ระบบแนะนำร้านอาหารอัตโนมัติบนสมาร์ทโฟนโดยใช้ข้อมูลเชิงตำแหน่งและรายการอาหาร

Restaurant Recommendation System on Smart Phone using Location Based Service and Menu

ปาลิตา แสงศิริ

Palita Sangsiri

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญา
วิทยาศาสตรมหาบัณฑิต สาขาวิชาการจัดการเทคโนโลยีสารสนเทศ
มหาวิทยาลัยสงขลานครินทร์
A Thesis Submitted in Partial Fulfillment of the Requirements for the
Degree of Master of Science in Management of Information Technology
Prince of Songkhla University
2561

ชื่อวิทยานิพนธ์ ระบบแนะนำร้านอาหารอัตโนมัติบนสมาร์ทโฟนโดยใช้ข้อมูลเชิงตำแหน่งและรายการอาหาร
ผู้เขียน ปาลิตา แสงศิริ
สาขาวิชา การจัดการเทคโนโลยีสารสนเทศ

อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก

....
(รองศาสตราจารย์ ดร.ทวีศักดิ์ เรืองพีระกุล)

อาจารย์ที่ปรึกษาวิทยานิพนธ์ร่วม

....
(ผู้ช่วยศาสตราจารย์ ดร.นิคม สุวรรณวร)

คณะกรรมการสอบ

....ประธานกรรมการ
(ผู้ช่วยศาสตราจารย์ ดร. วัชรวลี ตั้งคุปตานนท์)

....กรรมการ
(รองศาสตราจารย์ ดร.วัฒนพงศ์ เกิดทองมี)

....กรรมการ
(รองศาสตราจารย์ ดร.ทวีศักดิ์ เรืองพีระกุล)

....กรรมการ
(ผู้ช่วยศาสตราจารย์ ดร.นิคม สุวรรณวร)

บัณฑิตวิทยาลัย มหาวิทยาลัยสงขลานครินทร์ อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษา ตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต สาขาวิชาการจัดการเทคโนโลยีสารสนเทศ

....
(ศาสตราจารย์ ดร.ดำรงศักดิ์ ฟ้ารุ่งสาง)
คณบดีบัณฑิตวิทยาลัย

ขอรับรองว่า ผลงานวิจัยนี้มาจากการศึกษาวิจัยของนักศึกษาเอง และได้แสดงความขอบคุณบุคคลที่มีส่วนช่วยเหลือแล้ว

ลงชื่อ..

(รองศาสตราจารย์ ดร.ทวีศักดิ์ เรืองพีระกุล)

อาจารย์ที่ปรึกษาวิทยานิพนธ์

ลงชื่อ..

(นางสาวปาลิตา แสงศิริ)

นักศึกษา

ข้าพเจ้าขอรับรองว่า ผลงานวิจัยนี้ไม่เคยเป็นส่วนหนึ่งในการอนุมัติปริญญาในระดับใดมาก่อน และไม่ได้ถูกใช้ในการยื่นขออนุมัติปริญญาในขณะนี้

ลงชื่อ..

(นางสาวปาลิตา แสงศิริ)

นักศึกษา

ชื่อวิทยานิพนธ์ ระบบแนะนำร้านอาหารอัตโนมัติบนสมาร์ทโฟนโดยใช้ข้อมูลเชิงตำแหน่งและรายการอาหาร

ผู้เขียน นางสาวปาลิตา แสงศิริ

สาขาวิชา การจัดการเทคโนโลยีสารสนเทศ

ปีการศึกษา 2560

บทคัดย่อ

ปัจจุบันผู้บริโภคให้ความใส่ใจสุขภาพมากขึ้น นอกจากการออกกำลังกายแล้วการเลือกบริโภคอาหารที่มีประโยชน์ก็เป็นสิ่งสำคัญ ดังนั้นผู้บริโภคจึงให้ความสำคัญกับการเลือกรับประทานอาหาร โดยพิจารณาถึงส่วนประกอบและรสชาติของรายการอาหารมากขึ้น การเลือกรายการอาหารที่ต้องการรับประทานโดยพิจารณาถึงส่วนประกอบและรสชาตินั้นยังมีความยุ่งยาก การค้นหาร้านอาหารที่มีรายการอาหารที่ตรงกับความต้องการยังไม่มีความสะดวกรวดเร็ว อีกทั้งยังไม่สามารถเข้าถึงร้านอาหารที่อยู่บริเวณใกล้เคียงได้อย่างทั่วถึง

เทคโนโลยีที่นำมาใช้ในปัจจุบัน ตัวอย่าง เช่น เว็บแอปพลิเคชัน และแอปพลิเคชันบนสมาร์ทโฟน โดยส่วนใหญ่รองรับการค้นหาร้านอาหารโดยใช้คำค้นต่างๆ เช่น ข้อมูลตำแหน่งที่ตั้ง ประเภทร้าน ชื่อร้าน ระยะทาง เป็นต้น ระบบดังกล่าวยังไม่สามารถแนะนำรายการอาหารจากส่วนประกอบและรสชาติได้ และไม่สามารถค้นหาร้านอาหารจากรายการอาหารที่ต้องการได้ ดังนั้นระบบที่มีในปัจจุบันยังไม่สามารถแก้ปัญหาที่กล่าวมาข้างต้นได้ ผู้วิจัยจึงมีแนวคิดในการพัฒนาระบบแอปพลิเคชันบนสมาร์ทโฟนเพื่อแก้ปัญหาดังกล่าว ได้แก่

ระบบแนะนำร้านอาหารอัตโนมัติบนสมาร์ทโฟนโดยใช้ข้อมูลเชิงตำแหน่งและรายการอาหาร ประกอบด้วย 2 ส่วนหลักคือ ส่วนการกรองข้อมูลสำหรับแนะนำรายการอาหาร และส่วนแอปพลิเคชันแนะนำร้านอาหาร ส่วนการกรองข้อมูลสำหรับแนะนำรายการอาหาร ใช้หลักการ Content-Based Filtering ซึ่งให้ความสำคัญของข้อมูลรายการอาหารและค่าความคล้ายคลึงของรายการอาหารกับความต้องการของผู้ใช้งาน อีกทั้งมีการประยุกต์ใช้ระบบ Multi-criteria Rating เพื่อพิจารณาการกำหนดค่าคะแนนตามความชอบตามส่วนประกอบของรายการอาหาร สำหรับการหาค่าความคล้ายคลึงระหว่างรายการอาหารกับความชอบของผู้ใช้งาน ในระบบใช้วิธี K-NN (K-Nearest Neighbor) สำหรับการคำนวณระยะห่างระหว่างข้อมูล ส่วนแอปพลิเคชันแนะนำ

ร้านอาหารบนสมาร์ทโฟนโดยใช้ข้อมูลเชิงตำแหน่งและข้อมูลรายการอาหารบนโครงสร้าง Mobile Application Cross Platform ที่มีการสื่อสารแบบ Restful Protocol

Thesis Title Restaurant Recommendation System on Smart Phone using Location Based Service and Menu

Author Miss. Palita Sangsiri

Major Program Management of Information and Technology

Academic Year 2017

Abstract

At present, consumers pay attention to health more and more. In addition to exercise, food preferences are useful and important. So consumers will focus on the selection of food by considering components and flavor of the food. Menu selection by considering the components and taste that is complicated and finding a restaurant with a menu that meets the requirements is not easy and convenient. It also cannot access the nearby restaurant thoroughly.

The technology used at present, for example, web applications and applications on your smartphone by most support searching restaurant using search terms such as location, type of restaurants, restaurant's name, distance etc. That system can't guide menu from components and flavor of the food and can't find the restaurant from the menu that you want. So the system currently cannot solve the problems that mentioned above. Thus, the researcher has the concept in the development of application system on smart phone to solve such problems, including

The automatic restaurant recommendation system used on smart phone, through location and food data, is consisted of two key components which are 1) filter part for dish recommendation and 2) application part for restaurant recommendation. For the filter for dish recommendation, content-based filtering theory is applied which place an importance on dish data and similarity value of each food in accordance with user's demand. In addition, there is an application of multicriteria rating used for the furtherance of rating configure consideration aligning with preferences towards specific ingredients used in each dish. To find the similarity value between dish and user's personal taste, K-NN (K-Nearest Neighbor) is applied. Moreover, Standard Euclidean Distance is added in order to calculate distances between the data. In term of an application for restaurant recommendation, through location and food data, it is created on Mobile Application Cross Platform that is capable of accessing the Restful Protocol.

Keywords: Recommendation System, Similarity, Restaurant

กิตติกรรมประกาศ

วิทยานิพนธ์นี้สำเร็จลุล่วงด้วยดี ผู้วิจัยขอกราบขอบพระคุณบุคคลต่างๆ ที่ได้ให้คำปรึกษา และช่วยเหลืออย่างดียิ่ง ทั้งในด้านวิชาการ และด้านการดำเนินงานวิจัย ดังนี้

ขอขอบพระคุณรองศาสตราจารย์ ดร.ทวีศักดิ์ เรืองพีระกุล อาจารย์ที่ปรึกษาวิทยานิพนธ์ และผู้ช่วยศาสตราจารย์ ดร.นิคม สุวรรณวร อาจารย์ที่ปรึกษาร่วมวิทยานิพนธ์ ที่ให้คำปรึกษาและคำแนะนำต่างๆ รวมทั้งแนวทางและข้อคิดในการดำเนินงานวิจัยตลอดมา

ขอขอบพระคุณร้านอาหารใน อำเภอหาดใหญ่ ทุกๆ ร้าน ที่ให้ความร่วมมือ เอื้อเฟื้อข้อมูลร้านอาหารและรายการอาหารสำหรับใช้ในการทำวิจัยในครั้งนี้

ขอขอบพระคุณบัณฑิตวิทยาลัย มหาวิทยาลัยสงขลานครินทร์ ที่ให้ทุนอุดหนุนการวิจัยเพื่อวิทยานิพนธ์ ระดับปริญญาโท ปีการศึกษา 2557

ขอขอบคุณนายธีระยุทธ์ ถวิลวรรณ์ และเพื่อนๆ พี่ๆ น้องๆ สาขาการจัดการเทคโนโลยีสารสนเทศ และ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยสงขลานครินทร์ ที่คอยช่วยเหลือและให้กำลังใจมาโดยตลอด

ท้ายสุดผู้วิจัยขอกราบขอบพระคุณ บิดา มารดา ที่ให้การส่งเสริมในด้านการศึกษาและเป็นกำลังใจที่ดีตลอดมา จนประสบความสำเร็จในวันนี้

ปาลิตา แสงศิริ

สารบัญ

สารบัญ (ต่อ)

สารบัญ (ต่อ)

หน้า

รายการตาราง

หน้า

รายการตาราง (ต่อ)

หน้า

รายการตาราง (ต่อ)

หน้า

รายการภาพประกอบ

หน้า

รายการภาพประกอบ (ต่อ)

หน้า

รายการภาพประกอบ (ต่อ)

หน้า

บทที่ 1
บทนำ

1.1 ที่มาและความสำคัญของงานวิจัย

ปัจจุบันธุรกิจร้านอาหารประเภทต่างๆ ได้เติบโตและเพิ่มจำนวนขึ้นอย่างรวดเร็ว เพื่อตอบสนองความต้องการของผู้บริโภค และการแข่งขันทางธุรกิจ ทำให้ผู้บริโภคโดยทั่วไปมีตัวเลือกในการเลือกร้านอาหารเพื่อรับประทานมากขึ้น การค้นหาและตัดสินใจเลือกร้านอาหารหรือเมนูที่ถูกใจก็ทำได้ยากมากยิ่งขึ้น ส่งผลให้ผู้บริโภคไม่สามารถเข้าถึงร้านอาหารหรือรายการอาหารที่ถูกใจและอยู่บริเวณใกล้เคียงได้

ในชีวิตประจำวันผู้บริโภคที่ต้องการเลือกรับประทานอาหาร แต่ไม่ทราบว่าจะเริ่มต้นค้นหาหรือตัดสินใจเลือกอย่างไร โดยส่วนใหญ่จะค้นหาร้านอาหารและตัดสินใจเลือกอาหารเพื่อรับประทานโดยอ้างอิงจากประสบการณ์ที่เคยรับประทานที่ผ่านมา จากคำแนะนำของเพื่อนๆ หรือการประชาสัมพันธ์ต่างๆ ดังนั้นหากผู้บริโภคที่ยังไม่มีประสบการณ์หรือเคยรับประทานมาก่อน ผู้บริโภคจะตัดสินใจเลือกโดยทราบข้อมูลเพียงคร่าวๆ เท่านั้น เช่น ร้านอาหารชื่ออะไร เป็นอาหารประเภทไหน แต่ไม่สามารถทราบรายการอาหารที่แน่ชัด ชื่อรายการอาหารบางประเภทก็ไม่สามารถบ่งบอกส่วนประกอบหรือรสชาติของรายการอาหารนั้นได้ ผู้บริโภคจึงไม่สามารถเลือกอาหารที่มีส่วนประกอบที่ชอบหรือรสชาติที่ตนเองถูกใจได้ ดังนั้นทำให้มีข้อจำกัดต่างๆ เหล่านี้ในการตัดสินใจเลือกร้านอาหารเพื่อรับประทาน ทำให้ไม่สามารถตอบสนองความต้องการที่แท้จริงได้

เทคโนโลยีที่นำมาใช้ในร้านอาหารปัจจุบันยังไม่สนับสนุนหรือแก้ปัญหาดังกล่าว ได้แก่ ระบบบริหารจัดการร้านอาหาร ซึ่งระบบบริหารจัดการร้านอาหารเป็นระบบที่พัฒนาขึ้นเพื่อใช้งานเฉพาะเจาะจงสำหรับร้านใดร้านหนึ่งเท่านั้น ด้วยเทคโนโลยีที่มีการเปลี่ยนแปลงอย่างรวดเร็ว ส่งผลให้ร้านอาหารต้องมีการปรับเปลี่ยนระบบหรือโปรแกรมตามเทคโนโลยีที่เปลี่ยนแปลง มีค่าใช้จ่ายในการพัฒนาระบบที่สูง และระบบดังกล่าวยังไม่สามารถตอบสนองความต้องการของผู้บริโภคได้อย่างกว้างขวาง ระบบร้านอาหารบนเว็บ (Web Site, Web-based) ที่มีฟังก์ชันในการค้นหาร้านอาหารตามสถานที่ตั้งของร้าน ประเภทของอาหาร หรือราคา เว็บต่างๆ เหล่านี้จะมีสมาชิกเป็นของตัวเอง บางเว็บมีการสร้างลิงค์ไปยัง Facebook และ Twitter เป็นการเพิ่มโอกาสในการค้นหา แต่ระบบเหล่านี้ยังมีข้อจำกัดของความสะดวกและรวดเร็วในการใช้งาน เนื่องจากผู้ใช้จะต้องเข้าใช้งานระบบผ่านทางเว็บเบราเซอร์ และต้องใช้อินเตอร์เน็ตความเร็วสูง และยังไม่สามารถ

ตอบสนองหรือแก้ปัญหาของผู้บริโภคที่กล่าวมาข้างต้นได้ ระบบร้านอาหารในปัจจุบัน บนระบบปฏิบัติการ IOS และ android เป็นที่นิยม สมาร์ทโฟนจึงได้มีการพัฒนาสำหรับร้านอาหารมากขึ้น มีฟังก์ชันการทำงานในการค้นหาร้านอาหาร แสดงความคิดเห็นและจัดระดับความนิยมร้านอาหารนั้นๆ ตัวอย่างร้านอาหารที่มีการใช้แอปพลิเคชันบนสมาร์ทโฟน คือ KFC Thailand ซึ่งเป็นระบบร้าน KFC ที่พัฒนาให้รองรับการใช้งานบนสมาร์ทโฟน (KFC, 2018) และ Wongnai ระบบค้นหาร้านอาหาร มีฟังก์ชันสำหรับค้นหาร้านอาหารแสดงบนแผนที่ Google Maps แสดงความคิดเห็น และสามารถจัดระดับความนิยมให้กับร้านนั้นๆ ได้ (Wongnai, 2557) แต่ยังไม่สามารถดูรายการอาหารส่วนประกอบหรือรสชาติของอาหารแต่ละรายการและราคาได้

ทางผู้วิจัยจึงมีแนวคิดของระบบที่จะนำมาแก้ปัญหาดังกล่าวข้างต้น ได้แก่ ระบบแนะนำ (Recommender System) สำหรับแนะนำรายการอาหารตามความต้องการของผู้บริโภค (User Profile) ยกตัวอย่าง เช่น ผู้บริโภคประสงค์รับประทานกุ้ง ผักคะน้า หรือรายการอาหารที่มีรสชาติเผ็ด ระบบสามารถแนะนำรายการอาหารที่มีความคล้ายคลึงกับความต้องการของผู้บริโภคได้ และสามารถแนะนำร้านอาหารที่มีรายการอาหารที่ผู้บริโภคเลือกตามลักษณะของร้านอาหารที่ผู้บริโภคต้องการ (Restaurant Profile) โดยใช้เทคนิคการกรองแบบอิงเนื้อหา (Content-based Filtering) ทำให้ผู้บริโภคสามารถเข้าถึงรายการอาหารและรสชาติที่ถูกใจได้มากยิ่งขึ้น อีกทั้งสามารถค้นหาและเข้าถึงข้อมูลร้านอาหาร รายการอาหาร ราคา และตำแหน่งของร้านอาหาร ซึ่งมีความสะดวกในการใช้งานผ่านสมาร์ทโฟน โดยใช้การบริการเชิงตำแหน่ง Google Maps API ค้นหาพิกัดของร้านอาหาร มีการบริการและจัดเก็บข้อมูลที่อยู่บนคลาวด์ (Cloud) เพื่อลดการประมวลผลบนสมาร์ทโฟน ตอบสนองความต้องการของผู้ใช้บริการได้ทุกที่ทุกเวลา ลดความยุ่งยากในการเข้าถึงข้อมูลร้านอาหาร

1.2 วัตถุประสงค์ของงานวิจัย

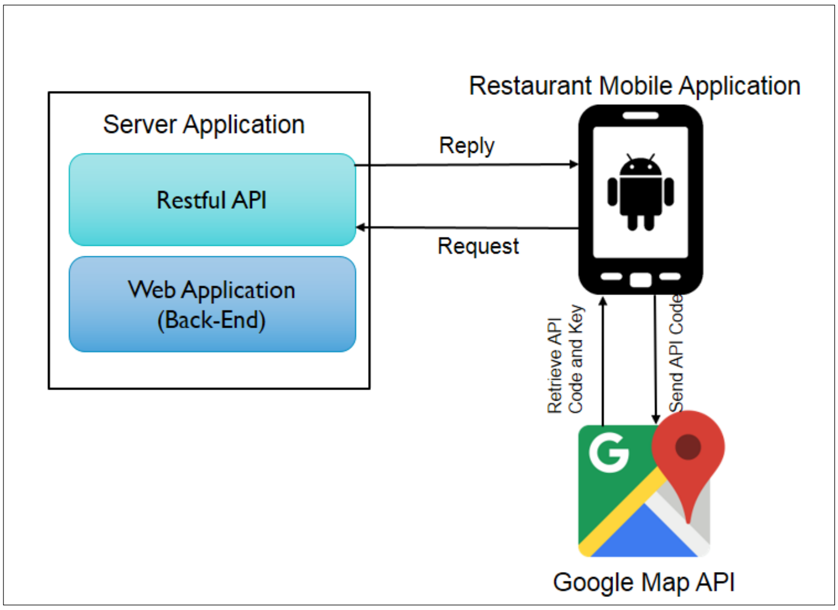
1.2.1 เพื่อพัฒนาระบบแนะนำรายการอาหารอัตโนมัติบนสมาร์ทโฟนโดยใช้ข้อมูลเชิงตำแหน่ง

1.2.2 วิเคราะห์วิจัยเพื่อหาวิธีการในการกรองข้อมูล (Content-based Filtering Technique) ที่เหมาะสมสำหรับการนำเสนอระบบแนะนำรายการอาหาร

1.3 ขอบเขตของงานวิจัย

1.3.1 ขอบเขตการพัฒนาระบบ

การพัฒนาระบบแนะนำรายการอาหารอัตโนมัติบนสมาร์ทโฟนโดยใช้ข้อมูลเชิงตำแหน่งและรายการอาหาร ระบบประกอบด้วย 2 ส่วนหลักได้แก่ Server Application และ Restaurant Mobile Application ดังรูปที่ 1-1



รูปที่ 1-1 ภาพรวมการทำงานของระบบโดยรวม

1.3.1.1 Server Application

1) Web Application (Back-End)

ผู้จัดการระบบ (Admin) ประกอบด้วยฟังก์ชันการทำงานดังนี้

a. ฟังก์ชัน Login สำหรับเข้าใช้งานระบบ

b. ฟังก์ชัน Manage User จัดการข้อมูลผู้ใช้

c. ฟังก์ชัน Crate/Delete/Update Profile Restaurant สำหรับเพิ่ม, ลบและแก้ไขข้อมูลร้านอาหาร เช่น ชื่อร้าน ที่อยู่ เบอร์ติดต่อ และข้อมูลรายละเอียดรายการอาหารและราคา ละติจูด ลองติจูด สำหรับกำหนดแผนที่ตั้งของร้านอาหาร เป็นต้น

d. ฟังก์ชัน Set Item Menu Profile สำหรับ สร้างข้อมูลพื้นฐานของแต่ละรายการอาหารที่จำเป็นในการประมวลผลของฟังก์ชันแนะนำรายการอาหารตามความต้องการของผู้ใช้

ผู้ใช้งานทั่วไป (User) ประกอบด้วยฟังก์ชันการทำงานดังนี้

a. ฟังก์ชัน Login สำหรับเข้าใช้งานระบบ

b. ฟังก์ชัน Crate/Delete/Update Profile Restaurant สำหรับเพิ่ม, ลบและแก้ไขข้อมูลร้านอาหาร เช่น ชื่อร้าน ที่อยู่ เบอร์ติดต่อ และข้อมูลรายละเอียดรายการอาหารและราคา ละติจูด ลองติจูด สำหรับกำหนดแผนที่ตั้งของร้านอาหาร เป็นต้น

2) **Restful API** สำหรับประมวลผลข้อมูล รับส่งข้อมูลระหว่าง Web Service และ Mobile Application

ตารางที่ 1-1 Restful API

ลำดับ	API Name	URL	อธิบาย
1	Register	/api/register	สมัครสมาชิก
2	Login	/api/login	เข้าสู่ระบบ,ออกจากระบบ
3	Set Profile	/api/setprofile	ตั้งค่าโปรไฟล์
4	Find Restaurant by Menu	/api/findrestaurant/menu	ค้นหาร้านอาหารโดยใช้รายการอาหาร
5	Find Restaurant by id	/api/findrestaurant/id	ค้นหาร้านอาหารโดยใช้รหัสของร้านอาหาร

ลำดับ	API Name	URL	อธิบาย
6	Recommend Menu	/api/recommendmenu	แนะนำรายการอาหาร
7	Recommend Restaurant	/api/recommendRestaurant	แนะนำร้านอาหาร
8	Home	/api/hotm	เรียกข้อมูลหน้าหลัก
9	Restaurant	/api/restaurant	เรียกข้อมูลร้านอาหาร
10	User Set Profile	/api/update-profile	แก้ไขโปรไฟล์

1.3.1.2 Restaurant Mobile Application ประกอบด้วยฟังก์ชันการทำงานดังนี้

a. ฟังก์ชัน Login สำหรับเข้าใช้งานระบบ

b. ฟังก์ชัน Register สำหรับลงทะเบียนเพื่อเป็นสมาชิกเข้าใช้งานระบบ

c. ฟังก์ชัน Setup Profile สำหรับการป้อนข้อมูล User Profile ของผู้ใช้การให้ค่าน้ำหนักแต่ละส่วนประกอบของรายการอาหารที่ชอบ

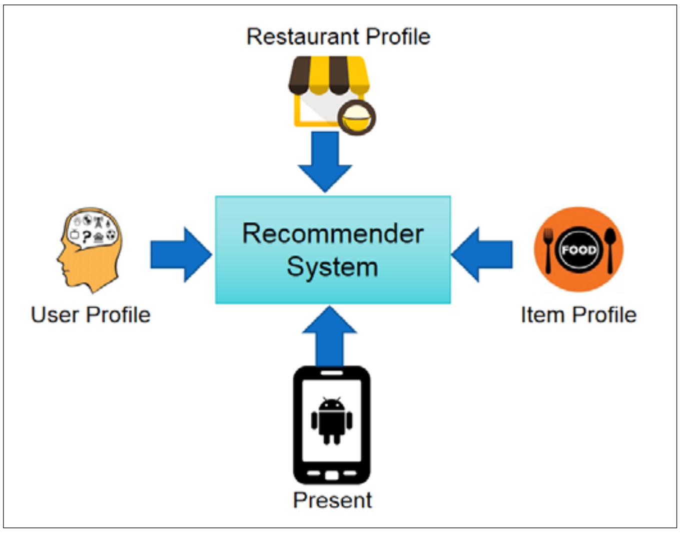
d. ฟังก์ชัน Present Menu แนะนำรายการอาหารที่ตรงตามความต้องการของผู้ใช้ และรายการอาหารใกล้เคียง ที่อยู่บริเวณโดยรอบตำแหน่งของผู้ใช้ ณ ขณะนั้น

e. ฟังก์ชัน Access Restaurant สำหรับการเข้าถึงข้อมูลร้านอาหารต่างๆ ได้โดยอ้างอิงจากการบริการเชิงตำแหน่งบนสมาร์ทโฟน

f. ฟังก์ชัน Search Restaurant สำหรับค้นหาร้านอาหารและเลือกดูข้อมูลต่างๆ ของร้านอาหาร เช่น ข้อมูลทั่วๆ ไป ชื่อร้าน ที่ตั้ง เบอร์ติดต่อ ข้อมูลรายละเอียดรายการอาหารและราคาอาหาร รวมทั้งการให้คะแนนความชอบกับร้านอาหาร

1.3.2 ขอบเขตการวิเคราะห์การกรองข้อมูล

โครงสร้างระบบ ส่วนแนะนำรายการอาหารจะประกอบด้วย 3 ส่วนหลัก ได้แก่ User Profile, Item Profile, Restaurant Profile ดังรูปที่ 1-2



รูปที่ 1-2 โครงสร้างระบบส่วนแนะนำรายการอาหาร

1.3.2.1 Item Profile (สร้างข้อมูลรายการอาหาร)

1) ส่วนประกอบที่เป็นเนื้อ ยกตัวอย่าง เช่น เนื้อไก่ เนื้อหมู เนื้อวัว ปลา กุ้ง หมึก ปู ไข่ อาหาทะเล เป็นต้น

2) ส่วนประกอบที่เป็นผัก ยกตัวอย่าง เช่น ผักคะน้า ผักบุ้ง ผักกาด ผักกวางตุ้ง ผักชี แตงกวา มะเขือ เป็นต้น

3) รสชาติ ได้แก่ เผ็ด เปรี้ยว หวาน เป็นต้น

4) ประเภทรายการอาหาร ได้แก่ แกง-ต้ม, ผัด-ทอด, ตุ๋น-นึ่ง

5) ช่วงราคา ได้แก่ น้อยกว่า100, 100-199, 200-299, 300-399, มากกว่า 400

1.3.2.2 User Profile (ความต้องการของผู้บริโภค) ความต้องการด้านรายการอาหาร ได้แก่

1) ส่วนประกอบที่เป็นเนื้อ ยกตัวอย่างเช่น ผู้บริโภคชอบเนื้อไก่ และ ซีฟู๊ด มากที่สุด

