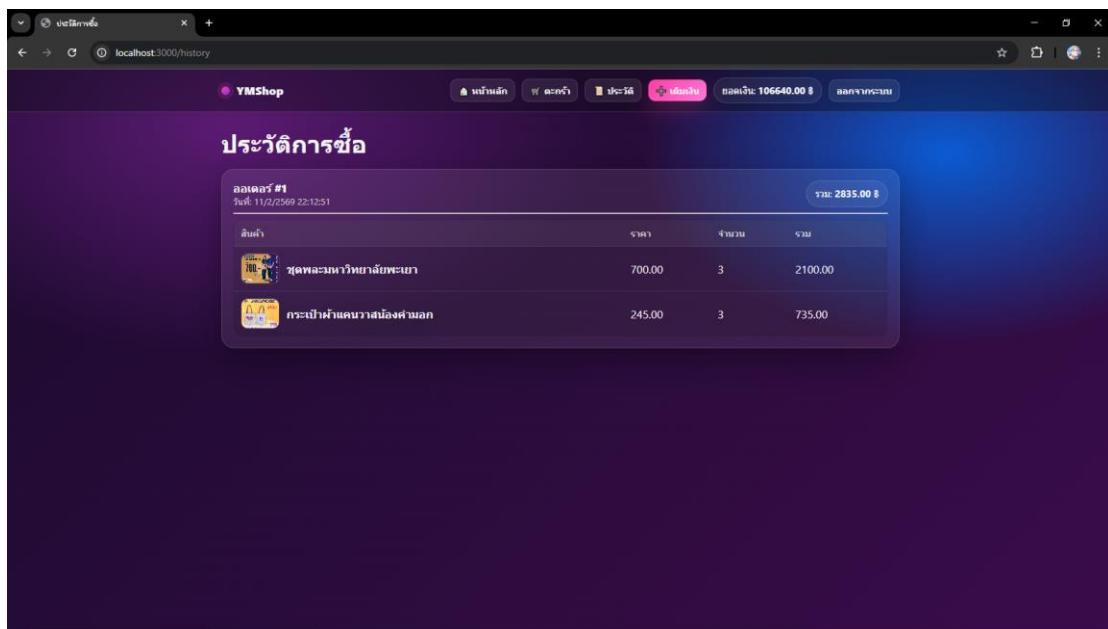
เพิ่มหรือลดจำนวนสินค้า

ลบสินค้าออกจากตะกร้า

เลือกสินค้าที่ต้องการชำระเงิน

ระบบจะคำนวณราคารวมอัตโนมัติ และเต็รีียมข้อมูลสำหรับขั้นตอนการชำระเงิน

6.หน้าประวัติการสั่งซื้อ (Order History Page)

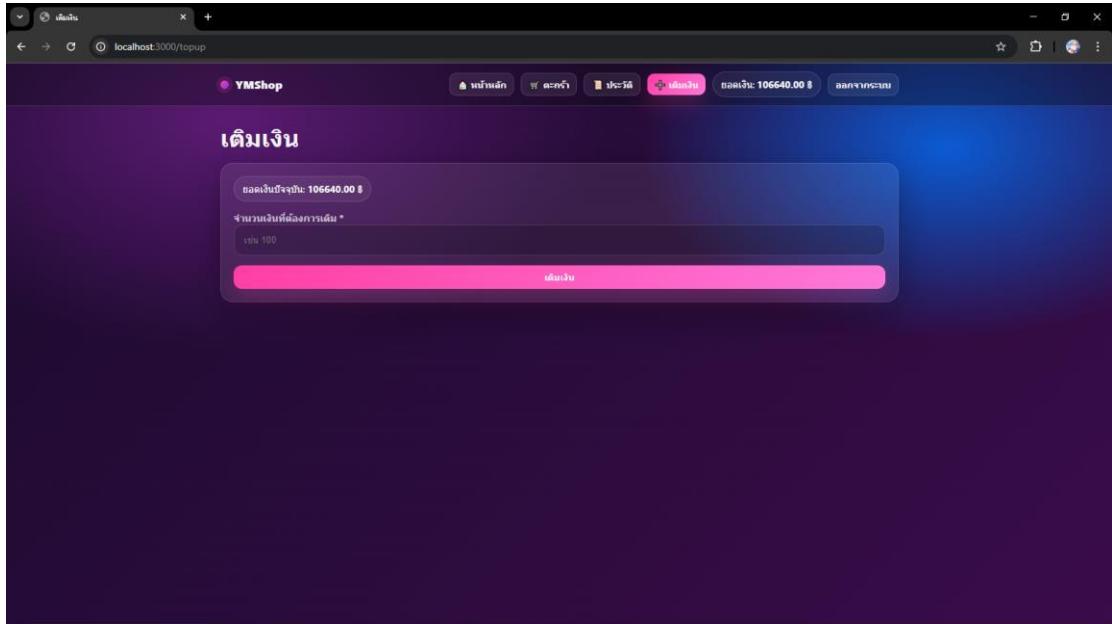


คำอธิบาย:

