

เว็บไซต์อาหารโรงสัสม

คู่มือผู้พัฒนาและการติดตั้ง

Date: 12 ตุลาคม พ.ศ. 2565

By

นางสาวกมลชนก ศรีหา 6220503163

นายวัชรศักดิ์ ชื่นชม 6220503341

นายเสฏฐนันท์ เลี่ยมयोगโย 6220503368

สารบัญ

เรื่อง	หน้า
สารบัญ	2
ที่มาและความสำคัญ	3
วัตถุประสงค์	3
ประเภทผู้ใช้งาน	4
แนวคิดเกี่ยวกับการออกแบบ	4
ส่วนของพัฒนาระบบ	5
รายละเอียดภาษาและโปรแกรมที่ใช้	5-13
การติดตั้ง	14-21
ภาพรวมของระบบ	22
การออกแบบ	23
1. การออกแบบฐานข้อมูล	23-26
2. การออกแบบเว็บไซต์	27-33
3. Use Case Diagram	34-37
การพัฒนาระบบและการติดตั้ง	38
1. ฐานข้อมูล	38
2. การเขียนหน้าเว็บไซต์	39
หลักการดึง API โดยใช้ Axios และ Await	39-43
การกำหนดหน้าตาม Path โดยใช้ Route ใน Library react-router-dom	44
รายละเอียดหน้าเว็บไซต์	45-49
3. การเขียนระบบหลังเว็บไซต์	50
ระบบล็อกอินและสมัครสมาชิก	50-54
การจัดการเมนูอาหารสำหรับเจ้าของร้าน	54-65
การแสดงผลร้านอาหารและเมนูอาหารตามร้าน	66-67
การสั่งอาหาร	68
การตรวจสอบสถานะรายการอาหารสำหรับลูกค้า	69-76
การตรวจสอบสถานะรายการอาหารลูกค้าสำหรับเจ้าของร้าน	76-85
การจัดการสถานะรายการคำสั่งซื้ออาหารของลูกค้าสำหรับเจ้าของร้าน	85-89
การจัดการข้อมูลร้านอาหารสำหรับเจ้าของร้าน	90-92
การแสดงผลรายงานสถิติยอดขายของร้านอาหาร	93-97
Project team Lesson Learned	98-99

เว็บไซต์อาหารโรงสั้

ที่มาและความสำคัญ

เนื่องจากทางมหาวิทยาลัยเกษตรศาสตร์ วิทยาเขตกำแพงแสน มีโรงอาหารสำหรับให้พ่อค้า หรือแม่ค้า ได้ประกอบอาชีพร้านอาหาร และจำหน่ายอาหารให้กับนิสิต อาจารย์ และบุคลากรทางมหาวิทยาลัย เนื่องจากในช่วงพักกลางวันเป็นช่วงเวลาหลังเลิกเรียน นิสิต อาจารย์ และบุคลากรทางมหาวิทยาลัยส่วนมากเลือกรับประทานอาหารภายในบริเวณมหาวิทยาลัย เพื่อความสะดวก และเตรียมตัวสำหรับการเรียนการสอนในวิชาถัดไปในช่วงบ่าย ทำให้มีลูกค้าจำนวนมากรอต่อแถวซื้ออาหารในช่วงเวลาพักกลางวัน คนขายอาหารจำเป็นต้องรับคำสั่งอาหารจากลูกค้าหลายรายการ อาจทำให้รายการอาหารตกหล่นได้ และลูกค้าที่มาสั่งอาหารไม่สามารถรู้ได้ว่ารายการอาหารที่ตนเองสั่งจะได้รับเมื่อไหร่ นอกจากเดินเข้าไปถามคนขายอาหารโดยตรง ทำให้ไม่สะดวกในการรอรับอาหาร และลูกค้าบางส่วนต้องการที่จะสั่งอาหารล่วงหน้าเพราะไม่สะดวกที่จะมาสั่งอาหาร ณ ช่วงเวลานั้นๆ เมื่อถึงเวลาจะสามารถเข้ามารับอาหารได้ที่ทันที โดยไม่จำเป็นต้องเดินเข้าไปสั่งอาหารใหม่ที่ร้านอีกครั้ง

ด้วยทางคณะผู้จัดทำเห็นว่าเทคโนโลยีระบบสั่งอาหารออนไลน์ กำลังเป็นที่นิยมในปัจจุบัน ทางคณะผู้จัดทำจึงได้มีความคิดสร้างเว็บไซต์สั่งอาหารโรงสั้เพื่อสำหรับให้คนขายอาหารสามารถจัดการ เพิ่ม/แก้ไข และลบเมนูอาหารภายในร้านบนเว็บไซต์ และจัดการรายการคำสั่งอาหารของลูกค้าผ่านทางเว็บไซต์ได้ และลูกค้าสามารถสั่งอาหาร และเช็คสถานะของรายการอาหารที่ตัวเองสั่งผ่านทางเว็บไซต์ได้

วัตถุประสงค์

- 2.1 เพื่อสร้างและพัฒนาเว็บไซต์สั่งอาหารโรงสั้
- 2.2 เพื่อศึกษาภาษา JavaScript , HTML/CSS , และ React ในการพัฒนาเว็บไซต์
- 2.3 เพื่อศึกษา และจัดทำฐานข้อมูลที่เกี่ยวข้องกับระบบสั่งอาหารออนไลน์

ประเภทของผู้ใช้งาน

การพัฒนาเว็บไซต์สั่งอาหารโรงสั่ม แบ่งประเภทของผู้ใช้เป็น 2 ส่วนดังนี้

1. ผู้ใช้ทั่วไป

- ผู้ใช้ทั่วไปสามารถสมัครสมาชิก และเข้าสู่ระบบเพื่อใช้งานเว็บไซต์สั่งอาหารโรงสั่มได้
- ผู้ใช้ทั่วไปสามารถกดเลือกร้านอาหาร และกดสั่งอาหารได้ครั้งละ 1 ร้าน
- ใช้ทั่วไปสามารถตรวจสอบเช็คสถานะของรายการอาหารที่ตัวเองสั่งได้

2. เจ้าของร้านอาหาร

- เจ้าของร้านอาหารสามารถเข้าสู่ระบบเพื่อใช้งานเว็บไซต์สั่งอาหารโรงสั่มได้
- เจ้าของร้านอาหารสามารถ ดู/แก้ไขสถานะ/ลบ รายการคำสั่งซื้ออาหารของผู้ใช้ทั่วไปในร้านอาหารของตนเองได้
- เจ้าของร้านอาหารสามารถ ดู/เพิ่ม/แก้ไข/ลบ รายการอาหารภายในร้านอาหารของตนเองได้
- เจ้าของร้านอาหารสามารถแก้ไขข้อมูลร้านอาหารของตนเองได้
- เจ้าของร้านอาหารสามารถเช็คสถิติยอดขายเมนูประจำร้าน/ยอดขายตามรายวัน/ยอดขายตามรายเดือน/ยอดขายตามรายปีได้

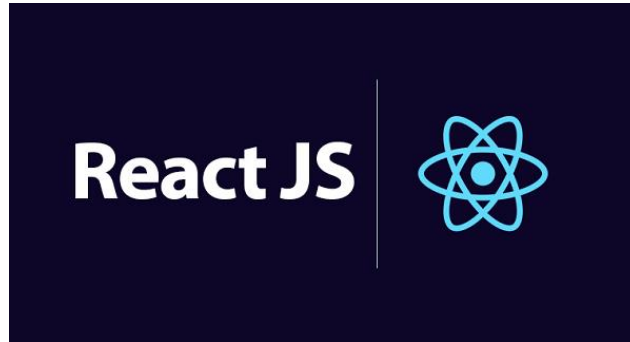
แนวคิดเกี่ยวกับการออกแบบโปรแกรม

คณะผู้จัดทำต้องการพัฒนาระบบเว็บไซต์สั่งอาหารโรงสั่มเพื่อให้นิสิต อาจารย์ และบุคลากรทางมหาวิทยาลัยได้รับความสะดวกในการสั่งอาหารในโรงอาหารของมหาวิทยาลัย ติดตามรายการคำสั่งซื้ออาหาร และมารับอาหารตามสถานะ เพื่อลดการต่อแถวและรอนำร้านอาหาร และเพื่อให้เจ้าของร้านอาหารสามารถจัดการเมนูอาหาร ดูสถิติภายในร้านอาหารของตนเอง และจัดการคำสั่งซื้ออาหารของลูกค้าได้ ทำให้เจ้าของร้านอาหารทำงานได้อย่างมีประสิทธิภาพมากขึ้น

ส่วนของการพัฒนาระบบ

รายละเอียดภาษาและโปรแกรมที่ใช้

1. React



ภาพที่ 1 โลโก้ React

React เป็น JavaScript library ที่ใช้สำหรับสร้าง user interface ที่ให้เราสามารถเขียนโค้ดในการสร้าง UI ที่มีความซับซ้อนแบ่งเป็นส่วนเล็กๆออกจากกันได้ ซึ่งแต่ละส่วนสามารถแยกการทำงานออกจากกันได้อย่างอิสระ และทำให้สามารถนำชิ้นส่วน UI เหล่านั้นไปใช้ซ้ำได้อีก

2. JavaScript



ภาพที่ 2 โลโก้ JavaScript

JavaScript คือ ภาษาคอมพิวเตอร์สำหรับการเขียนโปรแกรมบนระบบอินเทอร์เน็ต ที่กำลังได้รับความนิยมอย่างสูง Java JavaScript เป็น ภาษาสคริปต์เชิงวัตถุ (ที่เรียกกันว่า “สคริปต์” (script) ซึ่งในการสร้างและพัฒนาเว็บไซต์ (ใช้ร่วมกับ HTML) เพื่อให้เว็บไซต์ของเราดูมีการเคลื่อนไหว สามารถตอบสนองผู้ใช้งานได้มากขึ้น ซึ่งมีวิธีการทำงานในลักษณะ “แปลความและดำเนินงานไปที่ละคำสั่ง” (interpret) หรือเรียกว่า อ็อบเจ็กโอเรียนเตด (Object Oriented Programming) ที่มีเป้าหมายในการ ออกแบบและพัฒนาโปรแกรมในระบบอินเทอร์เน็ต สำหรับผู้เขียนด้วยภาษา HTML สามารถทำงานข้ามแพลตฟอร์มได้ โดยทำงานร่วมกับ ภาษา HTML และภาษา Java ได้ทั้งทางฝั่งไคลเอนต์ (Client) และ ทางฝั่งเซิร์ฟเวอร์ (Server)

3. ภาษา HTML/CSS

3.1 ภาษา HTML

HTML ย่อมาจาก HyperText Markup Language เป็นภาษาคอมพิวเตอร์ที่ใช้สร้างหน้าเว็บ (WebPage) ในรูปแบบของไฟล์ HTML (คือไฟล์ที่มีนามสกุลเป็น .htm หรือ .html) ซึ่งมีเว็บเบราว์เซอร์ (WebBrowser) เป็นโปรแกรมที่ใช้แปลงไฟล์ HTML เพื่อแสดงผลในรูปแบบของหน้าเว็บ ไฟล์ HTML เป็นไฟล์รหัสแอสกี (ASCII) ถูกบันทึกในรูปแบบของไฟล์เอกสาร (Text File) ที่สามารถถูกสร้างจากโปรแกรมสร้างไฟล์ข้อความ (Text Editor) เช่น Notepad หรือ Word Processing ทั่วๆ ไป ซึ่งลักษณะของไฟล์ HTML ประกอบไปด้วยแท็ก (Tag) ต่างๆ ที่เป็นคำสั่งของ HTML ซึ่งแท็กจะอยู่ภายในเครื่องหมาย < และ > แท็กใน HTML แบ่งเป็น 2 ประเภทคือคอนเทนเนอร์แท็ก (Container Tag) และแท็กเปล่า (Empty Tag) โดยที่คอนเทนเนอร์แท็กประกอบไปด้วยแท็กเปิด และแท็กปิด โดยที่แท็กปิดจะมีเครื่องหมาย/ นำหน้าแท็ก เช่น <H1> . . . </H1> ส่วนแท็กเปล่าจะมีแท็กเปิดอย่างเดียว เช่น <HR> ซึ่งแท็กจะถูกเขียนด้วยตัวอักษรพิมพ์ใหญ่หรือพิมพ์เล็กก็ได้จะไม่มีผลต่อการแสดงผลของเว็บเบราว์เซอร์ เช่น
,
,
 หรือ
 เว็บเบราว์เซอร์จะแปลความหมายเหมือนกัน

โครงสร้างไฟล์ HTML แบ่งออกเป็น 2 ส่วน คือ ส่วนหัวเรื่อง (Head Section) และส่วนเนื้อหา (Body Section) โดยจะมีแท็ก <HTML> และ </HTML> เป็นตัวกำหนดขอบเขตไฟล์ซึ่งส่วนหัวเรื่อง มีไว้กำหนดข้อมูลเฉพาะของหน้าเว็บ เช่น ชื่อเรื่องของเว็บภายในแท็ก <HEAD> และ </HEAD> และสำหรับส่วนเนื้อหามีไว้กำหนดรายละเอียดต่างๆ ที่ต้องการแสดงบนหน้าเว็บเช่น ข้อความ และรูปภาพภายในแท็ก <BODY> และ </BODY>

3.2 ภาษา CSS

CSS ย่อมาจาก Cascading Style Sheet มักเรียกโดยย่อว่า "สไตลชีต" คือภาษาที่ใช้เป็นส่วนของการจัดรูปแบบการแสดงผลเอกสาร HTML โดยที่ CSS กำหนดกฎเกณฑ์ในการระบุรูปแบบ (หรือ "Style") ของเนื้อหาในเอกสาร อันได้แก่ สีของข้อความ สีพื้นหลัง ประเภทตัวอักษร และการจัดวางข้อความ ซึ่งการกำหนดรูปแบบ หรือ Style นี้ใช้หลักการของการแยกเนื้อหาเอกสาร HTML ออกจากคำสั่งที่ใช้ในการจัดรูปแบบการแสดงผล กำหนดให้รูปแบบของการแสดงผลเอกสาร ไม่ขึ้นอยู่กับเนื้อหาของเอกสาร เพื่อให้ง่ายต่อการจัดรูปแบบการแสดงผลลัพท์ของเอกสาร HTML โดยเฉพาะในกรณีที่มีการเปลี่ยนแปลงเนื้อหาเอกสารบ่อยครั้ง หรือต้องการควบคุมให้รูปแบบการแสดงผลเอกสาร HTML มีลักษณะของความสม่ำเสมอทั่วกันทุกหน้าเอกสารภายในเว็บไซต์เดียวกัน โดยกฎเกณฑ์ในการกำหนดรูปแบบ (Style) เอกสาร HTML ถูกเพิ่มเข้ามาครั้งแรกใน

HTML 4.0 เมื่อปีพ.ศ. 2539 ในรูปแบบของ CSS level 1 Recommendations ที่กำหนดโดย องค์กร World Wide Web Consortium หรือ W3C

ประโยชน์ของ CSS

1. CSS มีคุณสมบัติมากกว่า tag ของ html เช่น การกำหนดกรอบให้ข้อความ รวมทั้งสี รูปแบบของข้อความที่กล่าวมาแล้ว
2. CSS นั้นกำหนดที่ต้นของไฟล์ html หรือตำแหน่งอื่น ๆ ก็ได้ และสามารถมีผล กับเอกสารทั้งหมด หมายถึงกำหนด ครั้งเดียวจุดเดียวก็มีผลกับการแสดงผลทั้งหมด ทำให้เวลาแก้ไขหรือปรับปรุงทำได้สะดวก ไม่ต้องไล่ตามแก้ tag ต่างๆ ทั้งทั้งเอกสาร
3. CSS สามารถกำหนดแยกไว้ต่างหากจาก ไฟล์เอกสาร html และสามารถนำมาใช้ร่วม กับเอกสารหลายไฟล์ได้ การแก้ไขก็แก้เพียง จุดเดียวก็มีผลกับเอกสารทั้งหมด
4. ภาษา MySQL



ภาพที่ 3 ภาพโลโก้ MySQL

MySQL คือ โปรแกรมระบบจัดการฐานข้อมูล ที่พัฒนาโดยบริษัท MySQL AB มีหน้าที่เก็บข้อมูลอย่างเป็นระบบ รองรับคำสั่ง SQL เป็นเครื่องมือสำหรับเก็บข้อมูล ที่ต้องใช้ร่วมกับเครื่องมือหรือโปรแกรมอื่นอย่าง บูรณาการ เพื่อให้ได้ระบบงานที่รองรับ ความต้องการของผู้ใช้ เช่นทำงานร่วมกับเครื่องบริการเว็บ (Web Server) เพื่อให้บริการแก่ภาษาสคริปต์ที่ทำงานฝั่งเครื่องบริการ (Server-Side Script) เช่น ภาษา php ภาษา aps.net หรือภาษาเจเอสพี เป็นต้น หรือทำงานร่วมกับโปรแกรมประยุกต์ (Application Program) เช่น ภาษา วิวอลเบสิกคอตเน็ต ภาษาจาวา หรือภาษาซีชาร์ป เป็นต้น โปรแกรมถูกออกแบบให้สามารถทำงานได้บนระบบปฏิบัติการที่หลากหลาย และเป็นระบบฐานข้อมูลโอเพนทเซอร์ท (Open Source) ที่ถูกนำไปใช้งานมาก

ที่สุด MySQL จัดเป็นระบบจัดการฐานข้อมูลเชิงสัมพันธ์ (RDBMS : Relational Database Management System) ซึ่งเป็นที่นิยมใช้กันมากในปัจจุบัน

MySQL เป็นระบบจัดการฐานข้อมูลโดยใช้ภาษา SQL. แม้ว่า MySQL เป็นซอฟต์แวร์โอเพนซอร์ส แต่มันแตกต่างจากซอฟต์แวร์โอเพนซอร์สทั่วไป โดยมีการพัฒนาภายใต้บริษัท MySQL AB ในประเทศสวีเดน โดยจัดการ MySQL ทั้งในแบบที่ให้ฟรี และแบบที่ใช้ในเชิงธุรกิจ MySQL สร้างขึ้นโดยชาวสวีเดน 2 คน และชาวฟินแลนด์ ชื่อ David Axmark, Allan Larsson และ Michael "Monty" Widenius. ปัจจุบันบริษัทซันไมโครซิสเต็มส์ (Sun Microsystems, Inc.) เข้าซื้อกิจการของ MySQL AB เรียบร้อยแล้ว ฉะนั้นผลิตภัณฑ์ภายใต้ MySQL AB ทั้งหมดจะตกเป็นของซัน

5. Node.js



ภาพที่ 4 ภาพโลโก้ Nodejs

NodeJs คือ เครื่องมือที่ใช้ในการ run ภาษา JavaScript หรือ เรียกว่า Environment ในการ run JavaScript โดย NodeJs ถูกสร้างเมื่อปี 2009 และถูกสร้างขึ้นบน chrome's v8 javascript engine และทุกๆ browser ก็จะมี engine หรือเครื่องมือที่ใช้ในการ compiles code ของเราไปยัง ภาษาเครื่อง หรือ machine code ซึ่ง chrome นั้นใช้เครื่องมือตรงนี้ที่เรียกว่า V8 ซึ่งโดยตัว Node เป็นเครื่องมือที่จะช่วยจัดการทั้งฝั่งของ front-end และ back-end โดยอยู่บนพื้นฐานของ JavaScriptซึ่งในการใช้งานของ NodeJs เราจะต้องอาศัยพื้นฐานจาก HTML, CSS และ JavaScript และถ้าเป็น JavaScript ต้องมีพื้นฐานความเข้าใจ เกี่ยว es6 callbacks promises และ sync await