2) ส่วนประกอบที่เป็นผัก ยกตัวอย่างเช่น ผู้บริโภคชอบหน่อไม้ และ ผักบุ้ง มากที่สุด

3) รสชาติ ยกตัวอย่างเช่น ผู้บริโภคชอบรสชาติเผ็ดมากที่สุด

4) ประเภทรายการอาหาร ยกตัวอย่างเช่น ผู้บริโภต้องการรายการอาหาร ประเภท แกง-ต้ม มากที่สุด

5) ช่วงราคา ยกตัวอย่างเช่น ผู้บริโภคต้องการรายการอาหารที่ราคาน้อยกว่า 100

เงื่อนไขความต้องการด้านอื่นๆ ที่เกี่ยวข้องกับร้านอาหาร ได้แก่

1) ระยะทางน้อยกว่า 20 Km
2) ระดับดาว ได้แก่ 5 ดาว
3) บรรยากาศแบบมีเครื่องปรับอากาศ (Close)
4) มีฮาลาล
5) มีลานจอดรถ

1.3.2.3 Restaurant Profile (ข้อมูลรายละเอียดของร้านอาหาร)

1) ระยะทาง ได้แก่ <1 Km, <2Km, <3Km, <5Km, <10Km, <20Km ไม่จำกัด
2) ระดับดาว ได้แก่ 5ดาว 4ดาว 3ดาว 2ดาว 1ดาว ไม่ระบุ
3) บรรยากาศ ได้แก่ ไม่มีเครื่องปรับอากาศ (Open) มีเครื่องปรับอากาศ (Close) ไม่ระบุ
4) ฮาลาล ได้แก่ ฮาลาล ไม่ฮาลาล ไม่ระบุ
5) ลานจอดรถ ได้แก่ มีลานจอดรถ ไม่มีลานจอดรถ ไม่ระบุ

รวมทั้งข้อมูลทั่วไปของร้านอาหาร เช่น ที่ตั้งของร้านอาหาร ชื่อร้าน ที่อยู่ เบอร์ติดต่อ และรายละเอียดรายการอาหาร

1.3.2.4 กระบวนการคำนวณหาค่าความคล้ายคลึง

ขั้นตอนการคำนวณหาค่าคลามคล้ายคลึงมี 7 ขั้นตอนดังนี้

1) ผู้ใช้กำหนดความชอบด้านรายการอาหาร (User Profile) โดยกำหนดค่าความชอบแต่ละ Item ในระดับต่างๆ ระหว่าง 0-5 หลังจากนั้นทำการ Normalizetion แปลงค่าความชอบแต่ะ Item ให้อยู่ในรูป 0-1

2) นำค่าน้ำหนักความสำคัญมาคูณแต่ละ Item โดยระบบคำนวณค่าน้ำหนักความสำคัญให้อัตโนมัติหากผู้ใช้ไม่ได้กำหนด

3) คำนวณหาค่า Standard Euclidean Distance ระหว่างแต่ละ Item ของ User กับแต่ละ Item ของรายการอาหารตามสูตร (1) ดังนี้

X_i คือ Item ที่ i ของ User และ Y_i คือ Item ที่ i ของ รายการอาหาร เช่น Item ที่ 1 คือ ไก่

$$d(X,Y)=\sqrt{\sum_{i=1}^{n}(C-W_i)(X_i-Y_i)^2} \qquad (1)$$

n คือค่าของจำนวน Criteria ทั้งหมด

C คือค่าคงที่

W คือน้ำหนักของแต่ละ Item

4) คำนวณหาค่าความคล้ายคลึง (Similarity Measure) ตามสูตร (2) ดังนี้

$$Sv=\frac{1}{1+d(X,Y)} \qquad (2)$$

ค่าที่ได้จะอยู่ในช่วง 0-1 ค่ายิ่งมากยิ่งมีความคล้ายคลึงมาก

5) แสดงรายการรายการอาหารที่มีค่าความคล้ายคลึงจากมากไปหาน้อยเรียงตามลำดับ

6) คำนวณหาความคล้ายคลึงของร้านอาหาร Restaurant Profile กับ User Profile วิธีการทำเช่นเดียวกันกับ Item Profile

7) แสดงร้านอาหารที่มีค่าความคล้ายคลึงมากไปหาน้อยเรียงตามลำดับ

ยกตัวอย่างการหาค่าความคล้ายคลึงดังต่อไปนี้

1) ผู้ใช้กำหนดค่าความชอบด้านรายการอาหาร (User Profile) โดยกำหนดค่าความชอบในรายละเอียดย่อยของ Item ต่างๆ ในช่วง 0-5 ยกตัวอย่างดังรูปที่ 1-3

ชอบกินไก่และซีฟู๊ดมากที่สุดให้ค่าน้ำหนักเท่ากับ 5 ชอบกินหมูรองลงมาให้ค่าน้ำหนักเท่ากับ 2

ชอบกินหน่อไม้มากที่สุดให้ค่าน้ำหนักเท่ากับ 5 ชอบกินผักบุ้งและผักคะน้าน้อยลงมาให้ค่าน้ำหนักเท่ากับ 4 และ 3 ตามลำดับ

ชอบกินรสชาติเผ็ดมาก ให้ค่าน้ำหนักเท่ากับ 4 ชอบกินรสชาติหวานและเปรี้ยวน้อยลงมาให้ค่าน้ำหนักเท่ากับ 3 และ 1 ตามลำดับ

ส่วนประกอบ (เนื้อ)

ไก่	หมู	ซีฟู๊ด	...
5	2	5	...

ส่วนประกอบ (ผัก)

คะน้า	ผักบุ้ง	หน่อไม้	...
3	4	5	...

รสชาติ

เผ็ด	เปรี้ยว	หวาน
4	1	3

อื่นๆ

...
...

รูปที่ 1-3 การกำหนดคะแนนความชอบให้กับรายการอาหาร

กำหนดให้ค่าน้ำหนักอยู่ในรูป 0-1 Normalizetion (หารด้วย 5) คือ นำค่าคะแนนความชอบหารด้วย 5 ทุกๆ Criteria จากรูปที่ 1-4 ไก่มีค่าคะแนนความชอบเท่ากับ 5 เมื่อหารด้วย 5 จะได้ผลลัพธ์เท่ากับ 1 หมูมีค่าคะแนนความชอบเท่ากับ 2 เมื่อหารด้วย 5 จะไดผลลัพธ์เท่ากับ 0.4 ซีฟู๊ดมีค่าคะแนนความชอบเท่ากับ 5 เมื่อหารด้วย 5 จะได้ผลลัพธ์เท่ากับ 1 ทำเช่นเดียวกันทุกๆ Item จนครบทุก Criteria

ส่วนประกอบ (เนื้อ)

ไก่	หมู	ซีฟู๊ด	...
1	0.4	1	...

ส่วนประกอบ (ผัก)

คะน้า	ผักบุ้ง	หน่อไม้	...
0.6	0.8	1	...

รสชาติ

เผ็ด	เปรี้ยว	หวาน
0.8	0.2	0.6

อื่นๆ

...
...

รูปที่ 1-4 Normalization

2) จะนำค่าน้ำหนักความสำคัญมาคูณแต่ละ Item โดยระบบคำนวณค่าน้ำหนักความสำคัญแต่ละ Item ให้อัตโนมัติหากผู้ใช้ไม่ได้กำหนด

ยกตัวอย่างเช่น ผู้ใช้กำหนดค่าน้ำหนักความสำคัญให้กับส่วนประกอบที่เป็นเนื้อเท่ากับ 0.36 ส่วนประกอบที่เป็นผักค่าน้ำหนักความสำคัญเท่ากับ 0.36 รสชาติค่าน้ำหนักความสำคัญเท่ากับ 0.29 เป็นต้น จากรูปที่ 1-5 ไก่มีค่าคะแนนความชอบเท่ากับ 1 เมื่อคูณด้วยค่าน้ำหนัก คือ 0.36 จะได้ผลลัพธ์เท่ากับ 0.36 หมูมีค่าคะแนนความชอบเท่ากับ 0.4 เมื่อคูณค่าน้ำหนักเท่ากับ 0.36 จะได้ผลลัพธ์เท่ากับ 0.14 ซีฟู๊ดมีค่าคะแนนความชอบเท่ากับ 1 เมื่อคูณค่าน้ำหนักเท่ากับ 0.36 จะได้ผลลัพธ์เท่ากับ 0.36 ทำเช่นเดียวกันทุกๆ Item จนครบทุก Criteria

ส่วนประกอบ (เนื้อ)*0.36

ไก่	หมู	ซีฟู๊ด	...
0.36	0.14	0.36	...

ส่วนประกอบ (ผัก)*0.36

คะน้า	ผักบุ้ง	หน่อไม้	...
0.22	0.29	0.36	...

รสชาติ*0.29

เผ็ด	เปรี้ยว	หวาน
0.23	0.06	0.17

อื่นๆ

...
...

รูปที่ 1-5 การคูณค่าน้ำหนักให้กับทุก criteria ในแต่ละ Item

3) คำนวณหาค่า Standard Euclidean Distance ระหว่างแต่ละ Item ของ User กับแต่ละ Item ของรายการอาหาร ตามสูตร (1) ดังนี้

X_i คือ Item ที่ i ของ User และ Y_i คือ Item ที่ i ของ รายการอาหาร เช่น Item ที่ 1 คือ ไก่

$$d(X,Y)=\sqrt{\sum_{i=1}^{n}(C-W_i)(X_i-Y_i)^2} \tag{1}$$

n คือค่าของจำนวน Criteria ทั้งหมด

C คือค่าคงที่

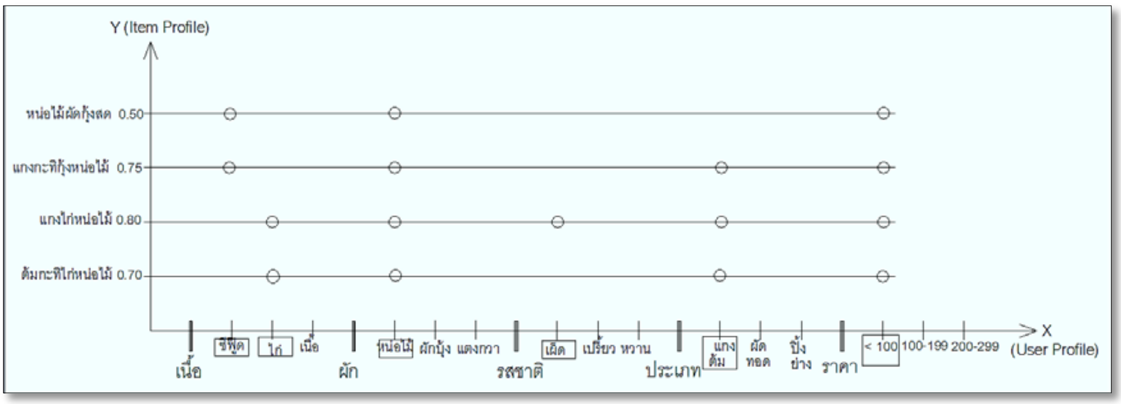
W คือน้ำหนักของแต่ละ Item

4) คำนวณหาค่าความคล้ายคลึง (Similarity Measure) ตามสูตร (2) ดังนี้

$$Sv = \frac{1}{1+d(X,Y)} \tag{2}$$

ค่าที่ได้จะอยู่ในช่วง 0-1 ค่ายิ่งมากยิ่งมีความคล้ายคลึงมาก

จากรูปที่ 1-6 กำหนดให้แกน X คือ ความต้องการของผู้ใช้ (User Profile) แกน Y คือ รายการอาหาร (Item Profile) ยกตัวอย่าง ผู้บริโภคประสงค์รับประทานไก่และซีฟู้ด หน่อไม้ รสชาติเผ็ดและหวาน และราคาน้อยกว่า 100 บาท จากนั้นจะทำการคำนวณหาค่าความต่าง Euclidean Distance และคำนวณหาค่าความคล้ายคลึง ระหว่าง User Profile และ Item Profile จะได้ค่าความคล้ายคลึงของรายการอาหารที่มีค่าสูงสุด 10 ลำดับ ยกตัวอย่าง 4 ลำดับดังต่อไปนี้ แกงไก่หน่อไม้ค่าความคล้ายคลึงเท่ากับ 0.80 แกงกะทิกุ้งหน่อไม้ค่าความคล้ายคลึงเท่ากับ 0.75 ต้มกะทิไก่หน่อไม้ค่าความคล้ายคลึงเท่ากับ 0.70 และหน่อไม้ผัดกุ้งสดค่าความคล้ายคลึงเท่ากับ 0.50



รูปที่ 1-6 การหาค่าความคล้ายคลึง ของแกน X,Y

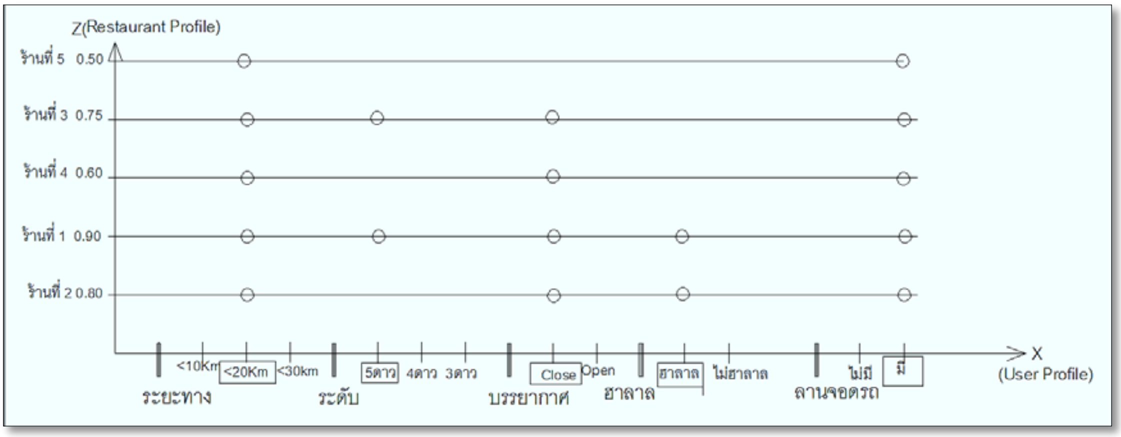
5) แสดงรายการอาหารที่มีค่าความคล้ายคลึงมากที่สุดเรียงตามลำดับ 10 รายการ ผู้บริโภคสามารถเลือกรายการอาหารที่ต้องการ เพื่อค้นหาร้านอาหาร โดยสามารถเลือกรายการอาหารที่ต้องการได้มากกว่า 1 รายการ เพื่อค้นหาร้านอาหารที่มีรายการอาหารทั้งหมดที่เลือก ยกตัวอย่างเช่น เลือก แกงไก่หน่อไม้ ต้มกะทิไก่หน่อไม้ หน่อไม้ผัดกุ้งสด เพื่อค้นหาร้านอาหาร ดังรูปที่ 1-7

ลำดับ	ชื่อเมนู	ค่าความคล้ายคลึง	เลือกเมนูที่ชอบ
1	แกงไก่หน่อไม้	0.80	√
2	แกงกะทิกุ้งหน่อไม้	0.75	
3	ต้มกะทิไก่หน่อไม้	0.70	√
4	หน่อไม้ผัดกุ้งสด	0.50	√
...	...	...	

รูปที่ 1-7 ผลลัพธ์ของการหาค่าความคล้ายคลึงของรายการอาหาร

6) คำนวณหาความคล้ายคลึงของร้านอาหาร Restaurant Profile กับ User Profile ที่มีรายการอาหาร แกงไก่หน่อไม้ ต้มกะทิไก่หน่อไม้ และ หน่อไม้ผัดกุ้งสด วิธีการทำเช่นเดียวกันกับ Item Profile ยกตัวอย่างดังนี้

จากรูปที่ 1-8 กำหนดให้แกน X คือ User Profile แสดงด้วยความต้องการของผู้ใช้ด้านความต้องการร้านอาหาร และแกน Z คือ Restaurant Profile แสดงข้อมูลต่างๆ ของร้านอาหาร จากนั้นจะทำการคำนวณหาค่าความต่าง Euclidean Distance และคำนวณหาค่าความคล้ายคลึงระหว่าง User Profile และ Restaurant Profile จะได้ค่าความคล้ายคลึงของร้านอาหารที่มีค่าสูงสุด 5 ลำดับที่มีรายการอาหารทั้งสามรายการดังต่อไปนี้ ร้านที่ 1 ค่าความคล้ายคลึงเท่ากับ 0.9 ร้านที่ 2 ค่าความคล้ายคลึงเท่ากับ 0.8 ร้านที่ 3 ค่าความคล้ายคลึงเท่ากับ 0.75 ร้านที่ 4 ค่าความคล้ายคลึงเท่ากับ 0.6 ร้านที่ 5 ค่าความคล้ายคลึงเท่ากับ 0.5



รูปที่ 1-8 การหาค่าความคล้ายคลึง ของแกน Y,Z

7) แสดงร้านอาหารที่มีค่าความคล้ายคลึงมากที่สุดเรียงตามลำดับจากมากไปหาน้อยดังรูปที่ 1-9

ชื่อร้าน	**ค่าความคล้ายคลึง**	**เมนูแกงไก่หน่อไม้**	**ต้มกะทิไก่หน่อไม้**	**หน่อไม้ผัดกุ้ง**
ร้านที่ 1	แกงไก่หน่อไม้	Sv =0.90	√	√
ร้านที่ 2	แกงกะทิกุ้งหน่อไม้	Sv =0.80	√	√
ร้านที่ 3	ต้มกะทิไก่หน่อไม้	Sv =0.75	√	√
ร้านที่ 4	หน่อไม้ผัดกุ้งสด	Sv =0.60	√	√
...	...	...		

รูปที่ 1-9 ผลลัพธ์การหาค่าความคล้ายคลึงของร้านอาหาร

1.3.3 ขอบเขตด้านประชากรและกลุ่มตัวอย่าง

1.3.3.1 ประชากร

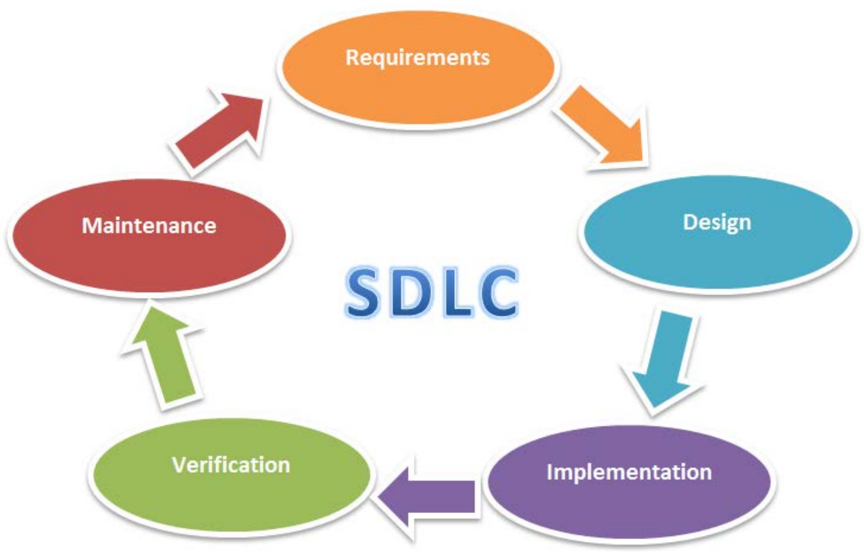
การวิจัยครั้งนี้มีประชากรที่ใช้ในการวิจัย ได้แก่ ร้านอาหารใน อ.หาดใหญ่ จ.สงขลา และบุคลากรและนักศึกษาคณะวิศวกรรมศาสตร์ มหาวิทยาลัยสงขลานครินทร์ วิทยาเขตหาดใหญ่

1.3.3.2 กลุ่มตัวอย่าง

กลุ่มตัวอย่างที่ใช้ในการศึกษาครั้งนี้ ได้แก่ ร้านอาหารใน อ.หาดใหญ่ จำนวน 60 ร้าน และบุคลากรและนักศึกษา คณะวิศวกรรมศาสตร์ มหาวิทยาลัยสงขลานครินทร์ วิทยาเขตหาดใหญ่ โดยใช้การสุ่มตัวอย่างแบบเจาะจง (Purposive Sampling) จำนวน 35 ราย

1.4 ขั้นตอนและวิธีการดำเนินงาน

ในงานวิจัยชิ้นนี้ ผู้วิจัยได้พัฒนาระบบโดยอาศัยหลักการของวงจรการพัฒนาระบบงาน (System Development Life Cycle: SDLC) ซึ่งประกอบด้วยกระบวนการดำเนินงาน แสดงดังรูปที่ 1-10



รูปที่ 1-10 ระเบียบวิธีการวิจัยด้วย (System Development Life Cycle: SDLC)

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1.5.1 ระบบรองรับร้านอาหารได้หลากหลายประเภทและหลายขนาด ไม่กำจัดเฉพาะร้านใดร้านหนึ่งเท่านั้น

1.5.2 โปรแกรมประยุกต์บนสมาร์ทโฟนสำหรับแนะนำรายการอาหารตามความต้องการของผู้บริโภค

1.5.3 ผู้ใช้สามารถเข้าถึงข้อมูลร้านอาหารได้สะดวกและรวดเร็ว

1.5.4 ระบบสามารถแนะนำรายการอาหารให้กับผู้บริโภคได้ตรงตามความต้องการ และเป็นการเพิ่มโอกาสทางธุรกิจให้กับร้านอาหาร

บทที่ 2
วรรณกรรมปริทัศน์

ผู้วิจัยได้ทำการศึกษาหลักการและทฤษฎีที่เกี่ยวข้อง และได้ทบทวนงานวิจัยที่เกี่ยวข้องทั้งในประเทศไทยและต่างประเทศ เพื่อนำไปพัฒนางานวิจัย โดยมีรายละเอียด ดังนี้

2.1 หลักการและทฤษฎีที่เกี่ยวข้อง

2.1.1 ระบบแนะนำ

Recommender Systems ระบบแนะนำข้อมูล ผลิตภัณฑ์ สินค้าและบริการ ให้กับผู้ใช้ระบบ โดยอ้างอิงจากสมมุติฐานการเรียนรู้ข้อมูลความชอบหรือความต้องการ ณ ขณะนั้นของผู้ใช้ ตัวอย่าง เช่น ระบบแนะนำในการเลือกซื้อหนังสือ ซีดีเพลง หรือผลิตภัณฑ์อื่นๆ ประกอบด้วย 4 ส่วนหลัก (เปรมฤดี ผลชอบ, 2553) ดังนี้

1) Item Profile คือ ข้อมูลพื้นฐานสำหรับใช้ในการประมวลผล โดยการสร้างโปรไฟล์ของแต่ละ Item บันทึกหรือเก็บรวบรวมข้อมูล ตัวอย่างเช่น การพิจารณาคุณสมบัติของภาพยนตร์ โดยพิจารณาจาก ผู้แต่งภาพยนตร์ นักแสดง ปี เป็นต้น

2) User Profile คือ เป็นข้อมูลที่ได้จากการป้อนของผู้ใช้ เช่นการให้คะแนนความชอบระดับความนิยม หรือประวัติการใช้งานในอดีต

3) ส่วนของอัลกอริธึม เป็นส่วนที่ใช้ประมวลผลข้อมูลเพื่อให้การแนะนำข้อมูลออกมาให้ตรงกับความต้องการของผู้ใช้

4) ส่วนในการนำเสนอให้แก่ผู้ใช้ เพื่อใช้เป็นตัวกลางในการติดต่อระหว่างผู้ใช้กับระบบ

2.1.1.1 Content-based Filtering

เป็นวิธีของการกรองข้อมูลเพื่อหาความคล้ายคลึง การทำงานของเทคนิควิธีนี้ คือ จะให้ความสำคัญต่อเนื้อหาของข้อมูลเป็นสำคัญ เพื่อค้นหาข้อมูลที่ผู้ใช้แต่ละคนสนใจ ซึ่งวิธีนี้จะทำการคำนวณหาค่าความคล้ายคลึงกัน (Similarity Measure) ระหว่างข้อมูลหรือสินค้าที่ระบบมีอยู่กับความต้องการในตัวข้อมูลหรือสินค้าที่ผู้ใช้ต้องการ สำหรับการให้คะแนนความชอบต่อข้อมูลยังไม่

ทั่วถึงหรือยังไม่ได้ให้คะแนนความชอบแก่ข้อมูลนั้นๆ จะไม่ส่งผลกระทบต่อเทคนิควิธีนี้ (เปรมฤดี ผลชอบ, 2553)

2.1.1.2 ระบบแนะนำที่ใช้ Multicriteria Rating มีความสำคัญอย่างไร

ระบบแนะนำในปัจจุบันนั้นทำการค้นหากลุ่มผู้ใช้ที่มีความชอบใกล้เคียงกัน โดยพิจารณาจากคะแนนความชอบเพียงค่าเดียวที่ผู้ใช้ได้ให้คะแนนต่อ Item นั้นๆ โดยไม่ได้คำนึงถึงคุณสมบัติที่ผู้ใช้ใช้ในการพิจารณาให้คะแนน ทำให้การค้นหาผู้ที่มีความชอบใกล้เคียงกัน (Neighbor) ของระบบมีความคลาดเคลื่อนและไม่สามารถค้นหาผู้ที่มีความชอบใกล้เคียงกันที่ตรงกับความต้องการของผู้ใช้ที่แท้จริงได้ (G. Admavicius and Y. Kown., 2007)

จากรูปที่ 2-1 ผู้ใช้ u_1 ต้องการหาค่าคะแนนความชอบของ i_5 จะเห็นได้ว่าผู้ใช้ u_1 มีคะแนนความชอบ i_1 ถึง i_4 เหมือนกับ u_2 และ u_3 ดังนั้นหากเราไม่พิจารณาคุณสมบัติย่อยของแต่ละ Item ก็จะสรุปได้ว่า u_1 น่าจะชอบสิ่งของ i_5 เท่ากับ 9 แต่ถ้าเราพิจารณาการให้คะแนนความชอบจากคุณสมบัติอื่นๆ ของ i_5เพิ่มเข้าไปแทนที่จะพิจารณาคะแนนโดยรวมเพียงค่าเดียวอาจพบได้ว่าผู้ใช้ u_1 อาจจะไม่ได้ชอบ i_5 เท่ากับ 9 คะแนนก็ได้

Target User

User most similar to the target user

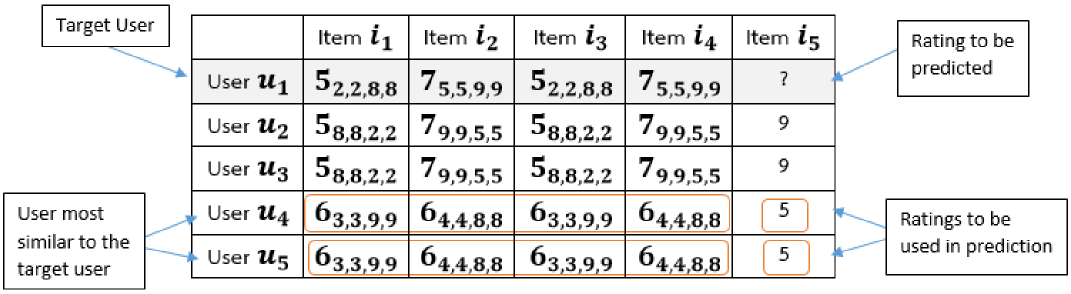
Rating to be predicted

Ratings to be used in prediction

	Item i_1	Item i_2	Item i_3	Item i_4	Item i_5
User u_1	5	7	5	7	?
User u_2	5	7	5	7	9
User u_3	5	7	5	7	9
User u_4	6	6	6	6	5
User u_5	6	6	6	6	5

