

ข้อเสนอโครงการเตรียมสหกิจศึกษา

กลุ่มที่ 2

เรื่อง เว็บไซต์สั่งอาหารโรงส้ม

Orange Food Shop

เสนอต่อ ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ กำแพงแสน มหาวิทยาลัยเกษตรศาสตร์

เพื่อทำการอนุมัติทำการประกอบวิชาเตรียมสหกิจศึกษา ภาควิชาการศึกษาด้าน ปีการศึกษา 2565

โดย	นางสาวกมลชนก	ศรिता	รหัสประจำตัวนิสิต	6220503163
	นายวัชรศักดิ์	ชื่นชม	รหัสประจำตัวนิสิต	6220503341
	นายเสกฐนันท์	เลี่ยมयोगโย	รหัสประจำตัวนิสิต	6220503368

บทนำ

1. ที่มาและความสำคัญ

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

ด้วยทางคณะผู้จัดทำเห็นว่าเทคโนโลยีระบบสั่งอาหารออนไลน์ กำลังเป็นที่นิยมในปัจจุบัน ทางคณะผู้จัดทำจึงได้มีความคิดสร้างเว็บไซต์สั่งอาหารโรงสั้มเพื่อสำหรับให้คนขายอาหารสามารถจัดการ เพิ่ม/แก้ไข และลบเมนูอาหารภายในร้านบนเว็บไซต์ และจัดการรายการคำสั่งอาหารของลูกค้าผ่านทางเว็บไซต์ได้ และลูกค้าสามารถสั่งอาหาร และเช็คสถานะของรายการอาหารที่ตัวเองสั่งผ่านทางเว็บไซต์ได้

2. วัตถุประสงค์

2.1 เพื่อสร้างและพัฒนาเว็บไซต์สั่งอาหารโรงสั้ม

2.2 เพื่อศึกษาภาษา JavaScript , HTML/CSS , และ React ในการพัฒนาเว็บไซต์

2.3 เพื่อศึกษา และจัดทำฐานข้อมูลที่เกี่ยวข้องกับระบบสั่งอาหารออนไลน์

3. ขอบเขตและข้อจำกัดของโครงการ

3.1 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

3.2 ขอบเขตของโครงการ

เป็นเว็บไซต์ที่สามารถสั่งอาหารและจัดการระบบข้อมูลได้ ภายในขอบเขตโรงอาหารกลาง 2 ในมหาวิทยาลัยเกษตรศาสตร์ วิทยาเขตกำแพงแสน

3.3 ข้อจำกัดของโครงการ

สิ่งที่โครงการสามารถทำได้

- ระบบสั่งอาหารสามารถกดเลือกสั่งหลายเมนู หลายจำนวน และกำหนดเวลาในการรับอาหารได้ ครั้งละ 1 ร้าน
- ระบบตรวจสอบสถานะของรายการอาหารที่ตัวเองสั่งได้
- ระบบเพิ่ม แก้ไข และลบเมนูอาหารสำหรับเจ้าของร้านอาหาร
- ระบบลบ และแก้ไขสถานะรายการคำสั่งอาหารของลูกค้าสำหรับเจ้าของร้านอาหาร
- ระบบตรวจสอบรายงานสถิติยอดขายในรูปแบบกราฟและตาราง

สิ่งที่โครงการไม่สามารถทำได้

- ระบบชำระเงินคำสั่งอาหารล่วงหน้า
- ระบบตรวจสอบคิวตามรายการอาหารล่าสุด

4. ประโยชน์ที่คาดว่าจะได้รับ

- 4.1 ได้เว็บไซต์สั่งอาหารโรงแรมที่สามารถใช้งาน และอำนวยความสะดวกให้กับผู้ใช้ได้
- 4.2 ได้เรียนรู้ภาษา JavaScript , HTML/CSS และการทำ API โดยใช้ Nodejs
- 4.3 ได้ทดลองการนำฐานข้อมูล Microsoft Azure มาใช้ และได้รับฐานข้อมูลที่เกี่ยวข้องกับระบบสั่งอาหารออนไลน์

5. วัสดุ อุปกรณ์และซอฟต์แวร์

5.1 ฮาร์ดแวร์

- 5.1.1 คอมพิวเตอร์พกพา 3 เครื่อง

5.2 ซอฟต์แวร์

- 5.2.1 ระบบปฏิบัติการ Microsoft Windows 10
- 5.2.2 โปรแกรม Visual Studio Code
- 5.2.3 ฐานข้อมูล Microsoft Azure

5.2.4 Canva สำหรับออกแบบกราฟิกตกแต่งเว็บไซต์

5.2.5 Post man ไว้สำหรับทดสอบเรียกใช้ API

6. ตารางแผนงาน

6.1 ตารางหน้าที่

การทำงาน	ชื่อ	ตำแหน่ง
สร้าง และดูแลฐานข้อมูล	เสฏฐนันท์ เลี่ยมยองใย	Database Administrator
ทำหน้าเว็บไซต์ , สร้าง และเชื่อมต่อ API	กมลชนก ศรีหา	Developer
สร้าง API	วัชรศักดิ์ ชื่นชม	Developer

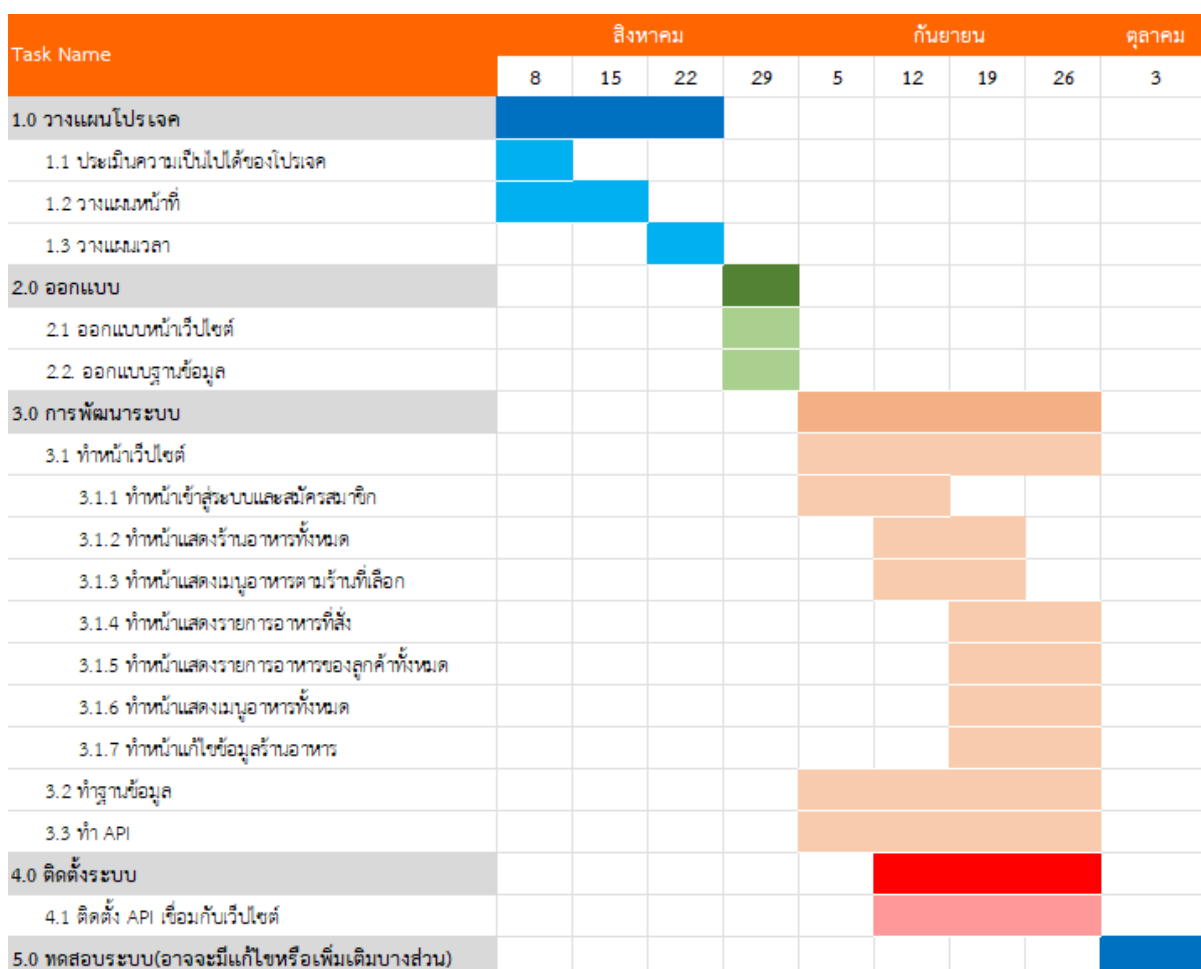
6.2 ตารางความคืบหน้าของโครงการ

สิ่งส่งมอบ	วันที่โดยประมาณ	บุคคลที่รับผิดชอบ
วางแผนการทำโครงการ แบ่งหน้าที่ และกำหนดระยะเวลา	8-22 สิงหาคม 2565	ทุกคน
ออกแบบหน้าเว็บไซต์	8-22 สิงหาคม 2565	ทุกคน
ออกแบบฐานข้อมูล	8-22 สิงหาคม 2565	ทุกคน
ทำหน้าเข้าสู่ระบบและสมัครสมาชิก, ทำ หน้าแสดงร้านอาหารทั้งหมด	5-18 กันยายน 2565	กมลชนก
ทำหน้าแสดงเมนูอาหารตามร้านที่เลือก , ทำหน้าแสดงรายการอาหารที่สั่ง	12-25 กันยายน 2565	กมลชนก
ทำหน้าแสดงรายการอาหารของลูกค้า ทั้งหมด, ทำหน้าแสดงเมนูอาหารทั้งหมด , ทำหน้าแก้ไขข้อมูลร้านอาหาร	19-26 กันยายน 2565	กมลชนก
ส่ง Proposal	30 กันยายน 2565	กมลชนก