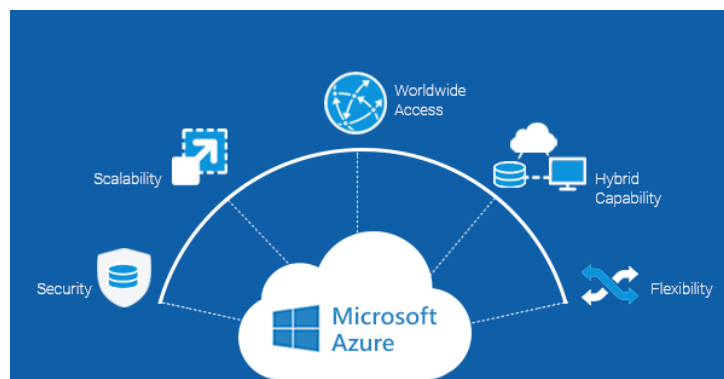
6. โปรแกรม Visual Studio Code



ภาพที่ 5 ภาพโลโก้ Visual Studio Code

VS Code หรือ Visual Studio Code จากบริษัทไมโครซอฟต์ เป็นโปรแกรมประเภท Editor ใช้ในการแก้ไขโค้ดที่มีขนาดเล็ก แต่มีประสิทธิภาพสูง เป็น OpenSource โปรแกรมจึงสามารถนำมาใช้งานได้โดยไม่ มีค่าใช้จ่าย เหมาะสำหรับนักพัฒนาโปรแกรมที่ต้องการใช้งานหลายแพลตฟอร์ม รองรับการใช้งานทั้งบน Windows , macOS และ Linux รองรับหลายภาษาทั้ง JavaScript, TypeScript และ Node.js ในตัว และสามารถเชื่อมต่อกับ Git ได้ง่าย สามารถนำมาใช้งานได้ง่ายไม่ซับซ้อน มีเครื่องมือและส่วนขยายต่าง ๆ ให้เลือกใช่มากมาย รองรับการใช้งานภาษาอื่น ๆ ทั้ง ภาษา C++ , C# , Java , Python , PHP หรือ Go สามารถปรับเปลี่ยน Themes ได้ มีส่วน Debugger และ Commands เป็นต้น

7. Microsoft Azure



ภาพที่ 6 ภาพ Microsoft Azure

Microsoft Azure เป็นแพลตฟอร์ม (Platform) สำหรับให้บริการศูนย์ข้อมูลสำหรับองค์กร ในรูปแบบคลาวด์ (Cloud) ซึ่งมีศูนย์ข้อมูลกระจายอยู่ตามภูมิภาคต่างๆ (Region) ทั่วโลก กว่า 60 ภูมิภาค มีบริการต่างๆ กว่า 200 บริการให้เลือกใช้ตามความเหมาะสมขององค์กร

รูปแบบการให้บริการของ Microsoft Azure

เป็นการให้บริการในรูปแบบการรับผิดชอบร่วมกัน (Shared Responsibility) ระหว่างลูกค้า และ Microsoft และมีการแบ่งรูปแบบการรับผิดชอบร่วมกันเป็น 3 รูปแบบคือ

7.1 IaaS (Infrastructure as a Service)

การทำงานที่ Microsoft รับผิดชอบการให้บริการโครงสร้างพื้นฐานต่างๆ ที่จำเป็นในศูนย์ข้อมูล (Datacenter) เช่นเครื่องแม่ข่าย, ระบบเครือข่าย, ตลอดจนพื้นที่เก็บข้อมูลต่างๆ ส่วนบริการอื่นๆ อันต่อเนื่องไปเช่น ระบบปฏิบัติการ ลูกค้าจะต้องทำการบริหารจัดการเอง ตัวอย่างของ IaaS เช่น Virtual Machine, Storage Account เป็นต้น การทำงานในรูปแบบนี้เป็นที่นิยมใช้มากในการย้ายระบบโครงสร้างพื้นฐานมาจาก ศูนย์ข้อมูลขององค์กร (On-premise)

ตัวอย่างการบริการรูปแบบ IaaS

- Azure Virtual Machine (VM)

เป็นการเปิดให้ใช้งานเครื่องคอมพิวเตอร์ในรูปแบบ Virtual Machine เช่นเดียวกับที่ใช้ใน On-premise โดย Azure Virtual Machine นั้น รองรับระบบปฏิบัติการหลากหลาย ทั้ง Windows Server, Linux จากผู้พัฒนาต่างๆ เช่น Redhat, Ubuntu, Debian และยังรองรับซอฟต์แวร์ชั้นนำต่างๆ เช่น SAP, Oracle อีกด้วย นอกจากนี้ Azure VM ยังมี VM หลากหลายรูปแบบให้เลือกใช้บริการเพื่อให้ตอบโจทย์การใช้งานอีกด้วย ซึ่ง Azure VM นั้นนับว่าเป็นรูปแบบที่ง่ายที่สุดในการเริ่มย้ายระบบขึ้นมาใช้งาน Microsoft Azure

- Virtual Network (VNET)

เป็นการสร้างระบบเครือข่ายเสมือน เพื่อรองรับการเชื่อมต่อต่างๆ ภายใน Microsoft Azure เช่น เชื่อมต่อ Azure VM เข้ากับระบบเครือข่ายอินเทอร์เน็ต และนอกจากนี้ VNET ยังเป็นส่วนสำคัญที่ช่วยให้เชื่อมต่อระหว่าง Microsoft Azure เข้ากับศูนย์ข้อมูลขององค์กร (On-premise Datacenter) ได้อย่างปลอดภัย ข้อมูลเพิ่มเติม

7.2 PaaS (Platform as a Service)

รูปแบบการให้บริการแบบ PaaS นี้ทาง Microsoft จะมีการดูแลระบบปฏิบัติการ และเครื่องมือต่างๆ ในทำงานของแอปพลิเคชันเพิ่มให้จาก IaaS การให้บริการในรูปแบบ PaaS เช่น Azure App Service, Azure

Dev Ops, Azure Function, Azure SQL Database เป็นต้น การทำงานในรูปแบบนี้เหมาะสำหรับการพัฒนาแอปพลิเคชันใหม่สำหรับใช้งานบน Microsoft Azure โดยเฉพาะ ลดภาระการทำงานของผู้ดูแลระบบลงได้มาก

ตัวอย่างการบริการรูปแบบ PaaS

- Azure App Service

เป็น Platform ในการให้บริการแอปพลิเคชันต่างๆ รองรับหลากหลายเครื่องมือในการพัฒนา เช่น .NET, .NET Core, Node.js, Java, Python, PHP สามารถขยายขนาดเพื่อตอบสนองการใช้งานที่มีผู้ใช้งานจำนวนมาก รองรับการรักษาความปลอดภัยในระดับสูง รองรับการทำงานในรูปแบบ Container ทั้ง Windows และ Linux ตลอดจนสามารถตอบสนองการพัฒนาแอปพลิเคชันในรูปแบบ CI/CD (Continuous Integration/Continuous Delivery) ได้เป็นอย่างดี

- Azure SQL

ฐานข้อมูลซึ่งมีพื้นฐานของ Microsoft SQL Server รองรับฐานข้อมูลขนาดใหญ่ โดยไม่ต้องกังวลในส่วนของการตั้งค่าต่างๆ รองรับการสำรองข้อมูล และมีการติดตั้ง Update ให้โดยอัตโนมัติ

- Azure Backup

การสำรองข้อมูลอันมีค่าขององค์กรไปเก็บไว้ในพื้นที่เก็บข้อมูลที่เชื่อถือได้ของ Microsoft Azure ทำให้มั่นใจได้ว่าข้อมูลที่สำรองไว้จะสามารถเรียกคืนได้ทันทีที่ต้องการ โดยมีระยะเวลาการเก็บข้อมูลสูงสุด 10 ปี

7.3 SaaS (Software as a Service)

เป็นการให้บริการโดย Microsoft รับผิดชอบการดูแลระบบต่างๆ ให้ทั้งหมด ทำให้การบริการจะอยู่ในรูปแบบสำเร็จรูป เช่น Microsoft 365 เป็นต้น

8. Postman



ภาพที่ 7 โลโก้ POSTMAN

Postman เป็นเครื่องมือที่ไว้ใช้สำหรับ API Developers ซึ่งหากให้อธิบายในความหมายง่ายๆก็คือ โดยปกติแล้วเมื่อเราเขียน API Service ขึ้นมาเรามักจะใช้ Postman ที่เป็น API Testing tool ในการส่ง Request และดู Response ที่ได้ต่างๆกลับมา

9. Canva



ภาพที่ 8 ภาพโลโก้ Canva

Canva เป็นแอปพลิเคชันสำหรับสร้างสื่อการนำเสนอหลากหลายรูปแบบ เช่น Presentation, Poster, Card, Resume, Certificate, Infographic เป็นต้น ซึ่ง Canva นั้นจะมีขนาดมาตรฐานให้เลือกหรือผู้ใช้สามารถกำหนดขนาดเองได้ Canva ใช้งานง่าย สวยงาม สามารถแบ่งปันให้แก่ผู้อื่นได้

10. Express



ภาพที่ 9 ภาพ Node.js Express

Express.js หรือ Express เป็น Web Framework จาก NPM ที่ใช้สำหรับพัฒนาเว็บแอปพลิเคชันหรือเว็บไซต์บน Node.js ที่ทำงานที่ฝั่งของ Backend ตัวของเฟรมเวิร์กนั้นถูกพัฒนามาจากโมดูล http ซึ่งเป็นโมดูลของ Node. Express ใช้เพื่อทำให้การพัฒนาเว็บแอปพลิเคชันบน Node.js ทำได้ง่ายขึ้น และ Express.js มีคุณสมบัติที่โดดเด่นดังนี้

- การจัดการ Routing ที่ง่าย
- ฟังก์ชันช่วยสำหรับ HTTP
- สนับสนุน Template engines สำหรับสร้าง View
- ทำงานได้รวดเร็วและมีประสิทธิภาพ
- สนับสนุน Middleware

Express.js เป็นแพ็คเกจที่ได้รับความนิยมเป็นอย่างมากในการพัฒนาเว็บแอปพลิเคชันบน Node.js และมันถูกเผยแพร่อยู่บน NPM ในการใช้งาน Express สามารถช่วยให้ทำงานได้สะดวกและรวดเร็วขึ้นเนื่องจาก API ส่วนมากของมันถูกออกแบบมาเพื่อการพัฒนาเว็บไซต์โดยเฉพาะและง่ายต่อการใช้งาน แทนที่จะใช้ Low-level APIs ของ Node.js โดยตรง

การติดตั้ง

1. วิธีการติดตั้ง Microsoft Azure

1.1 ติดตั้ง ubuntu ใน VMware

ดาวน์โหลด VMware ได้ที่เว็บไซต์ : <https://kb.vmware.com/s/article/2053973>

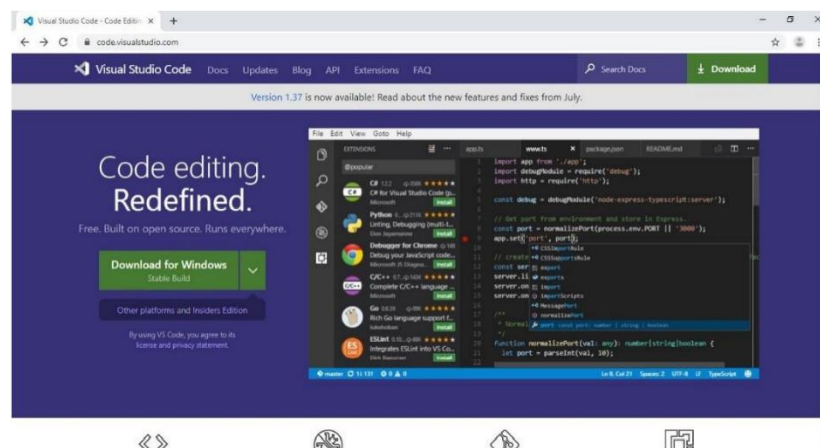
1.2 ลง Docker Compose บน ubuntu

1.3 ลง MySQL ใน Docker Compose

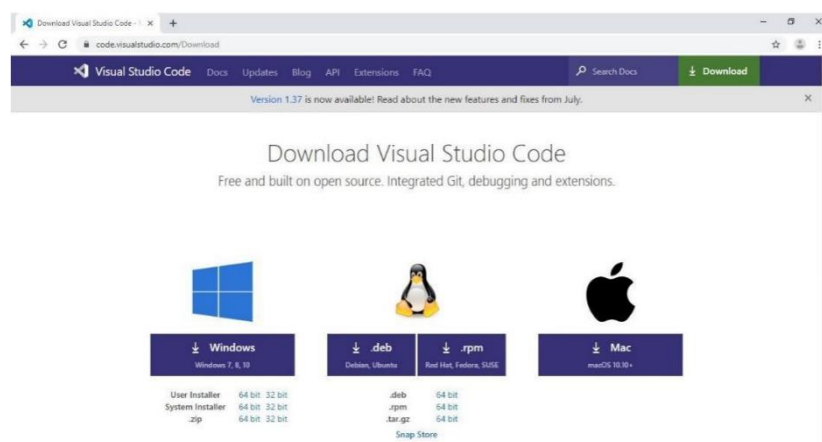
2. วิธีการติดตั้ง Visual Studio Code

ขั้นตอนการติดตั้งโปรแกรม Visual Studio Code หรือ VS Code

2.1 เข้าไปที่เว็บไซต์ <https://code.visualstudio.com/> และ Download โปรแกรม VS Code โดยเลือกให้ตรงกับ OS ของเครื่องคอมพิวเตอร์

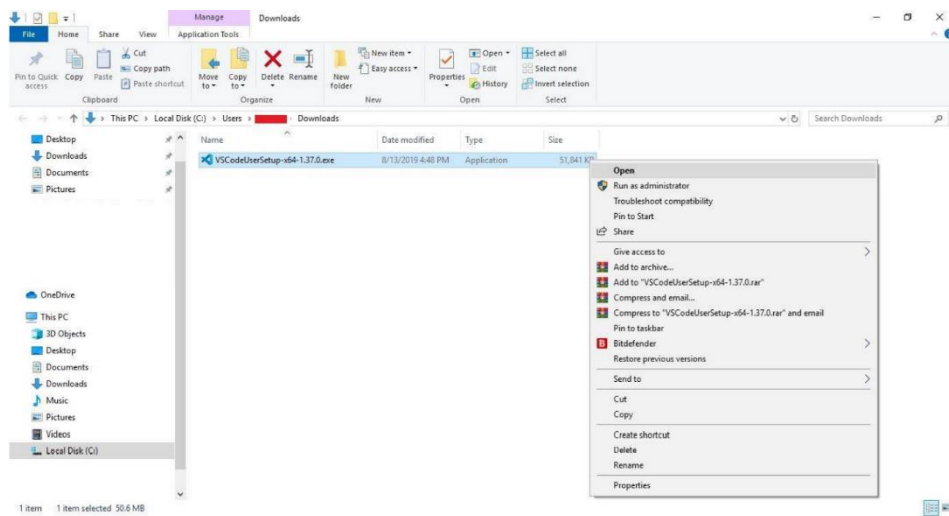


ภาพที่ 10 ภาพหน้าเว็บไซต์ติดตั้งของ Visual Studio Code



ภาพที่ 11 ภาพหน้าเว็บไซต์ติดตั้งของ Visual Studio Code ใน version OS ต่างๆ

2.2 ดับเบิลคลิก หรือคลิกขวาและกด “Open” โปรแกรมที่ดาวน์โหลดมา



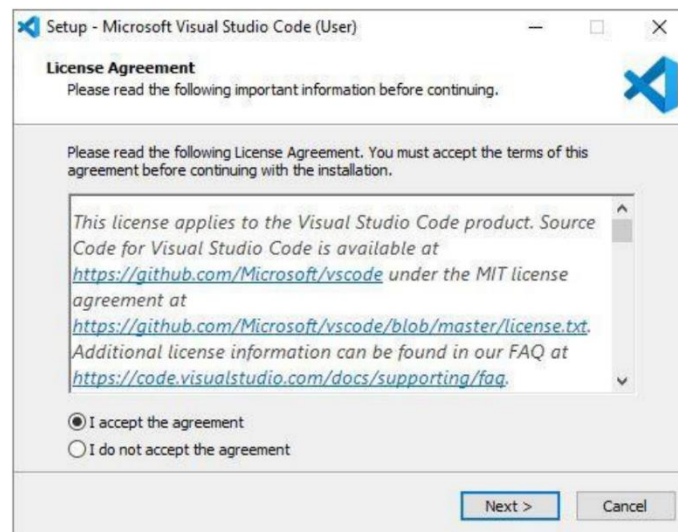
ภาพที่ 12 ภาพขั้นตอนที่ 2.2 ของการติดตั้ง Visual Studio Code

2.3 คลิกปุ่ม “Run”



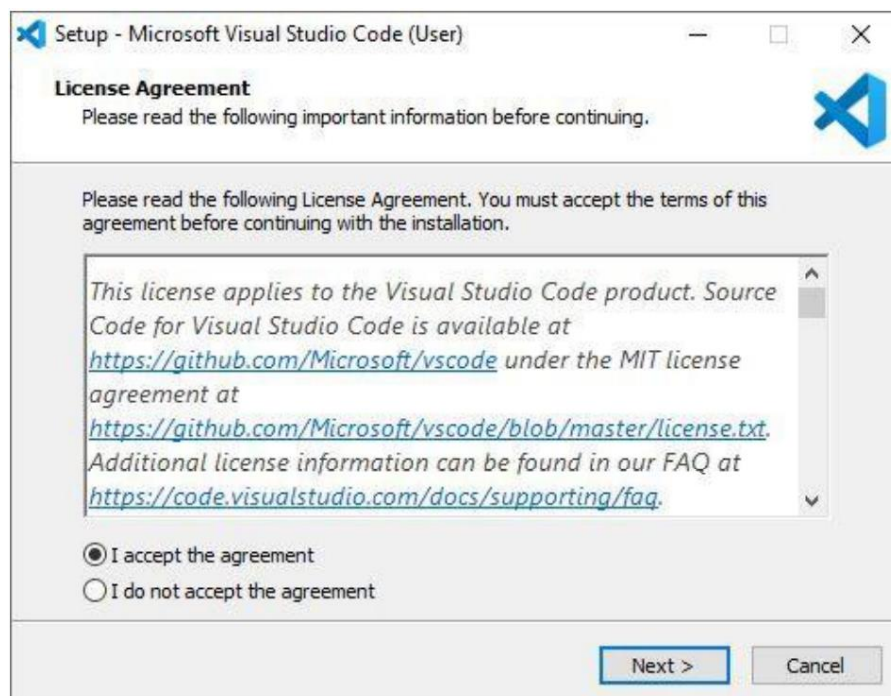
ภาพที่ 13 ภาพขั้นตอนที่ 2.3 ของการติดตั้ง Visual Studio Code

2.4 เลือก “I accept the agreement” และคลิกปุ่ม “Next >”