รูปที่ 2-1 การทำนายค่าคะแนนความชอบในระบบปัจจุบัน ที่มา: G. Admavicius and Y. Kown. (2007)

จากรูปที่ 2-2 แสดงให้เห็นว่าผู้ใช้ u_1นั้นมีคะแนนความชอบแต่ละคุณสมบัติของ i_1 ถึง i_4 แตกต่างจากผู้ใช้ u_2 และ u_3 แต่มีคะแนนความชอบของแต่ละคุณสมบัติของ i_1 ถึง i_4 ใกล้เคียงกับผู้ใช้ u_4 และ u_5 มากกว่า ดังนั้นคะแนนความชอบที่ระบบควรแนะนำจึงควรเป็น 5 ตามผู้ใช้ u_4 และ u_5 ซึ่งเรียกวิธีการนี้ว่า Multicriteria Rating



รูปที่ 2-2 การทำนายค่า Rating แบบ Multicriteria ที่มา: G. Admavicius and Y. Kown. (2007)

2.1.2 การหาค่าความคล้ายคลึง (Similarity measure)

2.1.2.1 K-NN (K-nearest Neighbor)

วิธีการหาสมาชิกที่มีความคล้ายคลึงกันมากที่สุด โดยการหาค่าระยะห่างระหว่างข้อมูล (Standard Euclidean Distance) เป็นวิธีการหาค่าความคล้ายคลึงที่ง่ายและนิยมใช้มากที่สุด มีฟังก์ชันการใช้งาน (เปรมฤดี ผลชอบ, 2553) ดังนี้

ข้อมูลที่นำมาคำนวณประกอบไปด้วย n แอททริบิวต์ โดยข้อมูลในเรคคอร์ด X ใดๆ แทนด้วยเวกเตอร์ได้ดังนี้

$$x = (x1, x2, x3, \ldots, xn)$$

โดย xi หมายถึงค่าข้อมูลในแอททริบิวต์ที่ i

การคำนวณระยะทาง $d(x, y)$ คือ ระยะห่างระหว่างข้อมูล x และ y โดยการวัดระยะทางของ Standard Euclidean Distance กับการให้น้ำหนักหรือการระดับให้ความสำคัญกับแอททริบิวต์แต่ละตัวซึ่งเรียกว่า Distance Weighted เพื่อให้เหมาะกับการคำนวณหาค่าความคล้ายคลึงกัน ดังแสดงในสมการ (1) และ (2)

$$d(X,Y)=\sqrt{(c-w_1)(x_1-y_1)^2+(c-w_2)(x_2-y_2)^2+\ldots+(c-w_n)(x_n-y_n)^2} \quad (1)$$

$$d(X,Y)=\sqrt{\sum_{i=1}^{n}(C-W_i)(X_i-Y_i)^2} \quad (2)$$

C คือ ค่าคงที่

X คือ เรคคอร์ดของข้อกำหนดของ User Profile ที่ผู้ใช้ต้องการ

Y คือ เรคคอร์ดของข้อกำหนดของ Item Profile ในระบบ

W คือ ค่าของน้ำหนักของข้อมูลแต่ละแอททริบิวต์

X_i คือ ค่าข้อมูลในแอททริบิวต์ที่ i ของเรคอร์ด x

Y_i คือ ค่าข้อมูลในแอททริบิวต์ที่ i ของเรคอร์ด y

W_i คือ ค่าน้ำหนักของข้อมูลในแอททริบิวต์ที่ i

วิธีการนี้เป็นวิธีหาความคล้ายคลึงระหว่างเอกสารในระบบกับโปรไฟล์ของผู้ใช้ โดยนำเนื้อหาหรือคุณลักษณะของข้อมูลมาสร้างเป็นโปรไฟล์ของผู้ใช้เพื่อค้นหาข้อมูลที่ตรงตามความต้องการของผู้ใช้

2.1.2.2 Naive Bayes Theorem

เป็นอัลกอริธึมในการหาค่าความคล้ายคลึง โดยการแบ่งเป็นคลาสของข้อมูลและหาค่าความคล้ายคลึง โดยอาศัยหลักสถิติและความน่าจะเป็น โดยค่าที่ให้ความน่าจะเป็นสูงสุด คือคำตอบนั่นเอง ค่าความน่าจะเป็นแต่ละแอตทริบิวต์จะต้องไม่ขึ้นต่อกัน จะได้ Bayes Rule ดังสมการ (3) (วลัยนุช สกุลนุ้ย, 2552)

$$P_c = \left(\frac{P(C = c)}{P(A_1 = a_1 \wedge \ldots \wedge A_k = a_k)} \right) . \prod_{i=1}^{k} P(A_i = a_i \mid C = c) \tag{3}$$

เปรียบเทียบแต่ละคลาสเพื่อหาว่าข้อมูล i ควรอยู่ในคลาสไหนสำหรับผู้ใช้ u ดังสมการ (4)

$$P_c = P(C) . \prod_{i=1}^{k} P(A_i = a_i \mid C = c) \tag{4}$$

หาค่าความน่าจะเป็นของเนื้อหาหรือแอตทริบิวต์ $a_1 P(A_i = a_i | C = c)$ อยู่ในคลาส c ดังสมการ (5)

$$P(A_i = a_i \mid C = c) = \frac{\text{count}(A_i = a_i \wedge C = c)}{\text{count}(C = c)} \qquad (5)$$

$count(A_i = a_i \wedge C = c)$ จำนวนครั้งที่แอตทริบิวต์อยู่ในคลาส c

$count(C = c)$ จำนวนคะแนนความชอบหมดที่อยู่ในคลาส c

2.1.2.3 Pearson Correlation

วิธีการหาค่าความคล้ายคลึงโดยวัดจากค่าสหสัมพันธ์ (Correlation) ระหว่างข้อมูล 2 ชุด ค่าที่ได้จะบอกถึงความสัมพันธ์ระหว่างข้อมูลทั้งสองชุด ว่าเหมือนกันมากน้อยแค่ไหน โดยค่าที่ได้จะอยู่ระหว่าง -1 ถึง 1 ยิ่งค่านี้มีค่าเป็นบวกมาก แสดงว่าข้อมูลทั้งสองมีความคล้ายคลึงกันมาก ในทางตรงกันข้าม หากค่าที่ได้เป็นลบก็จะแสดงว่าผู้ใช้ทั้งสองคนนี้ไม่มีความคล้ายกัน ดังสมการที่ (6) (วรชัญ เฉลิมพรพงศ์, 2555)

$$Pearson(u,v) = \frac{\sum_{i \epsilon I_{uv}} (u_i - \bar{u})(v_i - \bar{v})}{\sqrt{\sum_{i \epsilon I_{uv}} (u_i - \bar{u})^2} \cdot \sqrt{\sum_{i \epsilon I_{uv}} (v_i - \bar{v})^2}} \qquad (6)$$

I_{uv} คือ เซตสินค้าที่ผู้ใช้ u

u และ v เคยให้ค่าความพึงพอใจ (Co-rated Item) ต่อสินค้าชิ้นที่ i ส่วน $\bar{u}$, $\bar{v}$ คือค่าเฉลี่ยความพึงพอใจของของผู้ใช้ u และ v ดังสมการที่ (7)

$$\bar{u} = \frac{i}{n} \sum_{i \in I_u} u_i \qquad (7)$$

ทางผู้วิจัยจึงมีความสนใจที่จะพัฒนาระบบแนะนำรายการอาหารด้วยการประยุกต์ใช้ระบบแนะนำที่ใช้ Multicriteria Rating โดยพิจารณาว่าลักษณะของข้อมูลรายการอาหารนั้นตรงตามโปรไฟล์ของผู้ใช้หรือไม่ ถ้าไช่ก็จะนำเสนอทันที โดยใช้อัลกอริธึม K-NN (K-nearest Neighbor) ในการหาสมาชิกที่ใกล้เคียงกันมากที่สุด เนื่องจากเป็นวิธีการที่ไม่ซับซ้อนและเข้าใจง่าย อีกทั้งยังเป็นวิธีการที่นิยมใช้มากที่สุด เหมาะสำหรับข้อมูลประเภทตัวเลข อีกทั้งเงื่อนไขในการคำนวณที่ซับซ้อนจะไม่ส่งผลกระทบต่อวิธีนี้

ข้อดีของวิธีนี้ คือ การแนะนำผลลัพธ์ที่ได้มีความถูกต้องสูง รองรับข้อมูลจำนวนมาก และผู้ใช้สามารถเรียนรู้ได้เร็ว

2.1.3 Location Base Service

2.1.3.1 Web Services

เว็บเซอร์วิส (Web Services) เป็นการบริการ ซึ่งเป็นซอฟต์แวร์ที่ออกแบบมาเพื่อสนับสนุนการทำงานระหว่างเครื่องคอมพิวเตอร์หลายเครื่องผ่านระบบเครือข่าย ภาษาที่นิยมใช้ในการติดต่อสื่อสาร คือ ภาษาเอกซ์เอ็มแอล (XML) และเจซัน (JSON) ตัวอย่าง เว็บเซอร์วิส (Web Services) คือ เว็บ Amazon เป็นชุด Service ประมวลผลระยะไกลประกอบกันเป็นแพลตฟอร์มประมวลผลบนคลาวด์ Cloud

เว็บเซอร์วิส (Web Services) (ศูนย์สารสนเทศ กลุ่มสารสนเทศข้อมูล, 2557) ทำให้สามารถเชื่อมต่อบริการต่างๆ เข้าด้วยกันได้ ทำให้โปรแกรมที่ถูกพัฒนาด้วยภาษาหรือเทคโนโลยีที่ต่างกัน หรือพัฒนาบนแพลตฟอร์มที่ต่างกันสามารถสื่อสารกันได้ มาตรฐานที่ทำให้เว็บเซอร์วิสติดต่อกันได้ เช่น การใช้เอกสารภาษาวิสเดิล (Web Services Description Language) ซึ่งไม่ว่าจะเป็นมนุษย์หรือโปรแกรมสามารถอ่านเอกสารวิสเดิลได้ เนื่องจากเป็นเอกซ์เอ็มแอล (XML) ประเภทหนึ่ง รูปแบบของข้อมูลเอกซ์เอ็มแอลที่ใช้ในการติดต่อนี้เรียกว่าโซป (SOAP : Simple Object Access Protocol) หากเป็นซอฟต์แวร์ที่ส่งข้อมูลเอ็กซ์เอ็มแอลโดยใช้อินเทอร์เน็ตโปรโตคอลจะเป็นเว็บเซอร์วิสแบบเรสต์ (REST) เว็บเซอร์วิสแบบเรสต์นี้จะใช้เจซัน (JSON) มากกว่า เอกซ์เอ็มแอล (XML) เพราะแสดงข้อมูลได้สั้นและกระชับมากกว่า โดยใช้โปรโตคอลเฮชทีทีพี (HTTP) (Doug Tidwell, 2000)

1) Simple Object Access Protocol (SOAP)

Simple Object Access Protocol (SOAP) (Software Engineering, 2552) เป็นโปรโตคอลชนิดหนึ่งที่ใช้บนเว็บเซอร์วิส (Web Services) และนิยมใช้ HTTP เป็นโปรโตคอลร่วมสำหรับส่งผ่านข้อมูลบนระบบอินเตอร์เน็ต

รูปแบบการส่งข้อมูลผ่าน Simple Object Access Protocol (SOAP)

Client สร้าง SOAP Message เพื่อเรียกใช้เว็บเซอร์วิสที่ผู้ให้บริการ (Provider) ประกาศไว้ หลังจากผู้ให้บริการ (Provider) ได้รับ SOAP Message จาก Client เว็บเซอร์วิส (Web Services) จะประมวลผลและส่งผลลัพธ์มายัง Web Service Application หลังจากนั้นผู้ให้บริการ (Provider) สร้าง SOAP Message พร้อมแนบผลลัพธ์ (Output) ส่งให้ Client (Doug Tidwell, 2000)

โครงสร้างของ SOAP Message

- SOAP บรรจุข้อมูลทั้งหมดไว้ภายใน SOAP Message และช่วยจัดเตรียมบริการต่างๆ ใช้ RPC เพื่อเรียกดูข้อมูลที่อยู่ในรูปแบบ XML Message
- SOAP Intermediaries จะทำหน้าที่เป็นตัวกลางในการส่ง SOAP Message จากต้นทางไปยังปลายทาง

2) Representational State Transfer (REST)

Representational State Transfer (REST) (Software Engineering, 2552) ใช้ HTTP protocol ในการรับส่งข้อมูล เว็บเซอร์วิสแบบ REST จะมี URL กำกับเพื่อบอกที่อยู่ของแหล่งข้อมูล ซึ่งสามารถที่จะจัดการได้โดยวิธีการต่างๆ (Doug Tidwell, 2000) ดังนี้

- GET สำหรับดึงข้อมูล
- DELETE สำหรับลบการนำเสนอ
- PUT สำหรับการสร้างและเปลี่ยนแปลงการนำเสนอ
- POST สำหรับการแก้ไขข้อมูล

ข้อดีของ Representational State Transfer (REST) คือผู้พัฒนาเว็บเซอร์วิสสามารถสร้างและเรียกใช้งานเว็บเซอร์วิสได้ง่าย เพียงแค่ใช้ XML และ HTTP เท่านั้น

ดังนั้นผู้วิจัยจึงเลือกใช้ Representational State Transfer (REST) เนื่องจากเรียนรู้และเข้าใจง่าย สามารถสร้างและเรียกใช้งานเว็บเซอร์วิสได้ง่าย ลดความซ้ำซ้อนในการดำเนินงาน (Simplifying) เพิ่มความสามารถในการนำกลับมาใช้ใหม่ (Increasing Reused Ability) และเป็น Architectural Style ใช้ได้หลายรูปแบบเช่น HTTP, URL, XML/HTML/GIF/JPEG, Text/XML เป็นต้น

2.1.4 Cloud Server

กลุ่มคอมพิวเตอร์จำลองบนก้อนเมฆที่ให้บริการโดยใช้ทรัพยากร ซอฟต์แวร์ และข้อมูลร่วมกัน เป็นการทำงานร่วมกันของเซิร์ฟเวอร์จำนวนมาก แม้เซิร์ฟเวอร์ตัวใดเสียไปจะไม่ส่งผลกระทบต่อการใช้งาน เพราะระบบจะทำการสลับการใช้งานไปยังตัวอื่นอัตโนมัติ ซึ่งแตกต่างจากเว็บโฮสติ้งโดยทั่วไปเมื่ออุปกรณ์ตัวใดเสียก็จะส่งผลกระทบต่อการใช้งาน (Cymiz., 2555)

VPS (Virtual Private Server) คือเซิฟเวอร์เสมือนที่การทำงานของ CPU, Memory และ Disk แยกจากกัน จึงไม่ส่งผลกระทบต่อกัน ดังนั้นการทำงานจึงเป็นอิสระเหมือนใช้เซิร์ฟเวอร์ของตัวเอง และค่าใช้จ่ายน้อยกว่าการเช่าเซิร์ฟเวอร์ทั้งเครื่อง

2.1.4.1 ระบบ Cloud Server ดีกว่า Web Hosting อย่างไร (Cymiz., 2555)

1) ระบบคลาวด์ (Cloud) ใช้เครือข่ายเน็ตเวิร์คความเร็วสูง สามารถรับส่งข้อมูลได้อย่างรวดเร็ว

2) ระบบคลาวด์ (Cloud) มีการยืดหดตัวของเซิร์ฟเวอร์เสมือนเมื่อมีการใช้งานเพิ่มขึ้นหรือลดลง

3) ระบบคลาวด์ (Cloud) จะมีต้นทุนของระบบที่สูงกว่าระบบอื่นมาก จึงทำให้การทำงานไม่ติดขัดซึ่งแตกต่างจากเซิร์ฟเวอร์ธรรมดาทั่วไป หากมีอุปกรณ์ใดเสียก็จะส่งผลกระทบต่อการใช้งาน

4) เส้นทาง Network มีการทำสำรองทุกจุด ระบบทั่วไปมักมีเส้นทางสัญญาณเพียงเส้นเดียว แต่ระบบคลาวด์ (Cloud) จะมีเส้นทางสำรองกรณีเส้นทางหลักเสียหรือหนาแน่น และสามารถส่งข้อมูลได้ปริมาณมากตามความต้องการ

ดังนั้นผู้วิจัยจึงเลือกใช้ Cloud Server ในการพัฒนาระบบเนื่องจากสามารถจัดการกับเซิร์ฟเวอร์ได้เหมือนเซิร์ฟเวอร์ของตนเอง มีประสิทธิภาพทั้งด้านการทำงาน และความปลอดภัยที่สูงกว่าเซิร์ฟเวอร์ทั่วไป ไม่มีปัญหาหากมีการส่งข้อมูลปริมาณมากหรือการจราจรของข้อมูลที่หนาแน่น

2.1.4.2 เปรียบเทียบผู้ให้บริการ Cloud Server

Heroku และ AppFog (Heroku, 2018) เหมาะสำหรับงาน เช่น Web Site ทั่วไป โดยที่ไม่จำเป็นต้องเกี่ยวข้องกับ Server ไม่ต้องควบคุมติดตั้ง หรือตั้งค่าใดๆ ให้กับ Server เรามีหน้าที่ซื้อหรือเช่าทรัพยากรต่างๆ เช่น CPU, RAM และ Disk เท่านั้น เมื่อติดตั้ง Web Site ของเราเสร็จก็ใช้งานได้ทันที โดย Heroku และ AppFog จะเป็นผู้ดูแลทั้งหมด ส่วน Data Center ของ AppFog มีหลายหลายมากกว่า Heroku

Amazon Cloud เหมาะสำหรับงานที่ต้องการควบคุมติดตั้ง Server เองทั้งหมด จะมีระบบปฏิบัติการให้เลือกใช้ ซึ่งระบบติดตั้งเองอัตโนมัติตามความต้องการของผู้ใช้ มีทรัพยากรต่างๆ เช่น CPU, RAM และ Disk ให้เลือกใช้งาน ทำให้สามารถใช้งานได้เหมือนมี Web Server เป็นของตนเอง ส่วนของ Data Center อยู่ครอบคลุมทั่วโลก

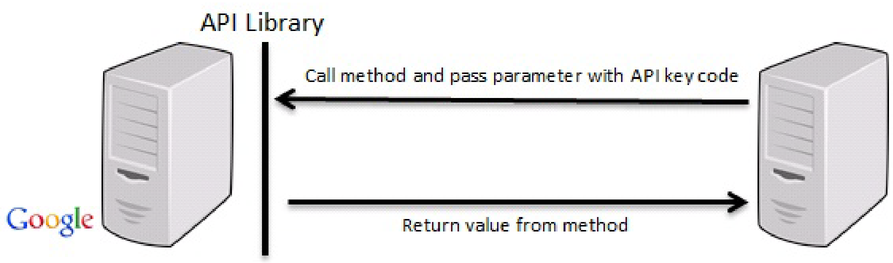
DigitalOcean (DigitalOcean, 2018) คือ Simple Cloud Hosting ผู้ใช้มีสิทธิ์กระทำทุกอย่างกับเครื่องได้เหมือนมีเครื่องคอมพิวเตอร์ของเราเอง (Root Access) จุดเด่นคือสามารถใช้งานได้ง่ายและรวดเร็ว ราคาถูก ใช้ฮาร์ดแวร์คุณภาพสูง และมี Datacenter ให้เลือกหลากหลายพื้นที่ ได้แก่ นิวยอร์กซานฟรานซิสโก อัมสเตอร์ดัม และสิงคโปร์ ระบบการสำรองข้อมูลของ DigitalOcean จะเรียกว่า “Snapshots” ไฟล์สำรองข้อมูลจะเรียกว่า “Images” เปรียบเทียบเหมือนกับการถ่ายรูปของเซิร์ฟเวอร์ไว้นั่นเอง ซึ่งเป็นการบันทึกค่าทั้งหมดของเซิร์ฟเวอร์ขณะนั้น

สามารถใช้งานผ่านแผงควบคุมในเว็บไซต์ได้เลย และยังสามารถกู้คืนไฟล์ “Images” นี้ใน Datacenter ที่ต่างจากตอนสำรองข้อมูลได้อีกด้วย

ดังนั้นทางผู้วิจัยจึงเลือกใช้งาน DigitalOcean เนื่องจากสามารถใช้งานได้ง่ายและติดตั้งได้อย่างรวดเร็ว ราคาถูก รวมทั้งมี Data Center อยู่ประเทศสิงคโปร์ทำให้สามารถเรียกใช้งานและตอบสนองได้อย่างรวดเร็ว ด้วยข้อจำกัดในด้านงบประมาณของผู้วิจัยเอง ทำให้ DigitalOcean จึงเหมาะสมที่สุดในการเลือกใช้งาน

2.1.5 Google Maps API

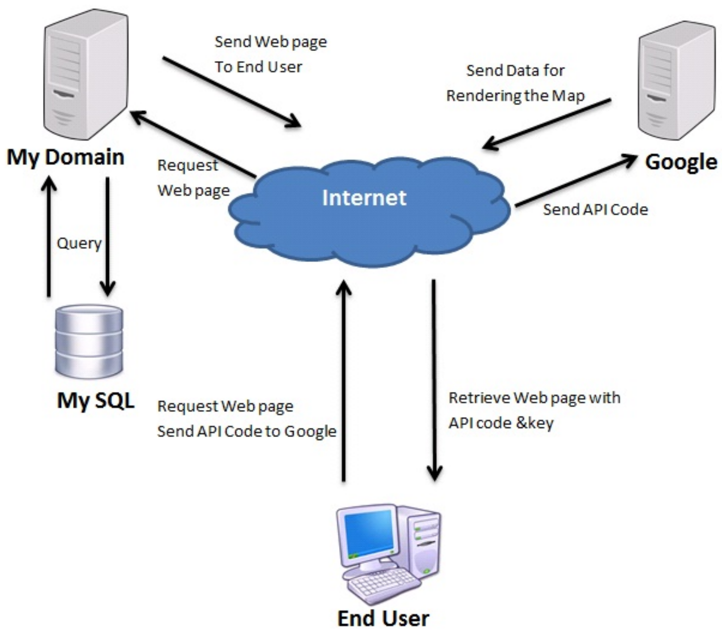
API (Application Programming Interface) (ปฐมพงษ์ ฉับพลัน, 2553) คือ การเขียนโปรแกรมสำหรับเชื่อมต่อผู้ให้บริการผ่านอินเตอร์เฟสของคลาสต่างๆ จากรูปที่ 2-3 แสดงการรับส่งข้อมูลระหว่างผู้ใช้กับผู้ให้บริการ โดยผู้สามารถเรียกใช้งาน ข้อมูล (Data) ความสามารถ (Method) จากผู้ให้บริการ ซึ่งผู้ให้บริการจะมีไลบารีหรือคลาสเพื่อให้บริการ



ที่มา: ปฐมพงษ์ ฉับพลัน, (2553)

รูปที่ 2-3 การเชื่อมต่อ API ของ Google

Google Maps API เป็นส่วนที่ใช้ในการเชื่อมต่อกับ Google เพื่อเรียกใช้ข้อมูลทางภูมิศาสตร์ด้วยภาษา JavaScript และรับส่งข้อมูลในรูปแบบของภาษา เอกซ์เอ็มแอล (XML)



ที่มา: ปฐมพงษ์ ฉับพลัน, (2553)

รูปที่ 2-4 การทำงานกับ Google Maps

จากรูปที่ 2-4 แสดงการการทำงานกับ Google API ผู้ใช้จะต้องมี Google Maps API Key ด้วยการลงทะเบียนกับ Google และในการเรียกดูแผนที่จะต้องส่ง Google Maps API Key เพื่อแจ้งความต้องการไปยัง Google ขอให้ Google Maps API ประมวลผล จากนั้น Google ก็จะส่งแผนที่กลับมาเพื่อแสดงข้อมูลแผนที่ให้กับผู้ร้องขอผ่านทางเว็บเพจในการสร้างแผนที่ให้ระบุละติจูดและลองติจูดของศูนย์กลางแผนที่ที่ต้องการ

1) GMapTypeControl() ใช้ในการสร้างตัวควบคุมแผน ซึ่ง Google Maps ได้ เตรียมไว้ให้ เพื่อกำหนดลักษณะการแสดงผลแผนที่แบบ Map, Satellite หรือ Hybrid

2) GMap2.removeMapType() ใช้ในการยกเลิกชนิดแผนที่ที่ไม่ต้องการออกไป

3) GMap2.addMapType() ใช้ในการเพิ่มชนิดแผนที่ต้องการ

ดังนั้นการใช้ GPolyline() สามารถสร้างขอบเขตแผนที่ที่ต้องการ เช่นการสร้างขอบเขตของบริเวณร้านอาหารได้

2.1.6 Mobile Application

ระบบปฏิบัติการบนอุปกรณ์สื่อสารแบบพกพาในปัจจุบัน (9'M webmaster@sourcecode.in.th, 2556) ได้แก่ Android OS (Android Operating System), IOS (iPhone Operating System) และ Windows Phone ในส่วนของระบบปฏิบัติการ Windows Phone นั้นไม่ได้รับความนิยมเท่าที่ควร และการพัฒนายังช้ากว่าระบบปฏิบัติการอื่นๆ ดังนั้นผู้วิจัยจึงเลือกทำการศึกษาและพิจารณาเฉพาะ Android OS (Android Operating System) และ IOS (iPhone Operating System)

2.1.6.1 Android OS (Android Operating System) (WDS PhoneTablet., 2555) คือระบบปฏิบัติการบนอุปกรณ์พกพา เช่น สมาร์ทโฟน แทบเล็ต เน็ตบุ๊ก เป็นระบบปฏิบัติการโอเพนซอร์ซที่มีพื้นฐานอยู่บนลินุกซ์

แอปพลิเคชันจะเขียนโดยใช้ภาษาจาวา และใช้แอนดรอยด์ซอฟต์แวร์เดเวล็อปเมนต์คิต (Android Software Development Kit) หรือ SDK โดย SDK จะประกอบด้วยชุดเครื่องมือต่างๆสำหรับใช้ในการพัฒนาแอปพลิเคชัน รวมไปถึงแหล่งรวมซอฟต์แวร์ โดยโครงสร้างของระบบปฏิบัติการ Android OS ประกอบด้วย Applications ส่วนบนสุดจะเป็นส่วนที่ผู้ใช้งานทำการติดต่อโดยตรง, Application Framework เป็นส่วนที่พัฒนาขึ้นเพื่อให้นักพัฒนาสามารถพัฒนาโปรแกรมได้สะดวก และมีประสิทธิภาพมากยิ่งขึ้น, Libraries เป็นส่วนของชุดคำสั่ง Android Runtime และ Linux Kernel คือส่วนที่เป็นหัวใจสำคัญ ในจัดการกับบริการหลักของระบบปฏิบัติการ เช่น เรื่องหน่วยความจำ พลังงาน ติดต่อกับอุปกรณ์ต่างๆ ความปลอดภัย และระบบปฏิบัติการ Android OS ยังมีดังนี้

1) ได้รับความนิยมในกลุ่มของสมาร์ทโฟนรุ่นใหม่ และอุปกรณ์จอสัมผัส (Touch Screen) ซึ่งมีความหลากหลายในส่วนของขนาดหน้าจอ ความเร็วของหน่วยประมวลผล ปริมาณหน่วยความจำ อื่นๆ

2) Google ได้พัฒนา Application Framework ไว้เพื่อให้นักพัฒนาได้ใช้งานได้อย่างสะดวก