สร้าง และดูแลจัดการฐานข้อมูล	5 กันยายน 2565 – 2 ตุลาคม 2565	เสกฐันนท์
ทำ API ไว้สำหรับประสานงานระหว่าง หน้าเว็บไซต์และฐานข้อมูล	5 กันยายน 2565 – 2 ตุลาคม 2565	วัชรศักดิ์ , กมลชนก
ติดตั้ง API เชื่อมกับเว็บไซต์	12 กันยายน 2565 – 2 ตุลาคม 2565	วัชรศักดิ์ , กมลชนก
ทดสอบระบบ	3 ตุลาคม 2565 – 12 ตุลาคม 2565	ทุกคน
นำเสนอ	13 ตุลาคม 2565	ทุกคน

ตารางที่ 1 ตารางความคืบหน้าของโครงการ

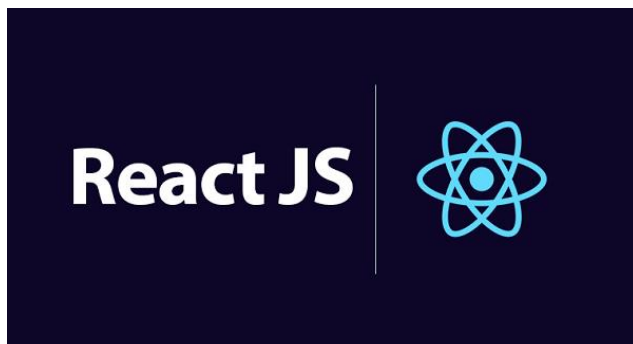
6.3 ตาราง Gantt chart



ตารางที่ 2 ตาราง Gantt Chart

ทฤษฎีและหลักการที่เกี่ยวข้อง

1. React



ภาพที่ 1 โลโก้ React

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

การเตรียมใช้งาน React

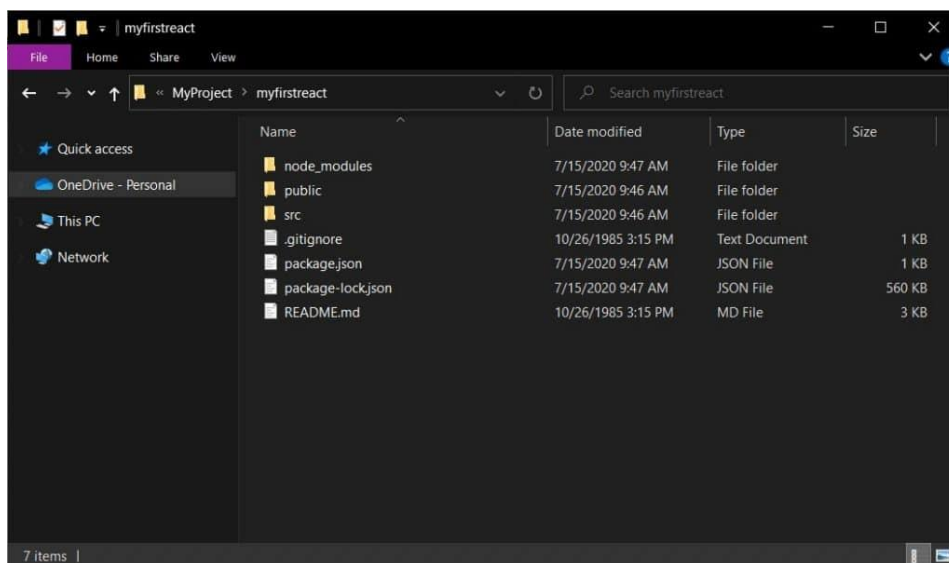
1. ติดตั้ง npm ไว้ในเครื่องให้เรียบร้อย เข้าไปดูว่านโหลดได้ดังนี้
<https://www.npmjs.com/get-npm>
2. สร้างโปรเจกต์เริ่มต้นขึ้นมาจากเครื่องมือที่ชื่อ create-react-app ที่จะสร้างไฟล์พื้นฐานต่างๆที่จำเป็นในการเขียน React ขึ้นมา ให้ทำการติดตั้งด้วยคำสั่ง

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

3. สร้างโปรเจกต์ใหม่ขึ้นมาด้วยคำสั่ง

```
npx create-react-app myreact
```

4. เราจะได้ไฟล์ต่างๆขึ้นมาประมาณนี้

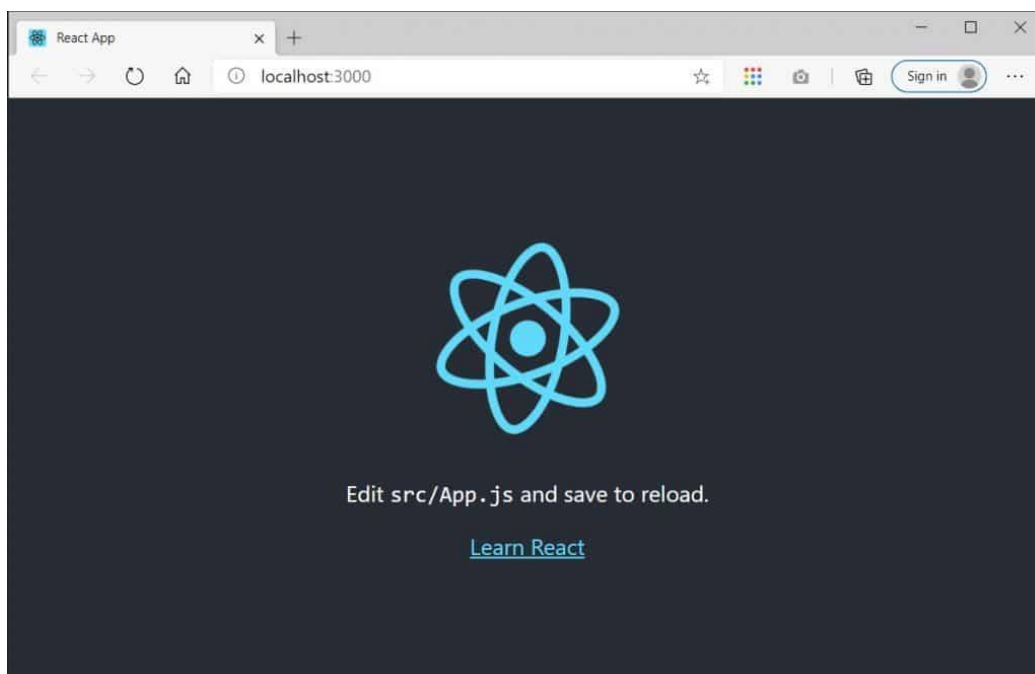


ภาพที่ 2 ภาพโฟลเดอร์หลังจากติดตั้ง React

5. รันด้วยคำสั่ง

```
npm start
```

6. เบราเซอร์จะเปิดขึ้นมาที่ <http://localhost:3000/> ถ้ามีหน้าต่างแบบนี้แสดงว่าพร้อมใช้งานแล้ว



ภาพที่ 3 ภาพหน้าเว็บไซต์หลังจากรันคำสั่ง npm start

2. HTML/CSS

2.1 HTML

HTML ย่อมาจาก HyperText Markup Language เป็นภาษาคอมพิวเตอร์ที่ใช้สร้างหน้าเว็บ (WebPage) ในรูปแบบของไฟล์ HTML (คือไฟล์ที่มีนามสกุลเป็น .htm หรือ .html) ซึ่งมีเว็บเบราว์เซอร์ (WebBrowser) เป็นโปรแกรมที่ใช้แปลงไฟล์ HTML เพื่อแสดงผลในรูปของหน้าเว็บ ไฟล์ HTML เป็นไฟล์รหัสแอสกี (ASCII) ถูกบันทึกในรูปของไฟล์เอกสาร (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>

2.2 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 ทั่วทั้งเอกสาร
3. CSS สามารถกำหนดแยกไว้ต่างหากจาก ไฟล์เอกสาร html และสามารถนำมาใช้ร่วม กับเอกสารหลายไฟล์ได้ การแก้ไขก็แก้เพียง จุดเดียวก็มีผลกับเอกสารทั้งหมด

3. REST

Representational state transfer หรือ REST คือ การสร้าง Webservice ชนิดหนึ่งที่ใช้สื่อสารกันบน Internet ใช้หลักการแบบ stateless คือไม่มี session ซึ่งต่างจาก webservice แบบอื่นเช่น WSDL และ SOAP การทำงานของ RESTful Webservice จะอาศัย URI/URL ของ request เพื่อค้นหาและประมวลผล แล้วตอบกลับไปในรูป XML, HTML, JSON โดย response ที่ตอบกลับจะเป็นการยืนยันผลของคำสั่งที่ส่งมา และสามารถพัฒนาด้วยภาษา programming ได้หลากหลาย คำสั่งก็จะมีตาม HTTP verbs ซึ่งก็คือ