หน้านี้แสดงรายการการคำสั่งซื้อย้อนหลังของผู้ใช้งาน โดยแสดงเลขที่คำสั่งซื้อ วันที่สั่งซื้อ และยอดรวม

ช่วยให้ผู้ใช้งานสามารถตรวจสอบประวัติการทำรายการได้อย่างสะดวก

7.หน้าเติมเงิน (Top-up Page)



คำอธิบาย:

หน้าเติมเงินใช้สำหรับเพิ่มยอดเงินเข้าสู่บัญชีของผู้ใช้งาน โดยผู้ใช้สามารถกรอกจำนวนเงิน เลือก

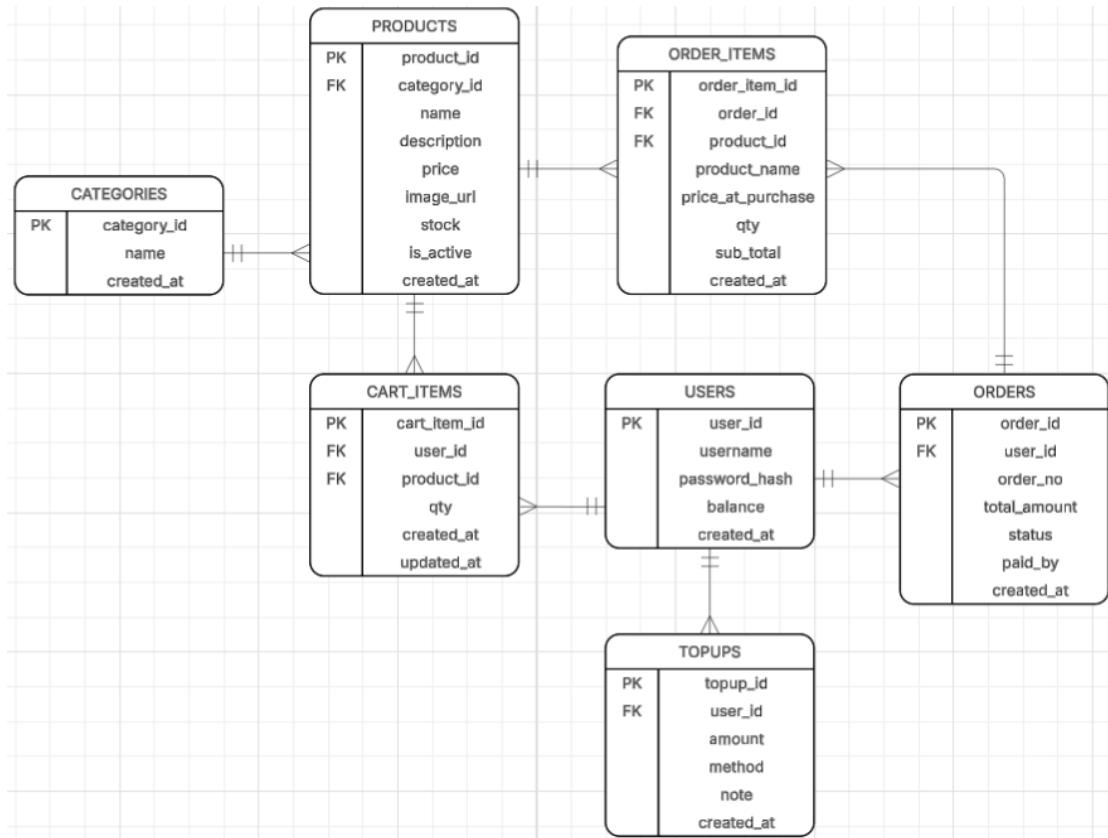
วิธีการชำระเงิน และกดปุ่มยืนยันรายการ

เมื่อทำการสมัครสมาชิก ระบบจะ:

บันทึกข้อมูลในตาราง Topups

อัปเดตยอดเงิน (balance) ในตาราง Users

Database Schema (ER Diagram)