3) ชุดโปรแกรมที่พัฒนาสามารถนำไปใช้ได้กับอุปกรณ์ที่ต่างกัน

4) Android เป็นระบบปฏิบัติการ ที่ พัฒนา Map และระบบ Navigation บนสมาร์ทโฟน ได้ดีที่สุดเพราะมี Google Map ที่รองรับการทำงานแบบติดตามเส้นทาง

2.1.6.2 IOS (iPhone Operating System) (Chaiyohosting, 2557) คือ ระบบปฏิบัติการสำหรับสมาร์ทโฟน ของบริษัทแอปเปิล เป็น Close Source สำหรับภาษาที่ใช้พัฒนาแอปพลิเคชันบนระบบปฏิบัติการ IOS คือ ภาษา Objective-c ในปัจจุบันนักพัฒนาแอปพลิเคชันบนระบบปฏิบัติการ IOS ยังมีจำนวนน้อยกว่าระบบปฏิบัติการ Android OS เนื่องจากนักพัฒนาต้องมีการศึกษาหาข้อมูลหลายอย่าง เช่น การหาไอเดียใหม่ๆ การสำรวจความสามารถของตัวเองรวมทั้งทักษะทางด้านการเขียนโปรแกรม และอุปกรณ์ที่ใช้ระบบปฏิบัติการ IOS ในปัจจุบันนั้นยังมีไม่หลากหลาย รวมทั้งราคาสูง

ดังนั้นผู้วิจัยจึงเลือกใช้ระบบปฏิบัติการ Android OS เนื่องจากเป็นระบบปฏิบัติการที่เป็น Open Source ติดตั้งได้กับสมาร์ทโฟนหรืออุปกรณ์ที่หลากหลาย สามารถทำงานร่วมกับ Hardware อย่างเป็นอิสระ และเป็นที่นิยมอย่างแพร่หลาย

2.1.7 โปรแกรมประยุกต์รองรับสมาร์ทโฟนหลายระบบปฏิบัติการ

โปรแกรมประยุกต์รองรับสมาร์ทโฟนหลายระบบปฏิบัติการ หรือ Mobile Application Cross Platform พัฒนาแอปพลิเคชันเพียงแค่ครั้งเดียว ด้วยภาษาทางคอมพิวเตอร์ 4 ชนิด ได้แก่ 1) Hypertext Markup Language version 5 หรือ HTML5 2) Visual C# 3) Cascading Style Sheet หรือ CSS และ 4) Javascript แต่สามารถรองรับการทำงานบนสมาร์ทโฟนได้หลายระบบปฏิบัติการ ซึ่งแอปพลิเคชันที่ได้ เรียกว่า “Hybrid Application” การพัฒนาหรือเรียกใช้ฟังก์ชันจะเรียกใช้ผ่านเครื่องมือ (Framework) ต่างๆ (Redda, 2012) Framework ที่นิยมใช้ในการพัฒนา Mobile Application Cross Platform ได้แก่

1) Apache Cordova เป็น Framework ที่ถูกพัฒนาโดย TASF (2014) ซึ่งใช้ภาษา HTML5, CSS3 และ Javascript โดยภาษา HTML5 ทำหน้าที่ในการแสดงผล CSS3 ทำหน้าที่ในการปรับแต่งส่วนติดต่อผู้ใช้ให้มีความสวยงามและน่าสนใจ และ Javascript สำหรับเรียกใช้ฟังก์ชันต่างๆ จากสมาร์ทโฟน Framework ของ Apache Cordova จะสามารถรวมชุดคำสั่งทั้งหมด แล้วส่งออกเป็นแอปพลิเคชันที่สามารถใช้งานได้บนระบบปฏิบัติการสมาร์ทโฟนต่างๆ ได้ทันที

Pimpler (2015) ได้กล่าวไว้ว่า ข้อดีของการใช้หลักการ Mobile Application Cross Platform นอกจากจะลดกระบวนการพัฒนาระบบที่ซ้ำซ้อนและช่วยประหยัดเวลาในการพัฒนาระบบแล้ว หากเลือกใช้เครื่องมือที่พัฒนาบนพื้นฐานของเทคโนโลยีเว็บ (Web Technology) จะเพิ่มความสะดวกในการนำแอปพลิเคชันไปประยุกต์ใช้กับระบบเทคโนโลยีสารสนเทศต่างๆ ได้ เช่น ระบบสารสนเทศภูมิศาสตร์ เป็นต้น

8) **Xamarin** เป็น Framework ที่ถูกพัฒนาโดย Xamarin Inc. (2015) สามารถพัฒนาแอปพลิเคชันบนสมาร์ทโฟนให้ทำงานได้บนระบบปฏิบัติการ Android และ IOS โดยการใช้ภาษา “Visual C#” ซึ่งเครื่องมือดังกล่าวมีการพัฒนามาจาก Framework ที่ชื่อ “Mono” และเปลี่ยนชื่อใหม่อย่างเป็นทางการว่า “Xamarin” ความสามารถของ Xamarin สามารถแปลงภาษาทางคอมพิวเตอร์ที่เขียนด้วย Visual C# ให้ทำงานได้เหมือนกับการพัฒนาแอปพลิเคชันด้วยภาษา Java และ Object-C แล้วส่งออกเป็นแอปพลิเคชันในระบบปฏิบัติการต่างๆ ได้

2.1.8 โปรแกรมประยุกต์สำหรับสร้างและบริหารจัดการเว็บไซต์

CMS (สำนักคอมพิวเตอร์ มหาวิทยาลัยทักษิณ, 2557) ย่อมาจาก Content Management System เป็นระบบที่ใช้ในการสร้างและบริหารจัดการข้อมูลของเว็บไซต์แบบสำเร็จรูป เช่น ข้อมูลตัวอักษร รูปภาพ หรือวีดีโอ ผู้ใช้ไม่จำเป็นต้องมีความรู้ในด้านการเขียนหรือพัฒนาเว็บไซต์ก็สามารถสร้างเว็บไซต์ได้

ข้อดีของ CMS

1) ผู้ใช้งานไม่จำเป็นต้องมีความรู้ด้านการสร้างเว็บไซต์ก็สามารถใช้งานได้
2) พัฒนาเว็บไซต์ได้อย่างรวดเร็ว
3) สามารถบริหารจัดการ และดูแลเว็บไซต์ได้ง่ายเพราะมีเครื่องมืออำนวยความสะดวกครบถ้วนอยู่แล้ว

ข้อเสียของ CMS

1) สำหรับผู้ที่ไม่ชำนาญด้านคอมพิวเตอร์ จะมีความยุ่งยากในการติดตั้งกับเว็บเซิร์ฟเวอร์
2) หากผู้ใช้ต้องการพัฒนาหน้าตาของเว็บ จะต้องมีความรู้พอสมควรเนื่องจาก CMS เป็นการนำระบบหลายๆ ระบบมารวมกัน

2.1.8.1 เปรียบเทียบข้อดีข้อเสียของ CMS

ตารางที่ 2-1 การเปรียบเทียบข้อดีข้อเสียของ CMS

ชื่อ CMS	**ข้อดี**	**ข้อเสีย**
WordPress	1. ใช้งานง่ายสำหรับมือใหม่ 2. มีลูกเล่นและ Module เสริมเยอะ 3. ใช้พื้นที่ในการติดตั้งน้อย	1. หากต้องการทำเว็บขนาดใหญ่มีเนื้อหาหลายชนิดอาจไม่รองรับ 2. การแก้ไข Template ทำได้ค่อนข้างยาก

ชื่อ CMS	ข้อดี	ข้อเสีย
Drupal	1. Module, Plugins ฟรีเกือบทั้งหมด 2. ระบบป้องกันความปลอดภัยค่อนข้างดี	1. มี Template ให้เลือกน้อย 2. ใช้งานค่อนข้างยากสำหรับผู้เริ่มใช้
Moodle	1. มีความยืดหยุ่นในการปรับแต่งแก้ไข 2. ใช้งานค่อนข้างง่ายสำหรับผู้เริ่มใช้ 3. รองรับมาตรฐาน e-learning เหมาะกับการทำสื่อแบบ e-learning	1. ยังมีผู้พัฒนาน้อย 2. มี Template ให้เลือกน้อย
October CMS	1. ใช้งานค่อนข้างง่ายและรวดเร็วในการสร้างเว็บไซต์ 2. มีเครื่องมืออำนวยความสะดวกครบถ้วน	1. หากต้องการทำเว็บขนาดใหญ่มีเนื้อหาหลายชนิดอาจไม่รองรับ 2. มี Template ให้เลือกน้อย

ทางผู้วิจัยเลือกใช้ October CMS ในการพัฒนาเว็บแอปพลิเคชันในส่วนของ Back-End เนื่องจากสามารถใช้งานได้ฟรี มีความสะดวกรวดเร็วในการพัฒนาระบบและง่ายต่อการพัฒนาโดยที่ผู้ใช้ไม่จำเป็นต้องมีความรู้ด้านการเขียนโปรแกรมมากนัก

2.2 งานวิจัยที่เกี่ยวข้อง

ในการทำวิจัยครั้งนี้ ผู้วิจัยได้ทำการศึกษางานวิจัยต่างๆที่เกี่ยวข้องกับการพัฒนาระบบแนะนำรายการอาหารอัตโนมัติตามบนสมาร์ทโฟนโดยใช้ข้อมูลเชิงตำแหน่ง รวมทั้งระบบแนะนำร้านอาหารในปัจจุบัน เพื่อให้เกิดความรู้ความเข้าใจและเป็นแนวทางในการพัฒนางานวิจัยดังนี้

2.2.1 ระบบแนะนำ (Recommender System)

1) ระบบให้คำแนะนำในการเลือกซื้อคอมพิวเตอร์โน๊ตบุ๊คด้วยเทคนิค Content-based Filtering

เปรมฤดี ผลชอบ และ สุพจน์ นิตย์สุวัฒน์ (เปรมฤดี ผลชอบ, 2553) งานวิจัยนี้ได้พัฒนาระบบและนำเสนอการนำเทคนิค Content-based Filtering มาประยุกต์ใช้ในการแนะนำ

การเลือกซื้อคอมพิวเตอร์โน๊ตบุ๊คเพื่อทำนายความคล้ายคลึงกันของคอมพิวเตอร์โน๊ตบุ๊คจากคุณลักษณะพื้นฐานของคอมพิวเตอร์โน๊ตบุ๊คที่มีอยู่ในระบบกับความต้องการของผู้ใช้

ผลจากการวิจัย ได้พัฒนาระบบแนะนำในการเลือกซื้อคอมพิวเตอร์โน๊ตบุ๊คด้วยเทคนิค Content-based Filtering ซึ่งวิธีนี้จะเน้นการหาความสัมพันธ์ของเนื้อหาเป็นหลัก เช่น คุณลักษณะของข้อมูล (Feature) ที่ผู้ใช้คนนั้นสนใจ โดยผู้ใช้จะต้องกรอกข้อมูลคุณลักษณะของคอมพิวเตอร์โน๊ตบุ๊คที่ต้องการพร้อมให้ค่าน้ำหนักความสำคัญของแต่ละอย่าง จากนั้นระบบจะแนะนำคอมพิวเตอร์โน้ตบุ๊คที่ใกล้เคียงกับความต้องการของผู้ใช้ ผลการประเมินความพึงพอใจของผู้ใช้อยู่ในระดับดี

จากงานวิจัยชิ้นนี้ สามารถนำหลักการที่ใช้ในการหาผู้ที่มีความชอบใกล้เคียงกัน (Neighbor) ไปประยุกต์ใช้เพื่อให้มีความเหมาะสมกับระบบแนะนำรายการอาหารให้ตรงกับความต้องการของผู้ใช้มากที่สุดได้

2) ระบบแนะนำภาพยนตร์ด้วยเทคนิคตัวกรองเชิงร่วมมือร่วมกับวิธีเคมีน (Movies Recommender System using Collaborative Filtering and K-Means)

ทัศนวรรณ แก้วใส และ สุพจน์ นิตย์สุวัฒน์ (ทัศนวรรณ แก้วใส, 2552) งานวิจัยนี้ได้พัฒนาระบบแนะนำภาพยนตร์โดยใช้เทคนิคตัวกรองเชิงร่วมมือ (Collaborative Filtering) ร่วมกับวิธีเคมีน (K-Mean) หรือการทำเหมืองข้อมูลเพื่อแบ่งกลุ่มข้อมูลก่อนใช้เทคนิคตัวกรองเชิงร่วมมือ เพื่อแนะนำภาพยนตร์ให้กับผู้ใช้งานระบบได้อย่างมีประสิทธิภาพ

ผลจากการวิจัย การพัฒนาระบบนะนำภาพยนตร์โดยใช้เทคนิคตัวกรองเชิงร่วมมือ (Collaborative Filtering) ร่วมกับวิธีเคมีน (K-Mean) หรือการทำเหมืองข้อมูลเพื่อแบ่งกลุ่มข้อมูลเป็นกลุ่มย่อยก่อนใช้เทคนิคตัวกรองเชิงร่วมมือ สามารถช่วยลดขนาดของข้อมูลได้ และสามารถแนะนำภาพยนตร์ให้กับผู้ใช้งานระบบได้รวดเร็ว และสร้างความพึงพอใจให้กับผู้ใช้งานต่อระบบอยู่ในระดับดี

จากงานวิจัยชิ้นนี้ สามารถนำหลักการและวิธีการไปประกอบการพิจารณาเพื่อประยุกต์ใช้ให้มีความเหมาะสมกับระบบแนะนำรายการอาหารตามความต้องการของผู้ใช้

2.2.2 ระบบบริการโดยใช้ข้อมูลเชิงตำแหน่ง

1) Cloud Based Application Development for Accessing Restaurant Information on Mobile Device using LBS

Keerthi S. Shetty and Sanjay Singh (Keerthi, 2009) ได้ศึกษาวิจัย พัฒนาแอพลิเคชัน สำหรับร้านอาหารบน Cloud โดยใช้ Location Based Service ผู้ใช้สามารถค้นหาร้านอาหารโดยใช้การป้อนข้อมูล Location แล้วทำการลงทะเบียนเพื่อเข้าถึงข้อมูลของร้านอาหารนั้นๆ โดยสามารถดูข้อมูลทั่วไปของร้านอาหาร เช่น ประเภทร้านอาหาร สถานที่ตั้ง เบอร์โทรศัพท์ โดยการพัฒนาบนระบบปฏิบัติการ Android บนสมาร์ทโฟน ด้วยภาษาจาวาสคริป

ผลของการวิจัย ได้พัฒนาแอพลิเคชันบนสมาร์ทโฟนที่ใช้สถานที่ Location ในการเข้าถึงข้อมูลของร้านอาหารได้ด้วยการระบุข้อมูลสถานที่ Location ผู้ใช้สามารถเข้าถึงบริการต่างๆ ได้โดยตรงเมื่อไหร่ก็ได้ตามความต้องการโดยไม่คำนึงถึงสถานที่ ณ ตอนนั้น การประมวลผลทำได้เร็ว และมีการนำความสามารถของ Cloud Computing มาประยุกต์ใช้ ซึ่งมีการจัดเก็บข้อมูลที่มีความปลอดภัยและความน่าเชื่อถือ

จะเห็นได้ว่า สามารถนำ Location-based Service ไปประยุกต์ใช้ให้เกิดประโยชน์ สามารถระบุพิกัดเพื่อแสดงตำแหน่งของร้านอาหารที่ได้จาการแนะนำของระบบบนสมาร์ทโฟนและประยุกต์ใช้ Cloud Computing สำหรับจัดเก็บข้อมูลที่มีความปลอดภัยมากยิ่งขึ้น และรองรับแพลตฟอร์มที่หลากหลาย

2) การประยุกต์ใช้ Google Maps API ในการเฝ้าระวังปัญหายาเสพติด

ปฐมพงษ์ ฉับพลัน และฐิมาพร เพชรแก้ว (ปฐมพงษ์ ฉับพลัน, 2553) ได้วิจัยและพัฒนา Web-based Application โดยการประยุกต์ใช้ Google Maps API ในการเฝ้าระวังปัญหายาเสพติดสำหรับสถานีตำรวจเพื่อใช้ในการเฝ้าระวัง ติดตามและจับกุมผู้ต้องหา ระบบสามารถกำหนดพื้นที่เสี่ยง ผู้ค้า ผู้เสพ สถานที่ค้า และจุดตรวจบนแผนที่ Google Maps ได้ และยังสามารถแสดงรายงานการจับกุม ผลการจับกุม กราฟสถิติการจับกุม และเปรียบเทียบการจับกุมได้ ระบบถูกพัฒนาขึ้นด้วยภาษา PHP V.5 ใช้ฐานข้อมูล MySQL V.5.0.22

ผลของการวิจัย ระบบนี้สามารถจัดเก็บข้อมูลต่างๆ เช่น ตำบล พื้นที่เสี่ยง ผู้ค้า ผู้เสพ สถานที่เสี่ยงและจุดตรวจ โดยแสดงบน Google Maps ทำให้ผู้ใช้สามารถมองเห็นภาพชัดเจน

มากขึ้น และมีประสิทธิภาพในการจับกุมผู้ต้องหา สามารถแสดงรายงานสถิติการจับกุมแต่ละประเภทในแต่ละเดือนได้ แสดงให้เห็นแนวโน้มของปัญหายาเสพติด

จะเห็นได้ว่า สามารถนำ Google Maps API ไปประยุกต์ใช้ให้เกิดประโยชน์ต่างๆ และมีประสิทธิภาพมากยิ่งขึ้น สามารถนำไปประยุกต์ใช้กับการค้นหาร้านอาหารบนสมาร์ทโฟนโดยการแสดงบนแผนที่ Google Maps เพื่อให้ผู้ใช้มองเห็นภาพได้ชัดเจนมากยิ่งขึ้น

3) On the Design of a LBS-based App for Searching Vegetarian Restaurant

You Geeng Neng, Chang Ching Jung and You Hsiao Chen (You Geeng Neng, 2012) ได้ออกแบบและพัฒนา LBS-based App สำหรับค้นหาร้านอาหารมังสวิรัติด้วยสมาร์ทโฟน เพื่อให้ผู้ใช้สามารถค้นหาร้านอาหารมังสวิรัติตามสถานที่ตั้งของผู้ใช้ โดยอาศัยฐานข้อมูลมังสวิรัติของภูมิภาค เนื่องจากร้านอาหารมังสวิรัติกำลังเป็นที่นิยมและร้านมีจำนวนน้อย ค้นหาได้ยาก

ผลของการวิจัย ได้ออกแบบพัฒนาและทดสอบเว็บต้นแบบบนสมาร์ทโฟนสำหรับค้นหาร้านอาหารมังสวิรัติ สามารถใช้เป็นต้นแบบในการพัฒนาและต่อยอดแอพลิเคชันการค้นหาร้านอาหารบนสมาร์ทโฟนโดยการใช้ LBS-based Service ต่อไป ผลประเมินและข้อเสนอแนะอยู่ในเชิงบวก

จากงานวิจัยฉบับนี้ ผู้วิจัยสามารถใช้งานวิจัยดังกล่าวเป็นต้นแบบในส่วนของการใช้ LBS-based Service สำหรับค้นหาร้านอาหาร และพัฒนาต่อยอดเพื่อให้มีความเหมาะสมมากยิ่งขึ้น

2.2.3 ระบบบนสมาร์ทโฟน

1) Have2eat: a Restaurant Finder with Review Summarization for Mobile Phones

Giuseppe Di Fabbrizio and Narendra Gupta (Giuseppe, 2010) ได้ศึกษาวิจัย Have2eat ซึ่งเป็น แอปพลิเคชันบนสมาร์ทโฟนที่ได้รับความนิยมสำหรับไอโฟนและระบบปฏิบัติการ Android ช่วยให้ผู้ใช้สามารถค้นหาและเข้าถึงร้านอาหารที่อยู่บริเวณใกล้ๆ โดยจะแสดงร้านอาหารทั้งหมดที่อยู่รอบๆ บริเวณนั้น

ผลของการวิจัย แอพลิเคชันบนสมาร์ทโฟนสามารถค้นหาร้านอาหารได้ และสามารถจัดคะแนนความชอบร้านอาหารได้จากการรีวิวของลูกค้า ผู้ใช้สามารถแสดงข้อความคิดเห็นที่มีต่อร้านอาหารได้โดยตรง โดยการพิมพ์ข้อความแสดงความคิดเห็นหรือกดปุ่มที่รูปมือได้

จะเห็นได้ว่า ระบบสามารถค้นหาและแสดงข้อมูลของร้านอาหาร ผู้ใช้สามารถแสดงความคิดเห็น และใช้เทคนิคในการจัดอันดับได้จากความคิดเห็นของลูกค้า สามารถนำไปประยุกต์ใช้ให้เกิดประโยชน์มากยิ่งขึ้น

2) Android Application of Restaurant Finder

Spoorthy Reddy Parne (Spoorthy Reddy Parne, 2010) ได้พัฒนาแอปพลิเคชันบน Android สำหรับค้นหาร้านอาหารโดยใช้ Location ผู้ใช้สามารถค้นหาร้านอาหารโดยการระบุข้อมูลสถานที่เพื่อค้นหาร้านอาหารบริเวณใกล้เคียงสถานที่นั้นๆ และสามารถแสดงข้อคิดเห็นที่มีต่อร้านอาหาร และสามารถอ่านข้อคิดเห็นที่ลูกค้าคนอื่นๆ ได้แสดงข้อคิดเห็นไว้ได้

ผลของการวิจัย ระบบสามารถแสดงร้านอาหารยอดนิยมสูงสุด 5 อันดับแรก บนแผนที่ Google Maps ได้ ผู้ใช้สามารถค้นหาร้านอาหารอย่างง่ายโดยการระบุตำแหน่ง เช่น เมือง ในการค้นหาร้านอาหารหรือสามารถค้นหาโดยการระบุเงื่อนไขอื่นๆ เพิ่มเติมได้ เช่น ประเภทร้านอาหาร ระดับความนิยมของร้านอาหาร เพื่อค้นหาร้านอาหารที่อยู่บริเวณใกล้เคียง และสามารถแสดงข้อคิดเห็นที่มีต่อร้านอาหาร สามารถอ่านข้อคิดเห็นที่ลูกค้าคนอื่นๆ ได้แสดงข้อคิดเห็นไว้ได้

จากงานวิจัยฉบับนี้ ผู้วิจัยสามารถนำฟังก์ชันการทำงานของระบบไปประยุกต์ใช้และต่อยอดได้ ให้มีความเหมาะสม ใช้งานได้สะดวกมากยิ่งขึ้น เช่น ฟังก์ชันการค้นหาร้านอาหารบริเวณใกล้เคียง การแสดงข้อคิดเห็นและการจัดระดับความนิยมที่มีต่อร้านอาหาร ที่ลูกค้ามีต่อร้านอาหาร

3) Digital Ordering System for Restaurant Using Android

Ashutosh Bhargave, Niranjan Jadhav, Apurva Joshi, Prachi Oke and Prof. Mr. S. R Lahane (Ashutosh Bhargave, 2013) ระบบสั่งอาหารใช้สำหรับ แท็บเล็ต พีซี พัฒนาขึ้นเพื่อช่วยในการจัดการร้านอาหารให้มีความสะดวก โดยลูกค้าสามารถสั่งอาหารได้ด้วยตัวเองเพื่อลดเวลาในการรอรับบริการ

ผลของการวิจัย ระบบสามารถสั่งอาหารได้ โดยลูกค้าสามารถสั่งอาหารด้วยตัวเองผ่านทางแท็บเล็ต สามารถช่วยลดเวลาในการรอรับบริการได้

จะเห็นได้ว่าระบบถูกพัฒนาขึ้นเฉพาะเจาะจงสำหรับร้านอาหารร้านใดร้านหนึ่งหรือร้านอาหารขนาดใหญ่ และแอปพลิเคชันบนสมาร์ทโฟนยังมีข้อจำกัดในเรื่องของประมวลผล และหน่วยความจำที่จำกัด ดังนั้นผู้วิจัยสามารถสามารถนำไปประกอบการพิจารณาในการปรับเปลี่ยนเพื่อหาวิธีในการแก้ปัญหาดังกล่าว ให้มีความเหมาะสมมากยิ่งขึ้น

4) e-Restaurant: Online Restaurant Management System for Android

Vivek Ranjan, Nikesh Masiwal and Nikita Verma (Vivek Ranjan, 2013) การสั่งซื้ออาหารในร้านอาหารด้วยแท็บเล็ต ใช้ฐานข้อมูลและServer ที่อยู่บน Cloud ซึ่งช่วยลดค่าใช้จ่ายและมีความปลอดภัย

ผลของการวิจัย ลูกค้าสามารถสั่งอาหารโดยใช้แท็บเล็ตได้ ซึ่งมีความสะดวกและรวดเร็วขึ้น และ อีกทั้งการใช้ฐานข้อมูลที่อยู่บน Cloud ทำให้ระบบมีความน่าเชื่อถือในเรื่องความปลอดภัยของระบบ

ดังนั้นผู้วิจัยสามารถสามารถนำไปประกอบการพิจารณาในการประยุกต์ใช้และพัฒนาให้มีความเหมาะสมมากยิ่งขึ้น

2.2.4 ระบบแนะนำร้านอาหารบนสมาร์ทโฟนในปัจจุบัน

1) ZAGAT (ZAGAT, 2556) แอปพลิเคชันแนะนำร้านอาหารในประเทศสหรัฐอเมริกา มีระบบค้นหาและแนะนำร้านอาหาร ทั้ง Web Site และ Mobile Application มีการให้คะแนนร้านอาหารใน 4 หัวข้อดังนี้ รสชาติ การตกแต่งร้าน การบริการ และราคา มีระดับการให้คะแนนดังนี้

26 – 30 พิเศษ-ดีเลิศ

21 – 25 ดีมาก-ดีเยี่ยม

16 – 20 ดี – ดีมาก

11 – 15 พอใช้-ดี

0 – 10 แย่- พอใช้

ระบบจะนำคะแนนมาประมวลผลและจัดทำหนังสือรับรองเพื่อการค้า ระบบมีข้อมูลรายละเอียดร้านอาหาร มีการให้ Rating ร้านอาหาร มีรีวิวร้านอาหารแต่ละร้าน แนะนำร้านอาหารตามที่ตั้งของร้านอาหาร ประเภทอาหาร สัญชาติอาหาร คะแนนร้านอาหาร ชื่อร้านอาหาร และราคาอาหาร

2) Wongnai (Wongnai, 2557) Application Wongnai รองรับระบบปฏิบัติการ Android OS จะช่วยค้นหาร้านอาหาร มีฟังก์ชันการทำงานดังนี้ ค้นหาร้านอาหารและร้านอื่นๆ ดูข้อมูลร้านอาหารและร้านอื่นๆ โพสและรีวิวรูปภาพของร้านอาหารเพื่อแบ่งปันให้สมาชิกคนอื่นๆ บันทึกร้านอาหารที่ชอบเพื่อเก็บไว้และเข้ามาดูในภายหลังได้สะดวก

3) Kinraidee (Thaiware, 2556) Application Kinraidee รองรับระบบปฏิบัติการ IOS จะช่วยแนะนำร้านอาหารเพื่อให้ผู้บริโภคเลือกดูเป็นแนวทางในการเลือก แบ่งเป็น 2 ส่วนคือ Food Delivery รวมเบอร์ร้านอาหาร Delivery สำหรับสั่งอหารมาเพื่อรับประทานที่บ้าน และ My Favorite สามารถเพิ่ม ข้อมูล ร้านอาหาร ที่คุณชื่นชอบไว้ใน List หรือจดบันทึกข้อความส่วนตัว เพื่อจดจำข้อมูลของร้านนั้นๆ เช่นเวลา เปิด-ปิด รายการอาหารที่ชื่นชอบ