1. GET ทำการดึงข้อมูลภายใน URI ที่กำหนด
2. POST สำหรับสร้างข้อมูล
3. PUT ใช้แก้ไขข้อมูล
4. DELETE สำหรับลบข้อมูล

ประวัติของ REST

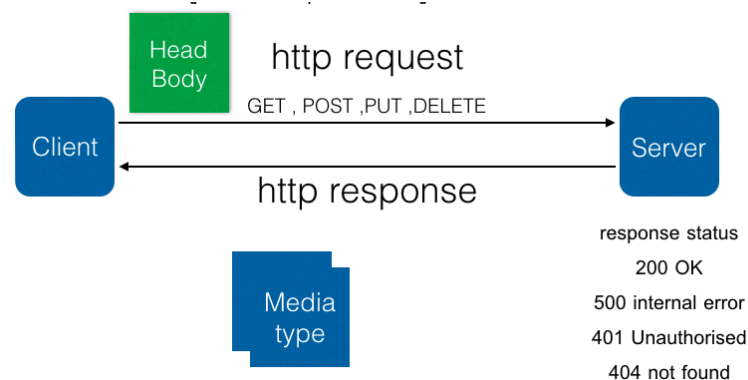
REST ถูกตั้งขึ้นโดย Roy Fielding ในปี 2000 ที่ University of California, Irvine ซึ่งได้ทำการพัฒนาขึ้นมาควบคู่กับ HTTP1.1 และ Uniform Resource Identifiers (URI)

คุณสมบัติของ REST

เป็น API อย่างหนึ่ง ซึ่งทุกๆ system ต่างใช้ resource ซึ่งเป็นได้ทั้ง image, video, web page หรือ ข้อมูลทางธุรกิจ ก็ได้ที่สามารถแสดงบนระบบ computer วัตถุประสงค์เพื่อให้ user สามารถเข้าถึง, ติดตั้ง, ปรับแต่ง, ขยาย resource เหล่านี้ได้ง่าย ซึ่งทาง RESTful ได้ออกแบบมาให้มีคุณสมบัติต่อไปนี้

- แสดงผล
- เก็บข้อมูล
- มี URIs
- Stateless ทำงานโดยไม่ต้องมี session
- เชื่อมต่อระหว่าง web service
- Caching

การทำงานของ REST



ภาพที่ 4 ภาพการทำงานของ Rest

จากรูป ฝั่ง Client จะส่งคำขอ ไปหาปลายทาง ซึ่งใน http โปรโตคอล จะเปรียบเสมือนเอกสารชุดหนึ่ง ที่มีหัวกระดาษ (Header) และเนื้อหาที่อยู่ข้างในตัวเอกสาร (Body) พร้อมประเภทคำสั่ง GET, POST, PUT, DELETE ไปยังเส้นทางที่ระบุถึงปลายทาง เช่น `http://myweb/getdata` ฝั่งปลายทางก็จะตอบรับคำขอ กลับไป พร้อมทั้งสถานะการตอบกลับที่เป็นมาตรฐาน ดังนี้

- 200 OK สถานะตอบไปยังต้นทางได้ปกติ
- 500 internal error มีข้อผิดพลาดที่ปลายทาง
- 401 Unauthorised ไม่มีสิทธิเข้าถึง URIs ดังกล่าว
- 404 not found หา URIs ดังกล่าวไม่เจอ

ตัวอย่าง response ให้รูปแบบต่างๆ

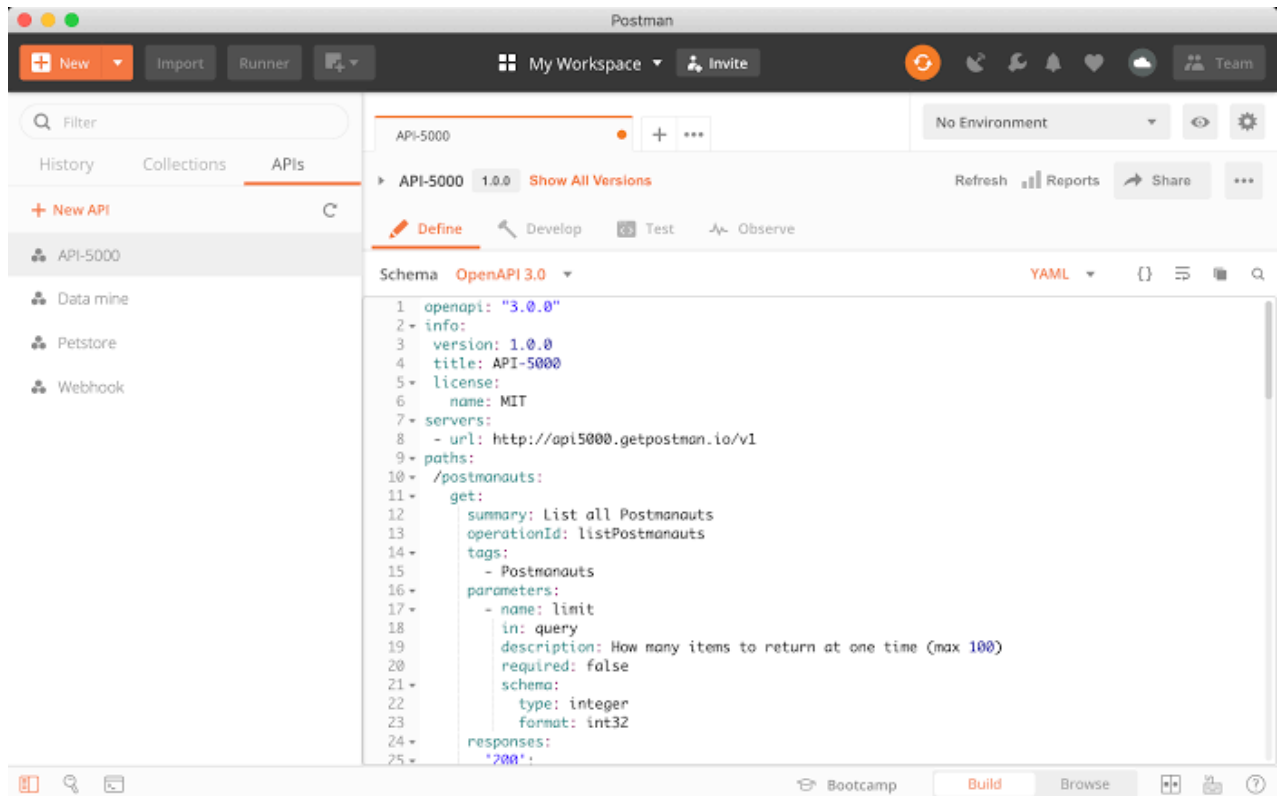
JSON

```
{  
  "ID": "1",  
  "Name": "M Vaqqas",  
  "Email": "m.vaqqas@gmail.com",  
  "Country": "India"  
}
```

JSON

```
<Person>  
<ID>1</ID>  
<Name>M Vaqqas</Name>  
<Email>m.vaqqas@gmail.com</Email>  
<Country>India</Country>  
</Person>
```

ตัวอย่างเครื่องมือที่ใช้ทดสอบ RESTful API



ภาพที่ 5 ภาพการใช้ Postman

4. Node.js

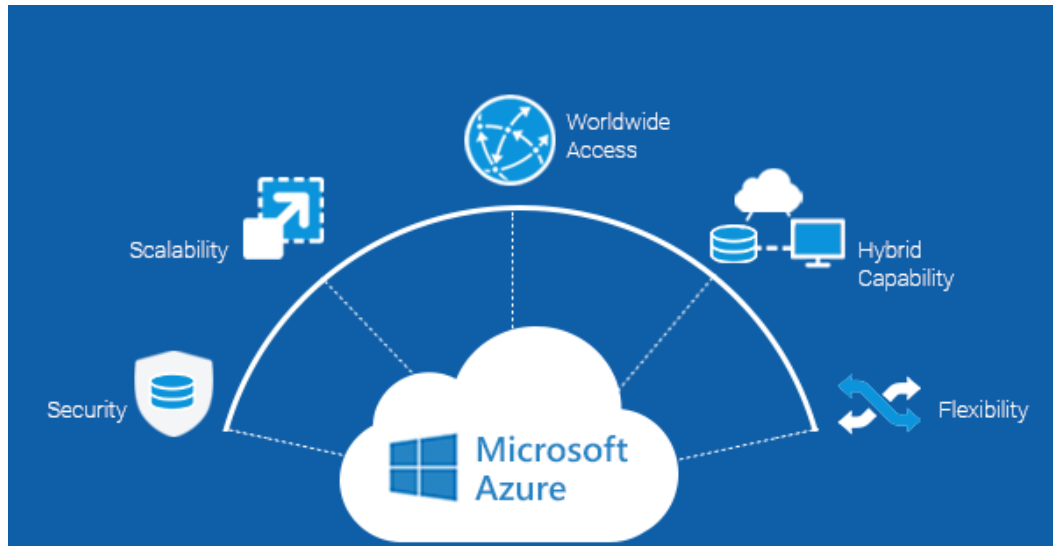


ภาพที่ 6 ภาพโลโก้ Nodejs

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

5. Microsoft Azure



ภาพที่ 7 ภาพ Microsoft Azure

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

รูปแบบการให้บริการของ Microsoft Azure

เป็นการให้บริการในรูปแบบการรับผิดชอบร่วมกัน (Shared Responsibility) ระหว่างลูกค้า และ Microsoft และมีการแบ่งรูปแบบการรับผิดชอบร่วมกันเป็น 3 รูปแบบคือ

5.1 IaaS (Infrastructure as a Service)

การทำงานที่ Microsoft รับผิดชอบการให้บริการโครงสร้างพื้นฐานต่างๆ ที่จำเป็นในศูนย์ข้อมูล (Datacenter) เช่น เครื่องแม่ข่าย, ระบบเครือข่าย, ตลอดจนพื้นที่เก็บข้อมูลต่างๆ ส่วนบริการอื่นๆ อันต่อเนื่อง ไปเช่น 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