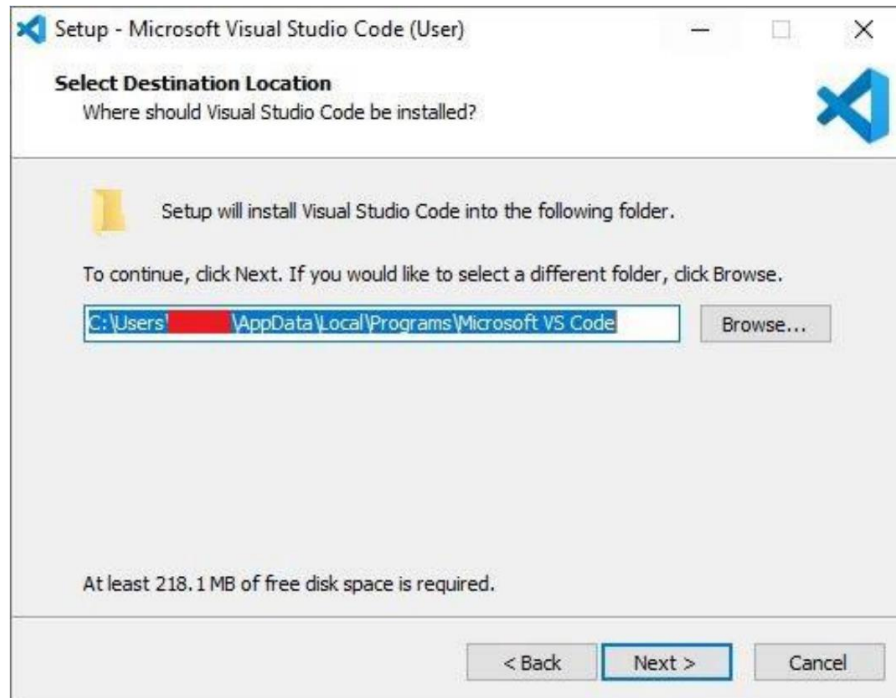
ภาพที่ 14 ภาพขั้นตอนที่ 2.4 ของการติดตั้ง Visual Studio Code

2.5 เลือก “I accept the agreement” และคลิกปุ่ม “Next >”



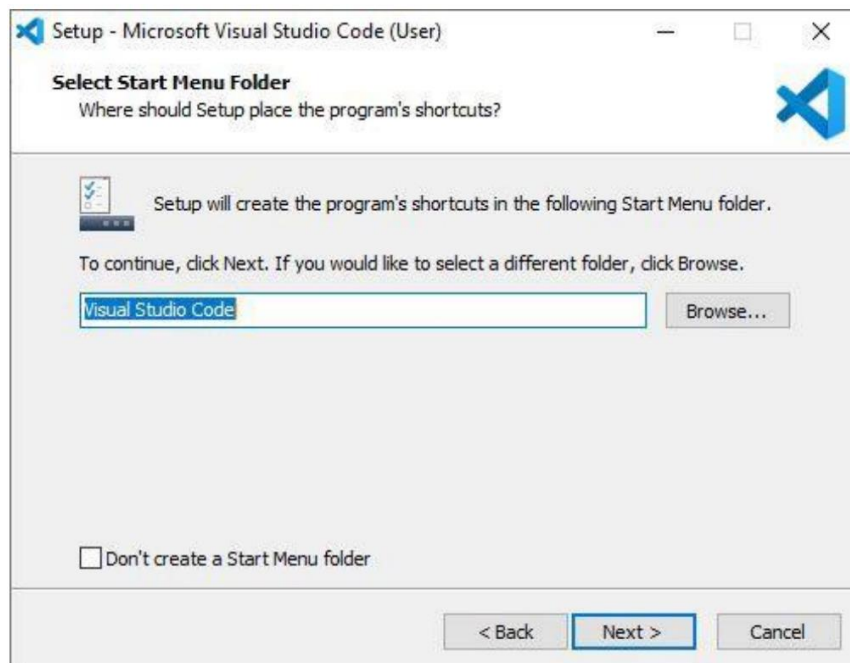
ภาพที่ 15 ภาพขั้นตอนที่ 2.5 ของการติดตั้ง Visual Studio Code

2.6 เลือกพื้นที่ในการจัดเก็บโปรแกรม (แนะนำให้ใช้ Default ที่ให้มา) และคลิกปุ่ม “Next >”



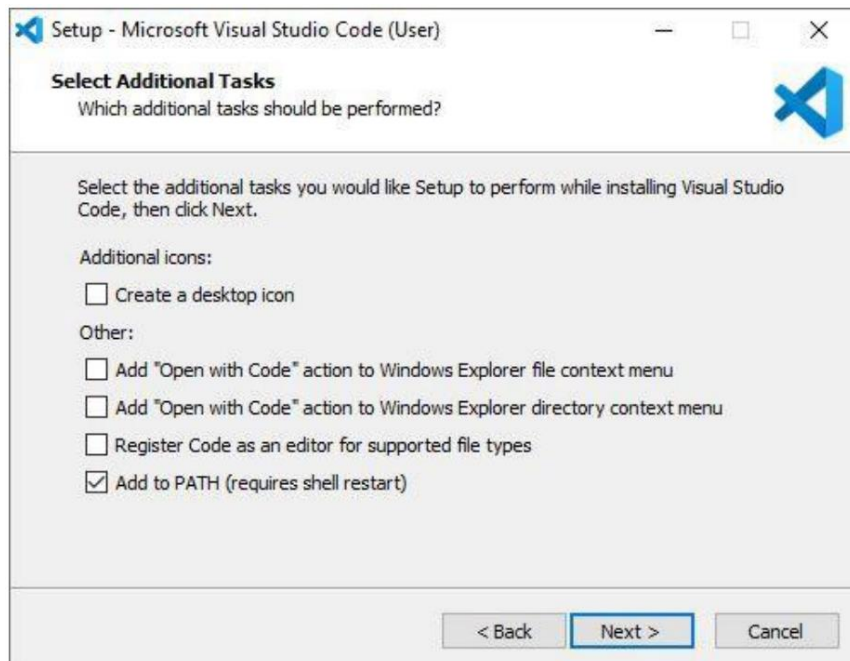
ภาพที่ 16 ภาพขั้นตอนที่ 2.6 ของการติดตั้ง Visual Studio Code

2.7 คลิกปุ่ม “Next >”



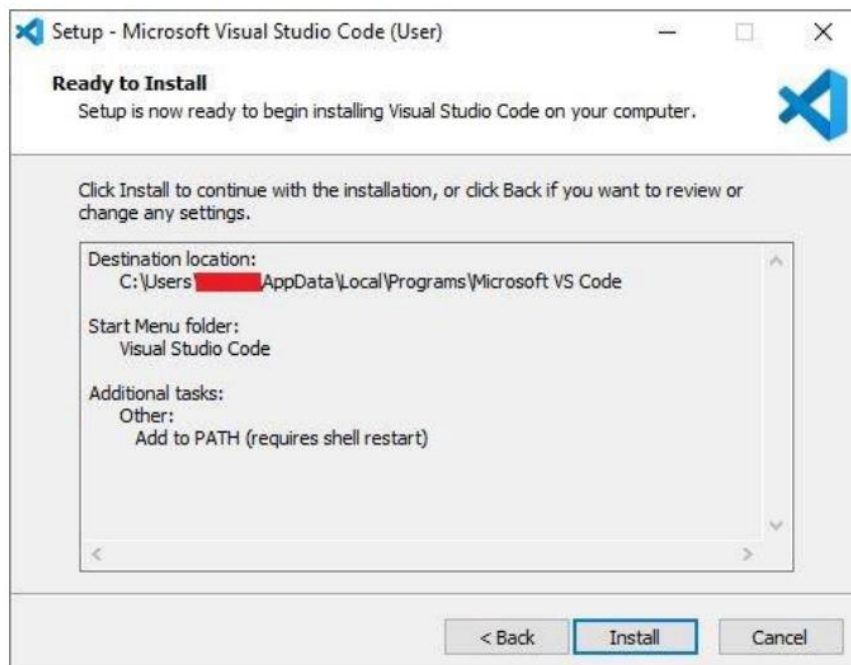
ภาพที่ 17 ภาพขั้นตอนที่ 2.7 ของการติดตั้ง Visual Studio Code

2.8 เลือกส่วนเพิ่มงานให้เลือก Create a desktop icon และ Add to PATH (requires shell restart) จากนั้นให้คลิกปุ่ม “Next >”



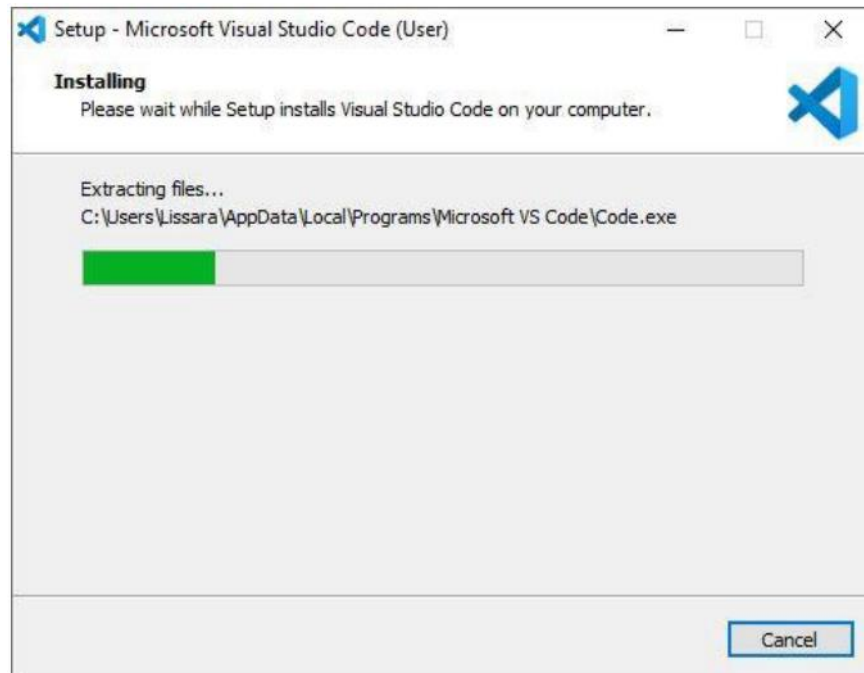
ภาพที่ 18 ภาพขั้นตอนที่ 2.8 ของการติดตั้ง Visual Studio Code

2.9 คลิกปุ่ม “Install” เพื่อติดตั้งโปรแกรม



ภาพที่ 19 ภาพขั้นตอนที่ 2.9 ของการติดตั้ง Visual Studio Code

2.10 คลิกปุ่ม “Finish” เสร้จ้สิ้นสุดการติดตั้งโปรแกรม VS Code



ภาพที่ 20 ภาพขั้นตอนที่ 2.10 ของการติดตั้ง Visual Studio Code

3. วิธีการติดตั้ง React

3.1 ติดตั้ง npm ไว้ในเครื่องให้เรียบร้อย เข้าไปดาวน์โหลดได้ลิงค์นี้

<https://www.npmjs.com/get-npm>

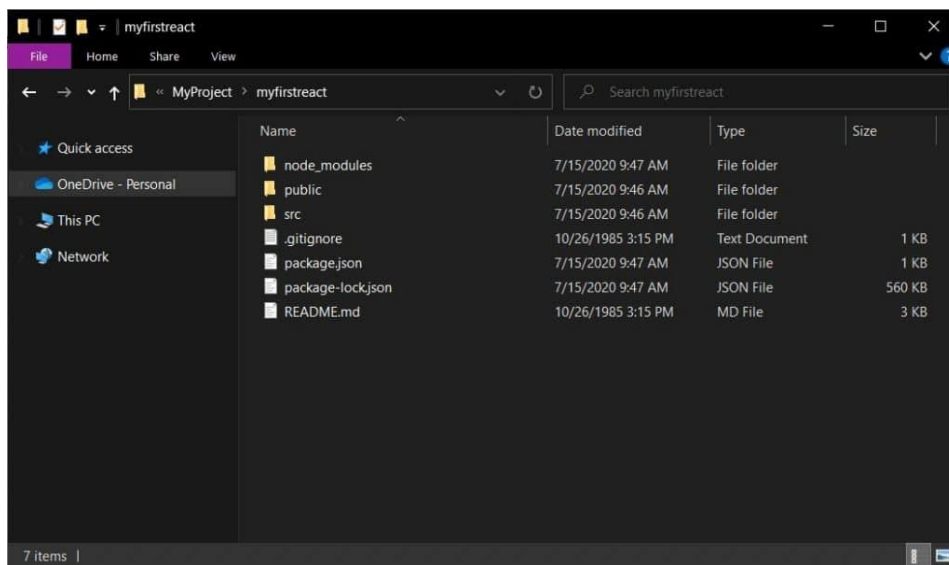
3.2 สร้างโปรเจกต์เริ่มต้นขึ้นมาจากเครื่องมือที่ชื่อ create-react-app ที่จะสร้างไฟล์พื้นฐานต่างๆที่จำเป็นในการเขียน React ขึ้นมา ให้ทำการติดตั้งด้วยคำสั่ง

```
npm install -g create-react-app
```

3.3 สร้างโปรเจกต์ใหม่ขึ้นมาด้วยคำสั่ง

```
npm create-react-app myreact
```

เราจะได้ไฟล์ต่างๆขึ้นมาประมาณนี้

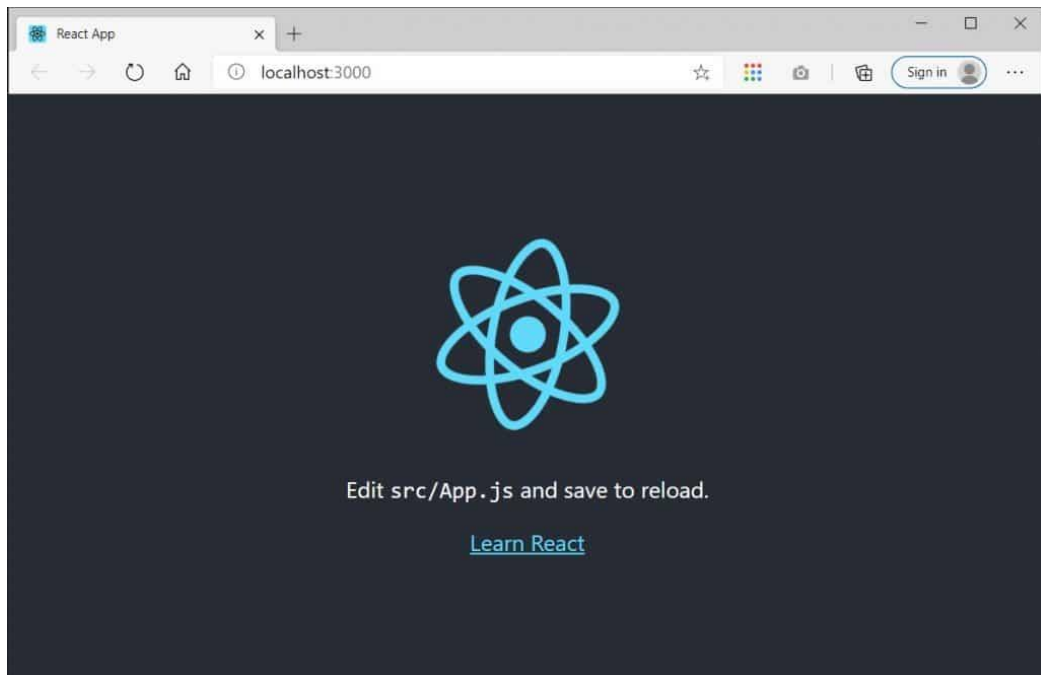


ภาพที่ 21 ภาพโฟลเดอร์หลังจากติดตั้ง React

3.4 รันด้วยคำสั่ง npm start บน Visual Studio Code

```
npm start
```

3.5 เบราเซอร์จะเปิดขึ้นมาที่ <http://localhost:3000/> ถ้ามีหน้าต่างแบบนี้แสดงว่าพร้อมใช้งาน



ภาพที่ 22 ภาพหน้าเว็บไซต์หลังจากรันคำสั่ง npm start

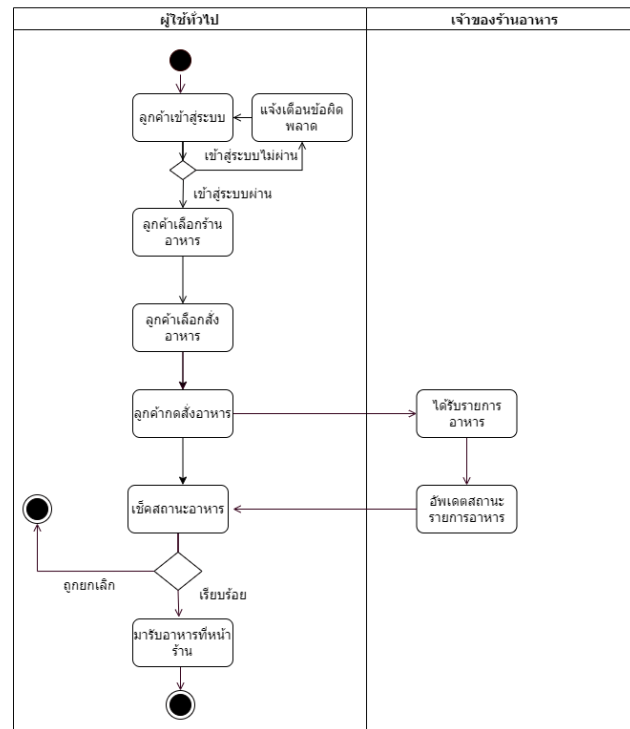
4. วิธีติดตั้ง Node.js

ดาวน์โหลด Node.js ได้ที่เว็บไซต์ : <https://nodejs.org/en/download/>

ภาพรวมของระบบ

1. Activity diagrams

ลำดับการทำงานการสั่งอาหารบนเว็บไซต์สั่งอาหารโรงสั่ม มีหลักการทำงานดังนี้



ภาพที่ 23 ภาพ Activity diagrams ของระบบสั่งอาหาร

จาก Activity diagrams เริ่มต้นจาก ผู้ใช้งานทั่วไปหรือลูกค้าเริ่มเข้าสู่ระบบ เมื่อกดเข้าสู่ระบบ ระบบจะทำการเช็ครหัสผ่าน ถ้าเข้าสู่ระบบผ่านลูกค้าจะสามารถเข้าสู่หน้าเว็บไซต์แบบมีผู้ใช้ใช้งานได้ แต่ถ้าเข้าสู่ระบบไม่ผ่าน ระบบจะทำการแจ้งเตือนข้อผิดพลาด และให้ลูกค้าทำการเข้าสู่ระบบหรือสมัครสมาชิกอีกครั้ง

หลังจากเข้าสู่ระบบผ่าน ลูกค้าสามารถกดเลือกร้านอาหาร และกดเลือกสั่งเมนูอาหารตามร้านที่เลือกได้ โดยเลือกสั่งอาหารได้ครั้งละ 1 ร้านเท่านั้น เมื่อทำการกดยืนยันการสั่งอาหารเสร็จ ระบบจะส่งข้อมูลไปพื้นฐานข้อมูลที่รวบรวมรายการคำสั่งซื้ออาหาร และเจ้าของร้านอาหารแต่ละร้านสามารถเช็ค และทำการอนุมัติรายการอาหารได้ เมื่อเจ้าของร้านทำการกดยืนยันสถานะเป็น “เรียบร้อย” แสดงว่าเมนูรายการนั้นทำเสร็จและพร้อมที่จะให้ลูกค้ามารับหน้าร้านอาหารเรียบร้อยแล้ว แต่ถ้าเจ้าของร้านทำการกดยืนยันสถานะเป็น “ถูกยกเลิก” และแจ้งเหตุผลในการถูกยกเลิก ลูกค้าจะไม่สามารถมารับอาหารดังกล่าวได้ อาจจะต้องทำการกดสั่งซื้อเมนูอื่น หรือเลือกสั่งซื้ออาหารร้านอื่นใหม่อีกครั้ง

โดยฐานข้อมูลมีรายละเอียดดังนี้

1. Customer

เป็น entity ไว้เก็บข้อมูลของลูกค้า ประกอบด้วย

- 1.1 CustomerID : เป็น Primary Key เก็บ ID
- 1.2 CustomerName : เก็บชื่อลูกค้า
- 1.3 CustomerUser : เก็บชื่อผู้ใช้ของลูกค้า
- 1.4 CustomerCol : เก็บรหัสผ่านของลูกค้า
- 1.5 CustomerPhone : เก็บเบอร์โทรของลูกค้า