4) Where To Eat (OnDemandWold, 2557) Application Where To Eat รองรับระบบปฏิบัติการ IOS ค้นหาร้านอาหารตามชนิดของอาหาร โดยใช้แผนที่ GPS สามารถค้นหาร้านอาหารได้

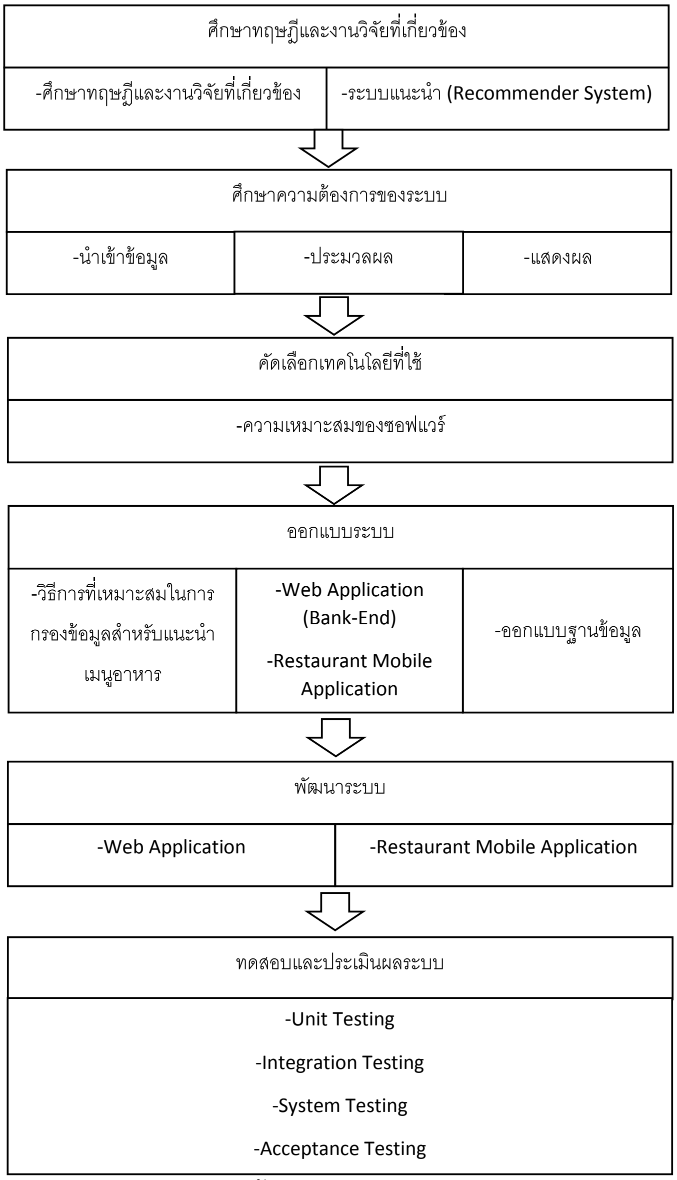
5) Yelp (Google Play, 2557) Application Yelp รองรับระบบปฏิบัติการ Android ช่วยในการค้นหาสถานที่กิน ร้านค้า เครื่องดื่ม โดยการเลือกหัวข้อที่ต้องการค้นหา ระบบจะแสดงรายการต่างๆ ตามสถานที่ตั้งของผู้ใช้บนแผนที่ สามารถดูข้อมูล เบอร์โทรและรายเอียดอื่นๆ ของร้านอาหาร รวมทั้งดูรีวิวร้านอาหารได้

แต่แอปพลิเคชันค้นหาร้านอาหารและแนะนำร้านอาหารในปัจจุบันทั้งหมดที่กล่าวมาข้างต้น ยังเป็นระบบค้นหาร้านอาหารสำหรับผู้บริโภคโดยทั่วไป ซึ่งจะค้นหาจากการเลือกประเภทอาหาร ประเภทร้าน สถานที่ตั้ง หรือราคา อย่างใดอย่างหนึ่ง ซึ่งระบบจะค้นหาออกมาทั้งหมดโดยไม่มีการคิดวิเคราะห์ความต้องการของผู้บริโภคร่วมด้วย ผู้บริโภคไม่สามารถระบุรายละเอียดความชอบส่วนตัวเกี่ยวกับรายการอาหารหรือร้านอาหารได้ ทำให้ไม่สามารถแนะนำตามความชอบส่วนตัวของผู้บริโภคแต่ละคนได้ ดังนั้นผู้วิจัยจึงต้องการพัฒนาระบบแนะนำร้านอาหารที่ผู้บริโภคนอกจากจะสามารถเลือกค้นหาร้านอาหารตามประเภทอาหาร ราคา สถานที่ตั้ง หรือหัวข้ออื่นๆ ได้แล้ว ระบบสามารถแนะนำรายการอาหารและร้านอาหารได้ตรงตามความต้องการ ตามความชอบส่วนตัว ของผู้ใช้แต่ละคนซึ่งอาจจะมีความชอบรายการอาหารหรือร้านอาหารที่ไม่เหมือนกันได้ และผู้บริโภคสามารถระบุรายละเอียดความต้องการ ความชอบส่วนตัว เกี่ยวกับรายการอาหารและร้านอาหาร

บทที่ 3
วิธีการดำเนินวิจัย

3.1 ภาพรวมการวิจัย

การวิจัยครั้งนี้เป็นการวิจัยและพัฒนา (Research and Development) เพื่อพัฒนาระบบแนะนำร้านอาหารอัตโนมัติบนสมาร์ทโฟนโดยใช้ข้อมูลเชิงตำแหน่งและรายการอาหาร ในบทนี้จะขอกล่าวถึงขั้นตอนในการวิจัยและพัฒนา โดยมีขั้นตอนการดำเนินงานดังแสดงในรูปที่ 3-1



รูปที่ 3-1 ขั้นตอนการดำเนินการวิจัย

3.2 การศึกษาบทความและงานวิจัยที่เกี่ยวข้อง

ผู้วิจัยได้ทำการศึกษาทฤษฎีและศึกษางานวิจัยที่เกี่ยวข้อง ซึ่งได้กล่าวไว้ในบทที่ 2 โดยได้ศึกษาจากเอกสารตามหัวข้อดังต่อไปนี้

1) เอกสารและงานวิจัยที่เกี่ยวข้องกับระบบแนะนำ (Recommender System)

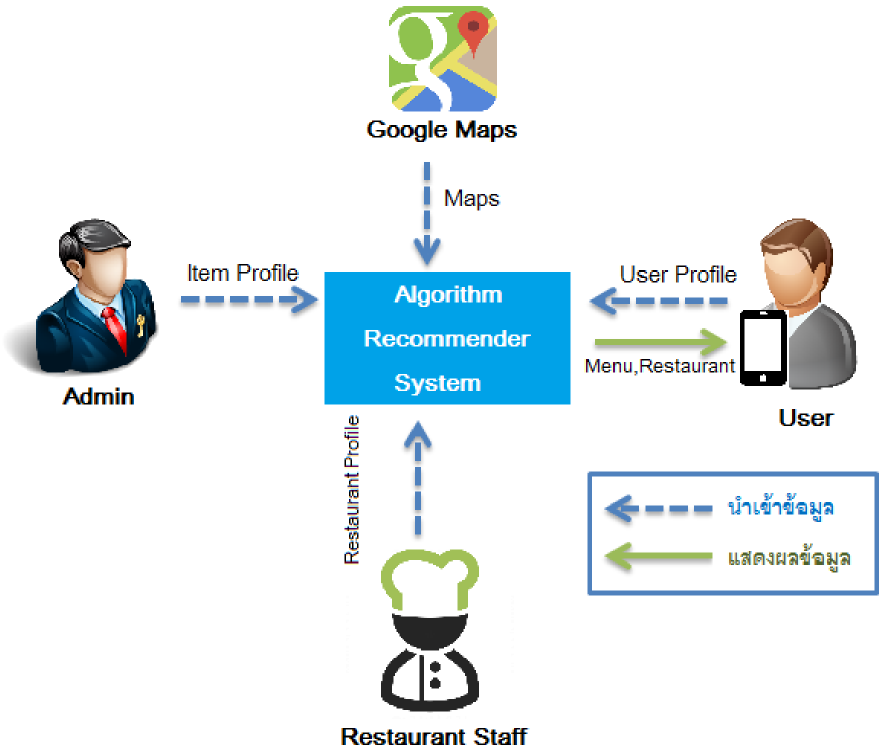
2) เอกสารและงานวิจัยที่เกี่ยวข้องกับระบบร้านอาหารบนเว็บไซต์ และร้านอาหารบนอุปกรณ์เคลื่อนที่แบบพพา

3) เอกสารและงานวิจัยที่เกี่ยวข้องกับระบบข้อมูลเชิงตำแหน่ง (Googles Map API)

4) เอกสารและงานวิจัยที่เกี่ยวข้องกับ Cloud และ Web Service

3.3 การศึกษาความต้องการของระบบ

ผู้วิจัยได้ศึกษาข้อมูลที่นำไปใช้ในการแนะนำรายการอาหารและร้านอาหาร โดยแบ่งออกเป็น 2 ส่วนคือ 1) ข้อมูลรายละเอียดของร้านอาหาร (Restaurant Profile) 2) ข้อมูลรายการอาหารและส่วนประกอบของรายการอาหาร (Item Profile) หลังจากนั้นจึงได้ออกแบบวิธีการกรองข้อมูลเพื่อหาค่าความคล้ายคลึงที่เหมาะสมและแนวคิดในการทำงานของระบบโดยแบ่งเป็น 3 ส่วนดังรูปที่ 3-2 ได้แก่ 1) การนำเข้าข้อมูล ได้แก่ ข้อมูลรายการอาหารตั้งต้น (Item Profile) ข้อมูลความต้องการด้านรายการอาหารและร้านอาหาร (User Profile) ข้อมูลร้านอาหารและตำแหน่งที่ตั้ง (Restaurant Profile) ข้อมูลแผนที่ (Maps) 2) Algorithm Recommender System สำหรับการประมวลผลหาค่าความคล้ายคลึงระหว่างข้อมูลตั้งต้น (Item Profile) กับความต้องการของผู้ใช้ (Active User) 3) ส่วนแสดงผลข้อมูลสำหรับติดต่อระหว่างระบบกับผู้ใช้งาน



รูปที่ 3-2 แนวคิดการทำงานของระบบส่วนแนะนำรายการอาหารและร้านอาหาร

3.3.1 การนำข้อมูลเข้า

3.3.1.1 ทำการสร้างข้อมูลรายการอาหาร (Item Profile) โดยแบ่ง Multicriteria ได้ 6 ประเภท ดังต่อไปนี้

ผู้วิจัยได้ทำการรวบรวมข้อมูลของร้านอาหารในอำเภอหาดใหญ่ จังหวัดสงขลา จำนวน 60 ร้าน และเก็บรวบรวมรายการอาหารจากร้านดังกล่าว จำนวนทั้งหมด 160 รายการ

1) ประเภทของร้านอาหาร แบ่งตามประเภทและลักษณะของอาหารที่แต่ละร้านให้บริการ จากการรวบรวมข้อมูลร้านอาหารในอำเภอหาดใหญ่จำนวน 60 ร้านนั้น สามารถแบ่งเป็น 9 fields ดังนี้

- ร้านอาหารไทย
- ร้านอาหารเช้า
- ร้านอาหารเพื่อสุขภาพและมังสวิรัติ
- เครื่องดื่มชา-กาแฟ เค้กและเบเกอรี่
- อาหารทะเล บุฟเฟ่-หมูกะทะ
- ขนมจีน
- สเต๊ก
- ก๋วยเตี๋ยว
- ไอศกรีม-ขนมหวาน

2) ส่วนประกอบ (เนื้อ) โดยรวบรวมประเภทของเนื้อจากรายการอาหารของร้านอาหารในอำเภอหาดใหญ่ จำนวน 160 รายการ ซึ่งแบ่งเป็น 5 fields ดังนี้

- เนื้อไก่
- เนื้อวัว
- เนื้อหมู
- เนื้อปลา
- กุ้ง
- ปู
- ปลาหมึก
- อาหารทะเล
- ไข่

3) ส่วนประกอบ (ผัก) โดยรวบรวมประเภทของผักจากรายการอาหารของร้านอาหารในอำเภอหาดใหญ่ จำนวน 160 รายการ ซึ่งแบ่งเป็น 33 fields ดังนี้

- มะเขือเทศ
- ขิง
- น้ำเต้า
- กะหล่ำปลี
- หน่อไม้ฝรั่ง
- เห็ด
- แตงกวา
- แครอท
- ผักคะน้า
- ใบชะพลู
- ผักสลัด
- ยอดมะพร้าวอ่อน
- ดอกขจร
- มะม่วงหิมพานต์
- ผักหวาน
- บวบ
- บล็อกเคอรี่
- ผักกาด
- หน่อไม้
- ฟักทอง
- ข้าวโพดอ่อน
- ถั่วลันเตา
- ถั่ว
- ถั่วฝักยาว
- หัวผัดกาด
- ผักบุ้ง
- ใบชะมวง
- ชะอม
- มะระ
- สะตอ
- เผือก
- ตำลึง
- ผักเหรียง

4) รสชาติ สามารถแยกแยะรสชาติหลักๆ ของรายการอาหารจากร้านอาหารในอำเภอหาดใหญ่ จำนวน 160 รายการ ซึ่งแบ่งเป็น 4 fields ดังนี้

- เผ็ด
- เปรี้ยว
- หวาน
- เค็ม

5) ประเภทการปรุงอาหาร สามารถแบ่งประเภทการปรุงอาหารของรายการอาหารจากร้านอาหารในอำเภอหาดใหญ่ จังหวัดสงขลา จำนวน 160 รายการ ซึ่งแบ่งเป็น 3 fields ดังนี้

- แกง-ต้ม
- ผัด-ทอด
- ตุ๋น-นึ่ง

6) ราคาอาหาร จากการรวบรวมรายการอาหารของร้านอาหารในอำเภอหาดใหญ่ จำนวน 160 รายการ สามารถแบ่งช่วงราคาเป็น 5 fields ดังนี้

- น้อยกว่า 100 บาท
- 100-199 บาท
- 200-299 บาท
- 300-399 บาท
- มากกว่า 400 บาท

3.3.1.2 ข้อมูลรายละเอียดของร้านอาหาร (Restaurants Profile)

1) ระยะทาง ซึ่งแบ่งเป็น 4 fields ดังต่อไปนี้

- น้อยกว่า 1 Km
- น้อยกว่า 2 Km
- น้อยกว่า 3 Km
- น้อยกว่า 5 Km
- น้อยกว่า 10 Km
- น้อยกว่า 20 Km

2) ระดับดาว ซึ่งแบ่งเป็น 5 fields ดังต่อไปนี้

- 5 ดาว
- 4 ดาว
- 3 ดาว
- 2 ดาว
- 1 ดาว

3) บรรยากาศภายในร้านอาหาร ซึ่งแบ่งเป็น 2 fields ดังต่อไปนี้

- ไม่มีเครื่องปรับอากาศ (Open)
- มีเครื่องปรับอากาศ (Close)

4) มีเครื่องหมายฮาลาล ซึ่งแบ่งเป็น 2 fields ดังต่อไปนี้

- มีเครื่องหมายฮาลาล
- ไม่มีเครื่องหมายฮาลาล

5) ลานจอดรถของร้านอาหาร ซึ่งแบ่งเป็น 2 fields ดังต่อไปนี้

- มีลานจอดรถ
- ไม่มีลาดจอดรถ

3.3.1.3 ข้อมูลความต้องการของผู้ใช้ (User Profile) ทำการเก็บข้อมูลจากการตั้งค่า Profile และการเก็บคะแนนความชอบ โดยการแยกออกเป็นสองส่วนดังต่อไปนี้

1) ความต้องการด้านรายการอาหาร ตัวอย่างเช่น

- ส่วนประกอบที่เป็นเนื้อ ยกตัวอย่างเช่น ผู้บริโภคชอบเนื้อไก่ และ ซีฟู๊ด มากที่สุด
- ส่วนประกอบที่เป็นผัก ยกตัวอย่างเช่น ผู้บริโภคชอบหน่อไม้ และ ผักบุ้ง มากที่สุด
- รสชาติ ยกตัวอย่างเช่น ผู้บริโภคชอบรสชาติเผ็ดมากที่สุด
- ประเภทอาหาร ยกตัวอย่างเช่น ผู้บริโภคเลือก ประเภทอาหาร แกง-ต้ม
- ราคา ได้แก่ ผู้บริโภคเลือกราคาน้อยกว่า 100

2) ความต้องการด้านร้านอาหาร ตัวอย่างเช่น

- ระยะทางน้อยกว่า 2 Km
- ระดับดาว ได้แก่ 5 ดาว
- บรรยากาศแบบมีเครื่องปรับอากาศ (Close)
- มีเครื่องหมายฮาลาล
- มีลานจอดรถ

3.3.1.4 การกำหนดคะแนนความชอบให้กับรายการอาหารตั้งต้น (Item Profile)

1) แยกแยะส่วนผสมของรายการอาหาร ตามสัดส่วนของส่วนผสมในการปรุงอาหารจากสูตรอาหารของรายการอาหารนั้นๆ (NLoveCooking.com, 2012) โดยแบ่งเป็น 3 ส่วน คือ

- ส่วนของเนื้อสัตว์ ซึ่งนำมาพิจารณาในส่วนของ ส่วนประกอบที่เป็นเนื้อ โดยแยกเป็นส่วนประกอบหลัก และส่วนประกอบรอง ตามอัตราส่วนจากมากไปหาน้อย
- ส่วนของผัก ซึ่งนำมาพิจารณาในส่วนของ ส่วนประกอบที่เป็นผัก โดยแยกเป็นส่วนประกอบหลัก และส่วนประกอบรอง ตามอัตราส่วนจากมากไปหาน้อย
- ส่วนเครื่องปรุง ซึ่งนำมาพิจารณาในส่วนของรสชาติ โดยเลือกเฉพาะส่วนผสมที่ให้รสชาติหลักเท่านั้น โดยแบ่งกลุ่มของ รสชาติออกเป็น 4 กลุ่ม ดังนี้ ได้แก่ เผ็ด หวาน เปรี้ยว เค็ม

สูตรอาหารที่นำมาใช้ในงานวิจัยนี้ มาจากหนังสือ อาหารไทยครบทุกภาค (สำนักพิมพ์แม่บ้าน, 2540) และ หนังสือเชฟอาหารไทย (พงษ์ศักดิ์ มิขุนทอง และ สุรศักดิ์ เจริญจันทร์ , 2540)

ตารางที่ 3-1 แยกแยะส่วนของเครื่องปรุงอาหารและการให้รสชาติ

ลำดับ	**เครื่องปรุง**	**รสชาติ**
1	พริก, พริกไทย, ขิง, ข่า, ผักและผลไม้ให้รสเผ็ด อื่นๆ	เผ็ด
2	น้ำตาล, วัตถุให้ความหวานใช้แทนน้ำตาล อื่นๆ	หวาน
3	น้ำส้มสายชู, ผักและผลไม้รสเปรี้ยว, ซอสปรุงรสให้รสเปรี้ยวต่างๆ อื่นๆ	เปรี้ยว

ลำดับ	เครื่องปรุง	รสชาติ
4.	เกลือ, น้ำปลา, กะปิ, ซอสปรุงรสให้รสเค็มต่างๆ อื่นๆ	เค็ม

2) กำหนดคะแนนความชอบให้กับส่วนประกอบของรายการอาหาร ยกตัวอย่างเช่น รายการอาหารแกงคั่วกะทิหอยแครง โดยมีส่วนผสมแกงคั่วกะทิหอยแครง ดังนี้

- พริกแห้ง ซอย 3 ช้อนโต้ะ
- พริกไทย 2 ช้อนโต้ะ
- ข่า ซอย 2 ช้อนโต้ะ
- หอมแดง ซอย 2 ช้อนโต้ะ
- ตะไคร้ ซอย 2 ช้อนโต้ะ
- ขมิ้น ซอย 2 ช้อนชา
- ผิวมะกรูด ซอย 1 ช้อนชา
- กระเทียม ซอย 2 ช้อนโต้ะ
- กะปิ 2 ช้อนโต้ะ
- หอยแครง 2 ถ้วย
- หัวกะทิ 1 ถ้วยตวง
- หางกะทิ 1 ถ้วยตวง
- น้ำตาลปี๊บ 2 ช้อนโต้ะ
- น้ำปลา 2 ช้อนโต้ะ
- ผงยี่หร่า 1 ช้อนชา
- ใบชะพลูซอย 4 ช้อนโต้ะ

ตารางที่ 3-2 แยกแยะส่วนผสมของรายการอาหาร แกงคั่วกะทิหอยแครง

ประเภท	ส่วนประกอบหลัก (คะแนนความชอบ=5)	ส่วนประกอบรอง (คะแนนความชอบ=4)	ส่วนประกอบรอง (คะแนนความชอบ=3)	ส่วนประกอบรอง (คะแนนความชอบ=2)	ส่วนประกอบรอง (คะแนนความชอบ=1)
เนื้อสัตว์	หอยแครง	-	-	-	-
ผัก	ใบชะพลู	-	-	-	-
รสชาติ	รสชาติเผ็ด ได้แก่ พริกแห้ง, พริกไทย,พริกหยวก, ข่า, ผงยี่หร่า	-	-	-	-

3) กำหนดค่าน้ำหนักให้กับ 3 ส่วนประกอบของรายการอาหารตั้งต้น (Item Profile) ได้แก่

- ส่วนของเนื้อสัตว์ เป็นส่วนประกอบหลักในรายการอาหาร ซึ่งพิจารณาเป็นหลัก กำหนดให้ค่าน้ำหนักเท่ากับ 5 คือมากที่สุด
- ส่วนของผัก เป็นส่วนประกอบรองในรายการอาหาร ซึ่งพิจารณาเป็นรองลงมา กำหนดให้ค่าน้ำหนักเท่ากับ 4 คือ มาก
- ส่วนของรสชาติ เป็นส่วนประกอบรองลงมาในรายการอาหาร ซึ่งพิจารณาเป็นรองลงมา กำหนดให้ค่าน้ำหนักเท่ากับ 3 คือ ปานกลาง

3.3.2 การประมวลผลข้อมูล

3.3.2.1 ขั้นตอนกระบวนการหาค่าความคล้ายคลึง

เมื่อผู้ใช้ทำการบันทึกข้อมูลความต้องการด้านรายการอาหารและความต้องการด้านร้านอาหาร (User Profile) เรียบร้อยแล้ว ยกตัวอย่างเช่น

1) ผู้ใช้ตั้งค่า User Profile โดยกำหนดค่าคะแนนความชอบในรายละเอียดย่อยของ Item ต่างๆ กำหนดสเกลเป็นลำดับเลข 0-5 (เรียงลำดับตามความชอบจากน้อยไปหามาก) ดังตัวอย่าง รูปที่ 3-3 จะอธิบายแค่เพียง 3 Criteria เพื่อง่ายในการอธิบาย

ชอบรับประทานไก่และซีฟู๊ดมากที่สุดให้ค่าน้ำหนักเท่ากับ 5 ชอบรับประทานหมูรองลงมาให้ค่าน้ำหนักเท่ากับ 2

ชอบรับประทานหน่อไม้มากที่สุดให้ค่าน้ำหนักเท่ากับ 5 ชอบรับประทานผักบุ้งและผักคะน้าน้อยลงมาให้ค่าน้ำหนักเท่ากับ 4 และ 3 ตามลำดับ

ชอบรับประทานรสชาติเผ็ดมาก ให้ค่าน้ำหนักเท่ากับ 4 ชอบรับประทานรสชาติหวานและเปรี้ยวน้อยลงมาให้ค่าน้ำหนักเท่ากับ 3 และ 1 ตามลำดับ

ส่วนประกอบ (เนื้อ)

ไก่	หมู	ซีฟู๊ด	...
5	2	5	...

ส่วนประกอบ (ผัก)

คะน้า	ผักบุ้ง	หน่อไม้	...
3	4	5	...

รสชาติ

เผ็ด	เปรี้ยว	หวาน
4	1	3

อื่นๆ

...
...

รูปที่ 3-3 การกำหนดค่าคะแนนความชอบรายละเอียดย่อยของ Item ต่างๆ

2) ทำการ Normalize ให้อยู่ในรูป 0-1 (หารด้วย 5) จากรูปที่ 3-4 ไก่มีค่าคะแนนความชอบเท่ากับ 5 เมื่อหารด้วย 5 จะได้ผลลัพธ์เท่ากับ 1 หมูมีค่าคะแนนความชอบเท่ากับ 2 เมื่อหารด้วย 5 จะไดผลลัพธ์เท่ากับ 0.4 ซีฟู๊ดมีค่าคะแนนความชอบเท่ากับ 5 เมื่อหารด้วย 5 จะได้ผลลัพธ์เท่ากับ 1 ทำเช่นเดียวกันทุกๆ Item จนครบทุก Criteria

ส่วนประกอบ (เนื้อ)

ไก่	หมู	ซีฟู๊ด	...
1	0.4	1	...

ส่วนประกอบ (ผัก)

คะน้า	ผักบุ้ง	หน่อไม้	...
0.6	0.8	1	...

รสชาติ

เผ็ด	เปรี้ยว	หวาน
0.8	0.2	0.6

อื่นๆ

...
...

รูปที่ 3-4 การ Normalization

3) หาค่าน้าหนักของแต่ละ Criteria โดยพิจารณาตำแหน่งที่มีค่าสูงสุดในแต่ละ Criteria จากนั้นคำนวนหาค่าน้ำหนักโดยวิธีการเทียบกับตำแหน่งที่มีค่าสูงสุดของ Criteria อื่น ดังสมการ

$$\text{ค่าน้ำหนักส่วนประกอบเนื้อ} = \frac{1}{1+1+0.8} = 0.36$$

$$\text{ค่าน้ำหนักส่วนประกอบผัก} = \frac{1}{1+1+0.8} = 0.36$$

$$\text{ค่าน้ำหนักรสชาติ} = \frac{0.8}{1+1+0.8} = 0.29$$

หลังจากนั้นเมื่อได้ค่าน้ำหนักของแต่ละ Criteria แล้ว นำค่าน้ำหนักมาคูณกับทุกค่าในแต่ละ criteria เพื่อถ่วงน้ำหนักให้มีความเหมาะสม เช่น ค่าน้ำหนักส่วนประกอบเนื้อเท่ากับ 0.36 จะนำไปคูณในแต่ละ Fields ของส่วนประกอบเนื้อ ค่าน้ำหนักส่วนประกอบผักเท่ากับ 0.36 จะนำไปคูณในแต่ละ Fields ของส่วนประกอบผัก และค่าน้ำหนักรสชาติเท่ากับ 0.29 จะนำไปคูณในแต่ละ Fields ของกลุ่มรสชาติ จากรูปที่ 3-5 ไก่ค่าเดิมคือ 1 เมื่อนำไปคูณกับค่าน้ำหนักคือ 0.36 จะได้ผลลัพธ์เท่ากับ 0.36 หมูค่าเดิมคือ 0.4 เมื่อไปคูณกับค่าน้ำหนักคือ 0.36 จะได้ผลลัพธ์เท่ากับ 0.14 ซีฟู๊ดค่าเดิมคือ 1 เมื่อนำไปคูณกับค่าน้ำหนักคือ 0.36 จะได้ผลลัพธ์เท่ากับ 0.36 คะน้าค่าเดิมคือ 0.6 เมื่อนำไปคูณกับค่าน้ำหนักคือ 0.36 จะได้ผลลัพธ์เท่ากับ 0.22 ผักบุ้งค่าเดิมคือ 0.8 เมื่อนำไปคูณกับค่าน้ำหนัก 0.36 จะได้ผลลัพธ์เท่ากับ 0.29 หน่อไม้ค่าเดิมคือ 1 เมื่อนำไปคูณกับค่าน้ำหนักคือ 0.36 จะได้ผลลัพธ์คือ 0.36 รสชาติเผ็ดค่าเดิมคือ 0.8 เมื่อนำไปคูณกับค่าน้ำหนักคือ 0.29 จะได้ผลลัพธ์

เท่ากับ 0.23 รสชาติเปรี้ยวค่าเดิมคือ 0.2 เมื่อนำไปคูณกับค่าน้ำหนักคือ 0.29 จะได้ผลลัพธ์คือ 0.06 รสชาติหวานค่าเดิมคือ 0.6 เมื่อนำไปคูณกับค่าน้ำหนักคือ 0.29 จะได้ผลลัพธ์เท่ากับ 0.17

ส่วนประกอบ (เนื้อ)*0.36

ไก่	หมู	ซีฟู้ด	...
0.36	0.14	0.36	...