5.2 PaaS (Platform as a Service)

รูปแบบการให้บริการแบบ PaaS นี้ทาง Microsoft จะมีการดูแลระบบปฏิบัติการ และเครื่องมือต่างๆ ในการทำงานของแอปพลิเคชันเพิ่มให้จาก 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 โดยไม่ต้องกังวลในในส่วนของการตั้งค่าต่างๆ รองรับการสำรองข้อมูล และมีการติดตั้ง Update ให้โดยอัตโนมัติ

- Azure Backup

การสำรองข้อมูลอันมีค่าขององค์กรไปเก็บไว้ในพื้นที่เก็บข้อมูลที่เชื่อถือได้ของ Microsoft Azure ทำให้มั่นใจได้ว่าข้อมูลที่สำรองไว้จะสามารถเรียกคืนได้ทันทีที่ต้องการ โดยมีระยะเวลาการเก็บข้อมูลสูงสุด 10 ปี

5.3 SaaS (Software as a Service)

เป็นการให้บริการโดย Microsoft รับผิดชอบการดูแลระบบต่างๆ ให้ทั้งหมด ทำให้การบริการจะอยู่ในรูปแบบสำเร็จรูป เช่น Microsoft 365 เป็นต้น

6. MySQL



ภาพที่ 8 ภาพโลโก้ MySQL

MySQL คือ โปรแกรมระบบจัดการฐานข้อมูล ที่พัฒนาโดยบริษัท MySQL AB มีหน้าที่เก็บข้อมูลอย่างเป็นระบบ รองรับคำสั่ง SQL เป็นเครื่องมือสำหรับเก็บข้อมูล ที่ต้องใช้ร่วมกับเครื่องมือหรือโปรแกรมอื่นอย่างบูรณาการ เพื่อให้ได้ระบบงานที่รองรับ ความต้องการของผู้ใช้ เช่นทำงานร่วมกับเครื่องบริการเว็บ (Web Server) เพื่อให้บริการแก่ภาษาสคริปต์ที่ทำงานฝั่งเครื่องบริการ (Server-Side Script) เช่น ภาษา php ภาษา asp.net หรือภาษาเจเอสพี เป็นต้น หรือทำงานร่วมกับโปรแกรมประยุกต์ (Application Program) เช่น ภาษาวิซวลเบสิกดอทเน็ต ภาษาจาวา หรือภาษาซีชาร์ป เป็นต้น โปรแกรมถูกออกแบบให้สามารถทำงานได้บนระบบปฏิบัติการที่หลากหลาย และเป็นระบบฐานข้อมูลโอเพนทเซอร์ท (Open Source) ที่ถูกนำไปใช้งานมากที่สุด MySQL จัดเป็นระบบจัดการฐานข้อมูลเชิงสัมพันธ์ (RDBMS : Relational Database Management System) ซึ่งเป็นที่นิยมใช้กันมากในปัจจุบัน

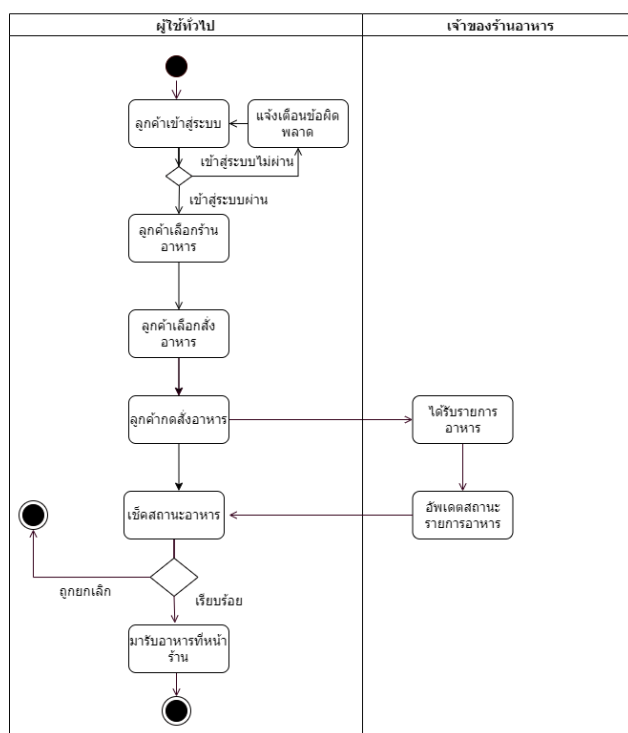
MySQL เป็นระบบจัดการฐานข้อมูลโดยใช้ภาษา SQL. แม้ว่า MySQL เป็นซอฟต์แวร์โอเพนซอร์ส แต่แตกต่างจากซอฟต์แวร์โอเพนซอร์สทั่วไป โดยมีการพัฒนาภายใต้บริษัท MySQL AB ในประเทศสวีเดน โดยจัดการ MySQL ทั้งในแบบที่ให้ใช้ฟรี และแบบที่ใช้ในเชิงธุรกิจ MySQL สร้างขึ้นโดยชาวสวีเดน 2 คน และชาวฟินแลนด์ ชื่อ David Axmark,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

ภาพรวมและการออกแบบระบบ

ภาพรวมของระบบ

1. Activity diagrams

ลำดับการทำงานการสั่งอาหารบนเว็บไซต์สั่งอาหารโรงแรม มีหลักการทำงานดังนี้



ภาพที่ 8 ภาพ Activity diagrams ของระบบสั่งอาหาร

จาก Activity diagrams เริ่มต้นจาก ผู้ใช้งานทั่วไปหรือลูกค้าเริ่มเข้าสู่ระบบ เมื่อกดเข้าสู่ระบบ ระบบจะทำการเช็ครหัสผ่าน ถ้าเข้าสู่ระบบผ่านลูกค้าจะสามารถเข้าสู่หน้าเว็บไซต์แบบมีผู้ใช้งานได้ แต่ถ้าเข้าสู่ระบบไม่ผ่าน ระบบจะทำการแจ้งเตือนข้อผิดพลาด และให้ลูกค้าทำการเข้าสู่ระบบหรือสมัครสมาชิกอีกครั้ง

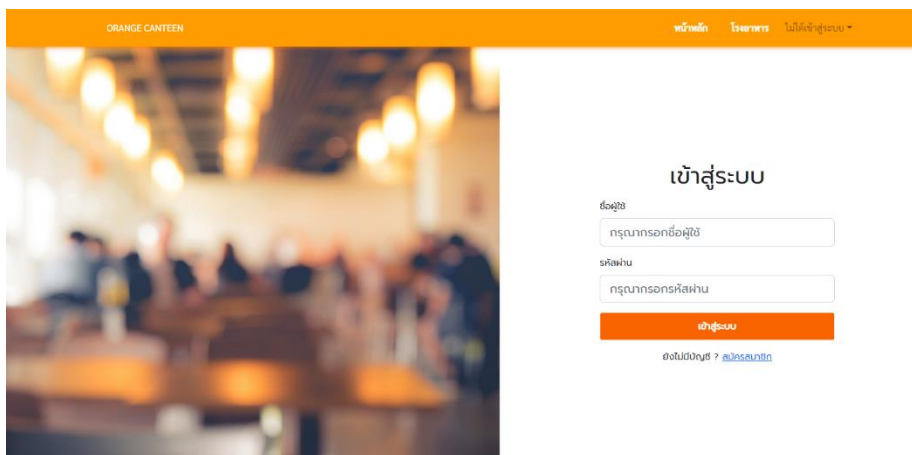
หลังจากเข้าสู่ระบบผ่าน ลูกค้าสามารถเลือกร้านอาหาร และกดเลือกสั่งเมนูอาหารตามร้านที่เลือกได้ โดยเลือกสั่งอาหารได้ครั้งละ 1 ร้านเท่านั้น เมื่อทำการกดยืนยันการสั่งอาหารเสร็จ ระบบจะส่งข้อมูลไปที่ฐานข้อมูลที่รวบรวมรายการคำสั่งซื้ออาหาร และเจ้าของร้านอาหารแต่ละร้านสามารถเช็ค และทำการอนุมัติรายการอาหารได้ เมื่อเจ้าของร้านทำการกดยืนยันสถานะเป็น “เรียบร้อย” แสดงว่าเมนูรายการนั้นทำเสร็จ และพร้อมที่จะให้ลูกค้ามารับหน้าร้านอาหารเรียบร้อยแล้ว แต่ถ้าเจ้าของร้านทำการกดยืนยันสถานะเป็น “ถูก

ยกเลิก” และแจ้งเหตุผลในการถูกยกเลิก ลูกค้าจะไม่สามารถมารับอาหารดังกล่าวได้ อาจจะต้องทำการจัดส่งซื้อเมนูอื่น หรือเลือกสั่งซื้ออาหารร้านอื่นใหม่อีกครั้ง