2. Orderfood

เป็น entity ไว้เก็บข้อมูลของลูกค้า ประกอบด้วย

- 2.1 OrderID : เป็น Primary Key เก็บ ID
- 2.2 MenuID : เป็น Foreign Key ที่ดึงมาจาก Menu ไว้เก็บเมนูอาหาร
- 2.3 CustomerID : เป็น Foreign Key ที่ดึงมาจาก Customer ไว้เก็บข้อมูลลูกค้าที่สั่ง
- 2.4 OrderDate: เก็บวันที่สั่งอาหาร
- 2.5 OrderTime : เก็บเวลาที่มารับอาหารหน้าร้าน
- 2.6 OrderStatus : เป็น Foreign Key ที่ดึงมาจาก OrderStatus ไว้เก็บสถานะของคำสั่งซื้ออาหาร
- 2.7 OrderQty : เก็บจำนวนอาหารที่สั่ง
- 2.8 OrderPrice : เก็บราคาอาหาร
- 2.9 OrderNoteCus: เก็บรายละเอียดหมายเหตุลูกค้า
- 2.10 OrderNoteOw : เก็บรายละเอียดหมายเหตุเจ้าของร้าน

3. OrderStatus

เป็น entity ไว้เก็บข้อมูลสถานะคำสั่งซื้ออาหาร ประกอบไปด้วย

3.1 OrderStatusID : เป็น Primary Key เก็บ ID

3.2 OrderNameStatus : เก็บชื่อสถานะ

4. Menu

เป็น entity ไว้เก็บเมนูอาหาร ประกอบไปด้วย

4.1 MenuID : เป็น Primary Key เก็บ ID

4.2 StoreID : เป็น Foreign Key ที่ดึงมาจาก Store ไว้เก็บร้านอาหาร

4.3 MenuName : เก็บชื่อเมนูอาหาร

4.4 MenuPrice: เก็บราคาเมนูอาหาร

4.5 MenuStatus : เป็น Foreign Key ที่ดึงมาจาก MenuStatus ไว้เก็บสถานะของเมนูอาหาร

4.6 MenuDate : เก็บวันที่เมนูอาหาร

4.7 MenuImage : เก็บชื่อรูปภาพเมนูอาหาร

5. OrderStatus

เป็น entity ไว้เก็บข้อมูลสถานะเมนูอาหาร ประกอบไปด้วย

5.1 MenuStatusID : เป็น Primary Key เก็บ ID

5.2 MenuNameStatus : เก็บชื่อสถานะ

6. Store

เป็น entity ไว้เก็บข้อมูลร้านอาหาร ประกอบไปด้วย

6.1 StoreID : เป็น Primary Key เก็บ ID

6.2 OwnerID : เป็น Foreign Key ที่ดึงมาจาก StoreStatus ไว้เก็บสถานะของร้านอาหาร

6.3 StoreName : เก็บชื่อร้านอาหาร

6.4 StoreStatus : เป็น Foreign Key ที่ตั้งมาจาก StoreStatus ไว้เก็บสถานะของร้านอาหาร

6.5 StoreImage : เก็บชื่อรูปภาพ

7. StoreStatus

เป็น entity ไว้เก็บข้อมูลสถานะร้านอาหาร ประกอบไปด้วย

7.1 StorerStatusID : เป็น Primary Key เก็บ ID

7.2 StoreNameStatus : เก็บชื่อสถานะ

8. Owner

เป็น entity ไว้เก็บข้อมูลของเจ้าของร้านอาหาร ประกอบไปด้วย

8.1 Owner ID : เป็น Primary Key เก็บ ID

8.2 OwnerName : เก็บชื่อเจ้าของร้าน

8.3 OwnerUser : เก็บชื่อผู้ใช้ของเจ้าของร้าน

8.4 OwnerCol : เก็บรหัสผ่านของเจ้าของร้าน

8.5 OwnerPhone : เก็บเบอร์โทรของเจ้าของร้าน

2. การออกแบบเว็บไซต์

2.1 ผู้ใช้งานทั่วไป

2.1.1 หน้าเข้าสู่ระบบและสมัครสมาชิก

ORANGE CANTEEN

[หน้าหลัก](#)
[โรงอาหาร](#)
[ไม่ได้เข้าสู่ระบบ ▼](#)



เข้าสู่ระบบ

ชื่อผู้ใช้

รหัสผ่าน

ยังไม่มีบัญชี ? [สมัครสมาชิก](#)

ภาพที่ 26 ภาพออกแบบหน้าเว็บไซต์เข้าสู่ระบบ

ORANGE CANTEEN

[หน้าหลัก](#)
[โรงอาหาร](#)
[ไม่ได้เข้าสู่ระบบ ▼](#)



สมัครสมาชิก

ชื่อ-นามสกุล

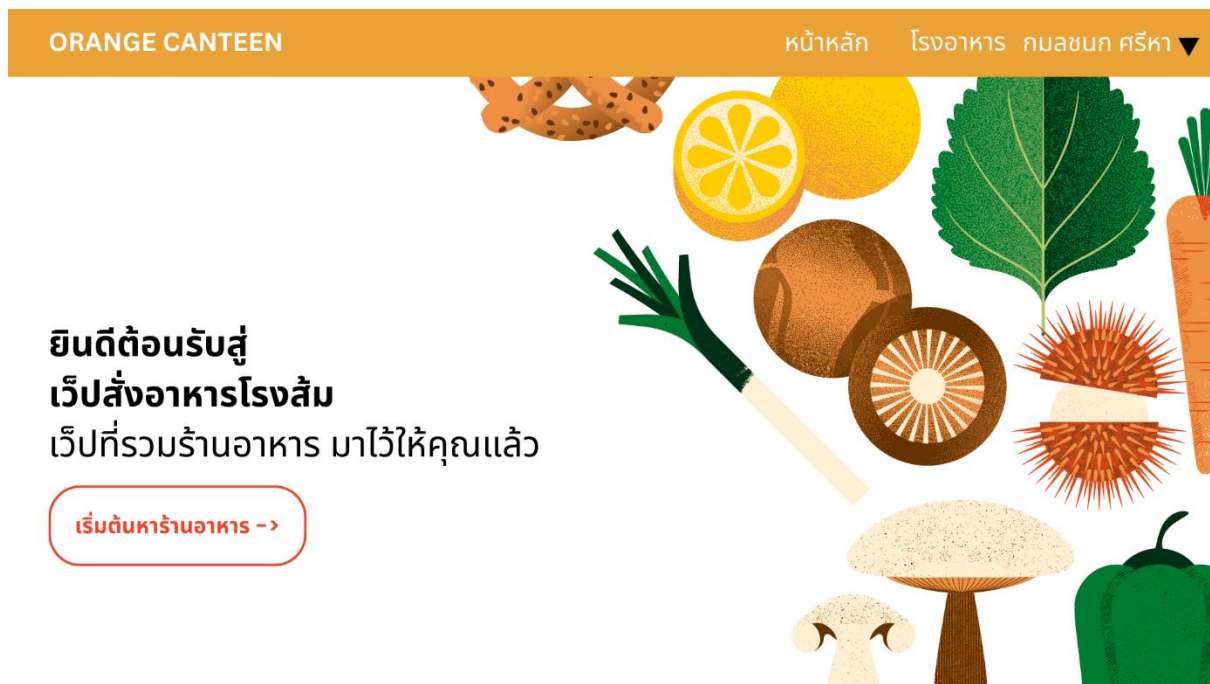
ชื่อผู้ใช้

รหัสผ่าน

ยืนยันรหัสผ่าน

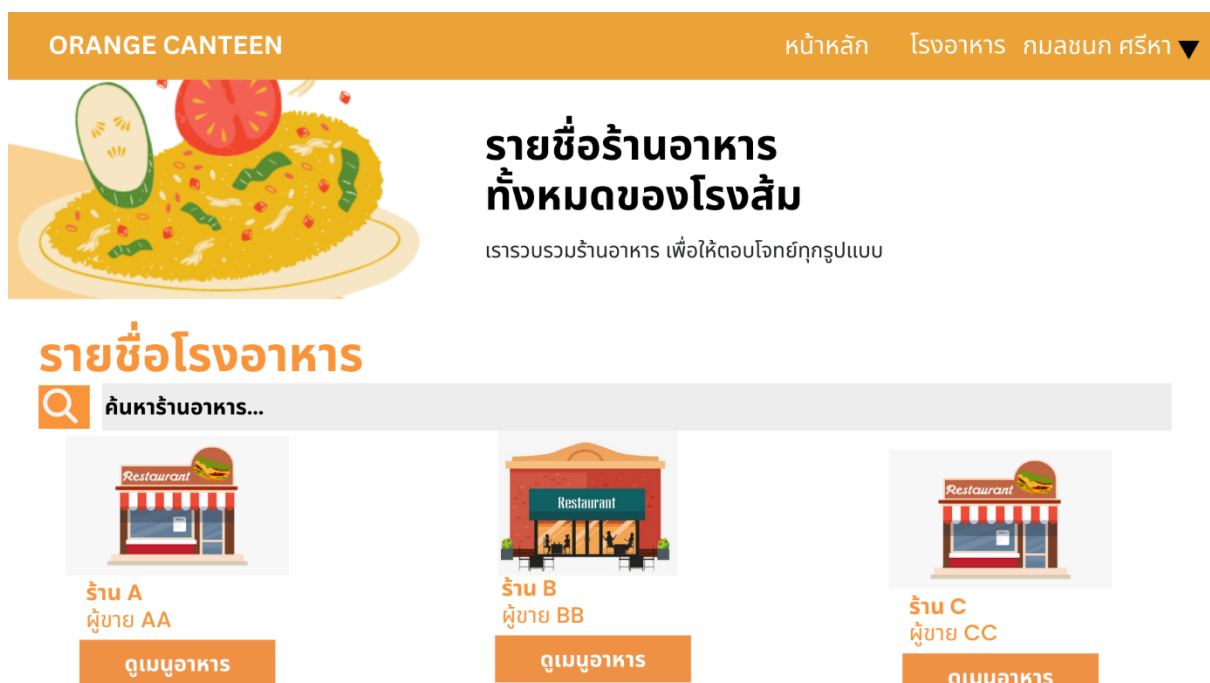
ภาพที่ 27 ภาพออกแบบหน้าเว็บไซต์สมัครสมาชิก

2.1.2 หน้าหลักของเว็บไซต์



ภาพที่ 28 ภาพออกแบบหน้าหลักเว็บไซต์

2.1.3 หน้าเว็บไซต์รวบรวมร้านอาหาร



ภาพที่ 29 ภาพออกแบบภาพหน้าเว็บไซต์รวบรวมร้านอาหาร

2.1.4 หน้าเว็บไซต์รวบรวมเมนูอาหารตามร้านอาหาร

ORANGE CANTEEN

หน้าหลัก

โรงอาหาร กมลชนก ศรีหา ▼

ร้านช้อนเงิน

ผู้ขาย AA

รายการอาหาร

ค้นหารายการอาหาร

ข้าวผัดไก่ผัดไข่เค็ม 35฿

สั่งอาหาร

ข้าวหมูกระเทียม 35฿

+ 1 -

ข้าวผัดกะเพราปลาหมึก 35฿

+ 2 -

ร้านช้อนเงิน

เพิ่มเมนูอาหาร ในตะกร้าของคุณ

ข้าวผัดกะเพราปลาหมึก + 2 - 70฿

ข้าวหมูกระเทียม + 1 - 35฿

ราคารวมทั้งหมด 105฿

เวลาบริการอาหาร

09.00 น. ▼

หมายเหตุ

แพ็กเกจตัวอย่าง

สั่งอาหาร

ภาพที่ 30 ภาพออกแบบภาพหน้าเว็บไซต์รวบรวมเมนูอาหารตามร้านอาหาร

2.1.5 หน้าเว็บไซต์รวบรวมสถานะรายการคำสั่งอาหาร

ORANGE CANTEEN

หน้าหลัก

โรงอาหาร กมลชนก ศรีหา ▼

สถานะรายการอาหาร

ค้นหาคำสั่งซื้อ...

วันที่	สถานะอาหาร	รายการอาหาร	จำนวน	ราคา	คนสั่ง	หมายเหตุลูกค้า	หมายเหตุเจ้าของร้าน
9 ตุลาคม 2565	กำลังปรุง	ข้าวผัดกะเพราปลาหมึก	2	70฿	กมลชนก ศรีหา	แพ็กเกจตัวอย่าง	
9 ตุลาคม 2565	ถูกยกเลิก	ข้าวหมูกระเทียม	1	35฿	กมลชนก ศรีหา	แพ็กเกจตัวอย่าง	หยุด

ภาพที่ 31 ภาพออกแบบภาพหน้าเว็บไซต์รวบรวมสถานะรายการคำสั่งอาหาร

คู่มือผู้พัฒนาและการติดตั้ง

หน้า 29

2.2 เจ้าของร้าน

2.2.1 หน้าเข้าสู่ระบบและสมัครสมาชิก

ORANGE CANTEEN
หน้าหลัก
โรงอาหาร
ไม่ได้เข้าสู่ระบบ ▼



เข้าสู่ระบบ

ชื่อผู้ใช้

รหัสผ่าน

เข้าสู่ระบบ

ยังไม่มีบัญชี ? [สมัครสมาชิก](#)

ภาพที่ 32 ภาพออกแบบหน้าเว็บไซต์เข้าสู่ระบบ

2.2.2 หน้าเว็บไซต์รวบรวมสถานะรายการคำสั่งอาหารทั้งหมด

ORANGE CANTEEN
หน้าหลัก
จัดการเมนูอาหาร
ดูคำสั่งอาหาร
วัชรศักดิ์ ชื่นชม ▼



สถานะรายการอาหาร

วันที่	สถานะอาหาร	รายการอาหาร	จำนวน	ราคา	คนสั่ง	หมายเหตุลูกค้า	หมายเหตุเจ้าของร้าน	แก้ไข	ลบ
9 ตุลาคม 2565	กำลังปรุง	ข้าวผัดกระเพรา ปลาหมึก	2	70฿	กมลชนก ศรีหา	แพ้งางคบางอย่าง			
9 ตุลาคม 2565	ถูกยกเลิก	ข้าวหมูกระเทียม	1	35฿	กมลชนก ศรีหา	แพ้งางคบางอย่าง	หมูหมด		

ภาพที่ 33 ภาพออกแบบภาพหน้าเว็บไซต์รวบรวมสถานะรายการคำสั่งอาหารทั้งหมด

ORANGE CANTEEN

[หน้าหลัก](#)
[จัดการเมนูอาหาร](#)
[ดูคำสั่งอาหาร](#)
[วัชรศักดิ์](#)
[ชั้นชม](#)

สถานะรายการอาหาร

วันที่

วันเสาร์ที่ 9 ตุลาคม 2565

เวลาที่รับอาหาร

15.00 น.

รายการอาหาร

ข้าวผัดกระเพราปลาหมึก

จำนวน

2

ราคา

70฿

รายชื่อคนสั่ง

กมลชนก ศรีหา

หมายเหตุของลูกค้า

แพ้ง่ายคนอย่างเธอ

☐ กำลังปรุง
 ☐ เรียบร้อย
 ☐ ได้รับแล้ว
 ☐ ถูกยกเลิก

อัปเดตสถานะ

ภาพที่ 34 ภาพออกแบบหน้าเว็บไซต์อัปเดตสถานะรายการอาหาร

2.2.3 หน้าเว็บไซต์รวบรวมเมนูภายในร้านอาหารทั้งหมด

ORANGE CANTEEN

[หน้าหลัก](#)
[จัดการเมนูอาหาร](#)
[ดูคำสั่งอาหาร](#)
[วัชรศักดิ์](#)
[ชั้นชม](#)

รายการอาหาร

+ เพิ่มรายการอาหาร

วันที่	รายการอาหาร	ราคา	รูปภาพ	แก้ไข	ลบ
9 ตุลาคม 2565	ข้าวผัดกระเพราปลาหมึก	70฿			
9 ตุลาคม 2565	ข้าวหมูกระเทียม	35฿			

ภาพที่ 35 ภาพออกแบบภาพหน้าเว็บไซต์รายการอาหารในร้าน

ORANGE CANTEEN

[หน้าหลัก](#)
[จัดการเมนูอาหาร](#)
[ดูคำสั่งอาหาร](#)
[วัชรศักดิ์](#)
[ชั้นชม](#)

เพิ่มเมนูอาหาร

วันที่เพิ่ม

วันศุกร์ที่ 30 กันยายน 2565

ราคา

35

ชื่อเมนูอาหาร

ข้าวผัดกะเพราปลาหมึก

อัปโหลดรูปภาพ

Choose File

รูปภาพ.png

เพิ่มเมนูอาหาร

ภาพที่ 36 ภาพออกแบบภาพหน้าเว็บไซต์เพิ่มเมนูอาหาร

ORANGE CANTEEN

[หน้าหลัก](#)
[จัดการเมนูอาหาร](#)
[ดูคำสั่งอาหาร](#)
[วัชรศักดิ์](#)
[ชั้นชม](#)

แก้ไขเมนูอาหาร

วันที่เพิ่ม

วันศุกร์ที่ 30 กันยายน 2565

ราคา

35

สถานะเมนูอาหาร

เปิด

ชื่อเมนูอาหาร

ข้าวผัดกะเพราปลาหมึก

อัปโหลดรูปภาพ

Choose File

รูปภาพ.png

รูปเดิม

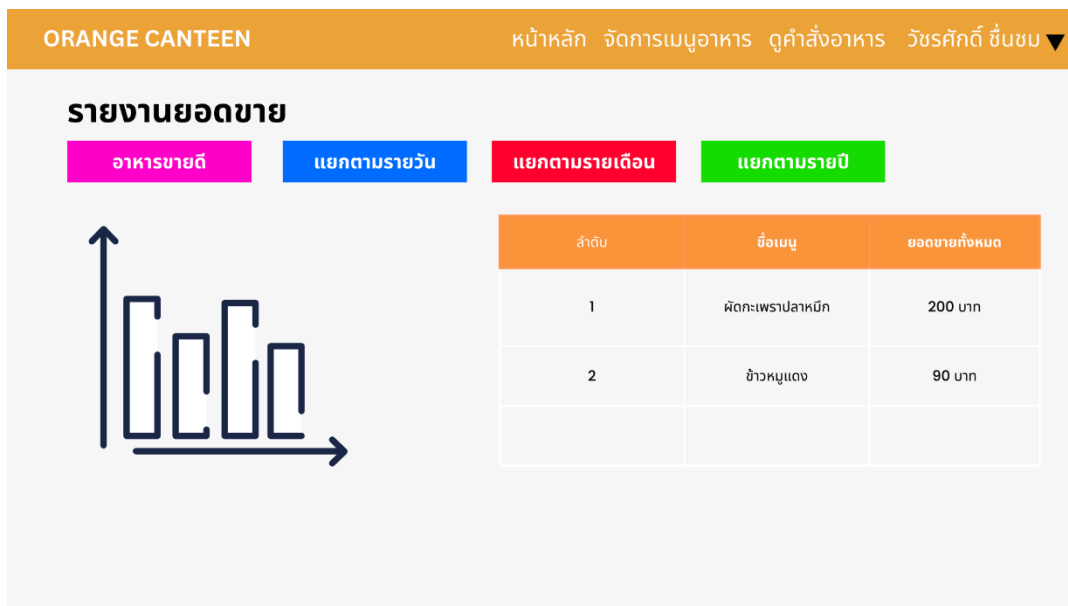


แก้ไขเมนูอาหาร

ภาพที่ 37 ภาพออกแบบภาพหน้าเว็บไซต์แก้ไขร้านอาหาร

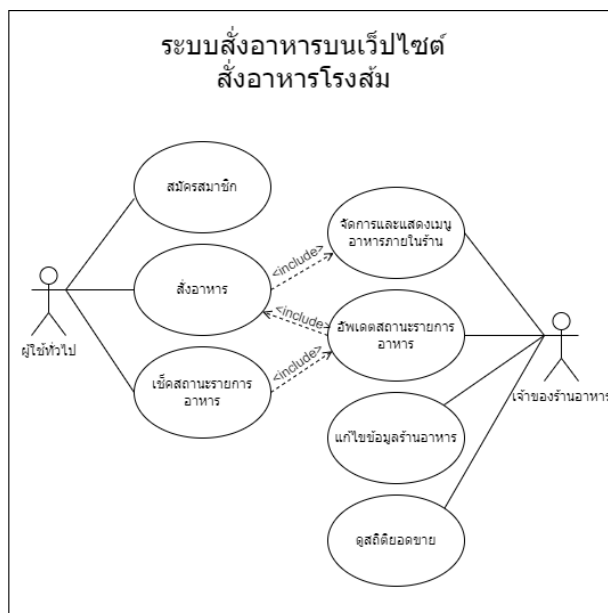
คู่มือผู้พัฒนาและการติดตั้ง

หน้า 32



ภาพที่ 38 ออกแบบภาพหน้าเว็บไซต์รายงานสถิติยอดขาย

3. Use Case Diagram



ภาพที่ 39 ภาพ Use Case Diagram ของระบบสั่งอาหารบนเว็บไซต์สั่งอาหารโรงแรม

Use Case Description

Use case	สมัครสมาชิก
Description	ลูกค้าต้องการสมัครสมาชิกเพื่อใช้งานเว็บไซต์สั่งอาหารโรงแรม
Actors	ลูกค้า
Pre-condition	ลูกค้ากรอกข้อมูลสมัครสมาชิก
Post-condition	ลูกค้าได้บัญชีในการเข้าสู่ระบบ
Related use case	-
Steps	<ol style="list-style-type: none"> 1. กดไปที่หน้าเข้าสู่ระบบ 2. กดไปที่ “ปุ่มสมัครสมาชิก” 3. กรอกข้อมูลในการสมัครสมาชิก 4. กดปุ่ม “สมัครสมาชิก”