1. ตาราง USERS

ใช้เก็บข้อมูลผู้ใช้งานในระบบ

ชื่อฟิลด์	ชนิดข้อมูล	รายละเอียด
user_id (PK)	INT	รหัสผู้ใช้งาน
username	VARCHAR	ชื่อผู้ใช้
password_hash	VARCHAR	รหัสผ่านที่เข้ารหัส
balance	DECIMAL	ยอดเงินคงเหลือ
created_at	DATETIME	วันที่สมัครสมาชิก

2. ตาราง CATEGORIES

ใช้เก็บหมวดหมู่สินค้า

ชื่อฟิลด์	ชนิดข้อมูล	รายละเอียด
category_id (PK)	INT	รหัสหมวดหมู่
name	VARCHAR	ชื่อหมวดหมู่
created_at	DATETIME	วันที่สร้าง

3. ตาราง PRODUCTS

ใช้เก็บข้อมูลสินค้า

ชื่อฟิลด์	ชนิดข้อมูล	รายละเอียด
product_id (PK)	INT	รหัสสินค้า
category_id (FK)	INT	อ้างอิงหมวดหมู่
name	VARCHAR	ชื่อสินค้า
description	TEXT	รายละเอียดสินค้า
price	DECIMAL	ราคาสินค้า
image_url	VARCHAR	รูปสินค้า
stock	INT	จำนวนคงเหลือ
is_active	BOOLEAN	สถานะสินค้า
created_at	DATETIME	วันที่เพิ่มสินค้า

4. ตาราง CART_ITEMS

เก็บข้อมูลสินค้าที่อยู่ในตะกร้า

ชื่อฟิลด์	ชนิดข้อมูล	รายละเอียด
cart_item_id (PK)	INT	รหัสรายการสินค้าในตะกร้า
user_id (FK)	INT	อ้างอิงผู้ใช้
product_id (FK)	INT	อ้างอิงสินค้า
qty	INT	จำนวนสินค้า
created_at	DATETIME	วันที่เพิ่มลงตะกร้า
updated_at	DATETIME	วันที่แก้ไขรายการ

5. ตาราง ORDERS

ใช้เก็บข้อมูลคำสั่งซื้อ

ชื่อฟิลด์	ชนิดข้อมูล	รายละเอียด
order_id (PK)	INT	รหัสคำสั่งซื้อ
user_id (FK)	INT	อ้างอิงผู้ใช้
order_no	VARCHAR	เลขที่คำสั่งซื้อ
total_amount	DECIMAL	ยอดรวมทั้งหมด
status	VARCHAR	สถานะคำสั่งซื้อ
paid_by	VARCHAR	วิธีการชำระเงิน
created_at	DATETIME	วันที่สร้างคำสั่งซื้อ

6. ตาราง ORDER_ITEMS

ใช้เก็บรายละเอียดสินค้าที่อยู่ในคำสั่งซื้อ

ชื่อฟิลด์	ชนิดข้อมูล	รายละเอียด
order_item_id (PK)	INT	รหัสรายการสินค้าในคำสั่งซื้อ
order_id (FK)	INT	อ้างอิงคำสั่งซื้อ
product_id (FK)	INT	อ้างอิงสินค้า
product_name	VARCHAR	ชื่อสินค้าขณะสั่งซื้อ
price_at_purchase	DECIMAL	ราคាសินค้าขณะซื้อ
qty	INT	จำนวนสินค้า
sub_total	DECIMAL	ราคารวมต่อรายการ
created_at	DATETIME	วันที่บันทึกรายการ

7. ตาราง TOPUPS

ใช้เก็บข้อมูลการเติมเงินของผู้ใช้

ชื่อฟิลด์	ชนิดข้อมูล	รายละเอียด
topup_id (PK)	INT	รหัสรายการเติมเงิน
user_id (FK)	INT	อ้างอิงผู้ใช้
amount	DECIMAL	จำนวนเงินที่เติม
method	VARCHAR	วิธีการเติมเงิน
note	TEXT	หมายเหตุเพิ่มเติม
created_at	DATETIME	วันที่เติมเงิน

ความสัมพันธ์ระหว่างตาราง

CATEGORIES 1:N PRODUCTS

USERS 1:N CART_ITEMS

USERS 1:N ORDERS

USERS 1:N TOPUPS

ORDERS 1:N ORDER_ITEMS

PRODUCTS 1:N CART_ITEMS

PRODUCTS 1:N ORDER_ITEMS

คู่มือการติดตั้ง (Install) + Create/Import Database + Launch Website

สิ่งที่ต้องติดตั้งก่อน (Prerequisites)

Node.js (แนะนำ v16 ขึ้นไป)

ตรวจสอบว่าเครื่องมีแล้วไหม: เปิด Terminal/PowerShell และพิมพ์

```
node -v
```

```
npm -v
```

MySQL (เลือกอย่างใดอย่างหนึ่ง)