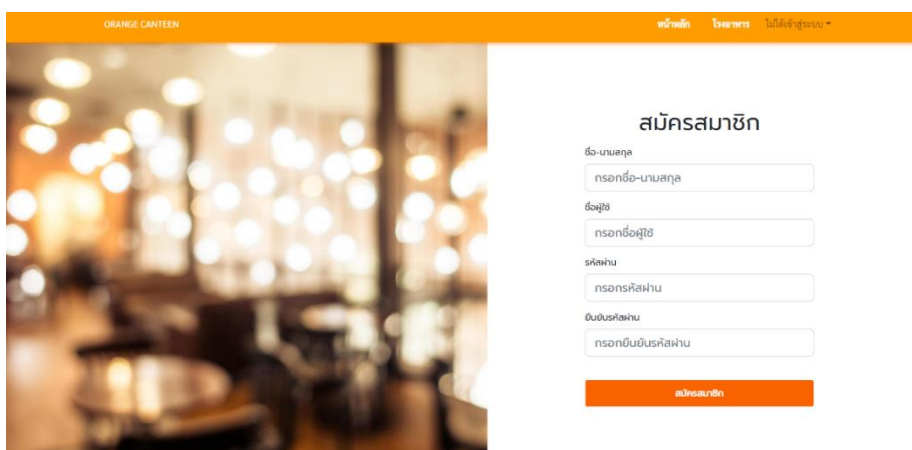
ส่วนการทำงานหลักของระบบ/ชิ้นงาน

1. สำหรับผู้ใช้ทั่วไป

1.1 ระบบสมัครสมาชิก และเข้าสู่ระบบ



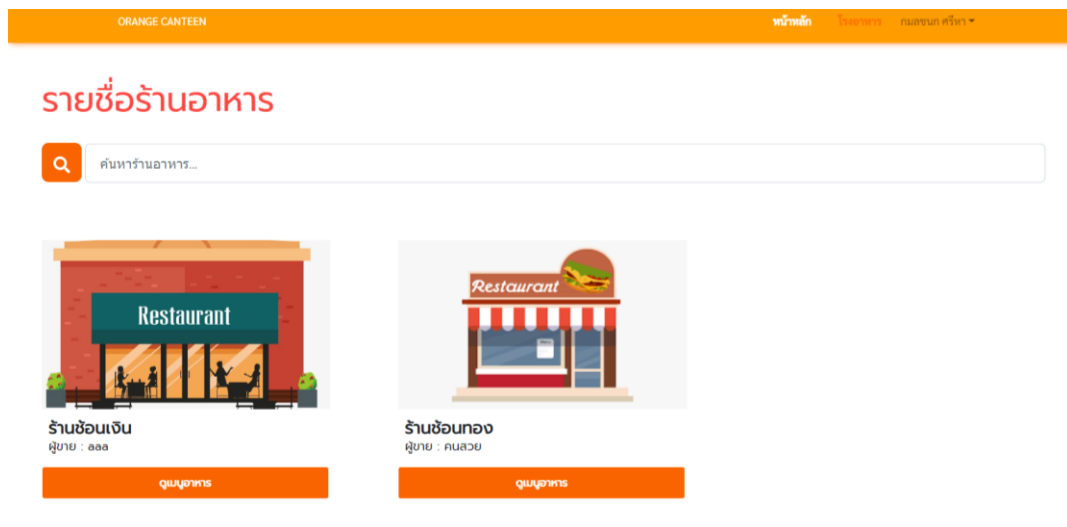
ภาพที่ 9 ภาพหน้าเว็บไซต์เข้าสู่ระบบ



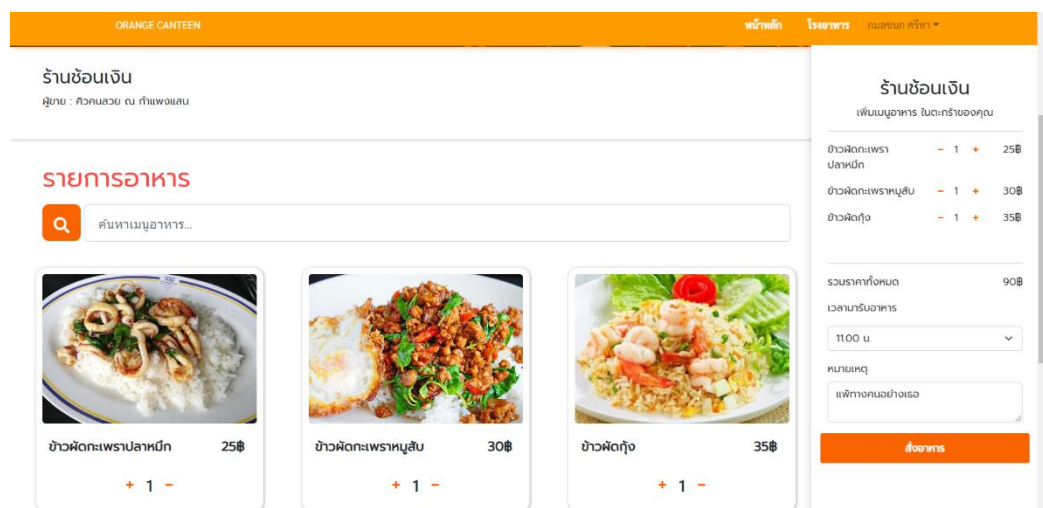
ภาพที่ 10 ภาพหน้าเว็บไซต์สมัครสมาชิก

ลูกค้าสามารถสมัครสมาชิก และเข้าสู่ระบบเพื่อเข้าสู่หน้าเว็บไซต์สั่งอาหารได้ โดยข้อมูลการสมัครสมาชิกจะถูกส่งไปในฐานข้อมูล และมีการเช็ครหัสผ่านก่อนเข้าสู่ระบบ

1.2 ระบบสั่งอาหาร



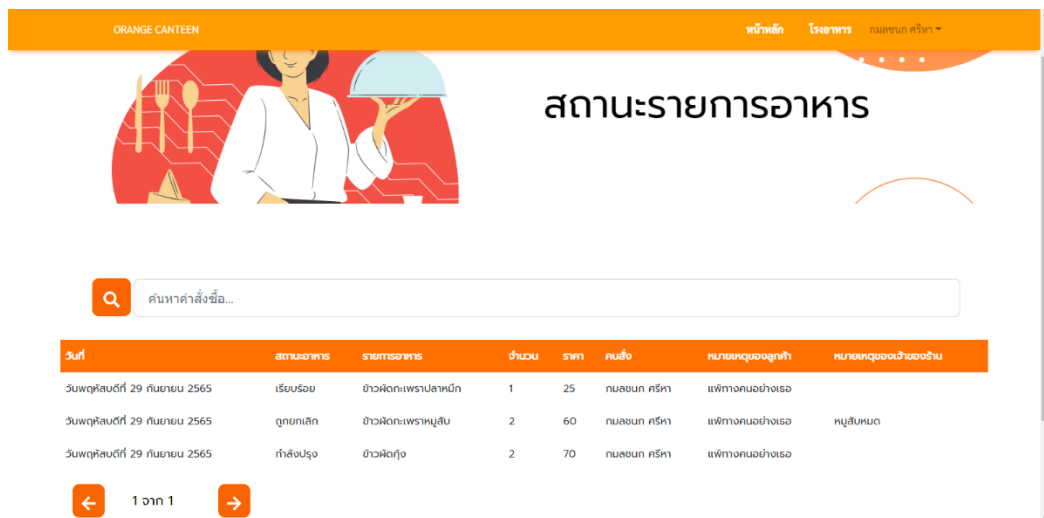
ภาพที่ 11 ภาพหน้าเว็บไซต์รวบรวมร้านอาหาร



ภาพที่ 12 ภาพหน้าเว็บไซต์รวบรวมเมนูอาหารตามร้านอาหาร

ลูกค้าสามารถเลือกร้านอาหารที่ต้องการที่จะเลือกสั่งเมนูอาหาร และสามารถเลือกเมนูอาหารจำนวน เวลาที่มารับอาหาร และหมายเหตุสำหรับใส่ในกรณีที่อยากเพิ่มเติมหรือข้อมูลเกี่ยวกับคนแพ้อาหารได้ เมื่อคำสั่งอาหารรายการคำสั่งอาหารดังกล่าวจะถูกส่งไปพื้นฐานข้อมูล

1.3 ระบบการตรวจสอบสถานะรายการคำสั่งอาหาร

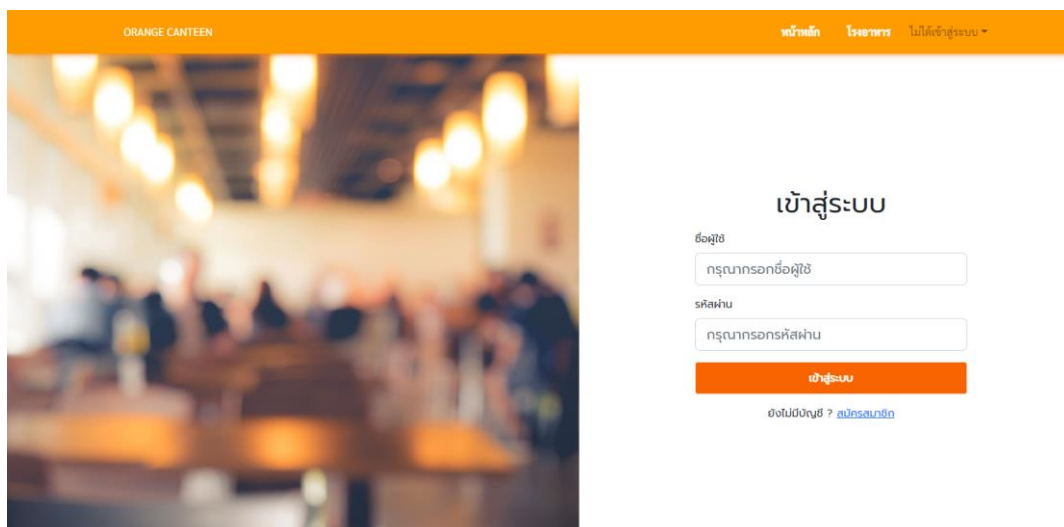


ภาพที่ 13 ภาพหน้าเว็บไซต์รวบรวมสถานะรายการคำสั่งอาหาร

ลูกค้าสามารถค้นหารายการอาหารที่ต้องการทราบสถานะ และสามารถเช็คได้ว่าสถานะรายการอาหารปัจจุบันเพื่อที่จะเตรียมความพร้อมในการรอรับอาหารจากร้าน