ส่วนประกอบ (ผัก)*0.36

คะน้า	ผักบุ้ง	หน่อไม้	...
0.22	0.29	0.36	...

รสชาติ*0.29

เผ็ด	เปรี้ยว	หวาน
0.23	0.06	0.17

อื่นๆ

...
...

รูปที่ 3-5 การคูณค่าน้ำหนักให้กับทุก Item ในแต่ละ criteria

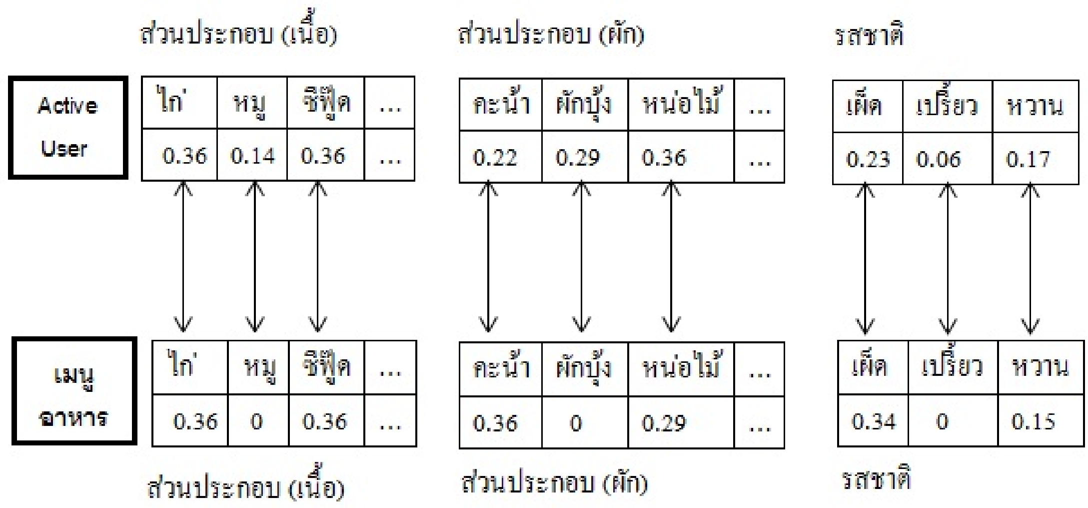
4) คำนวณหาค่า Euclidean Distance ซึ่งเป็นค่าที่แสดงความแตกต่างของลักษณะความชอบระหว่าง Active User กับรายการอาหาร ทำการหาค่าความต่างระหว่าง Active User กับรายการอาหารแต่ละรายการ โดยรายการอาหารแต่ละรายการคำนวณเช่นเดียวกับ Active User ดังขั้นตอนที่1-3 จากนั้นจึงหาผลต่างในแต่ละตำแหน่งที่ตรงกันระหว่าง Active User กับ รายการอาหารแต่ละรายการ แล้วนำมาหาค่าเฉลี่ยผลต่างทั้งหมด ตามสูตรที่ (1)

$$d(X,Y)=\sqrt{\sum_{i=1}^{n}(C-W_i)(X_i-Y_i)^2} \quad (1)$$

X_i คือ Item ที่ i ของ User และ Y_i คือ Item ที่ i ของ รายการอาหาร เช่น Item ที่ 1 คือ ไก่

W คือ ค่าน้ำหนักของแต่ละ Item

C คือ ค่าคงที่ กำหนดให้เท่ากับ 0 เนื่องจากให้ความสำคัญกับค่าน้ำหนักเป็นสำคัญโดยไม่สนใจค่าคงที่



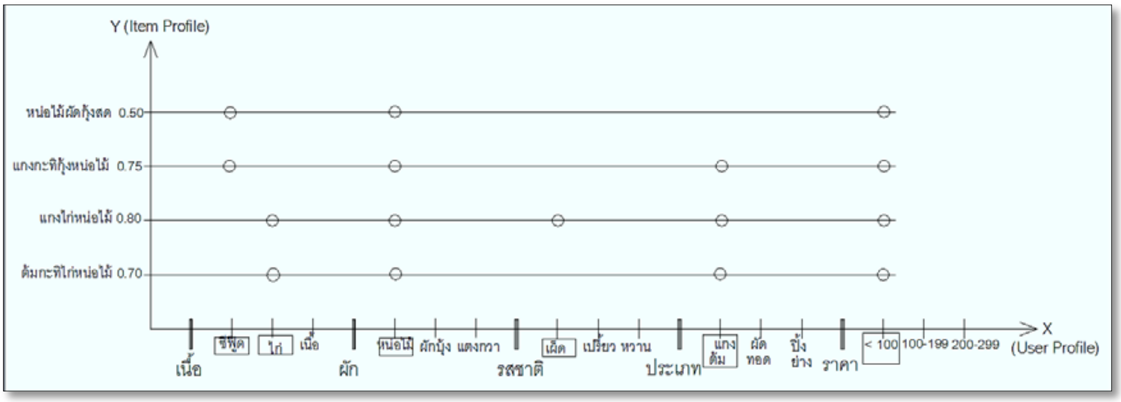
รูปที่ 3-6 การเปรียบเทียบความแตกต่างของลักษณะความชอบระหว่าง Active User กับรายการอาหาร

5) นำค่าที่ได้มาคำนวณในสูตรเพื่อหาค่าความเหมือน (Similarity Measure) ของ Active User และรายการอาหาร ตามสูตรที่ (2)

$$Sv = \frac{1}{1+d(X,Y)} \qquad (2)$$

Sv คือค่าความคล้ายคลึง ระหว่าง Active User และรายการอาหาร $d(X,Y)$ คือค่า Euclidean Distance ที่ได้จากการคำนวณ ค่าความคล้ายคลึงที่ได้จะอยู่ในช่วง 0-1 ค่ายิ่งมากยิ่งมีความคล้ายคลึงมาก

จากรูปที่ 3-7 กำหนดให้แกน X คือ ความต้องการของผู้ใช้ (User Profile) แกน Y คือ รายการอาหาร (Item Profile) ยกตัวอย่าง โดยความต้องการของผู้ใช้คือ ชอบกินไก่และซีฟู๊ด, หน่อไม้, ประเภทอาหารแกงต้ม, รสชาติเผ็ดและหวาน และราคาน้อยกว่า 100 บาท จากนั้นจะทำการคำนวณหาค่าความต่าง Euclidean Distance และคำนวณหาค่าความคล้ายคลึง ระหว่าง User Profile และ Item Profile จะได้ค่าความคล้ายคลึงของรายการอาหารที่มีค่าสูงสุด 10 ลำดับ ยกตัวอย่าง 4 ลำดับดังต่อไปนี้ แกงไก่หน่อไม้ค่าความคล้ายคลึงเท่ากับ 0.80 แกงกะทิกุ้งหน่อไม้ค่าความคล้ายคลึงเท่ากับ 0.75 ต้มกะทิไก่หน่อไม้ค่าความคล้ายคลึงเท่ากับ 0.70 และหน่อไม้ผัดกุ้งสดค่าความคล้ายคลึงเท่ากับ 0.50



รูปที่ 3-7 การหาค่าความคล้ายคลึง ของแกน X, Y

จากรูปทีี 3-8 แสดงรายการอาหารที่มีค่าความคล้ายคลึงมากที่สุดเรียงตามลำดับ 10 รายการ ผู้บริโภคสามารถเลือกรายการอาหารที่ต้องการ เพื่อค้นหาร้านอาหาร โดยสามารถเลือกรายการอาหารที่ต้องการได้มากกว่า1 รายการ เพื่อค้นหาร้านอาหารที่มีรายการอาหารทั้งหมดที่เลือก ยกตัวอย่างเช่น เลือก รายการอาหาร แกงไก่หน่อไม้ ต้มกะทิไก่หน่อไม้ หน่อไม้ผัดกุ้งสด เพื่อค้นหาร้านอาหาร

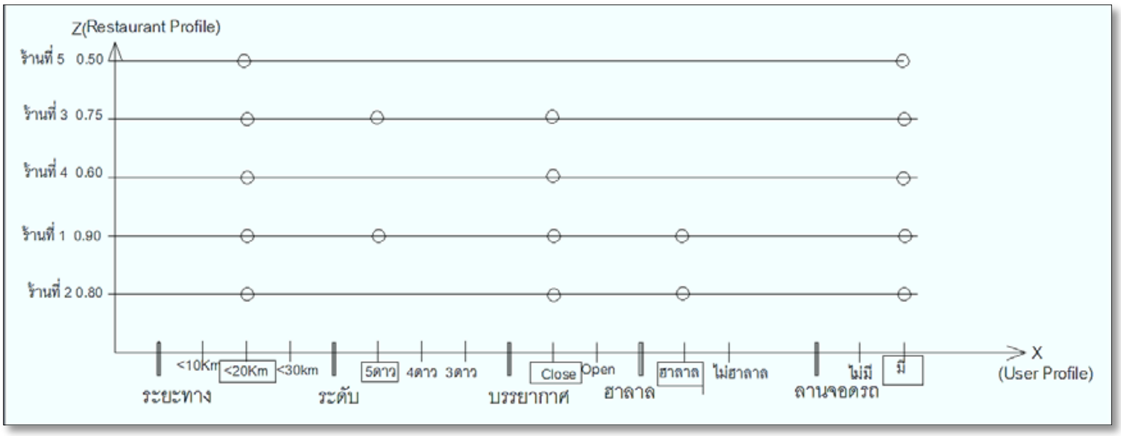
ลำดับ	ชื่อเมนู	ค่าความคล้ายคลึง	เลือกเมนูที่ชอบ
1	แกงไก่หน่อไม้	0.80	√
2	แกงกะทิกุ้งหน่อไม้	0.75	
3	ต้มกะทิไก่หน่อไม้	0.70	√
4	หน่อไม้ผัดกุ้งสด	0.50	√
...	...	...	

รูปที่ 3-8 ผลลัพธ์ของการหาค่าความคล้ายคลึงของรายการอาหาร

6) คำนวณหาความคล้ายคลึงของร้านอาหาร Restaurant Profile กับ User Profile ที่มีรายการอาหาร แกงไก่หน่อไม้ ต้มกะทิไก่หน่อไม้ และ หน่อไม้ผัดกุ้งสด วิธีการทำเช่นเดียวกันกับ Item Profile ยกตัวอย่างดังนี้

จากรูปที่ 3-9 กำหนดให้แกน X คือ User Profile แสดงด้วยความต้องการของผู้ใช้ด้านความต้องการร้านอาหาร และแกน Z คือ Restaurant Profile แสดงข้อมูลต่างๆ ของร้านอาหาร จากนั้นจะทำการคำนวณหาค่าความต่าง Euclidean Distance และคำนวณหาค่าความ

คล้ายคลึง ระหว่าง User Profile และ Restaurant Profile จะได้ค่าความคล้ายคลึงของร้านอาหารเรียงตามลำดับจากมากไปหาน้อยตามลำดับที่มีรายการอาหารทั้งสามรายการดังต่อไปนี้ ร้านที่ 1 ค่าความคล้ายคลึงเท่ากับ 0.9 ร้านที่ 2 ค่าความคล้ายคลึงเท่ากับ 0.8 ร้านที่ 3 ค่าความคล้ายคลึงเท่ากับ 0.75 ร้านที่ 4 ค่าความคล้ายคลึงเท่ากับ 0.6 เป็นต้น



รูปที่ 3-9 การหาค่าความคล้ายคลึง ของแกน Y,Z

จากรูปที่ 3-10 แสดงร้านอาหารที่มีรายการอาหารดังกล่าวและมีค่าความคล้ายคลึงจากมากไปหาน้อยเรียงตามลำดับ

ชื่อร้าน	ค่าความคล้ายคลึง	เมนูแกงไก่หน่อไม้	ต้มกะทิไก่หน่อไม้	หน่อไม้ผัดกุ้ง
ร้านที่ 1	แกงไก่หน่อไม้	Sv =0.90	√	√
ร้านที่ 2	แกงกะทิกุ้งหน่อไม้	Sv =0.80	√	√
ร้านที่ 3	ต้มกะทิไก่หน่อไม้	Sv =0.75	√	√
ร้านที่ 4	หน่อไม้ผัดกุ้งสด	Sv =0.60	√	√
...	...	...		

รูปที่ 3-10 ผลลัพธ์การหาค่าความคล้ายคลึงของร้านอาหาร

3.3.2.2 ผลการหาค่าความคล้ายคลึงของรายการอาหาร

คำนวณค่าความคล้ายคลึงของรายการอาหารด้วยวิธีการข้างต้นที่ได้กล่าวไปแล้ว โดยยกตัวอย่างรายการอาหารตั้งต้นทั้งหมด 36 รายการ กำหนดค่าคะแนนความชอบและค่าน้ำหนักให้กับ ส่วนประกอบที่เป็นเนื้อ ผัก และรสชาติ ดังตารางที่ 3-3 ถึง ตางรางที่ 3-5

ตารางที่ 3-3 การกำหนดค่าคะแนนความชอบและค่าน้ำหนักให้กับรายการอาหารตั้งต้นของส่วนประกอบเนื้อ

ลำดับ	ชื่อเมนู	ไก่	เนื้อวัว	อาหารทะเล	ปลาหมึก	ไข่	หมู	ปลา	กุ้ง	ปู	ค่าน้ำหนัก
1	สะตอผัดเครื่องแกงกุ้ง	0	0	0	0	0	0	0	5	0	5
2	ผัดเผ็ดกุ้งสะตอ	0	0	0	0	0	0	0	5	0	5
3	ฉู่ฉี่กุ้ง	0	0	0	0	0	0	0	5	0	5
4	แกงคั่วหน่อไม้ดองสะตอกุ้ง	0	0	0	0	0	0	0	5	0	5
5	สะตอผัดกะปิกุ้ง	0	0	0	0	0	0	0	5	0	5
6	บล็อกเคอรี่ผัดกุ้ง	0	0	0	0	0	0	0	5	0	5
7	คั่วกลิ้งไตปลา	0	0	0	0	0	0	5	0	0	5
8	ปลากระพงผัดขิง	0	0	0	0	0	0	5	0	0	5
9	กุ้งทอดกระเทียม	0	0	0	0	0	0	0	5	0	5
10	ทอดมันกุ้ง	0	0	0	0	0	0	0	5	0	5
11	ต้มยำกุ้งน้ำข้น	0	0	0	0	0	0	0	5	0	5
12	มะระผัดไข่	0	0	0	0	5	0	0	0	0	5
13	ข้าวผัดเครื่องแกง	0	0	5	0	0	0	0	0	0	5
14	ซีฟู้ดผัดฉ่า	0	0	5	0	0	0	0	0	0	5

ลำดับ	ชื่อเมนู	ไก่	เนื้อวัว	อาหารทะเล	ปลาหมึก	ไข่	หมู	ปลา	กุ้ง	ปู	ค่าน้ำหนัก
15	ไข่เจียวต้มยำ	0	0	0	0	5	0	0	0	0	5
16	ไข่เจียว	0	0	0	0	5	0	0	0	0	5
17	ปูม้าผัดต้นหอม	0	0	0	0	0	0	0	0	5	5
18	ปูนิ่มกระเพรากรอบ	0	0	0	0	0	0	0	0	5	5
19	ปูม้าผัดผงกะหรี่	0	0	0	0	0	0	0	0	5	5
20	แกงส้มปูม้า	0	0	0	0	0	0	0	0	5	5
21	คะน้าไก่กรอบ	5	0	0	0	0	0	0	0	0	5
22	ผัดคะน้าหมูกรอบ	0	0	0	0	0	5	0	0	0	5
23	ผัดคะน้าปลาเค็ม	0	0	0	0	0	0	5	0	0	5
24	ปูนิ่มกระเพรากรอบ	0	0	0	0	0	0	0	0	5	5
25	ต้มยำกุ้งมะพร้าวอ่อน	0	0	0	0	0	0	0	5	0	5
26	ปลากระพงซอสหวาน	0	0	0	0	0	0	5	0	0	5
27	กุ้งซอสมะขาม	0	0	0	0	0	0	0	5	0	5
28	หมึกไข่ผัดต้นหอม	0	0	0	5	0	0	0	0	0	5
29	หมึกไข่ผัดผงกะหรี่	0	0	0	5	0	0	0	0	0	5

ลำดับ	ชื่อเมนู	ไก่	เนื้อวัว	อาหารทะเล	ปลาหมึก	ไข่	หมู	ปลา	กุ้ง	ปู	ค่าน้ำหนัก
30	กุ้งผัดผงกะหรี่	0	0	0	0	0	0	0	5	0	5
31	กุ้งต้มยำมะม่วงเบา	0	0	0	0	0	0	0	5	0	5
32	กุ้งผัดน้ำพริกเผา	0	0	0	0	0	0	0	5	0	5
33	ต้มข่าซีฟู๊ดหัวปลี	0	0	5	0	0	0	0	0	0	5
34	ปูม้าผัดพริกไทยดำ	0	0	0	0	0	0	0	0	5	5
35	ต้มข่าไก่หัวปลี	5	0	0	0	0	0	0	0	0	5
36	ข้าวผัดหมู	0	0	0	0	0	5	0	0	0	5

ตารางที่ 3-4 การกำหนดค่าคะแนนความชอบและค่าน้ำหนักให้กับรายการอาหารตั้งต้นของส่วนประกอบผัก

ลำดับ	ชื่อเมนู	สะตอ	หน่อไม้	ชะพลู	มะเขือ	บล็อกเคอรี่	ขิง	มะเขือเทศ	เห็ด	มะระ	คะน้า	แครอท	หัวปลี	ค่าน้ำหนัก
1	สะตอผัดเครื่องแกงกุ้ง	5	0	0	0	0	0	0	0	0	0	0	0	4
2	ผัดเผ็ดกุ้งสะตอ	5	0	0	0	0	0	0	0	0	0	0	0	4
3	ฉู่ฉี่กุ้ง	0	0	0	0	0	0	0	0	0	0	0	0	4
4	แกงคั่วหน่อไม้ดอง	4	5	0	0	0	0	0	0	0	0	0	0	4

ลำดับ	ชื่อเมนู	สะตอ	หน่อไม้	ชะพลู	มะเขือ	บล็อกเคอรี่	ขิง	มะเขือเทศ	เห็ด	มะระ	คะน้า	แครอท	หัวปลี	ค่าน้ำหนัก
	สะตอกุ้ง													
5	สะตอผัดกะปิกุ้ง	5	0	0	0	0	0	0	0	0	0	0	0	4
6	บล็อกเคอรี่ผัดกุ้ง	0	0	0	0	5	0	0	0	0	0	0	0	4
7	คั่วกลิ้งไตปลา	0	0	0	0	0	0	0	0	0	0	0	0	4
8	ปลากระพงผัดขิง	0	0	0	0	0	5	0	0	0	0	0	0	4
9	กุ้งทอดกระเทียม	0	0	0	0	0	0	0	0	0	0	0	0	4
10	ทอดมันกุ้ง	0	0	0	0	0	0	0	0	0	0	0	0	4
11	ต้มยำกุ้งน้ำข้น	0	0	0	0	0	0	5	5	0	0	0	0	4
12	มะระผัดไข่	0	0	0	0	0	0	0	0	5	0	0	0	4
13	ข้าวผัดเครื่องแกง	0	0	0	0	0	0	0	0	0	5	3	0	4
14	ซีฟู๊ดผัดฉ่า	0	0	0	0	0	0	0	0	0	0	0	0	4
15	ไข่เจียวต้มยำ	0	0	0	0	0	0	0	0	0	0	0	0	4
16	ไข่เจียว	0	0	0	0	0	0	0	0	0	0	0	0	4
17	ปูม้าผัดต้นหอม	0	0	0	0	0	0	0	0	0	0	0	0	4
18	ปูนิ่มกระเพรากรอบ	0	0	0	0	0	0	0	0	0	0	0	0	4

ลำดับ	ชื่อเมนู	สะตอ	หน่อไม้	ชะพลู	มะเขือ	บล็อกเคอรี่	ขิง	มะเขือเทศ	เห็ด	มะระ	คะน้า	แครอท	หัวปลี	ค่าน้ำหนัก
19	ปูม้าผัดผงกะหรี่	0	0	0	0	0	0	0	0	0	0	0	0	4
20	แกงส้มปูม้า	0	0	0	0	0	0	0	0	0	0	0	0	4
21	คะน้าไก่กรอบ	0	0	0	0	0	0	0	0	0	5	0	0	4
22	ผัดคะน้าหมูกรอบ	0	0	0	0	0	0	0	0	0	5	0	0	4
23	ผัดคะน้าปลาเค็ม	0	0	0	0	0	0	0	0	0	5	0	0	4
24	ปูนิ่มกระเพรากรอบ	0	0	0	0	0	0	0	0	0	0	0	0	4
25	ต้มยำกุ้งมะพร้าวอ่อน	0	0	0	0	0	0	0	0	0	0	0	0	4
26	ปลากระพงซอสหวาน	0	0	0	0	0	0	0	0	0	0	0	0	4
27	กุ้งซอสมะขาม	0	0	0	0	0	0	0	0	0	0	0	0	4
28	หมึกไข่ผัดต้นหอม	0	0	0	0	0	0	0	0	0	0	0	0	4
29	หมึกไข่ผัดผงกะหรี่	0	0	0	0	0	0	0	0	0	0	0	0	4
30	กุ้งผัดผงกะหรี่	0	0	0	0	0	0	0	0	0	0	0	0	4
31	กุ้งต้มยำมะม่วงเบา	0	0	0	0	0	0	0	0	0	0	0	0	4
32	กุ้งผัดน้ำพริกเผา	0	0	0	0	0	0	0	0	0	0	0	0	4
33	ต้มข่าซีฟู๊ดหัวปลี	0	0	0	0	0	0	0	0	0	0	0	0	4

ลำดับ	ชื่อเมนู	สะตอ	หน่อไม้	ชะพลู	มะเขือ	บล็อกเคอรี่	ขิง	มะเขือเทศ	เห็ด	มะระ	คะน้า	แครอท	หัวปลี	ค่าน้ำหนัก
34	ปูม้าผัดพริกไทยดำ	0	0	0	0	0	0	0	0	0	0	0	0	4
35	ต้มข่าไก่หัวปลี	0	0	0	0	0	0	0	0	0	0	0	0	4
36	ข้าวผัดหมู	0	0	0	0	0	0	0	0	0	3	2	0	4

ตารางที่ 3-5 การกำหนดค่าคะแนนความชอบและค่าน้ำหนักให้กับรายการอาหารตั้งต้นของรสชาติอาหาร

ลำดับ	ชื่อเมนู	เผ็ด	หวาน	เค็ม	เปรี้ยว	ค่าน้ำหนัก
1	สะตอผัดเครื่องแกงกุ้ง	5	0	0	0	3
2	ผัดเผ็ดกุ้งสะตอ	5	0	0	0	3
3	ฉู่ฉี่กุ้ง	5	0	0	0	3
4	แกงคั่วหน่อไม้ดองสะตอกุ้ง	5	0	0	0	3
5	สะตอผัดกะปิกุ้ง	0	0	5	0	3
6	บล็อกเคอรี่ผัดกุ้ง	0	0	2	0	3
7	คั่วกลิ้งไตปลา	5	0	5	0	3
8	ปลากระพงผัดขิง	1	0	2	0	3
9	กุ้งทอดกระเทียม	0	0	5	0	3
10	ทอดมันกุ้ง	0	0	5	0	3
11	ต้มยำกุ้งน้ำข้น	5	0	0	5	3
12	มะระผัดไข่	0	0	2	0	3
13	ข้าวผัดเครื่องแกง	5	0	0	0	3
14	ซีฟู้ดผัดฉ่า	3	0	0	0	3
15	ไข่เจียวต้มยำ	2	0	2	2	3
16	ไข่เจียว	0	0	2	0	3

ลำดับ	ชื่อเมนู	เผ็ด	หวาน	เค็ม	เปรี้ยว	ค่าน้ำหนัก
17	ปูม้าผัดต้นหอม	0	0	2	0	3
18	ปูนิ่มกระเพรากรอบ	0	0	2	0	3
19	ปูม้าผัดผงกะหรี่	0	2	2	0	3
20	แกงส้มปูม้า	5	0	0	5	3
21	คะน้าไก่กรอบ	0	0	2	0	3
22	ผัดคะน้าหมูกรอบ	0	0	2	0	3
23	ผัดคะน้าปลาเค็ม	0	0	5	0	3
24	ปูนิ่มกระเพรากรอบ	0	0	2	0	3
25	ต้มยำกุ้งมะพร้าวอ่อน	5	0	0	5	3
26	ปลากระพงซอสหวาน	0	5	0	0	3
27	กุ้งซอสมะขาม	0	5	0	5	3
28	หมึกไข่ผัดต้นหอม	0	2	2	0	3
29	หมึกไข่ผัดผงกะหรี่	0	2	2	0	3
30	กุ้งผัดผงกะหรี่	0	2	2	0	3
31	กุ้งต้มยำมะม่วงเบา	5	0	0	5	3
32	กุ้งผัดน้ำพริกเผา	3	3	0	3	3
33	ต้มข่าซีฟู้ดหัวปลี	0	0	2	0	3
34	ปูม้าผัดพริกไทยดำ	2	0	2	0	3

ลำดับ	ชื่อเมนู	เผ็ด	หวาน	เค็ม	เปรี้ยว	ค่าน้ำหนัก
35	ต้มข่าไก่หัวปลี	0	2	2	0	3
36	ข้าวผัดหมู	0	0	2	0	3

จากตารางที่ 3-6 กำหนดความต้องการของผู้ใช้ ตัวอย่างที่ 1 กำหนดค่าคะแนนความชอบของเนื้อสัตว์ ได้แก่ กุ้ง เท่ากับ 5 ค่าคะแนนความชอบผัก ได้แก่ สะตอ เท่ากับ 5 ค่าคะแนนความชอบรสชาติ ได้แก่ เผ็ด เท่ากับ 5

ตารางที่ 3-6 กำหนดความต้องการของผู้ใช้ ตัวอย่างที่ 1

ตัวอย่างที่ 1	อยากทานมากที่สุด ค่าความชอบ=5	อยากทานมาก ค่าความชอบ=4	อยากทานปานกลาง ค่าความชอบ=3	อยากทานน้อย ค่าความชอบ=2	อยากทานน้อยมาก ค่าความชอบ=1
เนื้อสัตว์	กุ้ง	-	-	-	-
ผัก	สะตอ	-	-	-	-
รสชาติ	เผ็ด	-	-	-	-

ตารางที่ 3-7 ผลการคำนวณค่าความคล้ายคลึงของรายการอาหาร เรียงตามลำดับจากค่าความคล้ายคลึงมากไปหาน้อย จำนวน 5 รายการ