ตารางที่ 1 ตารางอธิบาย Use Case Description “สมัครสมาชิก”

Use case	สั่งอาหาร
Description	ลูกค้าต้องการสั่งอาหารบนเว็บไซต์สั่งอาหารโรงสั่ม
Actors	ลูกค้า,เจ้าของร้าน
Pre-condition	เจ้าของร้านจะเพิ่มเมนูอาหาร และลูกค้าจะทำการกดเลือกเมนูและกดสั่งอาหารบนเว็บไซต์หน้าร้านนั้นๆ
Post-condition	ลูกค้าได้สั่งอาหาร และข้อมูลสั่งอาหารจะถูกส่งไปที่ฐานข้อมูล
Related use case	เจ้าของร้านต้องมีการใช้ “จัดและแสดงเมนูอาหารภายในร้าน” ก่อน
Steps	<ol style="list-style-type: none"> 1. กดเลือกร้านอาหาร 2. กดเลือกเมนูและจำนวนอาหาร 3. กดเลือกเวลามารับอาหาร และกรณีมีหมายเหตุให้ใส่หมายเหตุ 4. กด “สั่งอาหาร”

ตารางที่ 2 ตารางอธิบาย Use Case Description “สั่งอาหาร”

Use case	เช็คสถานะรายการอาหาร
Description	ลูกค้าต้องการเช็คสถานะรายการอาหาร
Actors	ลูกค้า,เจ้าของร้านอาหาร
Pre-condition	ลูกค้ากดเช็คสถานะอาหารที่เมนู “เช็คสถานะรายการอาหาร”
Post-condition	ลูกค้าสามารถดูสถานะคำสั่งซื้ออาหารได้
Related use case	เจ้าของร้านจะมีการใช้ “อัปเดตสถานะรายการอาหาร”
Steps	<ol style="list-style-type: none"> 1. กดไปแท็บเมนูบัญชี 2. กดเลือกเมนู “เช็คสถานะรายการอาหาร” 3. ระบบแสดงสถานะทุกคำสั่งซื้ออาหาร ของบัญชีลูกค้า

ตารางที่ 3 ตารางอธิบาย Use Case Description “เช็คสถานะรายการอาหาร”

Use case	จัดและแสดงเมนูอาหารภายในร้าน
Description	เจ้าของร้านต้องการเพิ่มเมนูอาหารในร้าน เพื่อให้ลูกค้าสามารถเลือกสั่งได้
Actors	เจ้าของร้าน
Pre-condition	เจ้าของร้านกรอกรายละเอียดเมนูอาหาร
Post-condition	ข้อมูลเมนูอาหารถูกบันทึกไปในฐานข้อมูล
Related use case	-
Steps	<ol style="list-style-type: none"> 1. กดไปที่จัดการเมนูอาหาร 2. กดไปที่เพิ่มรายการอาหาร 3. กรอกข้อมูลเมนูอาหารที่ต้องการจะเพิ่ม 4. กด “เพิ่มเมนูอาหาร”

ตารางที่ 4 ตารางอธิบาย Use Case Description “จัดและแสดงเมนูอาหารภายในร้าน”

Use case	อัปเดตสถานะรายการอาหาร
Description	เจ้าของร้านต้องการอัปเดตสถานะคำสั่งซื้ออาหารเพื่อให้ลูกค้าทราบ
Actors	เจ้าของร้าน,ลูกค้า
Pre-condition	เจ้าของร้านกดอัปเดตสถานะตามรายการคำสั่งซื้ออาหารของลูกค้า
Post-condition	ข้อมูลอัปเดตสถานะถูกบันทึกไปในฐานข้อมูล
Related use case	ลูกค้าต้องมีการ “สั่งอาหาร” มาก่อน
Steps	<ol style="list-style-type: none"> 1. กดไปที่หน้าดูคำสั่งอาหาร 2. กดเลือกแก้ไขตามรายการคำสั่งซื้ออาหาร 3. กดปุ่ม “อัปเดตสถานะ” และกรณีมีหมายเหตุให้กรอกหมายเหตุ 4. กดปุ่ม “ยืนยัน”

ตารางที่ 5 ตารางอธิบาย Use Case Description “อัปเดตสถานะรายการอาหาร”

Use case	แก้ไขข้อมูลร้านอาหาร
Description	เจ้าของร้านต้องการปรับเปลี่ยนข้อมูลร้านอาหาร เพื่อปรับเปลี่ยนการแสดงผลบนเว็บไซต์
Actors	เจ้าของร้าน
Pre-condition	เจ้าของร้านกรอกข้อมูลแก้ไขร้านอาหาร
Post-condition	ข้อมูลแก้ไขร้านอาหารถูกบันทึกไปเป็นฐานข้อมูล
Related use case	-
Steps	<ol style="list-style-type: none"> 1. กดไปแท็บบัญชี 2. กดไปแก้ไขร้านอาหาร 3. กรอกข้อมูลในการแก้ไขร้านอาหาร 4. กดปุ่ม “ยืนยันการแก้ไข”

ตารางที่ 8 ตารางอธิบาย Use Case Description “แก้ไขข้อมูลร้านอาหาร”

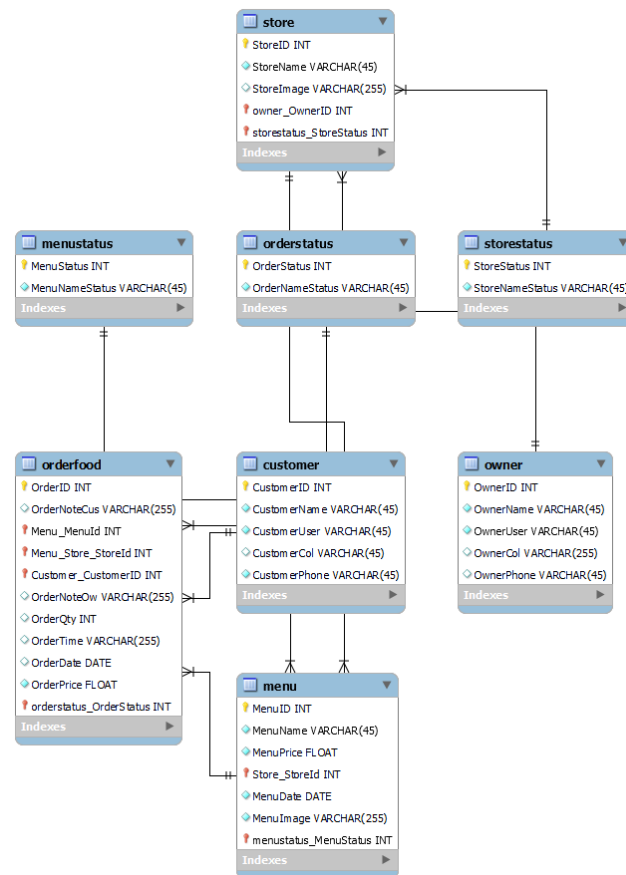
Use case	ดูสถิติยอดขาย
Description	เจ้าของร้านต้องการดูสถิติยอดขายในร้านอาหารของตนเอง เพื่อเอาไปพิจารณาและปรับปรุงร้านอาหารให้ดีขึ้น
Actors	เจ้าของร้าน
Pre-condition	เจ้าของรถดูที่ปุ่มรายงาน
Post-condition	เว็บไซต์แสดงรายงานสถิติยอดขายในร้านอาหาร
Related use case	-
Steps	<ol style="list-style-type: none"> 1. กดไปแท็บบัญชี 2. กดไปที่รายงาน 3. เลือกรูปแบบสถิติที่ต้องการจะตรวจสอบ

ตารางที่ 9 ตารางอธิบาย Use Case Description “ดูสถิติยอดขาย”

การพัฒนาาระบบและการติดตั้ง

1. ฐานข้อมูล

มีการทำฐานข้อมูลบน Microsoft Azure



ภาพที่ 40 การทำฐานข้อมูลบน Microsoft Azure

2. การเขียนหน้าเว็บไซต์

มีการเขียนโค้ดโดยใช้ภาษา JavaScript(ReactJS) HTML และ CSS โดยใช้หลักการดึง API และการสับเปลี่ยนหรือกำหนดหน้าตาม Path โดยใช้ Route ใน Library react-router-dom

หลักการดึง API โดยใช้ Axios และ Await

การดึง API จะประกอบไปด้วย GET,POST,PUT และ DELETE โดยทางคณะผู้จัดทำจะนำโค้ดตัวอย่างบางส่วนเพื่อให้เห็นหลักการทำงาน

- GET : เป็นการดึงข้อมูลจากฐานข้อมูลนำมาแสดง

รูปแบบที่ 1

```
const GetStore = async () => {  
  let res = await axios.get('http://localhost:8000/api/stores');  
  const data = res.data.data;  
  setdata(data)  
}
```

ตัวอย่างการดึงข้อมูลร้านอาหารทั้งหมด

รูปแบบที่ 2

```
const GetMenuByStoreID = async () => {  
  let StoreID = { StoreID: owner.StoreID };  
  let res = await axios.post('http://localhost:8000/api/menu', StoreID);  
  let data = res.data;  
  setbeforedata(data)  
  // console.log(beforedata);  
}
```

ตัวอย่างการดึงข้อมูลเมนูอาหารตาม ID ร้านอาหาร

- POST : เป็นการสร้างข้อมูลในฐานข้อมูล

รูปแบบที่ 1

```
let item = { CustomerCol, CustomerUser, CustomerName, CustomerPhone }
let result = await fetch("http://localhost:8000/api/signup", {
  method: 'POST',
  body: JSON.stringify(item),
  headers: {
    "Content-Type": 'application/json',
    "Accept": 'application/json'
  }
})
result = await result.json();
```

ตัวอย่างการสร้างข้อมูลสมัครสมาชิกของลูกค้า

รูปแบบที่ 2

```
let formData = new FormData()
const MenuDate = date.getFullYear() + "-" + parseInt(date.getMonth() + 1) + "-" +
date.getDate();
formData.append('file', file)
formData.append('MenuName', MenuName)
formData.append('MenuPrice', MenuPrice)
formData.append('MenuDate', MenuDate)
formData.append('MenuStatus', '1')
formData.append('StoreID', StoreID)

let uploadImg = await axios({
  method: 'post',
  url: 'http://localhost:8000/api/uploadsmenu',
  data: formData
})

.then((response) => {
  // console.log(response.data);
  localStorage.setItem("NmenuSc", "สำเร็จ")
  history("/managemenu")
});
```

ตัวอย่างการสร้างข้อมูลเมนูอาหารภายในร้าน

- PUT : เป็นการอัปเดตข้อมูล

```
const onClickUpload = async (MenuID, file, MenuName, MenuPrice, MenuImage, MenuStatus, history) => {
  let formData = new FormData()
  formData.append('file', file)
  formData.append('MenuName', MenuName)
  formData.append('MenuPrice', MenuPrice)
  formData.append('MenuID', MenuID)
  formData.append('MenuImage', MenuImage)
  formData.append('MenuStatus', MenuStatus)
  let uploadImg = await axios({
    method: 'put',
    url: 'http://localhost:8000/api/menu',
    data: formData
  })
  .then((response) => {
    localStorage.setItem("UPmenuSc", "สำเร็จ")
    history("/managemenu")
  });
}
```

ตัวอย่างการแก้ไขข้อมูลในเมนูรายการอาหาร

- DELETE : เป็นการลบข้อมูลตาม ID

```
let item = { id }  
  
let result = await fetch("http://localhost:8000/api/menu", {  
  method: 'DELETE',  
  body: JSON.stringify(item),  
  headers: {  
    "Content-Type": 'application/json',  
    "Accept": 'application/json'  
  }  
});  
  
result = await result.json();
```

ตัวอย่างการลบข้อมูลเมนูอาหาร

การกำหนดหน้าตาม Path โดยใช้ Route ใน Library react-router-dom

- ไฟล์ App.js

```

<div className="App">
  <AppHeader />
  <Wrapper>
    <Routes>
      <Route path="/" element={<PageHome />} />
      <Route path="listcanteen" element={<PageListCanteen />} />
      <Route path="login" element={<PageLogin />} />
      <Route path="signup" element={<PageSignup />} />

      /* ของผู้ใช้ทั่วไป */
      <Route path="food/:id" element={<PageListFood />} />
      <Route path="statusfood/:id" element={<Pagestatusfood />} />

      /* ของเจ้าของร้าน */
      <Route path="statusfoodofstaff" element={<Pagestatusfoodofstaff />} />
      <Route path="editstatusofstaff/:id" element={<Pageeditstatusofstaff />} />
      <Route path="managemenu" element={<PagesManageMenu />} />
      <Route path="newmenu" element={<PagesNewMenu />} />
      <Route path="editmenu/:id" element={<PagesEditMenu />} />
      <Route path="editstore/:id" element={<PageEditStore />} />
      <Route path="chart/" element={<PageChart />} />

      /* หา path ไม่เจอ */
      <Route path="*" element={<Page404 />} />
    </Routes>
  </Wrapper>
</div>

```

ตัวอย่างการ Route หน้าเว็บไซต์ในแต่ละหน้าให้อยู่ใน Path

รายละเอียดหน้าเว็บไซต์

โดยหน้าเว็บไซต์มีดังนี้

1. เว็บส่วนกลาง

1.1. ส่วนของหน้าหลัก : Path(/)

ประกอบไปด้วยไฟล์ดังนี้

1. PageHome.js : เป็นเว็บหน้าหลัก

Components ที่ประกอบอยู่ใน PageHome.js

1. One.js : เป็นส่วนหน้าหลักส่วนแรกของผู้ใช้ทั่วไป
2. Two.js: เป็นส่วนหน้าหลักส่วนที่สองของผู้ใช้ทั่วไป
3. Three.js : เป็นส่วนหน้าหลักส่วนสามของผู้ใช้ทั่วไป
4. Four.js : เป็นส่วนหน้าหลักส่วนแรกของผู้ใช้ทั่วไป
5. style.css : ควบคุมตกแต่งหน้าเว็บไซต์

1.2. ส่วนของหน้าแสดงโรงอาหาร : Path(/listcanteen)

ประกอบไปด้วยไฟล์ดังนี้

1. PageListCanteen.js : เป็นเว็บแสดงหน้าโรงอาหาร

Components ที่ประกอบอยู่ใน PageListCanteen.js

1. One.js : เป็นส่วนหน้าโรงอาหารส่วนแรกของโรงอาหาร
2. Two.js: เป็นส่วนหน้าโรงอาหารส่วนที่สองของโรงอาหาร แสดงช่องค้นหา
3. Canteenitem.js : เป็นส่วนแสดงข้อมูลร้านอาหาร
4. Testdatacanteen.js : เป็นตัวทดลองข้อมูล
5. style.css : ควบคุมตกแต่งหน้าเว็บไซต์

1.3. ส่วนของหน้าเข้าสู่ระบบ : Path(/login)

ประกอบไปด้วยไฟล์ดังนี้

1. PageLogin.js : เป็นเว็บแสดงหน้าเข้าสู่ระบบ

Components ที่ประกอบอยู่ใน PageLogin.js

1. PopUpLogin.Js : เป็นป๊อปอัพแจ้งเตือน
2. style.css : ไว้ตกแต่งหน้าเว็บไซต์

1.4. ส่วนของหน้าเข้าสู่ระบบ : Path(/Signup)

ประกอบไปด้วยไฟล์ดังนี้

1. PageSignup.js : เป็นเว็บแสดงหน้าสมัครสมาชิก

Components ที่ประกอบอยู่ใน PageSignup.js

1. PopUpLogin.Js : เป็นป๊อปอัพแจ้งเตือน
2. style.css : ไว้ตกแต่งหน้าเว็บไซต์

2. ผู้ใช้งานทั่วไป

2.1. ส่วนของหน้าสั่งอาหาร: Path (/food/:id)

ประกอบไปด้วยไฟล์ดังนี้

1. PageListFood.js : เป็นเว็บแสดงหน้าสั่งอาหาร

Components ที่ประกอบอยู่ใน PageListFood.js

1. PopUpLF.Js : เป็นป๊อปอัพแจ้งเตือน
2. Basket.Js: เป็นส่วนแสดงรายการอาหารที่สั่ง
3. Food.Js : เป็นส่วนแสดงเมนูอาหาร
4. Menu.Js : เป็นส่วนแสดงเมนูอาหารทั้งหมด
5. style.css : ไว้ตกแต่งหน้าเว็บไซต์

6. Testdata.js : ไว้ทดลองข้อมูล

2.2. ส่วนของหน้าตรวจสอบสถานะคำสั่งซื้ออาหาร: Path(/statusfood/:id)

ประกอบไปด้วยไฟล์ดังนี้

1. Pagestatusfood.js : เป็นเว็บแสดงหน้าตรวจสอบสถานะคำสั่งซื้ออาหาร

Components ที่ประกอบอยู่ใน Pagestatusfood.js

1. PopUpSc.js : เป็นป๊อปอัพแจ้งเตือน

2. GlobalFilter.js: เป็นส่วนแสดงช่องค้นหา

3. Testdata.js : ไว้ทดลองข้อมูล

4. style.css : ไว้ตกแต่งหน้าเว็บไซต์

3. เจ้าของร้านอาหาร

3.1. ส่วนของการจัดการเมนูอาหาร

3.1.1 ส่วนแสดงรายการอาหารภายในร้านทั้งหมด Path (/managemenu /:id)