2. สำหรับเจ้าของร้านอาหาร

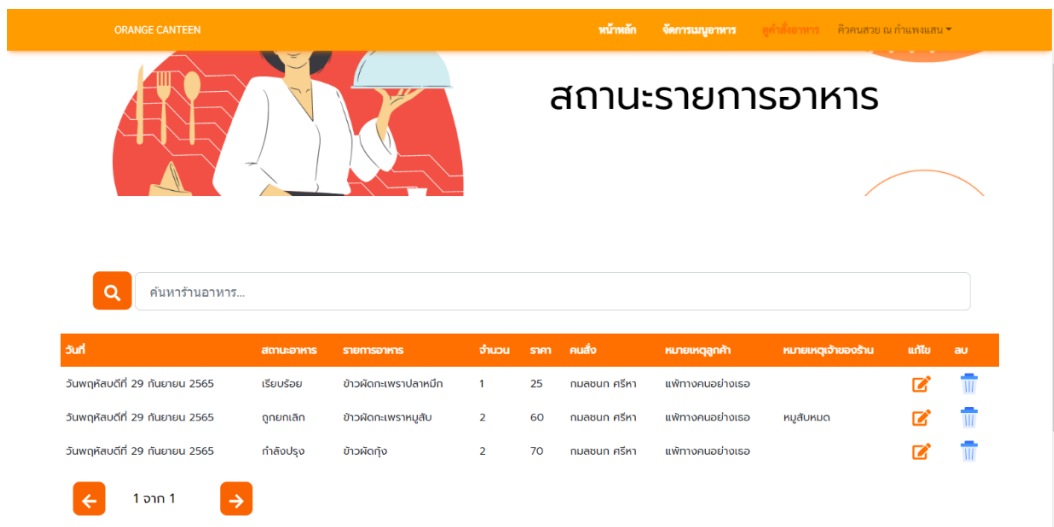
2.1 ระบบเข้าสู่ระบบ สำหรับเจ้าของร้านอาหาร



ภาพที่ 14 ภาพหน้าเว็บไซต์เข้าสู่ระบบ

เจ้าของร้านอาหาร สามารถเข้าสู่ระบบเพื่อเข้าสู่หน้าเว็บไซต์สั่งอาหารได้ โดยมีการเช็ครหัสผ่านก่อนเข้าสู่ระบบ และไม่มีระบบสมัครสมาชิกสำหรับเจ้าของร้านอาหาร

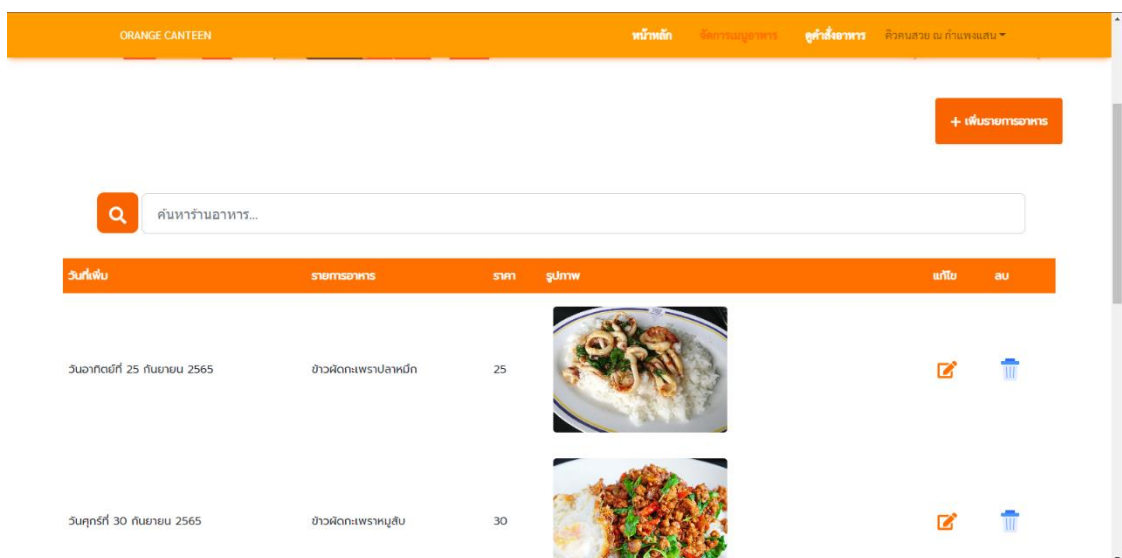
2.2 ระบบการตรวจสอบสถานะรายการคำสั่งอาหารทั้งหมดในร้าน สำหรับเจ้าของร้านอาหาร



ภาพที่ 15 ภาพหน้าเว็บไซต์ตรวจสอบสถานะรายการคำสั่งอาหารทั้งหมดในร้าน

เจ้าของร้านอาหารสามารถตรวจสอบและค้นหาคำสั่งซื้ออาหารของลูกค้า อัปเดตสถานะรายการอาหาร และลบรายการอาหารดังกล่าวได้ เมื่อบันทึกข้อมูล ข้อมูลจะถูกส่งไปในระบบ

2.3 ระบบการจัดการเมนูอาหารตามร้าน สำหรับเจ้าของร้านอาหาร



ภาพที่ 16 ภาพหน้าเว็บไซต์จัดการเมนูอาหาร

เจ้าของร้านอาหารสามารถเพิ่ม แก้ไข และเมนูอาหารภายในร้านบนเว็บไซต์ได้ เมื่อบันทึกข้อมูล ข้อมูลจะถูกส่งไปในระบบ

2.4 ระบบแก้ไขข้อมูลร้านอาหาร สำหรับเจ้าของร้านอาหาร

ORANGE CANTEEN

หน้าหลัก จัดการร้านอาหาร ดูค่าเชิงอาหาร ครัวคนสวย ณ กำแพงแสน

แก้ไขร้านอาหาร

ชื่อร้าน
ร้านชื่อนี้

สถานะร้าน
เปิด

ชื่อเจ้าของร้าน
ครัวคนสวย ณ กำแพงแสน

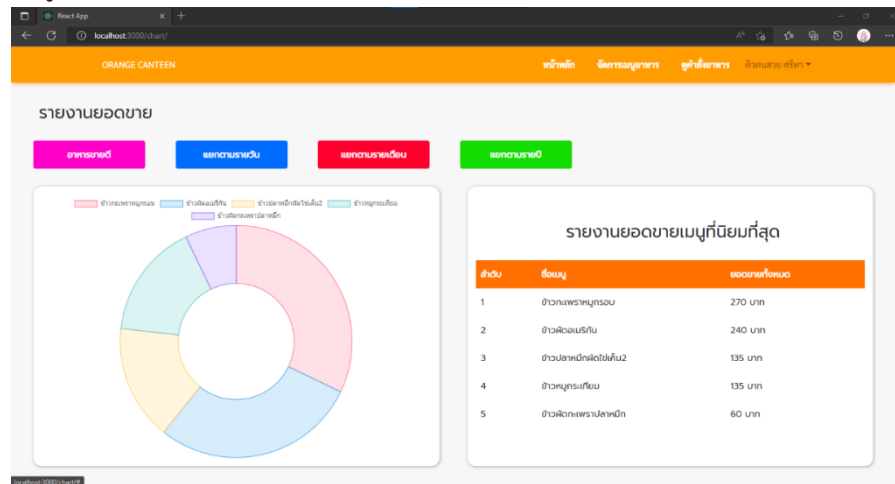
อัปโหลดรูปภาพ
Choose File No file chosen

รูปเดิม

ภาพที่ 17 ภาพหน้าเว็บไซต์แก้ไขร้านอาหาร

เจ้าของร้านอาหารแก้ไขข้อมูลร้านอาหารบนเว็บไซต์ได้ เมื่อบันทึกข้อมูล ข้อมูลจะถูกส่งไปในระบบ

2.5 ระบบดูยอดขายสถิติภายในร้าน สำหรับเจ้าของร้านอาหาร



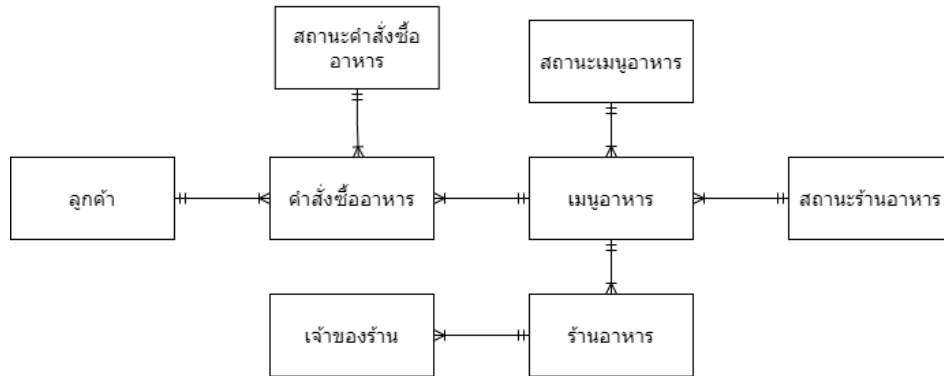
ภาพหน้าเว็บไซต์รายงาน

เจ้าของร้านอาหารสามารถตรวจสอบรายงานสถิติยอดขายภายในร้านอาหารได้ มี 4 รูปแบบดังนี้