ลำดับ	ชื่อเมนู	หาค่า Standard Euclidean Distance จากสูตร 1. $d(x,y) = \sqrt{\sum_{i=1}^{n}(c-w_i)\,(x_i-y_i)^2}$ $= \sum_{i=1}^{n}\lvert(c-w_i)\,(x_{i-}\;y_i)\rvert$	หาค่าความคล้ายคลึงจาก สูตร $S_v = \frac{1}{1+\mathrm{d(x,y)}}$	ผลลัพธ์ ค่าความคล้ายคลึง (0-1)
1	สตอผัดเครื่องแกงกุ้ง	|0+0+0|	1/(1+0)	1.00
2	ผัดเผ็ดกุ้งสะตอ	|0+0+0|	1/(1+0)	1.00
3	ฉู่ฉี่กุ้ง	|0+1+0|	1/(1+1)	0.50
4	แกงคั่วหน่อไม้ดอง สะตอกุ้ง	|0+1+0.2+0|	1/(1+1.2)	0.45
5	สะตอผัดกะปิกุ้ง	|0+0+1+1|	1/(1+2)	0.33

จากตารางที่ 3-8 กำหนดความต้องการของผู้ใช้ ตัวอย่างที่ 2 กำหนดค่าคะแนนความชอบของเนื้อสัตว์ ได้แก่ กุ้งเท่ากับ 5 และปลาเท่ากับ 4 ค่าคะแนนความชอบผัก ได้แก่ บล็อกเคอรี่เท่ากับ 5 ขิงเท่ากับ 4 ค่าคะแนนความชอบรสชาติ ได้แก่ เค็มเท่ากับ 5 เผ็ดเท่ากับ 4

ตารางที่ 3-8 การกำหนดความต้องการของผู้ใช้ ตัวอย่างที่ 2

ตัวอย่างที่ 2	**อยากทานมากที่สุด ค่าความชอบ=5**	**อยากทานมาก ค่าความชอบ=4**	**อยากทานปานกลาง ค่าความชอบ=3**	**อยากทานน้อย ค่าความชอบ=2**	**อยากทานน้อยมาก ค่าความชอบ=1**
เนื้อสัตว์	กุ้ง	ปลา	-	-	-
ผัก	บล็อกเคอรี่	ขิง	-	-	-
รสชาติ	เค็ม	เผ็ด	-	-	-

ตารางที่ 3-9 ผลการคำนวณค่าความคล้ายคลึงของรายการอาหาร เรียงตามลำดับจากค่าความคล้ายคลึงมากไปหาน้อย จำนวน 5 รายการ

ลำดับ	ชื่อเมนู	**หาค่า Standard Euclidean Distance จากสูตร** 1. $d(x,y) = \sqrt{\sum_{i=1}^{n}(c-w_i)\,(x_i-y_i)^2} = \sum_{i=1}^{n} \lvert(c-w_i)\,(x_{i-}\ y_i)\rvert$	สูตรการคำนวณ $s_v = \frac{1}{1+d(x,y)}$	**ผลลัพธ์ ค่าความคล้ายคลึง (0-1)**
1	บล็อกเคอรี่ผัดกุ้ง	|0+0.8+0+0.8+0.6+0.8|	1/(1+3.0)	0.25
2	คั่วกลิ้งไตปลา	|1+0.2+1+0.8+0+0.2|	1/(1+3.2)	0.24
3	ปลากระพงผัดขิง	|1+0.2+0.2+1+0.6+0.6|	1/(1+3.6)	0.22
4	กุ้งทอดกระเทียม	|0+0.8+1+0.8+0+0.8|	1/(1+3.4)	0.23
5	ทอดมันกุ้ง	|0+0.8+1+0.8+0+0.8|	1/(1+3.4)	0.23

จากตารางที่ 3-10 กำหนดความต้องการของผู้ใช้ ตัวอย่างที่ 3 กำหนดค่าคะแนนความชอบของเนื้อสัตว์ ได้แก่ กุ้งเท่ากับ 5 ปลาเท่ากับ 4 หมูเท่ากับ 3 ไข่เท่ากับ 2 เนื้อวัวเท่ากับ 1 ค่าคะแนนความชอบผัก ได้แก่ บล็อกเคอรี่เท่ากับ 5 ขิงเท่ากับ 4 เห็ดเท่ากับ 3 ข้าวโพดอ่อนเท่ากับ 2 มะเขือเทศเท่ากับ 1 ค่าคะแนนความชอบรสชาติ ได้แก่ เปรี้ยวเท่ากับ 5 เผ็ดเท่ากับ 4 เค็มเท่ากับ 3 หวานเท่ากับ 2

ตารางที่ 3-10 กำหนดความต้องการของผู้ใช้ ตัวอย่างที่ 3

ตัวอย่างที่ 3	**อยากทานมากที่สุด ค่าความชอบ=5**	**อยากทานมาก ค่าความชอบ=4**	**อยากทานปานกลาง ค่าความชอบ=3**	**อยากทานน้อย ค่าความชอบ=2**	**อยากทานน้อยมาก ค่าความชอบ=1**
เนื้อสัตว์	กุ้ง	ปลา	หมู	ไข่	เนื้อวัว
ผัก	บล็อกเคอรี่	ขิง	เห็ด	ข้าวโพดอ่อน	มะเขือเทศ
รสชาติ	เปรี้ยว	เผ็ด	เค็ม	หวาน	-

ตารางที่ 3-11 ผลการคำนวณค่าความคล้ายคลึงของรายการอาหาร เรียงตามลำดับจากค่าความคล้ายคลึงมากไปหาน้อย จำนวน 5 รายการ

ลำดับ	**ชื่อเมนู**	**หาค่า Standard Euclidean Distance จากสูตร** 1. $d(x,y) = \sqrt{\sum_{i=1}^{n}(c - w_i)\,(x_i - y_i)^2} = \sum_{i=1}^{n} \lvert (c - w_i)\,(x_{i-}\ y_i) \rvert$	**สูตรการคำนวณ** $s_v = \frac{1}{1+\mathrm{d(x,y)}}$	**ผลลัพธ์ ค่าความคล้ายคลึง (0-1)**
1	ต้มยำกุ้งน้ำข้น	|0+0.8+0.6+0.4+0.2+1+0.8+0.6+0.4+0.2+0+0.2+0.6+0.4|	1/(1+6.2)	0.14
2	ต้มยำกุ้งมะพร้าวอ่อน	|0+0.8+0.6+0.4+0.2+1+0.8+0.6+0.4+0.2+0+0.2+0.6+0.4|	1/(1+6.2)	0.14

ลำดับ	ชื่อเมนู	หาค่า Standard Euclidean Distance จากสูตร 1. $d(x,y) = \sqrt{\sum_{i=1}^{n}(c-w_i)\,(x_i-y_i)^2}$ $= \sum_{i=1}^{n} \lvert(c-w_i)\,(x_{i-}\ y_i)\rvert$	สูตรการคำนวณ $s_v = \frac{1}{1+d(x,y)}$	ผลลัพธ์ ค่าความคล้ายคลึง (0-1)
3	กุ้งต้มยำมะม่วงเบา	|0+0.8+0.6+0.4+0.2+1+0.8+0.6+0.4+0.2+0+0.2+0.6+0.4|	1/(1+6.2)	0.14
4	กุ้งผัดน้ำพริกเผา	|0+0.8+0.6+0.4+0.2+1+0.8+0.6+0.4+0.2+0.4+0.2+0.6+0.2|	1/(1+6.4)	0.14
5	บล็อกเคอรี่ผัดกุ้ง	|0+0.8+0.6+0.4+0.2+0+0.8+0.6+0.4+0.2+1+0.8+0.2+0.4|	1/(1+6.4)	0.14

จากตารางที่ 3-12 กำหนดความต้องการของผู้ใช้ ตัวอย่างที่ 4 กำหนดค่าคะแนนความชอบของเนื้อสัตว์ ได้แก่ อาหารทะเลและไข่เท่ากับ 5 ค่าคะแนนความชอบผัก ได้แก่ หัวผักกาดและมะระเท่ากับ 4 ค่าคะแนนความชอบรสชาติ ได้แก่ เผ็ดและเค็มเท่ากับ 5

ตารางที่ 3-12 การกำหนดความต้องการของผู้ใช้ ตัวอย่างที่ 4

ตัวอย่างที่ 4	อยากทานมากที่สุด ค่าความชอบ=5	อยากทานมาก ค่าความชอบ=4	อยากทานปานกลาง ค่าความชอบ=3	อยากทานน้อย ค่าความชอบ=2	อยากทานน้อยมาก ค่าความชอบ=1
เนื้อสัตว์	อาหารทะเล, ไข่	-	-	-	-
ผัก	หัวผักกาด, มะระ	-	-	-	-
รสชาติ	เผ็ด, เค็ม	-	-	-	-

ตารางที่ 3-13 ผลการคำนวณค่าความคล้ายคลึงของรายการอาหาร เรียงตามลำดับจากค่าความคล้ายคลึงมากไปหาน้อย จำนวน 5 รายการ

ลำดับ	ชื่อเมนู	หาค่า Standard Euclidean Distance จากสูตร 1. $d(x,y) = \sqrt{\sum_{i=1}^{n}(c-w_i)\,(x_i-y_i)^2} = \sum_{i=1}^{n}\left|(c-w_i)\,(x_{i-}\ y_i)\right|$	สูตรการคำนวณ $s_v = \frac{1}{1+\mathrm{d(x,y)}}$	ผลลัพธ์ ค่าความคล้ายคลึง (0-1)
1	มะระผัดไข่	|1+0+0+1+0.6+1|	1/(1+3.6)	0.22
2	ซีฟู้ดผัดฉ่า	|0+1+1+1+0.2+1|	1/(1+4.2)	0.19
3	ต้มข่าซีฟู้ดหัวปลี	|0+1+1+1+0.6+1|	1/(1+4.6)	0.18
4	ไข่เจียว	|0+1+1+1+0.6+1|	1/(1+4.6)	0.18
5	ไข่เจียวต้มยำ	|0+1+1+1+0.6+0.6+0.4|	1/(1+4.6)	0.18

จากตารางที่ 3-14 กำหนดความต้องการของผู้ใช้ ตัวอย่างที่ 5 กำหนดค่าคะแนนความชอบของเนื้อสัตว์ ได้แก่ปูเท่ากับ 5

ตารางที่ 3-14 การกำหนดความต้องการของผู้ใช้ ตัวอย่างที่ 5

ตัวอย่างที่ 5	อยากทานมากที่สุด ค่าความชอบ=5	อยากทานมาก ค่าความชอบ=4	อยากทานปานกลาง ค่าความชอบ=3	อยากทานน้อย ค่าความชอบ=2	อยากทานน้อยมาก ค่าความชอบ=1
เนื้อสัตว์	ปู	-	-	-	-
ผัก	-	-	-	-	-

ตัวอย่างที่ 5	อยากทานมากที่สุด ค่าความชอบ=5	อยากทานมาก ค่าความชอบ=4	อยากทานปานกลาง ค่าความชอบ=3	อยากทานน้อย ค่าความชอบ=2	อยากทานน้อยมาก ค่าความชอบ=1
รสชาติ	-	-	-	-	-

ตารางที่ 3-15 ผลการคำนวณค่าความคล้ายคลึงของรายการอาหาร เรียงตามลำดับจากค่าความคล้ายคลึงมากไปหาน้อย จำนวน 5 รายการ

ลำดับ	ชื่อเมนู	หาค่า Standard Euclidean Distance จากสูตร 1. $d(x,y) = \sqrt{\sum_{i=1}^{n}(c - w_i)\,(x_i - y_i)^2}$ $= \sum_{i=1}^{n} \lvert(c - w_i)\,(x_{i-}\ y_i)\rvert$	สูตรการคำนวณ $s_v = \dfrac{1}{1 + \mathrm{d(x,y)}}$	ผลลัพธ์ ค่าความคล้ายคลึง (0-1)
1	ปูม้าผัดพริกไทยดำ	|0|	1/(1+0)	1.00
2	ปูนิ่มกระเพรากรอบ	|0|	1/(1+0)	1.00
3	แกงส้มปูม้า	|0|	1/(1+0)	1.00
4	ปูม้าผัดผงกะหรี่	|0|	1/(1+0)	1.00
5	ปูม้าผัดต้นหอม	|0|	1/(1+0)	1.00

จากตารางที่ 3-16 กำหนดความต้องการของผู้ใช้ ตัวอย่างที่ 6 กำหนดค่าคะแนนความชอบของผัก ได้แก่คะน้าเท่ากับ 5

ตารางที่ 3-16 การกำหนดความต้องการของผู้ใช้ ตัวอย่างที่ 6

ตัวอย่างที่ 6	**อยากทานมากที่สุด ค่าความชอบ=5**	**อยากทานมาก ค่าความชอบ=4**	**อยากทานปานกลาง ค่าความชอบ=3**	**อยากทานน้อย ค่าความชอบ=2**	**อยากทานน้อยมาก ค่าความชอบ=1**
เนื้อสัตว์	-	-	-	-	-
ผัก	คะน้า	-	-	-	-
รสชาติ	-	-	-	-	-

ตารางที่ 3-17 ผลการคำนวณค่าความคล้ายคลึงของรายการอาหาร เรียงตามลำดับจากค่าความคล้ายคลึงมากไปหาน้อย จำนวน 5 รายการ

ลำดับ	**ชื่อเมนู**	**หาค่า Standard Euclidean Distance จากสูตร** 1. $d(x,y) = \sqrt{\sum_{i=1}^{n}(c-w_i)\,(x_i-y_i)^2} = \sum_{i=1}^{n}\lvert(c-w_i)\,(x_{i-}\ y_i)\rvert$	**สูตรการคำนวณ** $s_v = \frac{1}{1+\mathrm{d(x,y)}}$	**ผลลัพธ์ ค่าความคล้ายคลึง (0-1)**
1	คะน้าไก่กรอบ	|0|	1/(1+0)	1.00
2	ผัดคะน้าหมูกรอบ	|0|	1/(1+0)	1.00
3	ผัดคะน้าปลาเค็ม	|0|	1/(1+0)	1.00
4	ข้าวผัดเครื่องแกง	|0+0.6|	1/(1+0.6)	0.63
5	ข้าวผัดหมู	|0+0.4+0.4|	1/(1+0.8)	0.56

จากตารางที่ 3-18 กำหนดความต้องการของผู้ใช้ ตัวอย่างที่ 7 กำหนดค่าคะแนนความชอบของรสชาติ ได้แก่รสหวานเท่ากับ 5

ตารางที่ 3-18 การกำหนดความต้องการของผู้ใช้ ตัวอย่างที่ 7

ตัวอย่างที่ 7	อยากทานมากที่สุด ค่าความชอบ=5	อยากทานมาก ค่าความชอบ=4	อยากทานปานกลาง ค่าความชอบ=3	อยากทานน้อย ค่าความชอบ=2	อยากทานน้อยมาก ค่าความชอบ=1
เนื้อสัตว์	-	-	-	-	-
ผัก	-	-	-	-	-
รสชาติ	หวาน	-	-	-	-

ตารางที่ 3-19 ผลการคำนวณค่าความคล้ายคลึงของรายการอาหาร เรียงตามลำดับจากค่าความคล้ายคลึงมากไปหาน้อย จำนวน 5 รายการ

ลำดับ	ชื่อเมนู	หาค่า Standard Euclidean Distance จากสูตร 1. $d(x,y) = \sqrt{\sum_{i=1}^{n}(c-w_i)(x_i-y_i)^2} = \sum_{i=1}^{n}\lvert(c-w_i)(x_i - y_i)\rvert$	สูตรการคำนวณ $s_v = \frac{1}{1+\mathrm{d}(\mathrm{x},\mathrm{y})}$	ผลลัพธ์ ค่าความคล้ายคลึง (0-1)
1	ปลากระพงซอสหวาน	|0|	1/(1+0)	1.00
2	ต้มข่าไก่หัวปลี	|0.4+0.6|	1/(1+1)	0.50
3	หมึกไข่ผัดผงกะหรี่	|0.4+0.6|	1/(1+1)	0.50
4	ปูม้าผัดผงกะหรี่	|0.4+0.6|	1/(1+1)	0.50
5	กุ้งผัดผงกะหรี่	|0.4+0.6|	1/(1+1)	0.50

3.3.3 การแสดงผลข้อมูล

1) การแสดงผลการแนะนำรายการอาหารที่มีความคล้ายคลึงกับความต้องการของผู้ใช้ (User Profile) โดยเรียงตามลำดับรายการอาหารที่มีค่าความคล้ายคลึงจากมากไปหาน้อย ผู้ใช้สามารถเลือกรายการอาหารที่อยากรับประทานได้มากกว่า1รายการ เพื่อค้นหาร้านที่มีค่าความคล้ายคลึงกับความต้องการของผู้ใช้ (User Profile) และมีรายการอาหารดังกล่าว เรียงตามลำดับจากร้านที่มีค่าความคล้ายคลึงมากไปหาน้อย ผ่านทางแอปพลิเคชันบนสมาร์ทโฟนให้ผู้ใช้ได้รับทราบ

2) การแสดงผลการค้นหาร้านอาหาร ผู้ใช้สามารถค้นหาร้านอาหารได้ตามประเภทของร้านอาหาร ระบบจะแสดงข้อมูลของร้านอาหารต่างๆ ตามคำค้น (Keyword) ของผู้ใช้ โดยมีรายละเอียดดังนี้ ชื่อร้านอาหาร ระดับดาว รูปภาพร้านอาหาร ระยะทางระหว่างตำแหน่งปัจจุบันกับร้านอาหาร และผู้ใช้สามารถคลิกเพื่อดูรายละเอียดอื่นๆ ของร้านได้ ผ่านทางแอปพลิเคชันบนสมาร์ทโฟนให้ผู้ใช้ได้รับทราบ

3) การแสดงตำแหน่งของร้านอาหารบนแผนที่ Google Map ผู้ใช้สามารถ เลือกแสดงร้านอาหารบนแผนที่ Google Map ผ่านทางแอปพลิเคชันบนสมาร์ทโฟน

3.4 การคัดเลือกเทคโนโลยีในการพัฒนาระบบ

ผู้วิจัยได้ศึกษาและคัดเลือกเทคโนโลยีและเครื่องมือที่เหมาะสมสำหรับการพัฒนาระบบแนะนำร้านอาหารอัตโนมัติบนสมาร์ทโฟนโดยใช้ข้อมูลเชิงตำแหน่งและรายการอาหารโดยพิจารณาจากประสิทธิภาพของอุปกรณ์และต้นทุนค่าใช้จ่ายในการดำเนินการ โดยแบ่งการศึกษาออกเป็น 2 ส่วน คือ 1) เครื่องมือที่ใช้ในการพัฒนางานวิจัยด้านฮาร์ดแวร์ และ 2) เครื่องมือที่ใช้ในการพัฒนางานวิจัยด้านซอฟต์แวร์ ซึ่งมีรายละเอียด ดังนี้

3.4.1 เครื่องมือที่ใช้ในการพัฒนางานวิจัยด้านฮาร์ดแวร์ ประกอบด้วย

1) เครื่องคอมพิวเตอร์แม่ข่าย (Cloud Server)
2) เครื่องคอมพิวเตอร์ส่วนบุคคล
3) สมาร์ทโฟน ระบบปฏิบัติการ Android

3.4.2 ทรัพยากรด้านซอฟต์แวร์ ประกอบด้วย

1) Apache Cordova
2) Android Platform Components
3) October CMS
4) Visual Studio Code

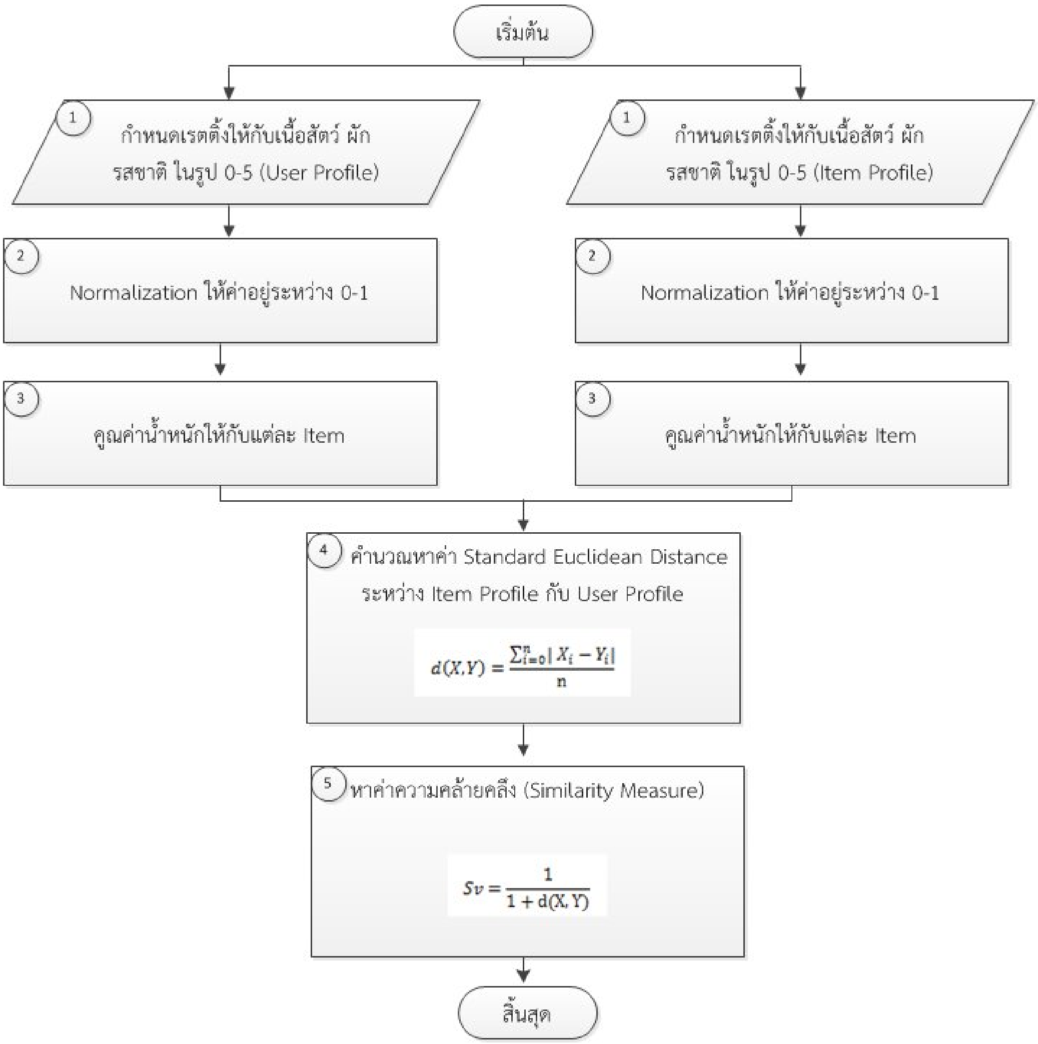
3.5 การออกแบบระบบ

หลังจากได้คัดเลือกเทคโนโลยีที่เหมาะสมในการพัฒนาระบบแล้ว ขั้นตอนต่อไปเป็นการออกแบบขั้นตอนการทำงานของระบบแนะนำร้านอาหารอัตโนมัติบนสมาร์ทโฟน โดยใช้ข้อมูลเชิงตำแหน่งและรายการอาหาร

3.5.1 ออกแบบขั้นตอนการทำงานของระบบ

1) วิธีการกรองข้อมูลสำหรับแนะนำรายการอาหาร

ผู้วิจัยออกแบบอัลกอริทึมสำหรับกรองข้อมูลเพื่อหาค่าความคล้ายคลึงระหว่างความต้องการของผู้ใช้ (User Profile) กับ รายการอาหาร (Item Profile) และ ร้านอาหาร (Restaurant Profile) ดังรูปที่ 3-11



รูปที่ 3-11 อัลกอริทึมสำหรับกรองข้อมูลเพื่อหาค่าความคล้ายคลึง

ขั้นตอนที่ 1 กำหนดค่าคะแนนความชอบของส่วนประกอบรายการอาหาร (Item Profile) ในรูป 0-5 คะแนน และกำหนดค่าคะแนนความชอบของความต้องการของผู้ใช้ (User Profile) ในรูป 0-5 คะแนน

ขั้นตอนที่ 2 ทำการ normalization Item Profile และ User Profile ให้อยู่ในรูป 0-1 คะแนน โดยนำค่าคะแนนความชอบของแต่ละ Item มาหารด้วย 5

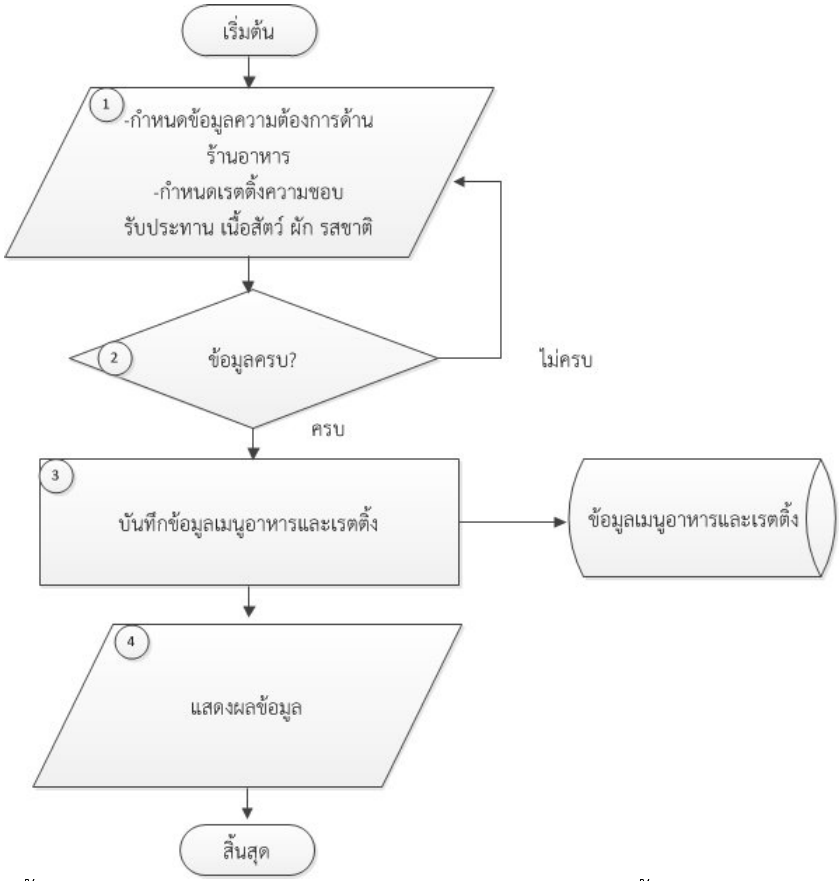
ขั้นตอนที่ 3 คูณค่าน้ำหนักให้กับ Item Profile และ User Profile

ขั้นตอนที่ 4 คำนวณหาค่า Standard Euclidean Distance ระหว่าง Item Profile และ User Profile

ขั้นตอนที่ 5 หาค่าความคล้ายคลึง Similarity Measure

2) ระบบบันทึกข้อมูลรายการอาหารตั้งต้นและการกำหนดคะแนนความชอบ (Item Profile)

ผู้วิจัยได้ออกแบบเว็บแอปพลิเคชันสำหรับระบบบันทึกข้อมูลรายการอาหารตั้งต้นและการกำหนดคะแนนความชอบ (Item Profile) โดยออกแบบเป็นผังงาน (Flow Chart) ดังแสดงในรูปที่ 3-12 สามารถอธิบายรายละเอียดขั้นตอนการทำงานได้ ดังนี้