ประกอบไปด้วยไฟล์ดังนี้

1. PagesManageMenu.js : เป็นหน้าเว็บส่วนแสดงรายการอาหารภายในร้านทั้งหมด

Components ที่ประกอบอยู่ใน PagesManageMenu.js

1. PopUpM.js : เป็นป๊อปอัพแจ้งเตือน

2. GlobalFilter.js: เป็นส่วนแสดงช่องค้นหา

3. style.css : ไว้ตกแต่งหน้าเว็บไซต์

3.1.2 ส่วนแก้ไขเมนูอาหาร Path (/editmenu/:id)

ประกอบไปด้วยไฟล์ดังนี้

1. PagesEditMenu.js : เป็นหน้าเว็บส่วนแสดงแก้ไขเมนูอาหาร 1 เมนู

Components ที่ประกอบอยู่ใน PagesEditMenu.js

- ## 2. style.css : ไม้ตกแต่งหน้าเว็บไซต์

ประกอบไปด้วยไฟล์ดังนี้

- ## Components ที่ประกอบอยู่ใน PagesNewMenu.js

- ## 2. style.css : ไม้ตกแต่งหน้าเว็บไซต์

3.2.1 ส่วนแสดงสถานะรายการคำสั่งซื้ออาหารของลูกค้าภายในร้านทั้งหมด Path (/statusfoodofstaff /:id)

ประกอบไปด้วยไฟล์ดังนี้

- ## Components ที่ประกอบอยู่ใน Pagestatusfoodofstaff.js

- ## 2. GlobalFilter.js: เป็นส่วนแสดงช่องค้นหา

- ### 3. style.css : ไว้ตกแต่งหน้าเว็บไซต์

ประกอบไปด้วยไฟล์ดังนี้

- ## Components ที่ประกอบอยู่ใน Pageeditstatusofstaff.js

- ## 2. style.css : ไม้ตกแต่งหน้าเว็บไซต์

3.3 ส่วนของการแก้ไขข้อมูลร้านอาหาร Path(/editstore/:id)

ประกอบไปด้วยไฟล์ดังนี้

1. PageEditStore.js : เป็นเว็บแสดงหน้าการแก้ไขข้อมูลร้านอาหาร

Components ที่ประกอบอยู่ใน PageEditStore.js

1. PopUpES.js : เป็นป๊อปอัพแจ้งเตือน
2. style.css : ไว้ตกแต่งหน้าเว็บไซต์

3.4 ส่วนของการแสดงรายงานสถิติยอดขาย Path(/chart)

ประกอบไปด้วยไฟล์ดังนี้

1. PageChart.js : เป็นเว็บแสดงหน้ารายงานสถิติยอดขาย

Components ที่ประกอบอยู่ใน PageChart.js

1. BarChartMostFood.js : เป็นส่วนแสดงสถิติเมนูยอดขายที่ดีที่สุดในร้าน
2. BarChartDay.js: เป็นส่วนแสดงสถิติเมนูยอดขายรายวันของร้าน
3. BarChartYear.js : เป็นส่วนแสดงสถิติเมนูยอดขายรายเดือนของร้าน
4. BarChartYear.js : เป็นส่วนแสดงสถิติเมนูยอดขายรายปีของร้าน
5. style.css : ไว้ตกแต่งหน้าเว็บไซต์

3. การเขียนระบบหลังเว็บไซต์

ระบบหลังเว็บไซต์ใช้ Node.js และ Express ในการทำ API โดยแต่ละ service มีหลักการทำงานดังนี้

3.1 ระบบล็อกอินและสมัครสมาชิก

การสมัครสมาชิก

1. สิ่งที่ต้องนำเข้า

ตัวแปร	รายละเอียด
CustomerUser	ชื่อผู้ใช้ลูกค้า
CustomerName	ชื่อ-นามสกุลลูกค้า
CustomerCol	รหัสผ่านของลูกค้า
CustomerPhone	เบอร์โทรของลูกค้า

2. URL ที่ใช้

Method	URL
POST	/api/signup

3. การตอบกลับ

เงื่อนไข	การตอบกลับ
กรณีใส่ข้อมูลไม่ครบ	{ "error": true, "message": "กรุณากรอกให้ครบ" }
กรณีใส่ข้อมูลครบถ้วน	{ "error": false, "data": { "CustomerUser": "222222", "CustomerName": "นางเอ วิตามิน", "CustomerCol": "111111" }, "message": "User successfully added" }

การเช็คชื่อผู้ใช้ก่อนสมัครสมาชิก

1. สิ่งที่ต้องนำเข้า

ตัวแปร	รายละเอียด
CustomerUser	ชื่อผู้ใช้ลูกค้า
CustomerName	ชื่อ-นามสกุลลูกค้า
CustomerCol	รหัสผ่านของลูกค้า

2. URL ที่ใช้

Method	URL
POST	/api/ checkmember

3. การตอบกลับ

เงื่อนไข	การตอบกลับ
กรณีใส่ข้อมูลไม่ครบ	{ "error": true, "message": "กรุณากรอกให้ครบ" }
กรณีใส่ข้อมูลครบถ้วน และสมาชิกไม่ซ้ำใน ระบบ	{ "error": false, "message": "ไม่มีสมาชิกในนี้" }
กรณีใส่ข้อมูลครบถ้วน และสมาชิกซ้ำในระบบ	{ "error": false, "data": { "CustomerID": 1, "CustomerName": "นางเอ วิตามิน", "CustomerUser": "aaaaaa", "CustomerCol": "aaaaaa", "CustomerPhone": "0922654387" }, "message": "เนื่องจากมีชื่อผู้ใช้นี้ในระบบแล้ว" }

การเข้าสู่ระบบสำหรับลูกค้า

1. สิ่งที่ต้องนำเข้า

ตัวแปร	รายละเอียด
User	ชื่อผู้ใช้ลูกค้า
Col	รหัสผ่านของลูกค้า

2. URL ที่ใช้

Method URL
POST /api/login

3. การตอบกลับ

เงื่อนไข	การตอบกลับ
กรณีกรอกชื่อผู้ใช้หรือรหัสผ่านไม่ครบ	{ "error": true, "message": "กรุณากรอกให้ครบ" }
กรณีชื่อผู้ใช้หรือรหัสผ่านไม่มีอยู่ในระบบ	{ "error": false, "message": "เช็คกรณีพนักงาน" }
กรณีกรอกรหัสผ่านถูกต้อง	{ "error": false, "data": { "CustomerID": 1, "CustomerName": "นางเอ วิตามิน", "CustomerUser": "aaaaaa", "CustomerCol": "aaaaaa", "CustomerPhone": "0922654387" }, "message": "Successfully retrieved User data" }

การเข้าสู่ระบบสำหรับเจ้าของร้านอาหาร

1. สิ่งที่ต้องนำเข้า

ตัวแปร	รายละเอียด
User	ชื่อผู้ใช้เจ้าของร้าน
Col	รหัสผ่านของเจ้าของร้าน

2. URL ที่ใช้

Method URL
POST /api/loginOwner

3. การตอบกลับ

เงื่อนไข	การตอบกลับ
กรณีกรอกชื่อผู้ใช้หรือรหัสผ่านไม่ครบ	{ "error": true, "message": "กรุณากรอกชื่อผู้ใช้และรหัสผ่าน" }
กรณีชื่อผู้ใช้หรือรหัสผ่านไม่มีอยู่ในระบบ	{ "error": false, "message": "เนื่องจากชื่อผู้ใช้หรือรหัสผ่านไม่ถูกต้อง" }
กรณีกรอกรหัสผ่านถูกต้อง	{ "error": false, "data": { "OwnerID": 1, "OwnerName": "คิวนคนสวย ศรีหา", "OwnerUser": "Kamonz", "OwnerCol": "Nami12755", "OwnerPhone": "0922654387", "StoreID": 2, "StoreName": "ซั๊อนทอง", "StoreImage": "1665617244109.png", "owner_OwnerID": 1, "storestatus_StoreStatus": 2 }

	<pre> }, "message": "Successfully retrieved User data" } </pre>
--	---

3.2 การจัดการเมนูอาหารสำหรับเจ้าของร้าน

การแสดงผลเมนูอาหารภายในร้านอาหารทั้งหมด

1. สิ่งที่ต้องนำเข้า

ตัวแปร	รายละเอียด
StoreID	ID ร้านอาหาร

2. URL ที่ใช้

Method	URL
POST	/api/menu

3. การตอบกลับ

เงื่อนไข	การตอบกลับ
กรณีไม่ได้กรอก ID ร้านอาหาร	<pre> { "error": true, "message": "หาเลขไม่เจอ" } </pre>
กรณีค้นหาเมนูอาหาร ตามเลขร้านอาหารไม่ เจอ	ไม่ได้คืนค่ากลับ
กรณีค้นหาเมนูอาหาร ตามเลขร้านอาหารเจอ	<pre> [{ "MenuID": 3, "MenuName": "ข้าวปลาหมึกผัดไข่เค็ม2", "MenuPrice": 45, "Store_StoreID": 2, "MenuDate": "2022-10-07T17:00:00.000Z", "MenuImage": "1665619469562.png", "menustatus_MenuStatus": 1, </pre>

	<pre> "StoreID": 2, "StoreName": "ซั้องทอง", "StoreImage": "1665617244109.png", "owner_OwnerID": 1, "storestatus_StoreStatus": 2, "MenuStatus": 1, "MenuNameStatus": "เป้ด" }, { "MenuID": 1, "MenuName": "ซั้วผั้ดกะเพราปลาหมึก", "MenuPrice": 30, "Store_StoreID": 2, "MenuDate": "2022-10-07T17:00:00.000Z", "MenuImage": "1665619428351.png", "menustatus_MenuStatus": 2, "StoreID": 2, "StoreName": "ซั้องทอง", "StoreImage": "1665617244109.png", "owner_OwnerID": 1, "storestatus_StoreStatus": 2, "MenuStatus": 2, "MenuNameStatus": "ป้ด" },]</pre>
--	---

การแสดงผลเมนูอาหารภายในร้านอาหารเฉพาะสถานะเมนูอาหารเปิด

1. สิ่งที่ต้องนำเข้า

ตัวแปร	รายละเอียด
StoreID	ID ร้านอาหาร

2. URL ที่ใช้

Method URL
POST /api/menu1

3. การตอบกลับ

เงื่อนไข	การตอบกลับ
กรณีไม่ได้กรอก ID ร้านอาหาร	{ "error": true, "message": "หาเลขไม่เจอ" }
กรณีค้นหาเมนูอาหาร ตามเลขร้านอาหารไม่ เจอ	ไม่ได้คืนค่ากลับ
กรณีค้นหาเมนูอาหาร ตามเลขร้านอาหารเจอ	[{ "MenuID": 3, "MenuName": "ข้าวปลาหมึกผัดไข่เค็ม2", "MenuPrice": 45, "Store_StoreID": 2, "MenuDate": "2022-10-07T17:00:00.000Z", "MenuImage": "1665619469562.png", "menustatus_MenuStatus": 1, "StoreID": 2, "StoreName": "ซั๊อนทอง", "StoreImage": "1665617244109.png", "owner_OwnerID": 1, "storestatus_StoreStatus": 2, "MenuStatus": 1, }

	<pre> "MenuNameStatus": "เปิด" }, { "MenuID": 1, "MenuName": "ข้าวผัดกะเพราปลาหมึก", "MenuPrice": 30, "Store_StoreId": 2, "MenuDate": "2022-10-07T17:00:00.000Z", "MenuImage": "1665619428351.png", "menustatus_MenuStatus": 1, "StoreID": 2, "StoreName": "ซั๋นทอง", "StoreImage": "1665617244109.png", "owner_OwnerID": 1, "storestatus_StoreStatus": 1, "MenuStatus": 2, "MenuNameStatus": "เปิด" },]</pre>
--	---

การแสดงผลเมนูอาหารภายในร้านอาหารเฉพาะสถานะเมนูอาหารปิด

1. สิ่งที่ต้องนำเข้า

ตัวแปร	รายละเอียด
StoreID	ID ร้านอาหาร

2. URL ที่ใช้

Method	URL
POST	/api/menu2

3. การตอบกลับ

เงื่อนไข	การตอบกลับ
กรณีไม่ได้กรอก ID ร้านอาหาร	{ "error": true, "message": "หาเลขไม่เจอ" }
กรณีค้นหาเมนูอาหาร ตามเลขร้านอาหารไม่ เจอ	ไม่ได้คืนค่ากลับ
กรณีค้นหาเมนูอาหาร ตามเลขร้านอาหารเจอ	[{ "MenuID": 7, "MenuName": "ข้าวกะเพราหมูสับ", "MenuPrice": 25, "Store_StoreID": 2, "MenuDate": "2022-10-12T17:00:00.000Z", "MenuImage": "1665619647965.png", "menustatus_MenuStatus": 2, "StoreID": 2, "StoreName": "ซั๊นทอง", "StoreImage": "1665617244109.png", "owner_OwnerID": 1, "storestatus_StoreStatus": 2, "MenuStatus": 2, }

	<pre>"MenuNameStatus": "ปิด" }]</pre>
--	--

การสร้างเมนูอาหาร

1. สิ่งที่ต้องนำเข้า

ตัวแปร	รายละเอียด
MenuName	ชื่อเมนูอาหาร
MenuPrice	ราคาอาหาร
MenuDate	วันที่เพิ่มอาหาร
MenuStatus	ID สถานะเมนูอาหาร
StoreID	ID ร้านอาหาร
filename(เป็นชนิดไฟล์ที่ดึงชื่อรูปภาพ)	ชื่อไฟล์รูปภาพ

2. URL ที่ใช้

Method URL
POST /api/uploadsmenu

3. การตอบกลับ

เงื่อนไข	การตอบกลับ
กรอกข้อมูลครบถ้วน	<pre>{ "MenuName": "ข้าวผัดกะเพราปลาหมึก", "MenuPrice": 30, "MenuDate": "2022-10-07T17:00:00.000Z", "MenuDate": "2022-10-07T17:00:00.000Z", "MenuStatus": "1", "MenuImage": "1665619428351.png", "StoreID": 2, }</pre>

การอัปเดตเมนูอาหาร

1. สิ่งที่ต้องนำเข้า

ตัวแปร	รายละเอียด
MenuID	ID เมนูอาหาร
MenuName	ชื่อเมนูอาหาร
MenuPrice	ราคาอาหาร
MenuStatus	ID สถานะเมนูอาหาร
filename(เป็นชนิดไฟล์ที่ตั้งชื่อรูปภาพ) *ไม่จำเป็นต้องใส่*	ชื่อไฟล์รูปภาพ
MenuImage	ชื่อรูปภาพเก่า

2. URL ที่ใช้

Method	URL
PUT	/api/menu

3. การตอบกลับ

เงื่อนไข	การตอบกลับ
ไม่ได้กรอก MenuName , MenuStatus และ MenuPrice	{ "error": true, "message": "กรุณาใส่ข้อมูลให้ครบถ้วน" }
ใส่ ID เมนูอาหารไม่ ตรงในฐานข้อมูล	{ "error": false, "data": { "fieldCount": 0, "affectedRows": 0, "insertId": 0, "info": "Rows matched: 0 Changed: 0 Warnings: 0", "serverStatus": 2, "warningStatus": 0, "changedRows": 0 }, "message": "หาเมนูไม่เจอ"

	}
กรอกข้อมูลครบถ้วน	<pre>{ "error": false, "data": { "fieldCount": 0, "affectedRows": 1, "insertId": 0, "info": "Rows matched: 1 Changed: 1 Warnings: 0", "serverStatus": 2, "warningStatus": 0, "changedRows": 1 }, "message": "อัปเดตสำเร็จ" }</pre>

การลบเมนูอาหาร

1. สิ่งที่ต้องนำเข้า

ตัวแปร	รายละเอียด
id	ID เมนูอาหาร

2. URL ที่ใช้

Method URL
DELETE /api/menu

3. การตอบกลับ

เงื่อนไข	การตอบกลับ
ไม่ได้กรอก id	{ "error": true, "message": "กรุณาใส่ id " }
ใส่ ID เมนูอาหารไม่ตรงในฐานข้อมูล	{ "error": false, "data": { "fieldCount": 0, "affectedRows": 0, "insertId": 0, "info": "", "serverStatus": 2, "warningStatus": 0 }, "message": "ไม่สามารถลบได้ ไม่มี id นี้อยู่ในระบบ" }
มี ID อยู่ในฐานข้อมูล แต่ id นั้นอยู่ในสถานะรายการคำสั่งซื้ออาหารของลูกค้า	{ "message": "ER_ROW_IS_REFERENCED_2" }
มี ID อยู่ในฐานข้อมูล	{ "error": false,

	<pre> "data": { "fieldCount": 0, "affectedRows": 1, "insertId": 0, "info": "", "serverStatus": 2, "warningStatus": 0 }, "message": "ลบสำเร็จ" } </pre>
--	--

การค้นหาเมนูอาหารตาม ID เมนูอาหาร

1. สิ่งที่ต้องนำเข้า

ตัวแปร	รายละเอียด
Id : ใส่ไว้ผ่าน path	ID เมนูอาหาร

2. URL ที่ใช้

Method URL
GET /api/menu/:id

3. การตอบกลับ

เงื่อนไข	การตอบกลับ
ไม่ได้กรอก id	จะเป็นการส่งรายการอาหารทั้งหมดทุกร้านมาให้แทน
ใส่ ID เมนูอาหารไม่ตรงในฐานข้อมูล	<pre> { "error": false, "message": "หาเมนูไม่เจอ" } </pre>
กรอก ID ตรงในฐานข้อมูล	<pre> { "error": false, "data": { "MenuID": 10, "MenuName": "ข้าวขาหมู", </pre>

	<pre> "MenuPrice": 45, "Store_StoreId": 4, "MenuDate": "2022-10-12T17:00:00.000Z", "MenuImage": "1665623792685.png", "menustatus_MenuStatus": 1 }, "message": "เจอเมนู" } </pre>
--	--

การแสดงผลข้อมูลเมนูอาหารภายในร้านอาหารทั้งหมด

1. สิ่งที่ต้องนำเข้า

ตัวแปร	รายละเอียด
StoreID	ID ร้านอาหาร

2. URL ที่ใช้

Method URL
POST /api/menu

3. การตอบกลับ

เงื่อนไข	การตอบกลับ
กรณีไม่ได้กรอก ID ร้านอาหาร	<pre> { "error": true, "message": "หาเลขไม่เจอ" } </pre>
กรณีค้นหาเมนูอาหาร ตามเลขร้านอาหารไม่ เจอ	ไม่ได้คืนค่ากลับ
กรณีค้นหาเมนูอาหาร ตามเลขร้านอาหารเจอ	<pre> [{ "MenuID": 3, "MenuName": "ข้าวปลาหมึกผัดไข่เค็ม2", "MenuPrice": 45, "Store_StoreId": 2, "MenuDate": "2022-10-07T17:00:00.000Z", </pre>

	<pre> "MenuImage": "1665619469562.png", "menustatus_MenuStatus": 1, "StoreID": 2, "StoreName": "ซั้อนทอง", "StoreImage": "1665617244109.png", "owner_OwnerID": 1, "storestatus_StoreStatus": 2, "MenuStatus": 1, "MenuNameStatus": "เปิด" }, { "MenuID": 1, "MenuName": "ซั้วผั้ดกะเพราปลาหมึก", "MenuPrice": 30, "Store_StoreID": 2, "MenuDate": "2022-10-07T17:00:00.000Z", "MenuImage": "1665619428351.png", "menustatus_MenuStatus": 2, "StoreID": 2, "StoreName": "ซั้อนทอง", "StoreImage": "1665617244109.png", "owner_OwnerID": 1, "storestatus_StoreStatus": 2, "MenuStatus": 2, "MenuNameStatus": "ปิด" },] </pre>
--	--