ใช้ XAMPP (ง่าย) → เปิด MySQL ผ่าน XAMPP Control Panel

หรือใช้ MySQL ติดตั้งแยก (MySQL Server)

Git (ถ้าใช้ clone จาก GitHub)

ตรวจสอบ: git --version

ดาวน์โหลด/Clone โปรเจคจาก GitHub

เปิด Terminal/PowerShell ไปที่โฟลเดอร์ที่ต้องการเก็บงาน เช่น Desktop

ใช้คำสั่ง:

```
git clone https://github.com/67021860/YMShop cd YMShop
```

ติดตั้ง Library ผ่าน npm (Dependencies)

เปิด Terminal/PowerShell เข้าโฟลเดอร์โปรเจคที่มี package.json เช่น

```
cd C:\Users\<ชื่อคุณ>\Documents\PJWEB\YMShop\project
```

ติดตั้งแพ็กเกจ:

```
npm install
```

ถ้าสำเร็จจะมีโฟลเดอร์ node_modules โผล่มา

การสร้างฐานข้อมูล (Create Database)

กรณีใช้ XAMPP + phpMyAdmin (ง่ายที่สุด)

เปิด XAMPP Control Panel → กด Start ที่ MySQL

เปิดเบราว์เซอร์เข้า: <http://localhost/phpmyadmin>

ไปที่เมนู New (ซ้ายมือ)

ตั้งชื่อฐานข้อมูล เช่น ymshop

เลือก Collation แนะนำ: utf8mb4_unicode_ci

กด Create

Import ฐานข้อมูลจากไฟล์ .sql (Create & Import)

โปรเจคคุณมีไฟล์ SQL อยู่ที่:

db/schema.sql

Import ผ่าน phpMyAdmin

ใน phpMyAdmin เลือกฐานข้อมูลที่สร้างไว้ (เช่น ymshop)

กดแท็บ Import

กด Choose File และเลือกไฟล์ db/schema.sql

เลื่อนลงล่างสุด กด Import

รอจนเข็ญว่า Import สำเร็จ → จะเห็นตารางในฐานข้อมูล

หลัง Import ควรเห็นอย่างน้อย 5 ตารางขึ้นไป เช่น

users, categories, products, cart_items, orders, order_items, topups (ตาม ERD ของคุณ)

ถ้า Import แล้ว “ไม่มีข้อมูล” แปลว่าไฟล์ schema.sql อาจสร้างแค่โครงสร้างตาราง (ไม่มี INSERT)

ซึ่งไม่ได้

ตั้งค่า Environment (.env)

ในโปรเจคคุณมีไฟล์ตัวอย่าง: .env.example

ทำให้ถูกต้อง (แนะนำสุด)

ก็อปปี้ .env.example และเปลี่ยนชื่อเป็น .env

เปิดไฟล์ .env และกรอกค่าตามเครื่องคุณ เช่น

PORT=3000

DB_HOST=localhost

DB_USER=root

DB_PASSWORD=... (ถ้า XAMPP ปกติ root นักว่าง แต่แล้วแต่เครื่อง)

DB_NAME=ymshop

SESSION_SECRET=... (ตั้งอะไรมีได้ให้เดายาก)

Launch Website (การรันเว็บ)

เปิด Terminal/PowerShell ที่ไฟล์เดอร์โปรเจค

สั่งรัน:

npm start

หรือถ้าใน package.json ไม่มี start script ใช้:

node app.js

ถ้า Terminal ขึ้นประมาณ:

Server running: http://localhost:3000

ให้เข้าเว็บที่:

<http://localhost:3000>

อธิบายการทำงาน “แต่ละฟังก์ชัน” ของเว็บไซต์ (ละเอียด)

หน้าเข้าสู่ระบบ (Login)

views/login.ejs

วัตถุประสงค์: ให้ผู้ใช้เข้าสู่ระบบเพื่อใช้งานระบบทะกร้า/สั่งซื้อ/เติมเงิน

การทำงานหลัก

ผู้ใช้กรอก ชื่อผู้ใช้งาน และ รหัสผ่าน

ระบบตรวจสอบข้อมูลกับตาราง USERS

หากถูกต้อง ระบบสร้าง session และพาไปหน้าแรก/หน้าหมวดหมู่

มีตัวเลือก แสดงรหัสผ่าน (checkbox) เพื่อสลับ password/text

ข้อมูลที่เกี่ยวข้องตาราง: users

ฟิลด์สำคัญ: username, password_hash

หน้าสมัครสมาชิก (Register) — views/register.ejs

วัตถุประสงค์: เพิ่มผู้ใช้ใหม่เข้าสู่ระบบ

การทำงานหลัก

ผู้ใช้กรอก Username ตามเงื่อนไข (ตัวอักษร+ตัวเลข, ชื่นต้นด้วยตัวอักษร ฯลฯ)