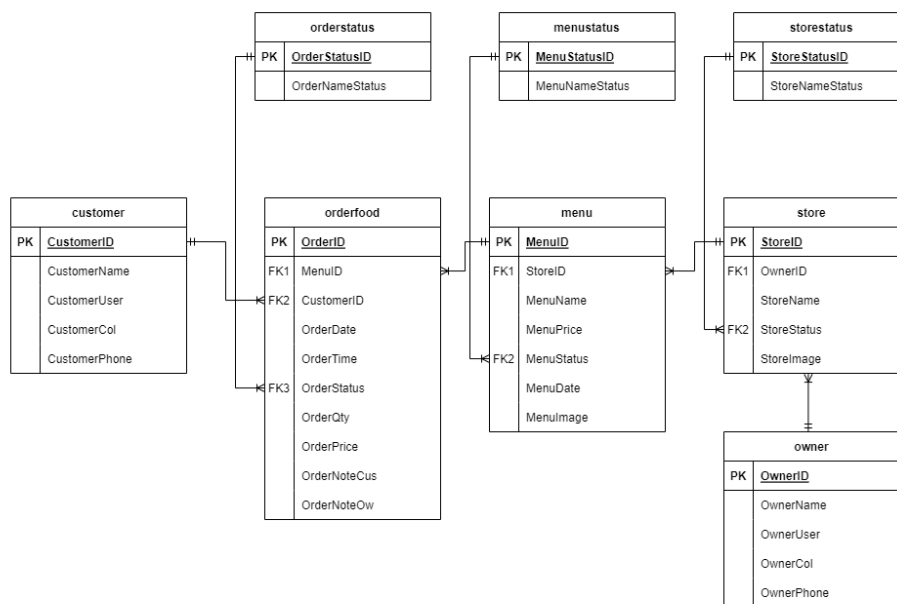
1. สถิติยอดขายเมนูขายดีที่สุด
2. สถิติยอดขายตามรายวัน
3. สถิติยอดขายตามรายเดือน
4. สถิติยอดขายตามรายปี

การออกแบบระบบ

1. การออกแบบฐานข้อมูล



ภาพที่ 18 ภาพการออกแบบ E-R Diagram



ภาพที่ 19 ภาพการนำ E-R Diagram มาแปลงเป็นโครงสร้างฐานข้อมูล

2. การออกแบบเว็บไซต์

2.1 ผู้ใช้งานทั่วไป

2.1.1 หน้าเข้าสู่ระบบและสมัครสมาชิก

สมัครสมาชิก' (Don't have an account? Sign up)." data-bbox="182 179 893 464"/>

ORANGE CANTEEN หน้าหลัก โรงอาหาร ไม่ได้เข้าสู่ระบบ ▼

เข้าสู่ระบบ

ชื่อผู้ใช้
กรณารอกชื่อผู้ใช้

รหัสผ่าน
กรณารอกรหัสผ่าน

เข้าสู่ระบบ

ยังไม่มีบัญชี? [สมัครสมาชิก](#)

ภาพที่ 20 ภาพออกแบบหน้าเว็บไซต์เข้าสู่ระบบ



ORANGE CANTEEN หน้าหลัก โรงอาหาร ไม่ได้เข้าสู่ระบบ ▼

สมัครสมาชิก

ชื่อ-นามสกุล
กรอกชื่อ-นามสกุล

ชื่อผู้ใช้
กรอกชื่อผู้ใช้

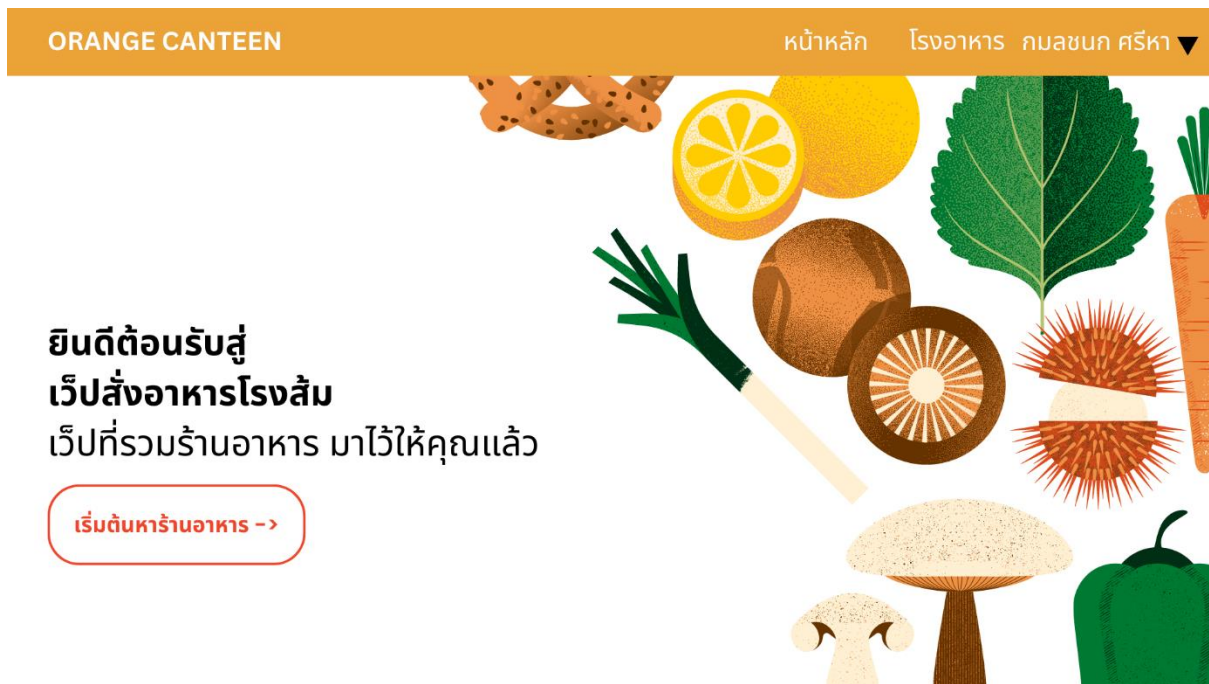
รหัสผ่าน
กรอกรหัสผ่าน

ยืนยันรหัสผ่าน
กรอกยืนยันรหัสผ่าน

สมัครสมาชิก

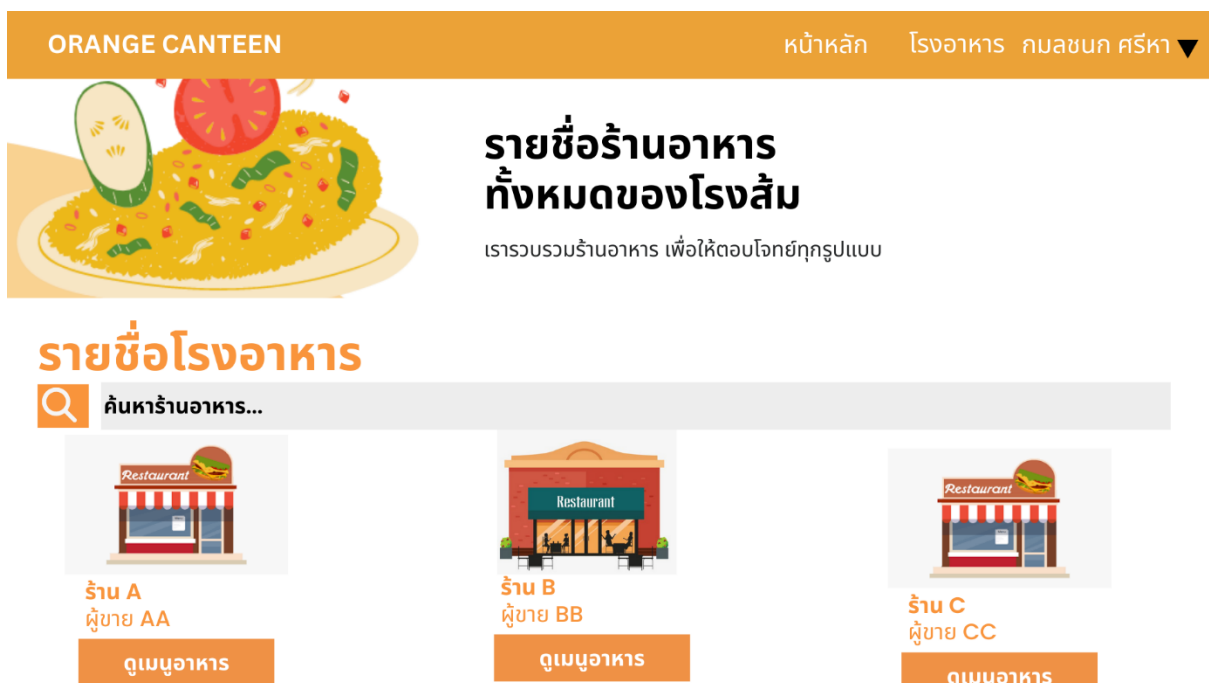
ภาพที่ 21 ภาพออกแบบหน้าเว็บไซต์สมัครสมาชิก