3.3 การแสดงผลร้านอาหารและเมนูอาหารตามร้าน

การแสดงผลร้านอาหารทั้งหมด

1. URL ที่ใช้

Method	URL
GET	/api/ stores

2. การตอบกลับ

การตอบกลับ
<pre>{ "error": false, "data": [{ "StoreID": 4, "StoreName": "คุณยายสมศรี", "StoreImage": "1665618475746.png", "owner_OwnerID": 3, "storestatus_StoreStatus": 1, "OwnerID": 3, "OwnerName": "สมศรี เสนิงค์ ณ อยุรยา", "OwnerUser": "Somsri", "OwnerCol": "123456", "OwnerPhone": "025846434" }, { "StoreID": 2, "StoreName": "ซัอนทอง", "StoreImage": "1665617244109.png", "owner_OwnerID": 1, "storestatus_StoreStatus": 2, "OwnerID": 1, "OwnerName": "คิวนสวย ศรีหา", </pre>

```

    "OwnerUser": "Kamonz",
    "OwnerCol": "Nami12755",
    "OwnerPhone": "0922654387"
  }
]

```

การแสดงผลข้อมูลร้านอาหาร 1 ร้าน

1. สิ่งที่ต้องนำเข้า

ตัวแปร	รายละเอียด
StoreID	ID ร้านอาหาร

2. URL ที่ใช้

Method URL
POST /api/storename

3. การตอบกลับ

เงื่อนไข	การตอบกลับ
ไม่ได้กรอก StoreID	{ "error": true, "message": "หาเลขไม่เจอ" }
ใส่ StoreID ไม่ตรงในฐานข้อมูล	ไม่มีการส่งค่ากลับมา
กรอก StoreID ตรงในฐานข้อมูล	{ "StoreID": 2, "StoreName": "ซั๊นทอง", "StoreImage": "1665617244109.png", "owner_OwnerID": 1, "storestatus_StoreStatus": 2, "OwnerID": 1, "OwnerName": "คิวนสวย ศรีหา", "OwnerUser": "Kamonz", "OwnerCol": "Nami12755", "OwnerPhone": "0922654387" }

	}
--	---

3.4 การสั่งอาหาร

การเพิ่มรายการคำสั่งซื้ออาหารสำหรับลูกค้า

1. สิ่งที่ต้องนำเข้า

ตัวแปร	รายละเอียด
OrderDate	วันที่เพิ่มเมนูอาหาร
OrderTime	เวลามารับอาหาร
OrderQty	จำนวนอาหารที่สั่ง
OrderPrice	ราคาอาหารที่สั่ง
CustomerID	ID ลูกค้า
OrderNoteCus	หมายเหตุลูกค้า
MenuID	ID เมนู
StoreID	ID ร้านอาหาร

2. URL ที่ใช้

Method URL
POST /api/order

3. การตอบกลับ

การตอบกลับ
<pre>{ "OrderTime": "12.00 น.", "OrderStatus": "1", "OrderQty": 10, "OrderPrice": 100, "CustomerID": 1, "OrderNoteCus": "ใส่กล่อง", "MenuID": "2" }</pre>

3.5 การตรวจสอบสถานะรายการอาหารสำหรับลูกค้า

การตรวจสอบสถานะรายการอาหารสำหรับลูกค้าทุกสถานะ

1. สิ่งที่ต้องนำเข้า

ตัวแปร	รายละเอียด
CustomerID	ID ผู้ใช้ทั่วไป

2. URL ที่ใช้

Method URL
POST /api/statusfoodcus

3. การตอบกลับ

เงื่อนไข	การตอบกลับ
ไม่ได้กรอก CustomerID	{ "error": true, "message": "หาเลขไม่เจอ" }
กรอก CustomerID ที่ไม่มีอยู่ในฐานข้อมูล	ไม่มีการส่งค่ากลับ
กรอก CustomerID ที่มีอยู่ในฐานข้อมูล	[{ "OrderID": 22, "OrderNoteCus": "ใส่กล่อง", "Menu_Menulid": 21, "Menu_Store_Storeid": 6, "Customer_CustomerID": 1, "OrderNoteOw": " ", "OrderQty": 1, "OrderTime": "09.00 น.", "OrderDate": "2022-10-12T17:00:00.000Z", "OrderPrice": 15, "orderstatus_OrderStatus": 3, "CustomerID": 1, "CustomerName": "นางเอ วิตามิน", "CustomerUser": "aaaaaa", }

	<pre> "CustomerCol": "aaaaaa", "CustomerPhone": "0922654387", "MenuID": 21, "MenuName": "ซูปมิโสะ", "MenuPrice": 15, "Store_StoreId": 6, "MenuDate": "2022-10-12T17:00:00.000Z", "MenuImage": "1665624462586.png", "menustatus_MenuStatus": 1, "StoreID": 6, "StoreName": "แม่มยุวดี", "StoreImage": "1665650314097.png", "owner_OwnerID": 5, "storestatus_StoreStatus": 2, "OrderStatus": 3, "OrderNameStatus": "ได้รับแล้ว" }, { "OrderID": 6, "OrderNoteCus": "", "Menu_MenuId": 8, "Menu_Store_StoreId": 2, "Customer_CustomerID": 1, "OrderNoteOw": " ", "OrderQty": 5, "OrderTime": "11.00 น.", "OrderDate": "2022-10-12T17:00:00.000Z", "OrderPrice": 150, "orderstatus_OrderStatus": 3, "CustomerID": 1, "CustomerName": "นางเอ วิตามิน", "CustomerUser": "aaaaaa", </pre>
--	--

	<pre> "CustomerCol": "aaaaaa", "CustomerPhone": "0922654387", "MenuID": 8, "MenuName": "ข้าวผัดอเมริกัน", "MenuPrice": 30, "Store_StoreId": 2, "MenuDate": "2022-10-12T17:00:00.000Z", "MenuImage": "1665619718572.png", "menustatus_MenuStatus": 1, "StoreID": 2, "StoreName": "ซ็อนทอง", "StoreImage": "1665617244109.png", "owner_OwnerID": 1, "storestatus_StoreStatus": 2, "OrderStatus": 3, "OrderNameStatus": "ได้รับแล้ว" }] </pre>
--	--

การตรวจสอบสถานะรายการอาหารสำหรับลูกค้าเฉพาะสถานะกำลังปรุง

1. สิ่งที่ต้องนำเข้า

ตัวแปร	รายละเอียด
CustomerID	ID ผู้ใช้ทั่วไป

2. URL ที่ใช้

Method URL
POST /api/statusfoodcus1

3. การตอบกลับ

เงื่อนไข	การตอบกลับ
ไม่ได้กรอก CustomerID	{ "error": true, "message": "หาเลขไม่เจอ" }
กรอก CustomerID ที่ไม่มีอยู่ในฐานข้อมูล	ไม่มีการส่งค่ากลับ
กรอก CustomerID ที่มีอยู่ในฐานข้อมูล	{ "OrderID": 31, "OrderNoteCus": "ใส่กล่อง", "Menu_Menuld": 10, "Menu_Store_StoreId": 4, "Customer_CustomerID": 1, "OrderNoteOw": null, "OrderQty": 1, "OrderTime": "11.00 น.", "OrderDate": "2022-10-13T17:00:00.000Z", "OrderPrice": 45, "orderstatus_OrderStatus": 1, "CustomerID": 1, "CustomerName": "นางเอ วิตามิน", "CustomerUser": "aaaaaa", "CustomerCol": "aaaaaa", "CustomerPhone": "0922654387", }

	<pre> "MenuID": 10, "MenuName": "ข้าวขาหมู", "MenuPrice": 45, "Store_StoreId": 4, "MenuDate": "2022-10-12T17:00:00.000Z", "MenuImage": "1665623792685.png", "menustatus_MenuStatus": 1, "StoreID": 4, "StoreName": "คุณยายสมศรี", "StoreImage": "1665618475746.png", "owner_OwnerID": 3, "storestatus_StoreStatus": 1, "OrderStatus": 1, "OrderNameStatus": "กำลังปรุง" }] </pre>
--	--

การตรวจสอบสถานะรายการอาหารสำหรับลูกค้าเฉพาะสถานะเรียบร้อย

1. สิ่งที่ต้องนำเข้าไป

ตัวแปร	รายละเอียด
CustomerID	ID ผู้ใช้ทั่วไป

2. URL ที่ใช้

Method URL
POST /api/statusfoodcus2

3. การตอบกลับ

เงื่อนไข	การตอบกลับ
ไม่ได้กรอก CustomerID	<pre> { "error": true, "message": "หาเลขไม่เจอ" } </pre>
กรอก CustomerID ที่ไม่มีอยู่ในฐานข้อมูล	ไม่มีการส่งค่ากลับ

<p>กรอก CustomerID ที่มีอยู่ในฐานข้อมูล</p>	<pre>[{ "OrderID": 1, "OrderNoteCus": "ใส่กล่อง", "Menu_MenuId": 3, "Menu_Store_StoreId": 2, "Customer_CustomerID": 1, "OrderNoteOw": " ", "OrderQty": 1, "OrderTime": "12.00 น.", "OrderDate": "2022-10-12T17:00:00.000Z", "OrderPrice": 45, "orderstatus_OrderStatus": 2, "CustomerID": 1, "CustomerName": "นางเอ วิตามิน", "CustomerUser": "aaaaaa", "CustomerCol": "aaaaaa", "CustomerPhone": "0922654387", "MenuID": 3, "MenuName": "ข้าวปลาหมึกผัดไข่เค็ม2", "MenuPrice": 45, "Store_StoreId": 2, "MenuDate": "2022-10-07T17:00:00.000Z", "MenuImage": "1665619469562.png", "menustatus_MenuStatus": 1, "StoreID": 2, "StoreName": "ซ็อนทอง", "StoreImage": "1665617244109.png", "owner_OwnerID": 1, "storestatus_StoreStatus": 2, "OrderStatus": 2, "OrderNameStatus": "เรียบร้อย" }</pre>
---	--

	<pre>}]</pre>
--	----------------

การตรวจสอบสถานะรายการอาหารสำหรับลูกค้าเฉพาะสถานะได้รับแล้ว

1. สิ่งที่ต้องนำเข้า

ตัวแปร	รายละเอียด
CustomerID	ID ผู้ใช้ทั่วไป

2. URL ที่ใช้

Method URL
POST /api/statusfoodcus3

3. การตอบกลับ

เงื่อนไข	การตอบกลับ
ไม่ได้กรอก CustomerID	<pre>{ "error": true, "message": "หาเลขไม่เจอ" }</pre>
กรอก CustomerID ที่ไม่มีอยู่ในฐานข้อมูล	ไม่มีการส่งค่ากลับ
กรอก CustomerID ที่มีอยู่ในฐานข้อมูล	<pre>[{ "OrderID": 1, "OrderNoteCus": "ใส่กล่อง", "Menu_MenuId": 3, "Menu_Store_StoreId": 2, "Customer_CustomerID": 1, "OrderNoteOw": " ", "OrderQty": 1, "OrderTime": "12.00 น.", "OrderDate": "2022-10-12T17:00:00.000Z", "OrderPrice": 45, "orderstatus_OrderStatus": 3, "CustomerID": 1,</pre>

	<pre> "CustomerName": "นางเอ วิตามิน", "CustomerUser": "aaaaaa", "CustomerCol": "aaaaaa", "CustomerPhone": "0922654387", "MenuID": 3, "MenuName": "ข้าวปลาหมึกผัดไข่เค็ม2", "MenuPrice": 45, "Store_StoreId": 2, "MenuDate": "2022-10-07T17:00:00.000Z", "MenuImage": "1665619469562.png", "menustatus_MenuStatus": 1, "StoreID": 2, "StoreName": "ซ็อนทอง", "StoreImage": "1665617244109.png", "owner_OwnerID": 1, "storestatus_StoreStatus": 2, "OrderStatus": 3, "OrderNameStatus": "ได้รับแล้ว" }] </pre>
--	--

3.6 การตรวจสอบสถานะรายการอาหารลูกค้าสำหรับเจ้าของร้าน

การตรวจสอบสถานะรายการอาหารลูกค้าสำหรับเจ้าของร้านทุกสถานะ

1. สิ่งที่ต้องนำเข้า

ตัวแปร	รายละเอียด
StoreID	ID ผู้ใช้ทั่วไป

2. URL ที่ใช้

Method URL
POST /api/statusfoodow

3. การตอบกลับ

เงื่อนไข	การตอบกลับ
ไม่ได้กรอก StoreID	{

	<pre> "error": true, "message": "หาเลขไม่เจอ" } </pre>
กรอก StoreIDที่ไม่มีอยู่ในฐานข้อมูล	ไม่มีการส่งค่ากลับ
กรอก StoreID ที่มีอยู่ในฐานข้อมูล	<pre> [{ "OrderID": 1, "OrderNoteCus": "ใส่กล่อง", "Menu_MenuId": 3, "Menu_Store_StoreId": 2, "Customer_CustomerID": 1, "OrderNoteOw": " ", "OrderQty": 1, "OrderTime": "12.00 น.", "OrderDate": "2022-10-12T17:00:00.000Z", "OrderPrice": 45, "orderstatus_OrderStatus": 2, "MenuID": 3, "MenuName": "ข้าวปลาหมักผัดไข่เค็ม2", "MenuPrice": 45, "Store_StoreId": 2, "MenuDate": "2022-10-07T17:00:00.000Z", "MenuImage": "1665619469562.png", "menustatus_MenuStatus": 1, "StoreID": 2, "StoreName": "ซ็อนทอง", "StoreImage": "1665617244109.png", "owner_OwnerID": 1, "storestatus_StoreStatus": 2, "CustomerID": 1, "CustomerName": "นางเอ วิตามิน", "CustomerUser": "aaaaaa", </pre>

	<pre> "CustomerCol": "aaaaaa", "CustomerPhone": "0922654387", "OrderStatus": 2, "OrderNameStatus": "เรียบร้อย" }, { "OrderID": 7, "OrderNoteCus": "sfdgfhg", "Menu_MenuId": 3, "Menu_Store_StoreId": 2, "Customer_CustomerID": 2, "OrderNoteOw": " ", "OrderQty": 1, "OrderTime": "13.00 น.", "OrderDate": "2022-10-12T17:00:00.000Z", "OrderPrice": 45, "orderstatus_OrderStatus": 4, "MenuID": 3, "MenuName": "ข้าวปลาหมึกผัดไข่เค็ม2", "MenuPrice": 45, "Store_StoreId": 2, "MenuDate": "2022-10-07T17:00:00.000Z", "MenuImage": "1665619469562.png", "menustatus_MenuStatus": 1, "StoreID": 2, "StoreName": "ซ้อนทอง", "StoreImage": "1665617244109.png", "owner_OwnerID": 1, "storestatus_StoreStatus": 2, "CustomerID": 2, "CustomerName": "น้องปี วิตามิน", "CustomerUser": "bbbbbb", </pre>
--	---

	<pre>"CustomerCol": "bbbbbb", "CustomerPhone": "0999999999", "OrderStatus": 4, "OrderNameStatus": "ถูกยกเลิก" }</pre>
--	---

การตรวจสอบสถานะรายการอาหารลูกค้าสำหรับเจ้าของร้านเฉพาะสถานะกำลังปรุง

1. สิ่งที่ต้องนำเข้า

ตัวแปร	รายละเอียด
StoreID	ID ผู้ใช้ทั่วไป

2. URL ที่ใช้

Method URL
POST /api/statusfoodow1

3. การตอบกลับ

เงื่อนไข	การตอบกลับ
ไม่ได้กรอก StoreID	<pre>{ "error": true, "message": "หาเลขไม่เจอ" }</pre>
กรอก StoreID ที่ไม่มีอยู่ในฐานข้อมูล	ไม่มีการส่งค่ากลับ
กรอก StoreID ที่มีอยู่ในฐานข้อมูล	<pre>[{ "OrderID": 1, "OrderNoteCus": "ใส่กล่อง", "Menu_MenulId": 3, "Menu_Store_StoreId": 2, "Customer_CustomerId": 1, "OrderNoteOw": " ", "OrderQty": 1, "OrderTime": "12.00 น.", "OrderDate": "2022-10-12T17:00:00.000Z", "OrderPrice": 45,</pre>

	<pre> "orderstatus_OrderStatus": 1, "MenuID": 3, "MenuName": "ข้าวปลาหมักผัดไข่เค็ม2", "MenuPrice": 45, "Store_StoreId": 2, "MenuDate": "2022-10-07T17:00:00.000Z", "MenuImage": "1665619469562.png", "menustatus_MenuStatus": 1, "StoreID": 2, "StoreName": "ซ็อนทอง", "StoreImage": "1665617244109.png", "owner_OwnerID": 1, "storestatus_StoreStatus": 2, "CustomerID": 1, "CustomerName": "นางเอ วิตามิน", "CustomerUser": "aaaaaa", "CustomerCol": "aaaaaa", "CustomerPhone": "0922654387", "OrderStatus": 1, "OrderNameStatus": "กำลังปรุง" }]</pre>
--	---

การตรวจสอบสถานะรายการอาหารลูกค้าสำหรับเจ้าของร้านเฉพาะสถานะเรียบร้อย

1. สิ่งที่ต้องนำเข้า

ตัวแปร	รายละเอียด
StoreID	ID ผู้ใช้ทั่วไป

2. URL ที่ใช้

Method URL
POST /api/statusfoodow2

3. การตอบกลับ

เงื่อนไข	การตอบกลับ
ไม่ได้กรอก StoreID	{ "error": true, "message": "หาเลขไม่เจอ" }
กรอก StoreID ที่ไม่มีอยู่ในฐานข้อมูล	ไม่มีการส่งค่ากลับ
กรอก StoreID ที่มีอยู่ในฐานข้อมูล	[{ "OrderID": 2, "OrderNoteCus": "ใส่กล่อง", "Menu_MenuId": 1, "Menu_Store_StoreId": 2, "Customer_CustomerID": 1, "OrderNoteOw": " ", "OrderQty": 2, "OrderTime": "12.00 น.", "OrderDate": "2022-10-12T17:00:00.000Z", "OrderPrice": 60, "orderstatus_OrderStatus": 2, "MenuID": 1, "MenuName": "ข้าวผัดกะเพราปลาหมึก", "MenuPrice": 30, "Store_StoreId": 2, "MenuDate": "2022-10-07T17:00:00.000Z", "MenuImage": "1665619428351.png", "menustatus_MenuStatus": 1, "StoreID": 2, "StoreName": "ซ็อนทอง", "StoreImage": "1665617244109.png", "owner_OwnerID": 1, "storestatus_StoreStatus": 2, "CustomerID": 1, }

	<pre> "CustomerName": "นางเอ วิตามิน", "CustomerUser": "aaaaaa", "CustomerCol": "aaaaaa", "CustomerPhone": "0922654387", "OrderStatus": 2, "OrderNameStatus": "เรียบร้อย" }]</pre>
--	---

การตรวจสอบสถานะรายการอาหารลูกค้าสำหรับเจ้าของร้านเฉพาะสถานะได้รับแล้ว

1. สิ่งที่ต้องนำเข้า

ตัวแปร	รายละเอียด
StoreID	ID ผู้ใช้ทั่วไป

2. URL ที่ใช้

Method URL
POST /api/statusfoodow3

3. การตอบกลับ

เงื่อนไข	การตอบกลับ
ไม่ได้กรอก StoreID	<pre> { "error": true, "message": "หาเลขไม่เจอ" }</pre>
กรอก StoreID ที่ไม่มีอยู่ในฐานข้อมูล	ไม่มีการส่งค่ากลับ
กรอก StoreID ที่มีอยู่ในฐานข้อมูล	<pre> [{ "OrderID": 3, "OrderNoteCus": "ใส่กล่อง", "Menu_MenuId": 9, "Menu_Store_StoreId": 2, "Customer_CustomerID": 1, "OrderNoteOw": " ", </pre>