กรอกรหัสผ่าน และยืนยันรหัสผ่าน

ระบบตรวจสอบว่า username ซ้ำหรือไม่

ระบบเข้ารหัสรหัสผ่าน (hash) และบันทึก users

ข้อมูลที่เกี่ยวข้อง

ตาราง: users

หน้าเลือกหมวดหมู่ (Categories)

views/categories.ejs

วัตถุประสงค์: ให้ผู้ใช้เลือกหมวดเพื่อคัดลอกสินค้า

การทำงานหลัก

ดึงข้อมูลหมวดหมู่จากตาราง categories

แสดงรายการหมวดหมู่เป็น tile/card

เมื่อกดหมวด ระบบจะไปหน้าแสดงสินค้าในหมวดนั้น

ข้อมูลที่เกี่ยวข้อง

ตาราง: categories

ใช้สัมพันธ์กับ products.category_id

หน้ารายการสินค้าในหมวด (Products List)

views/products.ejs

วัตถุประสงค์: ให้ผู้ใช้ดูสินค้าในหมวด และหยิบเข้าตะกร้า

การทำงานหลัก

ระบบดึงรายการสินค้าจาก products ตาม category_id

แสดงรูปสินค้า ชื่อ ราคา

ผู้ใช้สามารถเลือกจำนวน และกด “หยิบใส่ตะกร้า”

เมื่อยิบใส่ตะกร้า ระบบบันทึก/อัปเดตในตาราง cart_items

ถ้าสินค้านั้นมีในตะกร้าแล้ว → เพิ่มจำนวน (update qty)

ถ้ายังไม่มี → เพิ่มรายการใหม่ (insert)

ข้อมูลที่เกี่ยวข้อง

ตาราง: products, cart_items

หน้ารายละเอียดสินค้า (Product Detail)

views/product_detail.ejs

วัตถุประสงค์: ให้ผู้ใช้ดูรายละเอียดสินค้าแบบเต็มก่อนเพิ่มลงตะกร้า

การทำงานหลัก

รับ product_id จาก URL

ดึงข้อมูลสินค้าจาก products

แสดงรายละเอียด (รูป, ชื่อ, ราคา, คำอธิบาย, stock)

สามารถกดเพิ่มลงตะกร้าได้เมื่อนหน้ารายการสินค้า

ข้อมูลที่เกี่ยวข้อง

ตาราง: products, cart_items

หน้าต่างกร้าสินค้า (Cart)

views/cart.ejs

+

views/cart_items.ejs

วัตถุประสงค์: ให้ผู้ใช้ตรวจสอบรายการสินค้า ปรับจำนวน และเลือกชำระเงิน

การทำงานหลัก

ระบบดึงรายการจาก cart_items ตาม user_id

join กับ products เพื่อเอา ชื่อ/รูป/ราคา

ผู้ใช้:

กด + / - เพื่อปรับจำนวน → update cart_items.qty

ติ๊กเลือกสินค้าเฉพาะชิ้นได้

กด “เลือกทั้งหมด” เพื่อร่วมยอด

ระบบคำนวณราคารวมจากการที่ติ๊ก ($qty \times price$)

ข้อมูลที่เกี่ยวข้อง

ตาราง: cart_items, products

หน้าชำระเงิน (Checkout)

views/checkout.ejs

วัตถุประสงค์: สร้างคำสั่งซื้อจากสินค้าที่เลือกในตะกร้า

การทำงานหลัก

รับรายการที่เลือกจากตะกร้า (selected cart ids)

ระบบตรวจสอบยอดเงินใน users.balance ว่าพอหรือไม่

ถ้ายอดเงินพอ:

สร้างคำสั่งซื้อใหม่ใน orders

สร้างรายการสินค้าใน order_items

หักยอดเงินออกจาก users.balance

ลบรายการที่ checkout แล้วออกจาก cart_items

ถ้ายอดเงินไม่พอ:

แจ้งเตือน และแนะนำไปหน้าเติมเงิน

ข้อมูลที่เกี่ยวข้อง

ตาราง: orders, order_items, users, cart_items

หน้าเติมเงิน (Topup)

views/topup.ejs

วัตถุประสงค์: เพิ่มเครดิต/ยอดเงินให้ผู้ใช้เพื่อใช้สั่งซื้อ

การทำงานหลัก

ผู้ใช้กรอกจำนวนเงินและวิธีเติม (ถ้ามี)