2.1.2 หน้าหลักของเว็บไซต์



ภาพที่ 22 ภาพออกแบบหน้าหลักเว็บไซต์

2.1.3 หน้าเว็บไซต์รวบรวมร้านอาหาร



ภาพที่ 23 ภาพออกแบบภาพหน้าเว็บไซต์รวบรวมร้านอาหาร

2.1.4 หน้าเว็บไซต์รวบรวมเมนูอาหารตามร้านอาหาร

ORANGE CANTEEN

หน้าหลัก โรงอาหาร กมลชนก ศรีหา ▼

ร้านช้อนเงิน

ผู้ชาย AA

รายการอาหาร

ค้นหารายการอาหาร

ข้าวผัดไก่ผัดไข่เค็ม 35฿

สั่งอาหาร

ข้าวหมูกระเทียม 35฿

+ 1 -

ข้าวผัดกะเพราปลาหมึก 35฿

+ 2 -

ร้านช้อนเงิน

เพิ่มเมนูอาหาร ในตะกร้าของคุณ

ข้าวผัดกะเพราปลาหมึก + 2 70฿

ข้าวหมูกระเทียม + 1 35฿

ราคารวมทั้งหมด 105฿

เวลารับประทานอาหาร

09.00 น. ▼

หมายเหตุ

แพ็กคณอย่างเธอ

สั่งอาหาร

ภาพที่ 24 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	แพ็กคณอย่างเธอ	
9 ตุลาคม 2565	ถูกยกเลิก	ข้าวหมูกระเทียม	1	35฿	กมลชนก ศรีหา	แพ็กคณอย่างเธอ	หยุด

ภาพที่ 25 ภาพออกแบบภาพหน้าเว็บไซต์รวบรวมสถานะรายการคำสั่งอาหาร

2.2 เจ้าของร้าน

2.2.1 หน้าเข้าสู่ระบบและสมัครสมาชิก

สมัครสมาชิก' (Don't have an account? [Sign up](#)). On the left side of the login form is a blurred image of a canteen interior with warm lighting."/>

ภาพที่ 26 ภาพออกแบบหน้าเว็บไซต์เข้าสู่ระบบ

2.2.2 หน้าเว็บไซต์รวบรวมสถานะรายการคำสั่งอาหารทั้งหมด

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 27 ภาพออกแบบภาพหน้าเว็บไซต์รวบรวมสถานะรายการคำสั่งอาหารทั้งหมด

ORANGE CANTEEN

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

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 28 ภาพออกแบบหน้าเว็บไซต์อัปเดตสถานะรายการอาหาร

2.2.3 หน้าเว็บไซต์รวบรวมเมนูภายในร้านอาหารทั้งหมด

ORANGE CANTEEN

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



รายการอาหาร

+ เพิ่มรายการอาหาร

ค้นหารายการอาหาร...

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

ภาพที่ 29 ภาพออกแบบภาพหน้าเว็บไซต์รายการอาหารในร้าน

ORANGE CANTEEN

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

เพิ่มเมนูอาหาร

วันที่เพิ่ม

วันศุกร์ที่ 30 กันยายน 2565

ราคา

35

ชื่อเมนูอาหาร

ข้าวผัดกะเพราปลาหมึก

อัปโหลดรูปภาพ

Choose File

รูปภาพ.png

เพิ่มเมนูอาหาร

ภาพที่ 30 ภาพออกแบบภาพหน้าเว็บไซต์เพิ่มเมนูอาหาร

ORANGE CANTEEN

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

แก้ไขเมนูอาหาร

วันที่เพิ่ม

วันศุกร์ที่ 30 กันยายน 2565

ราคา

35

สถานะเมนูอาหาร

เปิด

ชื่อเมนูอาหาร


ข้าวผัดกะเพราปลาหมึก

อัปโหลดรูปภาพ

Choose File

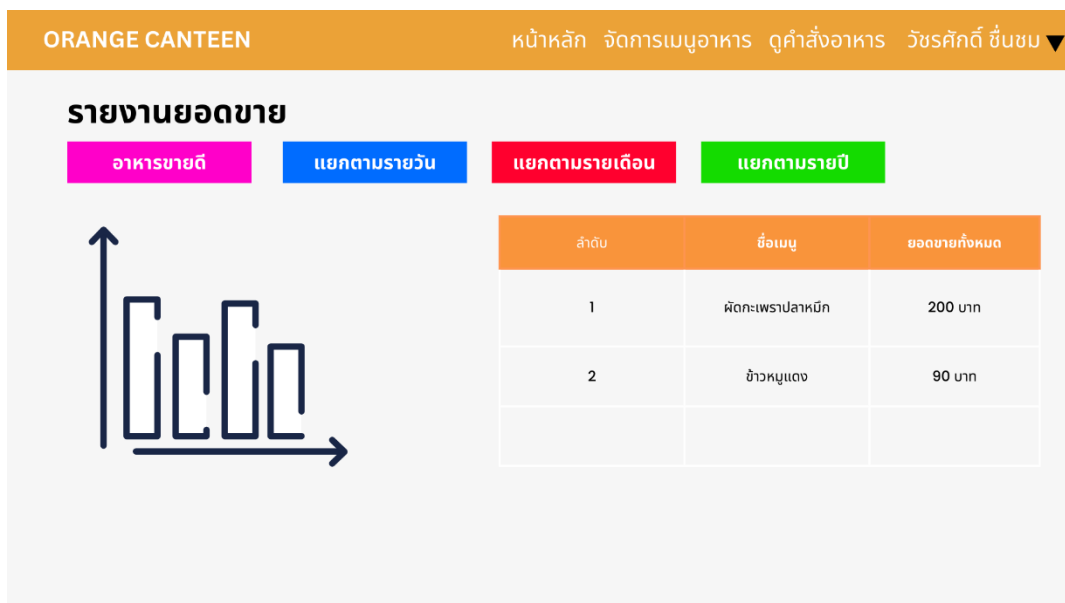
รูปภาพ.png

รูปเดิม

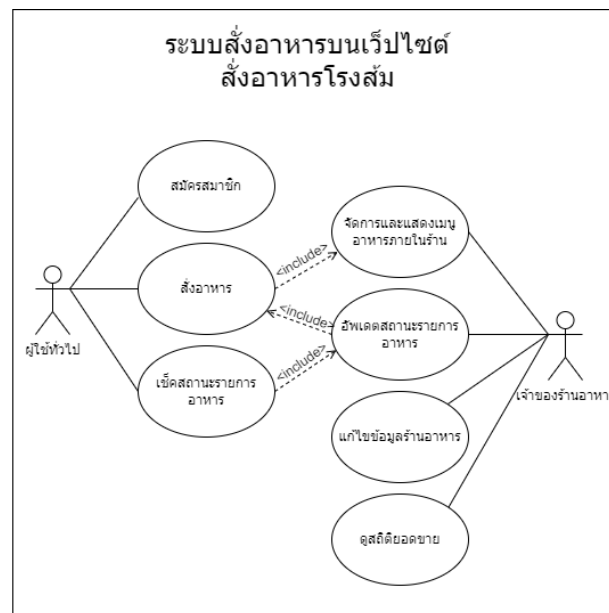


แก้ไขเมนูอาหาร

ภาพที่ 31 ภาพออกแบบภาพหน้าเว็บไซต์แก้ไขร้านอาหาร



3. Use Case Diagram



ภาพที่ 32 ภาพ Use Case Diagram ของระบบสั่งอาหารบนเว็บไซต์สั่งอาหารโรงสั้ม

Use Case Description

Use case	สมัครสมาชิก
Description	ลูกค้าต้องการสมัครสมาชิกเพื่อใช้งานเว็บไซต์สั่งอาหารโรงสั้ม
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 3 ตารางอธิบาย Use Case Description “สมัครสมาชิก”

Use case	สั่งอาหาร
Description	ลูกค้าต้องการสั่งอาหารบนเว็บไซต์สั่งอาหารโรงแรม
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 4 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 5 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 6 ตารางอธิบาย Use Case Description “จัดและแสดงเมนูอาหารภายในร้าน”

Use case	อัปเดตสถานะรายการอาหาร
Description	เจ้าของร้านต้องการอัปเดตสถานะคำสั่งซื้ออาหารเพื่อให้ลูกค้าทราบ
Actors	เจ้าของร้าน,ลูกค้า
Pre-condition	เจ้าของร้านกดอัปเดตสถานะตามรายการคำสั่งซื้ออาหารของลูกค้า
Post-condition	ข้อมูลอัปเดตสถานะถูกบันทึกไปที่ฐานข้อมูล
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 7 ตารางอธิบาย Use Case Description “อัปเดตสถานะรายการอาหาร”

Use case	แก้ไขข้อมูลร้านอาหาร
Description	เจ้าของร้านต้องการปรับเปลี่ยนข้อมูลร้านอาหาร เพื่อปรับเปลี่ยนการแสดงผลบนเว็บไซต์
Actors	เจ้าของร้าน
Pre-condition	เจ้าของร้านกรอกข้อมูลแก้ไขร้านอาหาร
Post-condition	ข้อมูลแก้ไขร้านอาหารถูกบันทึกไปในฐานข้อมูล
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	เจ้าของรถกดดูที่ปุ่มรายงาน
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”

เอกสารอ้างอิง

แนะนำ NodeJS คืออะไร (Online). https://www.mindphp.com/developer/dev-node-js/8542-1-step-website-nodejs.html?bb_limitstart=56 , 29 กันยายน 2565

รู้จักกับ Microsoft Azure (Online). <https://business.ais.co.th/news-activities/azure-geographies.html>, 29 กันยายน 2565

พิชิต วิจิตรบุญยรักษ์. 2554. **HTML: ภาษาเขียนเว็บ (Online).** https://www.bu.ac.th/knowledgecenter/executive_journal/july_sep_11/pdf/aw32.pdf, 29 กันยายน 2565

Css คืออะไร หลายคนคงสงสัย?? (Online). <https://shorturl.asia/cDb9q>, 29 กันยายน 2565

Admin ITGenius. 2557. **(MySQL) คืออะไร(Online).** <https://shorturl.asia/Oqly7>, 30 กันยายน 2565

saixiii. 2560. **RESTful คืออะไร REST คือ การสื่อสารแลกเปลี่ยนข้อมูลผ่าน webservice (Online).** <https://shorturl.asia/AmjPH>, 29 กันยายน 2565