	<pre>"OrderQty": 3, "OrderTime": "12.00 น.", "OrderDate": "2022-10-12T17:00:00.000Z", "OrderPrice": 135, "orderstatus_OrderStatus": 3, "MenuID": 9, "MenuName": "ชี้อ", "MenuPrice": 55, "Store_StoreId": 2, "MenuDate": "2022-10-12T17:00:00.000Z", "MenuImage": "1665619759345.png", "menustatus_MenuStatus": 1, "StoreID": 2, "StoreName": "ซ้อหนอง", "StoreImage": "1665617244109.png", "owner_OwnerID": 1, "storestatus_StoreStatus": 2, "CustomerID": 1, "CustomerName": "นางเอ วิตามิน", "CustomerUser": "aaaaaa", "CustomerCol": "aaaaaa", "CustomerPhone": "0922654387", "OrderStatus": 3, "OrderNameStatus": "ได้รับแล้ว" }]</pre>
--	---

การตรวจสอบสถานะรายการอาหารลูกค้าสำหรับเจ้าของร้านเฉพาะสถานะถูกยกเลิก

1. สิ่งที่ต้องนำเข้า

ตัวแปร	รายละเอียด
StoreID	ID ผู้ใช้ทั่วไป

2. URL ที่ใช้

Method URL
POST /api/statusfoodow4

3. การตอบกลับ

เงื่อนไข	การตอบกลับ
ไม่ได้กรอก StoreID	{ "error": true, "message": "หาเลขไม่เจอ" }
กรอก StoreID ที่ไม่มีอยู่ในฐานข้อมูล	ไม่มีการส่งค่ากลับ
กรอก StoreID ที่มีอยู่ในฐานข้อมูล	{ "OrderID": 7, "OrderNoteCus": "sfdgfhg", "Menu_MenulD": 3, "Menu_Store_StoreId": 2, "Customer_CustomerID": 2, "OrderNoteOw": " ", "OrderQty": 1, "OrderTime": "13.00 น.", "OrderDate": "2022-10-12T17:00:00.000Z", "OrderPrice": 45, "orderstatus_OrderStatus": 4, "MenuID": 3, "MenuName": "ข้าวปลาหมึกผัดไข่เค็ม2", "MenuPrice": 45, "Store_StoreId": 2, "MenuDate": "2022-10-07T17:00:00.000Z", }

	<pre> "MenuImage": "1665619469562.png", "menustatus_MenuStatus": 1, "StoreID": 2, "StoreName": "ซั๋อนทอง", "StoreImage": "1665617244109.png", "owner_OwnerID": 1, "storestatus_StoreStatus": 2, "CustomerID": 2, "CustomerName": "นั๋องปี วิตามิน", "CustomerUser": "bbbbbb", "CustomerCol": "bbbbbb", "CustomerPhone": "0999999999", "OrderStatus": 4, "OrderNameStatus": "ถูกยกเลิก" }]</pre>
--	--

3.7 การจัดการสถานะรายการคำสั่งซื้ออาหารของลูกค้าสำหรับเจ้าของร้าน

การอัปเดตสถานะรายการคำสั่งซื้ออาหารของลูกค้า

1. สิ่งที่ต้องนำเข้า

ตัวแปร	รายละเอียด
OrderID	ID รายการคำสั่งซื้ออาหาร
OrderStatus	สถานะรายการคำสั่งซื้ออาหาร
OrderNoteOw (ไม่จำเป็นต้องใส่)	หมายเหตุของเจ้าของร้าน

2. URL ที่ใช้

Method URL
PUT /api/statusfoodow

3. การตอบกลับ

เงื่อนไข	การตอบกลับ
ไม่ได้กรอก OrderID, OrderStatus	{ "error": true,

	<pre>"message": "กรุณาใส่ข้อมูลให้ครบถ้วน" }</pre>
ใส่ OrderID ไม่ตรงในฐานข้อมูล	<pre>{ "error": false, "data": { "fieldCount": 0, "affectedRows": 0, "insertId": 0, "info": "Rows matched: 0 Changed: 0 Warnings: 0", "serverStatus": 2, "warningStatus": 0, "changedRows": 0 }, "message": "หาคำสั่งซื้อไม่เจอ" }</pre>
กรอกข้อมูลครบถ้วน	<pre>{ "error": false, "data": { "fieldCount": 0, "affectedRows": 1, "insertId": 0, "info": "Rows matched: 1 Changed: 1 Warnings: 0", "serverStatus": 2, "warningStatus": 0, "changedRows": 1 }, "message": "อัปเดตสำเร็จ" }</pre>

การลบรายการคำสั่งซื้ออาหารของลูกค้า

1. สิ่งที่ต้องนำเข้า

ตัวแปร	รายละเอียด
id	ID คำสั่งซื้ออาหาร

2. URL ที่ใช้

Method URL
DELETE /api/statusfoodow

3. การตอบกลับ

เงื่อนไข	การตอบกลับ
ไม่ได้กรอก id	{ "error": true, "message": "กรุณาใส่ id " }
ใส่ ID คำสั่งซื้ออาหาร ไม่ตรงในฐานข้อมูล	{ "error": false, "data": { "fieldCount": 0, "affectedRows": 0, "insertId": 0, "info": "", "serverStatus": 2, "warningStatus": 0 }, "message": "ไม่สามารถลบได้ ไม่มี id นี้อยู่ในระบบ" }
มี ID อยู่ในฐานข้อมูล แต่ id นั้นอยู่ในสถานะ รายการคำสั่งซื้อ อาหารของลูกค้า	{ "message": "ER_ROW_IS_REFERENCED_2" }
มี ID อยู่ในฐานข้อมูล	{ "error": false,

	<pre> "data": { "fieldCount": 0, "affectedRows": 1, "insertId": 0, "info": "", "serverStatus": 2, "warningStatus": 0 }, "message": "ลบสำเร็จ" } </pre>
--	--

การค้นหาคำสั่งซื้ออาหารตาม ID รายการคำสั่งซื้ออาหาร

1. สิ่งที่ต้องนำเข้า

ตัวแปร	รายละเอียด
Id : ใส่ไว้ผ่าน path	ID รายการคำสั่งซื้ออาหาร

2. URL ที่ใช้

Method URL
GET /api/statusfoodow/:id

3. การตอบกลับ

เงื่อนไข	การตอบกลับ
ใส่ ID เมนูอาหารไม่ตรงในฐานข้อมูล	<pre> { "error": false, "message": "หาคำสั่งซื้อไม่เจอ" } </pre>
กรอก ID ตรงในฐานข้อมูล	<pre> { "error": false, "data": { "OrderID": 1, "OrderNoteCus": "ใส่กล่อง", "Menu_MenuId": 3, "Menu_Store_StoreId": 2, </pre>

	<pre> "Customer_CustomerID": 1, "OrderNoteOw": " ", "OrderQty": 1, "OrderTime": "12.00 น.", "OrderDate": "2022-10-12T17:00:00.000Z", "OrderPrice": 45, "orderstatus_OrderStatus": 1, "MenuID": 3, "MenuName": "ข้าวปลาหมึกผัดไข่เค็ม2", "MenuPrice": 45, "Store_StoreId": 2, "MenuDate": "2022-10-07T17:00:00.000Z", "MenuImage": "1665619469562.png", "menustatus_MenuStatus": 1, "StoreID": 2, "StoreName": "ซั๊นทอง", "StoreImage": "1665617244109.png", "owner_OwnerID": 1, "storestatus_StoreStatus": 2, "CustomerID": 1, "CustomerName": "นางเอ วิตามิน", "CustomerUser": "aaaaaa", "CustomerCol": "aaaaaa", "CustomerPhone": "0922654387" }, "message": "เจอละ" } </pre>
--	---

3.8 การจัดการข้อมูลร้านอาหารสำหรับเจ้าของร้าน

การค้นหาร้านอาหารตาม ID ร้านอาหาร

1. สิ่งที่ต้องนำเข้า

ตัวแปร	รายละเอียด
StoreID	ID ร้านอาหาร

2. URL ที่ใช้

Method URL
POST /api/storepw

3. การตอบกลับ

เงื่อนไข	การตอบกลับ
กรณีไม่ได้กรอก ID ร้านอาหาร	{ "error": true, "message": "หาเลขไม่เจอ" }
กรณีค้นหาเมนูอาหาร ตามเลขร้านอาหารไม่ เจอ	ไม่ได้คืนค่ากลับ
กรณีค้นหาเมนูอาหาร ตามเลขร้านอาหารเจอ	{ "StoreID": 2, "StoreName": "ซั๋นทอง", "StoreImage": "1665617244109.png", "owner_OwnerID": 1, "storestatus_StoreStatus": 2, "OwnerID": 1, "OwnerName": "คิวนสวย ศรีหา", "OwnerUser": "Kamonz", "OwnerCol": "Nami12755", "OwnerPhone": "0922654387" }

การอัปเดตร้านอาหาร

1. สิ่งที่ต้องนำเข้า

ตัวแปร	รายละเอียด
StoreID	ID ร้านอาหาร
StoreStatus	ID สถานะร้านอาหาร
StoreName	ชื่อร้านอาหาร
filename(เป็นชนิดไฟล์ที่ดึงชื่อรูปภาพ) *ไม่จำเป็นต้องใส่*	ชื่อไฟล์รูปภาพ

2. URL ที่ใช้

Method URL
PUT /api/ storepw

3. การตอบกลับ

เงื่อนไข	การตอบกลับ
ไม่ได้กรอก StoreName และ StoreStatus	{ "error": true, "message": "กรุณาใส่ข้อมูลให้ครบถ้วน" }
ใส่ ID เมนูอาหารไม่ ตรงในฐานข้อมูล	{ "error": false, "data": { "fieldCount": 0, "affectedRows": 0, "insertId": 0, "info": "Rows matched: 0 Changed: 0 Warnings: 0", "serverStatus": 2, "warningStatus": 0, "changedRows": 0 }, "message": "หาร้านอาหารไม่เจอ" }
กรอกข้อมูลครบถ้วน	{

	<pre> "error": false, "data": { "fieldCount": 0, "affectedRows": 1, "insertId": 0, "info": "Rows matched: 1 Changed: 1 Warnings: 0", "serverStatus": 2, "warningStatus": 0, "changedRows": 1 }, "message": "อัปเดตสำเร็จ" } </pre>
--	--

3.9 การแสดงผลรายงานสถิติยอดขายของร้านอาหาร

การแสดงผลรายงานสถิติอาหารขายดีที่สุดของร้าน

1. สิ่งที่ต้องนำเข้า

ตัวแปร	รายละเอียด
StoreID	ID ร้านอาหาร

2. URL ที่ใช้

Method	URL
POST	/api/mostfood

3. การตอบกลับ

เงื่อนไข	การตอบกลับ
กรณีไม่ได้กรอก ID ร้านอาหาร	<pre> { "error": true, "message": "หาเลขไม่เจอ" } </pre>
กรณีค้นหาเมนูอาหาร ตามเลขร้านอาหารไม่ เจอ	ไม่ได้คืนค่ากลับ

<p>กรณีค้นหาเมนูอาหาร ตามเลขร้านอาหารเจอ (ไฟล์ข้อมูลสูงสุด 5 อันดับ)</p>	<pre>[{ "MostfoodRow": 1, "MenuName": "ข้าวกะเพราหมูกรอบ", "OrderMostPrice": 270 }, { "MostfoodRow": 2, "MenuName": "ข้าวผัดอเมริกัน", "OrderMostPrice": 240 }, { "MostfoodRow": 3, "MenuName": "ข้าวปลาหมึกผัดไข่เค็ม", "OrderMostPrice": 135 }, { "MostfoodRow": 4, "MenuName": "ข้าวหมูกระเทียม", "OrderMostPrice": 135 }, { "MostfoodRow": 5, "MenuName": "ข้าวผัดกะเพราปลาหมึก", "OrderMostPrice": 30 }]</pre>
--	---

การแสดงผลรายงานสถิติอาหารขายตามรายวัน

1. สิ่งที่ต้องนำเข้า

ตัวแปร	รายละเอียด
StoreID	ID ร้านอาหาร

2. URL ที่ใช้

Method URL
POST /api/dayfood

3. การตอบกลับ

เงื่อนไข	การตอบกลับ
กรณีไม่ได้กรอก ID ร้านอาหาร	{ "error": true, "message": "หาเลขไม่เจอ" }
กรณีค้นหาเมนูอาหารตามเลขร้านอาหารไม่เจอ	ไม่ได้คืนค่ากลับ
กรณีค้นหาเมนูอาหารตามเลขร้านอาหารเจอ (โผล่ข้อมูลสูงสุด 5 อันดับ)	[{ "DayRow": 7, "OrderDate": "2022-10-14T17:00:00.000Z", "OrderPrice": 30 }, { "DayRow": 6, "OrderDate": "2022-10-12T17:00:00.000Z", "OrderPrice": 330 }, { "DayRow": 5, "OrderDate": "2022-09-12T17:00:00.000Z", "OrderPrice": 45 }

	<pre> }, { "DayRow": 4, "OrderDate": "2022-09-08T17:00:00.000Z", "OrderPrice": 45 }, { "DayRow": 3, "OrderDate": "2022-08-08T17:00:00.000Z", "OrderPrice": 135 }] </pre>
--	---

การแสดงผลรายงานสถิติอาหารขายตามรายเดือน

1. สิ่งที่ต้องนำเข้า

ตัวแปร	รายละเอียด
StoreID	ID ร้านอาหาร

2. URL ที่ใช้

Method URL
POST /api/ monthfood

3. การตอบกลับ

เงื่อนไข	การตอบกลับ
กรณีไม่ได้กรอก ID ร้านอาหาร	<pre> { "error": true, "message": "หาเลขไม่เจอ" } </pre>
กรณีค้นหาเมนูอาหาร ตามเลขร้านอาหารไม่ เจอ	ไม่ได้คืนค่ากลับ
กรณีค้นหาเมนูอาหาร ตามเลขร้านอาหารเจอ	<pre> [{ </pre>

<p>(ไฟล์ข้อมูลสูงสุด 5 อันดับ)</p>	<pre> "MonthRow": 1, "Month": 10, "OrderPrice": 360 }, { "MonthRow": 2, "Month": 9, "OrderPrice": 90 }, { "MonthRow": 3, "Month": 8, "OrderPrice": 135 }, { "MonthRow": 4, "Month": 7, "OrderPrice": 45 }, { "MonthRow": 5, "Month": 6, "OrderPrice": 45 } }]</pre>
------------------------------------	--

การแสดงผลรายงานสถิติอาหารขายตามรายปี

1. สิ่งที่ต้องนำเข้า

ตัวแปร	รายละเอียด
StoreID	ID ร้านอาหาร

2. URL ที่ใช้

Method URL
POST /api/yearfood

3. การตอบกลับ

เงื่อนไข	การตอบกลับ
กรณีไม่ได้กรอก ID ร้านอาหาร	{ "error": true, "message": "หาเลขไม่เจอ" }
กรณีค้นหาเมนูอาหาร ตามเลขร้านอาหารไม่ เจอ	ไม่ได้คืนค่ากลับ
กรณีค้นหาเมนูอาหาร ตามเลขร้านอาหารเจอ (ไฟล์ข้อมูลสูงสุด 5 อันดับ)	[[{ "YearRow": 1, "Year": 2022, "OrderPrice": 675 }, { "YearRow": 2, "Year": 2021, "OrderPrice": 135 }]

Project team Lesson Learned

เว็บไซต์ อาหารโรง สั้	ทำอะไร	มีวิธีการอย่างไร	ผลลัพธ์	แนวทางแก้ไขปัญหา
การวางแผน	วางแผนว่าจะโปรเจค ทำ อะไรได้หรือไม่ได้ บ้าง	มีการสื่อสารกับสมาชิก ในกลุ่มเสนอหัวข้อ และรับฟังความคิดเห็น	ได้ว่าเว็บไซต์ของเรา สามารถทำอะไรได้บ้าง	ไม่มี
แบ่งหน้าที่ ทำงาน	วางแผนตกลงกันคน ใน กลุ่มว่าใครจะทำ อะไรบ้าง	มีการสื่อสารกับสมาชิก ในกลุ่ม ทำข้อตกลงกัน ว่าใครทำอะไร สามารถทำอะไรได้บ้าง	สมาชิกทุกคนได้มี หน้าที่ เป็นของตัวเอง และทำ หน้าที่ของ ตัวเอง ไม่มี	ไม่มี
กำหนด ระยะเวลา	วางแผนกับคนใน กลุ่ม ว่างานในแต่ละ ส่วนควร จะเสร็จภายใน วันไหน	มีการสื่อสารและทำ ข้อตกลงกันว่า งานแต่ละ ส่วนควร เสร็จวัน ไหน	ได้ระยะเวลาในการ ทำงาน แต่งาน บางส่วน ไม่สามารถทำ ทันตาม กำหนดการได้	เวลากำหนดระยะเวลาต้อง ดู ความเหมาะสมต่อการ ทำงานแต่ละส่วน และเมื่อ มี สมาชิกคนไหนไม่ สามารถทำ ทันตาม กำหนดการได้ จำเป็นต้อง ให้สมาชิกคนอื่น เข้าไป ช่วย
ออกแบบ	เป็นการออกแบบ หน้าเว็บไซต์เพื่อให้ ง่าย และรู้ว่าควรทำ อะไร	สื่อสารกับคนในกลุ่ม และร่วมมือกัน ออกแบบ	ได้รับโครงร่างการ ออกแบบ แต่ออกแบบ ได้ไม่ครบฟังก์ชันที่ได้ ทำ จริง	จำเป็นต้องทำข้อตกลงกับ คนในกลุ่มใหม่ เมื่อมีการ ออกแบบในครั้งถัดไป
การพัฒนา	เป็นส่วนในการทำ เว็บไซต์ตามการ ออกแบบ	ผู้ทำหน้าที่เป็น ผู้พัฒนาจะต้องทำเว็บไซต์ตามการออกแบบ	ได้เว็บไซต์ตรงตามการ ออกแบบ แต่มี บางส่วน ไม่ใช่สามารถใช้ การได้จึง เลี่ยงไปใช้ แบบอื่นแทน	จำเป็นต้องศึกษาความรู้ที่ เกี่ยวกับการเขียนเว็บไซต์ มากกว่านี้

การติดตั้ง	เป็นส่วนในการนำเว็บ ไซต์มาเชื่อมต่อ database	ผู้ที่ทำหน้าที่เป็นผู้พัฒนาจะต้องนำเว็บ ไซต์มาเชื่อมต่อกับ database	สามารถเชื่อมต่อกันได้สำเร็จ	ไม่มี
ทดสอบระบบ	เป็นการทดสอบเว็บไซต์	สมาชิกทุกคนจะต้องร่วมมือกันหาจุดบกพร่องของชิ้นงาน และแก้ไข	ตัวชิ้นงานยังมีจุดบกพร่องค่อนข้างมาก แต่คณะผู้จัดทำไม่สามารถแก้ไขได้ทัน ระยะเวลาที่กำหนด และไม่รอบคอบ	จำเป็นจะต้องให้บุคคลภายนอกช่วยทดสอบเว็บไซต์และเพิ่มเวลาในการทดสอบระบบ
ทำเอกสาร	ทำเอกสารคู่มือใช้งาน และคู่มือการพัฒนาและการติดตั้ง	สมาชิกทุกคนมีการวางแผนในการทำเอกสาร	ได้เอกสาร 1.คู่มือใช้งาน 2. คู่มือการพัฒนาและการติดตั้ง	ไม่มี
การนำเสนอ	เป็นการนำเสนอชิ้นงาน	สมาชิกทุกคนได้วางแผน ฝึกซ้อมการนำเสนอ และนำเสนอจริง	นำเสนอได้สำเร็จและตรงเวลา	ไม่มี

ตารางที่ 10 ตารางอธิบาย Project team Lesson Learned