ระบบบันทึกประวัติลงตาราง topups

ระบบอัปเดตยอดเงินใน users.balance (+amount)

แจ้งผลสำเร็จ และสามารถลับไปชื่อสินค้าได้

ข้อมูลที่เกี่ยวข้อง

ตาราง: topups, users

หน้าประวัติคำสั่งซื้อ/ประวัติการทำรายการ (History)

views/history.ejs

วัตถุประสงค์: ให้ผู้ใช้ดูประวัติการสั่งซื้อย้อนหลัง

การทำงานหลัก

ดึงข้อมูลจาก orders ตาม user_id

ดึงรายละเอียดแต่ละออเดอร์จาก order_items

แสดงวันที่ ยอดรวม สถานะ และรายการสินค้าที่ซื้อ

ข้อมูลที่เกี่ยวข้อง

ตาราง: orders, order_items

ปัญหาที่พบบ่อย + วิธีแก้

ปัญหา: เข้าเว็บไม่ได้ / Error “Failed to lookup view ‘login’”

สาเหตุที่พบบ่อย:

ชื่อไฟล์ view ไม่ตรงกับที่ res.render('login') เรียก (ตัวพิมพ์เล็ก/ใหญ่สำคัญ)

โฟลเดอร์ views ไม่อยู่ตำแหน่งที่ app กำหนด

วิธีแก้:

ตรวจสอบว่ามี views/login.ejs จริง และในโค้ดใช้ res.render('login')

ตรวจใน app.js ว่ามี:

```
app.set('views', path.join(__dirname, 'views'))
```

```
app.set('view engine', 'ejs')
```

คูมีการใช้งานเว็บไซต์แยกตามลิทธิ์

ระบบเว็บไซต์ร้านค้าออนไลน์ (YMShop) ถูกออกแบบให้มีการควบคุมสิทธิ์การใช้งาน เพื่อให้การเข้าถึงข้อมูลและการทำงานแต่ละส่วนเป็นไปอย่างเหมาะสม โดยในระบบมีผู้ใช้งานหลัก

2 ประเภท ได้แก่

1. ผู้ใช้งานทั่วไป (User / Customer)

2. ผู้ดูแลระบบ (Admin)

1. ผู้ใช้งานทั่วไป (User / Customer)

ผู้ใช้งานทั่วไปคือสมาชิกที่สมัครและเข้าสู่ระบบเพื่อซื้อสินค้าในเว็บไซต์ โดยมีสิทธิ์ดังนี้

1.1 การสมัครสมาชิก (Register)

ขั้นตอนการใช้งาน:

เข้าเว็บไซต์ที่ URL: <http://localhost:3000>

กดปุ่ม “สมัครสมาชิก”

กรอกข้อมูล:

ชื่อผู้ใช้งาน (Username)

รหัสผ่าน

ยืนยันรหัสผ่าน

กดปุ่ม “สมัครสมาชิก”

หากข้อมูลถูกต้อง ระบบจะบันทึกข้อมูลลงฐานข้อมูลและสามารถเข้าสู่ระบบได้

เงื่อนไข: ชื่อผู้ใช้ต้องไม่ซ้ำกับผู้ใช้อื่นรหัสผ่านต้องตรงกับช่องยืนยันรหัสผ่าน

1.2 การเข้าสู่ระบบ (Login)

ขั้นตอนการใช้งาน:

กรอก Username และ Password

กดปุ่ม “เข้าสู่ระบบ”

หากข้อมูลลูกค้าต้อง ระบบจะสร้าง Session และพาเข้าสู่หน้าแรก

กรณีข้อมูลผิด: ระบบจะแจ้งเตือนว่า Username หรือ Password “ไม่ถูกต้อง”

1.3 การเลือกหมวดหมู่สินค้า (Categories)

ผู้ใช้สามารถ:

ดูรายการหมวดหมู่สินค้า

คลิกเลือกหมวดหมู่เพื่อดูสินค้าในหมวดนั้น

ระบบจะดึงข้อมูลจากตาราง categories และแสดงรายการสินค้าในตาราง

products

1.4 การดูรายละเอียดสินค้า (Product Detail)

ผู้ใช้สามารถ:

ดูรูปสินค้า

ดูรีวิวสินค้า

ดูรายละเอียดสินค้า

ดูราคา

สามารถกด “เพิ่มลงตะกร้า” ได้จากหน้านี้

1.5 การเพิ่มสินค้าใส่ตะกร้า (Add to Cart)

ขั้นตอน:

เลือกจำนวนสินค้า

กดปุ่ม “เพิ่มลงตะกร้า”

การทำงานของระบบ:

ถ้ายังไม่มีสินค้านั้นในตะกร้า → เพิ่มรายการใหม่

ถ้ามีอยู่แล้ว → เพิ่มจำนวนสินค้า (qty)

ข้อมูลถูกบันทึกในตาราง cart_items

1.6 การจัดการตะกร้าสินค้า (Cart)

ผู้ใช้สามารถ:

ดูรายการสินค้าในตะกร้า

ปรับเพิ่ม/ลดจำนวนสินค้า

ลบสินค้าออกจากตะกร้า

เลือกสินค้าเพื่อทำการชำระเงิน

ดูยอดรวมของสินค้าที่เลือก

ระบบจะคำนวณยอดรวม = ราคา × จำนวน

1.7 การชำระเงิน (Checkout)

ขั้นตอน:

เลือกสินค้าที่ต้องการซื้อ

กดปุ่ม “ชำระเงิน”

ระบบจะตรวจสอบยอดเงินในบัญชีผู้ใช้ (balance)

หากยอดเงินเพียงพอ:

สร้างคำสั่งซื้อในตาราง orders

สร้างรายการสินค้าใน order_items

หักยอดเงินจาก users.balance

ลบสินค้าที่ซื้อออกจาก cart_items

หากยอดเงินไม่พอ:

แจ้งเตือน และแนะนำให้เปลี่ยนต่อไป

1.8 การเติมเงิน (Top-up)

ขั้นตอน:

เข้าเมนู “เติมเงิน”

กรอกจำนวนเงิน

กดปุ่ม “ยืนยัน”

ระบบจะ:

บันทึกข้อมูลในตาราง topups

เพิ่มยอดเงินใน users.balance

หลังเติมเงินสำเร็จ ผู้ใช้สามารถกลับไปทำการสั่งซื้อได้

1.9 การดูประวัติการสั่งซื้อ (Order History)

ผู้ใช้สามารถ:

ดูรายการคำสั่งซื้อย้อนหลัง

ดูวันที่สั่งซื้อ

ดูยอดรวม

ดูสถานะคำสั่งซื้อ

ดูรายการสินค้าที่ซื้อในแต่ละออเดอร์

ข้อมูลดึงจาก:

ตาราง orders

ตาราง order_items

2. สิทธิ์: ผู้ดูแลระบบ (Admin)

2.1 จัดการหมวดหมู่สินค้า

สามารถ:

ເພີ່ມໜາວດໜູ້ໃໝ່

ແກ້ໄຂຫຼືອໜາວດໜູ້

ລບ່ມໜາວດໜູ້

ຂໍ້ມູນລັບຈັດເກີບໃນຕາຮາງ categories

2.2 ຈັດກາຮສິນຄ້າ (Products Management)

ສາມາຮດ:

ເພີ່ມສິນຄ້າໃໝ່

ແກ້ໄຂຮາຍລະເອີຍດສິນຄ້າ

ແກ້ໄຂຮາຄາ

ແກ້ໄຂຈຳນວນ stock

ລບສິນຄ້າ

ຂໍ້ມູນລັບຈັດເກີບໃນຕາຮາງ products

2.3 ຕຽບສອບຄໍາສັ່ງຫຼືອ

ສາມາຮດ:

ດູຮາຍກາຮຄໍາສັ່ງຫຼືອທັງໝາດ

ເປົ້າຍືນສຖານະຄໍາສັ່ງຫຼືອ (ເຫັນ ຂໍາຮະແລ້ວ / ຈັດສ່າງແລ້ວ)

ຂໍ້ມູນຈາກຕາຮາງ orders

สรุปผลโครงการ

โครงการเว็บไซต์ร้านค้าออนไลน์ (YMSShop) นี้เป็นการพัฒนาระบบ Web Application เพื่อรับรองการซื้อขายสินค้าออนไลน์ โดยมีวัตถุประสงค์เพื่อออกแบบและพัฒนาระบบที่สามารถจัดการข้อมูลสินค้า ผู้ใช้งาน คำสั่งซื้อ และธุกรรมทางการเงินได้อย่างเป็นระบบ โดยใช้เทคโนโลยี Node.js และ Express Framework ในส่วน Backend และใช้ EJS ร่วมกับ HTML, CSS และ JavaScript ในส่วน Frontend พร้อมเชื่อมต่อฐานข้อมูล MySQL เพื่อจัดเก็บและบริหารจัดการข้อมูลทั้งหมดของระบบ

จากการพัฒนาโครงการ พบร่วมกัน พบว่าสามารถดำเนินการพัฒนาระบบได้ครบตามขอบเขตที่กำหนด โดยระบบสามารถรองรับฟังก์ชันหลัก ได้แก่ การสมัครสมาชิก การเข้าสู่ระบบ การแสดงหมวดหมู่สินค้า การแสดงรายการสินค้า การดูรายละเอียดสินค้า การเพิ่มสินค้าเข้าสู่ตะกร้า การจัดการตะกร้าสินค้า การเติมเงินเข้าสู่ระบบ การทำรายการสั่งซื้อ และการแสดงประวัติคำสั่งซื้อย้อนหลัง ซึ่งทุกฟังก์ชันมีการเชื่อมโยงกับฐานข้อมูลผ่านโครงสร้างตารางที่ออกแบบไว้ใน ER Diagram จำนวน 7 ตาราง ได้แก่ users, categories, products, cart_items, orders, order_items และ topups โดยมีการกำหนด Primary Key และ Foreign Key อย่างเหมาะสม เพื่อให้เกิดความสมัมพันธ์ของข้อมูลอย่างถูกต้องและป้องกันความซ้ำซ้อน

ในด้านการจัดการฐานข้อมูล ระบบสามารถทำงานได้อย่างถูกต้องตามกระบวนการ CRUD (Create, Read, Update, Delete) เช่น การเพิ่มข้อมูลสินค้า การปรับปรุงจำนวนสินค้า การลบรายการสินค้าออกจากตะกร้า และการบันทึกคำสั่งซื้อ โดยมีการคำนวณยอดรวมอัตโนมัติและอัปเดตยอดเงินคงเหลือของผู้ใช้งานเมื่อทำการชำระเงินสำเร็จ อีกทั้งยังมีระบบเติมเงินที่ช่วยเพิ่มยอดเงินเข้าสู่บัญชีผู้ใช้ และบันทึกประวัติธุกรรมไว้อย่างชัดเจน

ในส่วนของความปลอดภัย ระบบมีการเข้ารหัสผ่านผู้ใช้งานก่อนจัดเก็บในฐานข้อมูล (password hashing) และมีการใช้ session สำหรับจัดการสถานะการเข้าสู่ระบบของผู้ใช้งาน เพื่อป้องกันการเข้าถึงข้อมูลโดยไม่ได้รับอนุญาต ส่งผลให้ระบบมีความปลอดภัยในระดับพื้นฐานที่เหมาะสมกับงานพัฒนาในระดับรายวิชา

จากการทดสอบระบบ พบร่วมระบบสามารถทำงานได้อย่างถูกต้องครบถ้วนตามขั้นตอนการใช้งานจริง โดยผู้ใช้งานสามารถดำเนินการตั้งแต่การสมัครสมาชิกจนถึงการสั่งซื้อสินค้าได้โดยไม่มีข้อผิดพลาดที่ส่งผลต่อการทำงานหลักของระบบ ทั้งนี้ยังสามารถแสดงผลข้อมูลได้อย่างถูกต้องและเชื่อมโยงข้อมูลระหว่างตารางฐานข้อมูลได้อย่างมีประสิทธิภาพอย่างไรก็ตาม โครงการนี้ยังมีข้อจำกัดบางประการ เช่น ยังไม่มีระบบชำระเงินผ่านช่องทางภายนอก (เช่น Payment Gateway จริง) ยังไม่มีระบบจัดการสิทธิ์หลายระดับ (Role Management) และยังไม่มีระบบรายงานเชิงวิเคราะห์ (Report Dashboard) สำหรับผู้ดูแลระบบ ซึ่งสามารถพัฒนาเพิ่มเติมในอนาคตได้

โดยรวมแล้ว โครงการเว็บไซต์ร้านค้าออนไลน์นี้บรรลุวัตถุประสงค์ที่ตั้งไว้ สามารถประยุกต์ใช้ความรู้ด้าน Web Programming, Database Design และการพัฒนา Web Application แบบ Full-stack ได้อย่างครบถ้วน อีกทั้งยังช่วยเสริมทักษะด้านการวิเคราะห์ระบบ การออกแบบฐานข้อมูล และการทำงานร่วมกันเป็นทีม ซึ่งเป็นพื้นฐานสำคัญสำหรับการพัฒนาระบบสารสนเทศในระดับที่สูงขึ้นต่อไป

ข้อมูลผู้พัฒนา (Developer Information)

ชื่อโครงการ: ระบบร้านค้าออนไลน์ (YMSShop)

รหัสนักศึกษา ชื่อ-นามสกุล หน้าที่รับผิดชอบ

67021860 นันทพงศ์ ปวงศ์ ตำแหน่ง: Frontend Developer

- ChatgptPlus Backend Developer