รูปที่ 3-12 ขั้นตอนการทำงานของระบบบันทึกข้อมูลรายการอาหารตั้งต้นและการกำหนดคะแนนความชอบ

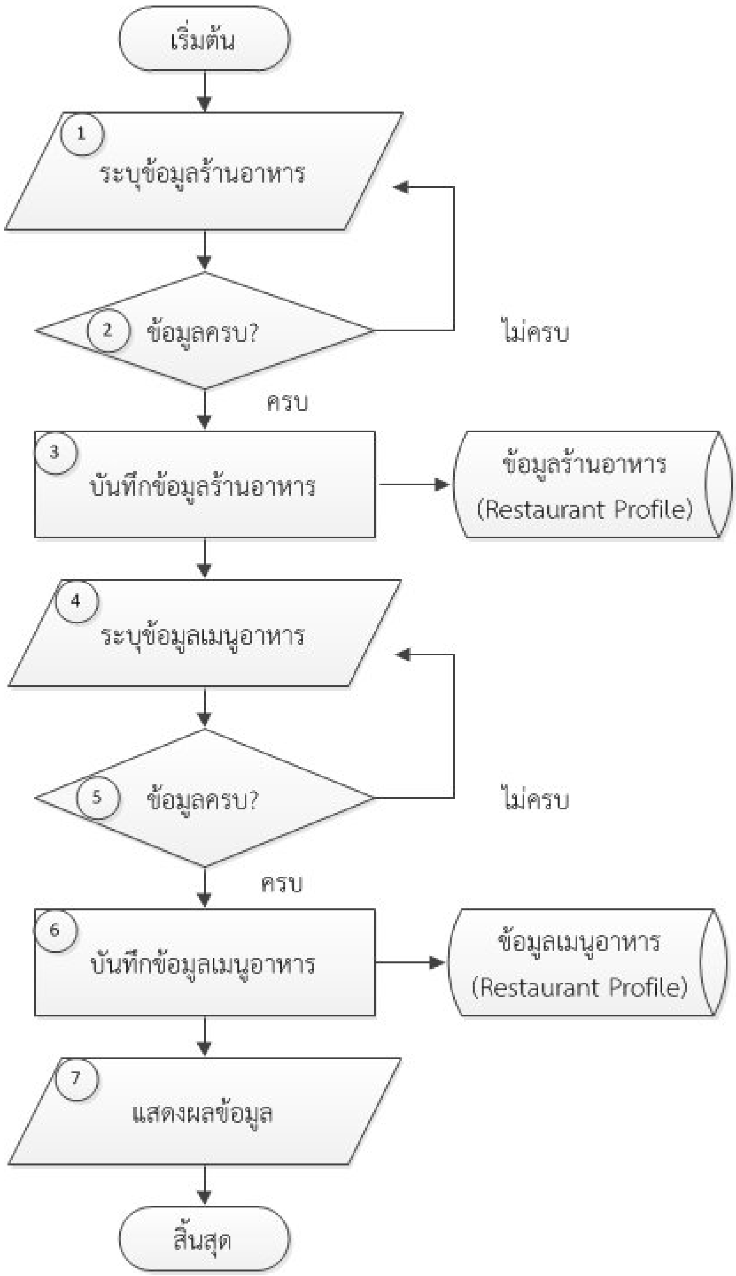
ขั้นตอนที่ 1 แอดมินระบุข้อมูลรายการอาหาร (Item Profile) โดยระบุชื่อรายการอาหาร ประเภทอาหาร รูปภาพ และ กำหนดข้อมูลคะแนนความชอบให้กับรายการอาหาร ได้แก่ กำหนดคะแนนความชอบเนื้อสัตว์ ผัก และรสชาติ

ขั้นตอนที่ 2 ระบบตรวจสอบข้อมูลครบถ้วนหรือไม่

ขั้นตอนที่ 3 กรณีระบุข้อมูลครบถ้วน สามารถบันทึกข้อมูลลงในฐานข้อมูลได้ กรณีที่ระบุข้อมูลไม่ครบถ้วนต้องกลับไประบุข้อมูลให้ครบถ้วน

ขั้นตอนที่ 4 แสดงผลข้อมูลการบันทึก

3) ระบบบันทึกข้อมูลร้านอาหารและรายการอาหาร (Restaurant Profile)



รูปที่ 3-13 ระบบบันทึกข้อมูลร้านอาหารและรายการอาหาร (Restaurant Profile)

ขั้นตอนที่ 1 ผู้ใช้ระบุข้อมูลร้านอาหาร (Restaurant Profile) โดยระบุชื่อร้านอาหาร ประเภทร้านอาหาร รูปภาพ ฮาลาล ลานจอดรถ แอร์ ระดับดาว ที่อยู่ เบอร์ติดต่อ ละติจูด ลองติจูด และเวลาเปิดให้บริการ

ขั้นตอนที่ 2 ระบบตรวจสอบข้อมูลครบถ้วนหรือไม่

ขั้นตอนที่ 3 กรณีระบุข้อมูลครบถ้วน สามารถบันทึกข้อมูลลงในฐานข้อมูลได้ กรณีที่ระบุข้อมูลไม่ครบถ้วนต้องกลับไประบุข้อมูลให้ครบถ้วน

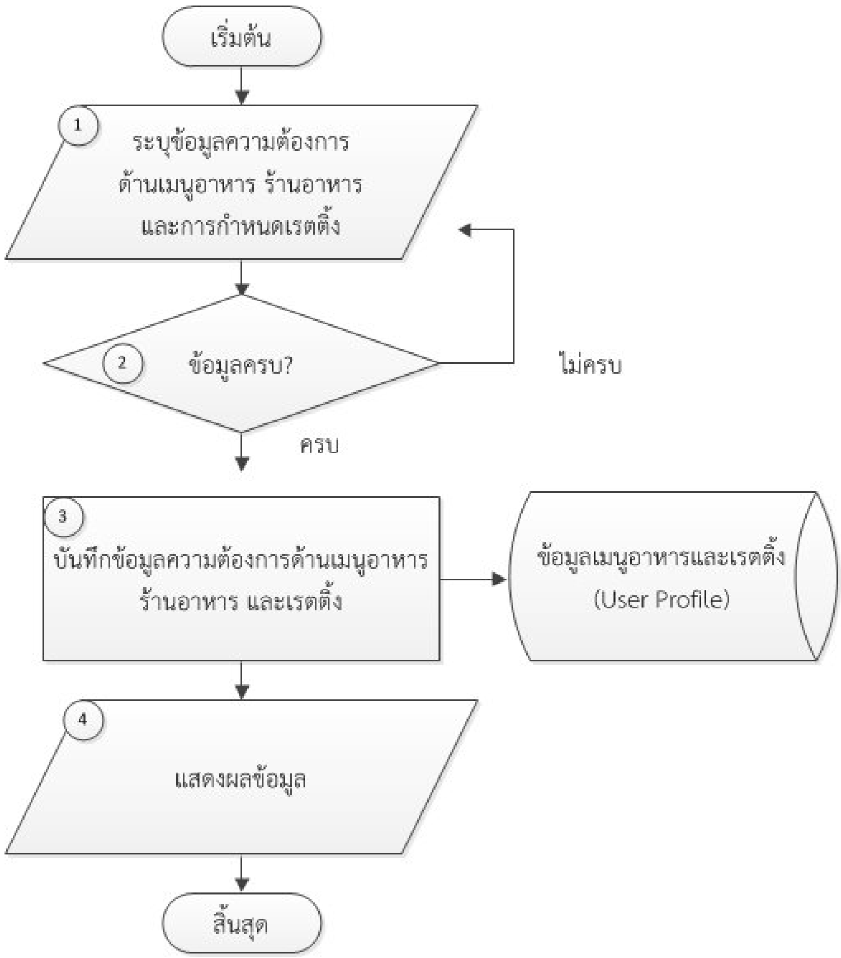
ขั้นตอนที่ 4 ผู้ใช้ระบุข้อมูลรายการอาหารของร้าน(Restaurant Profile) โดยระบุชื่อรายการอาหาร กลุ่มรายการอาหาร รูปภาพ และราคา

ขั้นตอนที่ 5 ระบบตรวจสอบข้อมูลครบถ้วนหรือไม่

ขั้นตอนที่ 6 กรณีระบุข้อมูลครบถ้วน สามารถบันทึกข้อมูลลงในฐานข้อมูลได้ กรณีที่ระบุข้อมูลไม่ครบถ้วนต้องกลับไประบุข้อมูลให้ครบถ้วน

ขั้นตอนที่ 7 แสดงผลข้อมูลการบันทึก

4) ระบบบันทึกความต้องการด้านรายการอาหารและความต้องการด้านร้านอาหารของผู้ใช้ (User Profile)



รูปที่ 3-14 ระบบบันทึกความต้องการด้านรายการอาหารและความต้องการด้านร้านอาหารของผู้ใช้ (User Profile)

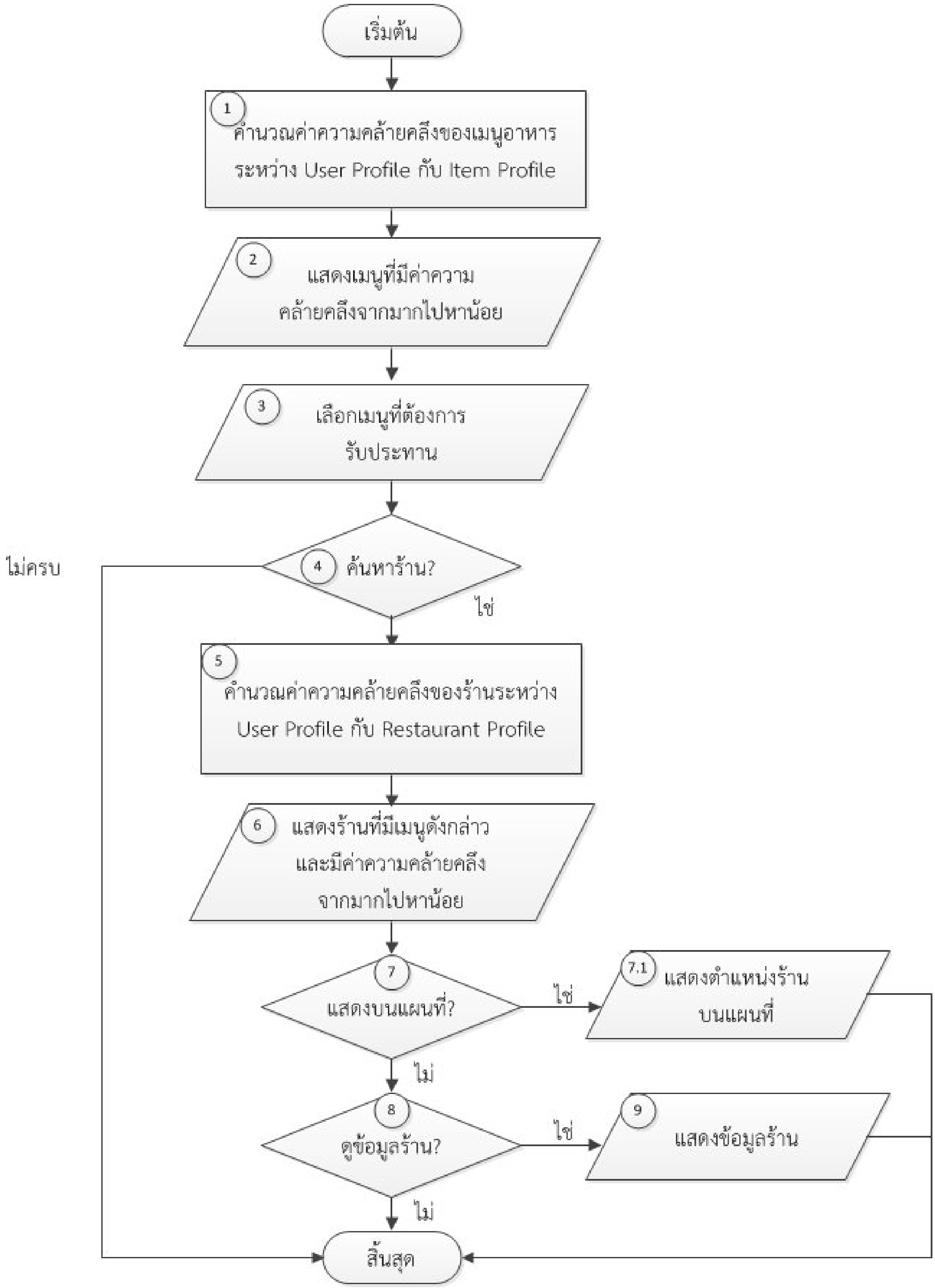
ขั้นตอนที่ 1 ผู้ใช้ระบุความต้องการด้านร้านอาหาร (User Profile) ระดับดาว ฮาลาล ลานจอดรถ แอร์ ระยะทาง และราคา รวมทั้งกำหนดคะแนนความชอบรับประทานให้กับเนื้อสัตว์ ผัก และรสชาติ

ขั้นตอนที่ 2 ระบบตรวจสอบข้อมูลครบถ้วนหรือไม่

ขั้นตอนที่ 3 กรณีระบุข้อมูลครบถ้วน สามารถบันทึกข้อมูลลงในฐานข้อมูลได้ กรณีที่ระบุข้อมูลไม่ครบถ้วนต้องกลับไประบุข้อมูลให้ครบถ้วน

ขั้นตอนที่ 4 แสดงผลข้อมูลการบันทึก

5) ระบบแนะนำรายการอาหารและร้านอาหาร (Recommender Systems)



รูปที่ 3-15 ระบบแนะนำรายการอาหารและร้านอาหาร (Recommender Systems)

ขั้นตอนที่ 1 ระบบประมวลผล โดยการหาค่าความคล้ายคลึงของรายการอาหาร

ขั้นตอนที่ 2 แสดงผลรายการอาหารที่มีค่าความคล้ายคลึงจากมากไปหาน้อย

ขั้นตอนที่ 3 ผู้ใช้เลือกรายการรายการอาหารที่ต้องการ สามารถเลือกได้มากกว่า1 รายการ

ขั้นตอนที่ 4 เลือกค้นหาร้านอาหารที่มีรายการอาหารดังกล่าวหรือไม่

ขั้นตอนที่ 5 ระบบประมวลผล โดยการหาค่าความคล้ายคลึงของร้านอาหารที่มีรายการอาหารดังกล่าวและมีค่าความคล้ายคลึงของร้านอาหารจากมากไปหาน้อย

ขั้นตอนที่ 6 แสดงรายการร้านอาหารที่มีค่าความคล้ายคลึงจากมากไปหาน้อย

ขั้นตอนที่ 7 เลือกดูร้านอาหารบนแผนที่ Google Map หรือไม่

ขั้นตอนที่ 7.1 แสดงร้านอาหารบนแผนที่ Google Map ระบบจะทำการปักหมุดร้านอาหารดังกล่าวบนแผนที่

ขั้นตอนที่ 8 เลือกดูข้อมูลรายละเอียดร้านอาหารหรือไม่

ขั้นตอนที่ 9 แสดงข้อมูลรายละเอียดของร้านอาหาร

3.5.2 ส่วนติดต่อผู้ใช้ของระบบ (User Interface)

1) หน้าเข้าสู่ระบบ (LogIn) และ หน้าสมัครสมาชิก (Register)

- หน้าเข้าสู่ระบบ สามารถเข้าสู่ระบบด้วยอีเมล (Email Address) และรหัสผ่าน (Password)
- หน้าสมัครสมาชิก ผู้ใช้สมัครสมาชิกโดยใช้ข้อมูลดังต่อไปนี้ ชื่อ (Name) อีเมล (Email Address) รหัสผ่าน (Password) เพศ (Gender) และเบอร์โทรศัพท์ (Phone Number)

รูปที่ 3-16 หน้าเข้าสู่ระบบ (LogIn) และ หน้าสมัครสมาชิก (Register)

2) หน้าหลัก (Home) และหน้าเมนูหลัก (Main Menu)

- หน้าหลัก (Home) แสดงร้านอาหารมาใหม่ และตำแหน่ง ณ ปัจจุบัน ของผู้ใช้

- หน้าเมนูหลัก (Main Menu) ประกอบด้วยเมนูค้นหาร้านอาหาร เมนูแนะนำร้านอาหาร เมนูตั้งค่าโปรไฟล์ เมนูแสดงสถานที่ เมนูเข้าสู่ระบบ (Log In) และเมนูออกจากระบบ (Log Out)

รูปที่ 3-17 หน้าหลัก (Home) และหน้าเมนูหลัก (Main Menu)

3) หน้าตั้งค่าโปรไฟล์ Profile

- หน้าตั้งค่าโปรไฟล์ (Profile) ประกอบด้วย 2 ส่วนหลัก ได้แก่ กำหนดความต้องการด้านรายการอาหาร และกำหนดความต้องการด้านร้านอาหาร

รูปที่ 3-18 หน้าตั้งค่าโปรไฟล์ (Profile)

4) หน้าแนะนำรายการอาหารและร้านอาหาร

- หน้าแนะนำรายการอาหาร แสดงรายการอาหารที่มีความคล้ายคลึงกับความต้องการของผู้ใช้

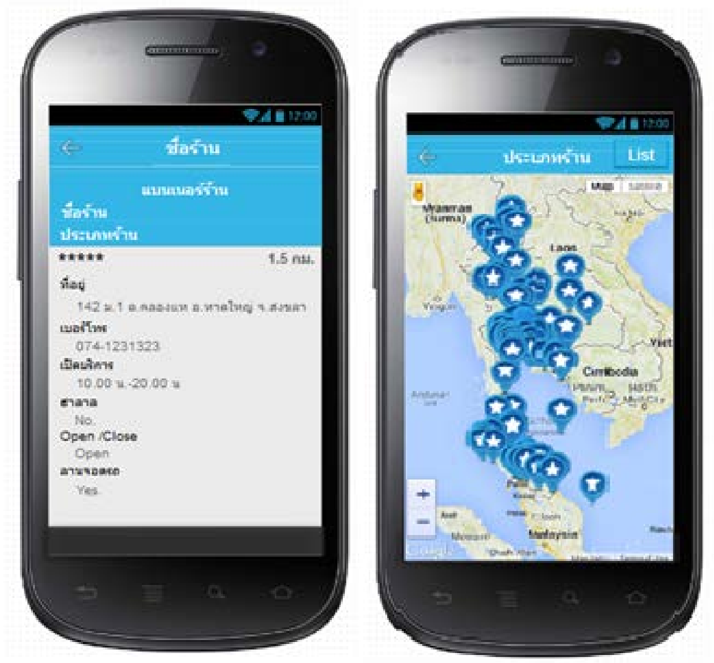
- หน้าแนะนำร้านอาหาร แสดงร้านอาหารที่มีความคล้ายคลึงกับความต้องการของผู้ใช้

รูปที่ 3-19 หน้าแนะนำรายการอาหารและร้านอาหาร

5) หน้าแสดงรายละเอียดร้านอาหาร และ แสดงที่ตั้งของร้านอาหารบนแผนที่

- หน้าแสดงรายละเอียดร้านอาหาร แสดงข้อมูลเบื้องต้นของร้านอาหาร ได้แก่ชื่อร้าน ที่อยู่ เบอร์โทรศัพท์ เวลาเปิดให้บริการ ลักษณะของร้าน และเครื่องหมายฮาลาล

- หน้าแสดงร้านอาหารบนแผนที่ แสดงตำแหน่งที่ตั้งของร้านอาหารบนแผนที่ Google Maps



รูปที่ 3-20 หน้าแสดงรายละเอียดร้านอาหาร และแสดงที่ตั้งของร้านอาหารบนแผนที่

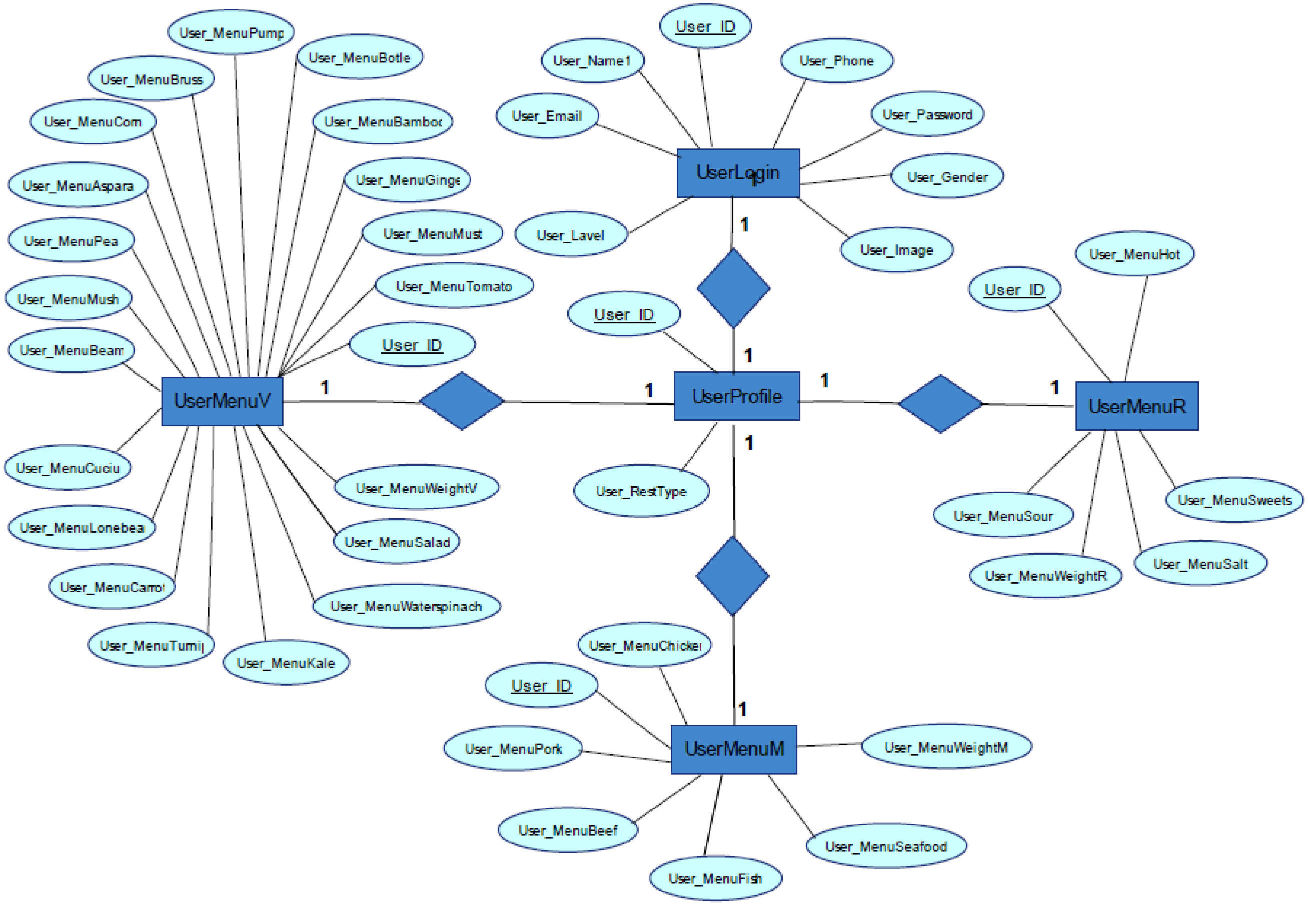
6) หน้าค้นหาร้านอาหาร

- หน้าค้นหาร้านอาหาร แสดงประเภทของร้านอาหาร ผู้ใช้สามารถเลือกประเภทของร้านอาหารเพื่อค้นหาชื่อร้านต่างๆ ได้

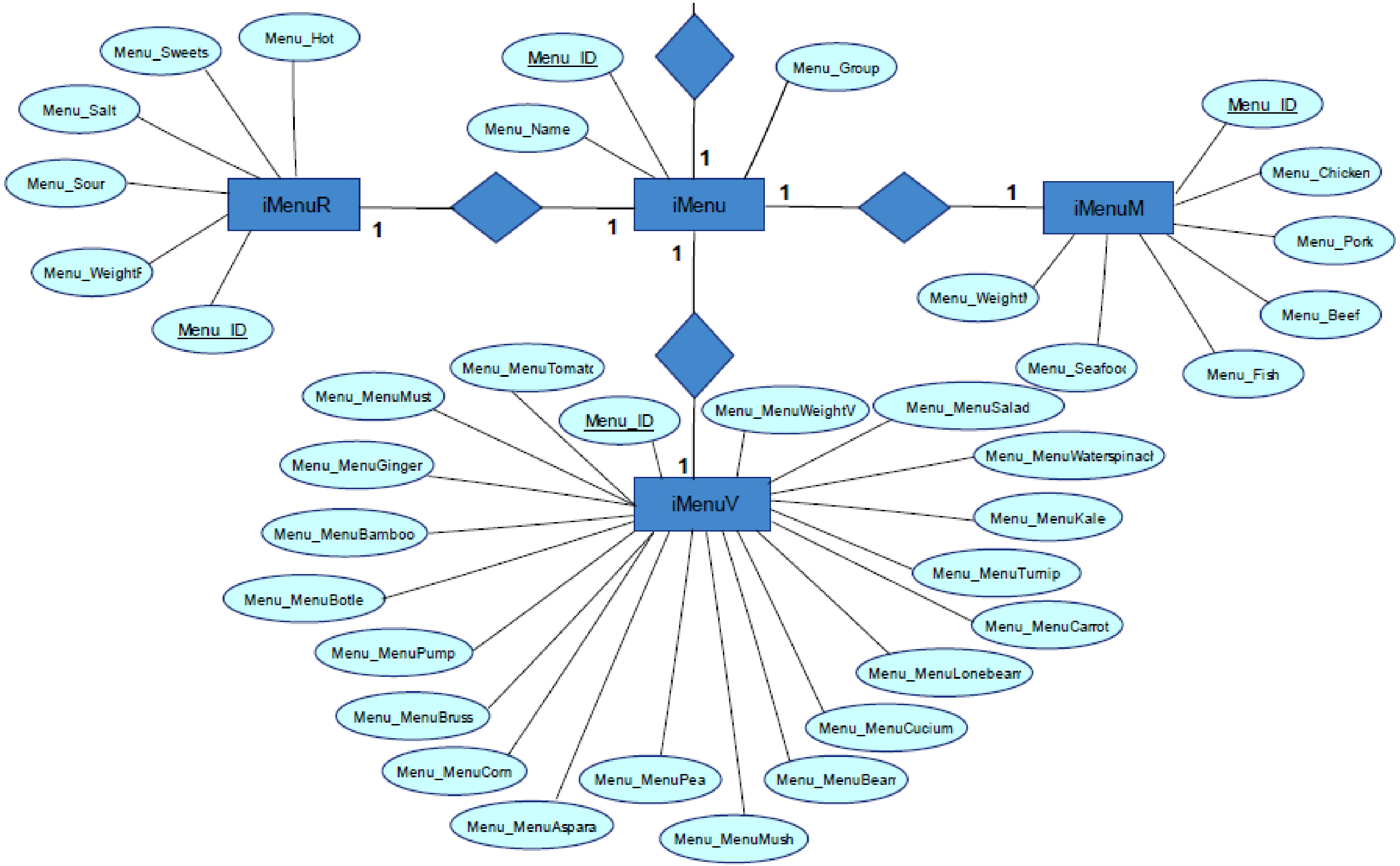
รูปที่ 3-21 หน้าค้นหาร้านอาหาร

3.5.3 การออกแบบฐานข้อมูล ประกอบด้วย แผนภาพแสดงความสัมพันธ์ (E-R Diagram) ตารางพจนานุกรมข้อมูล (Data Dictionary)

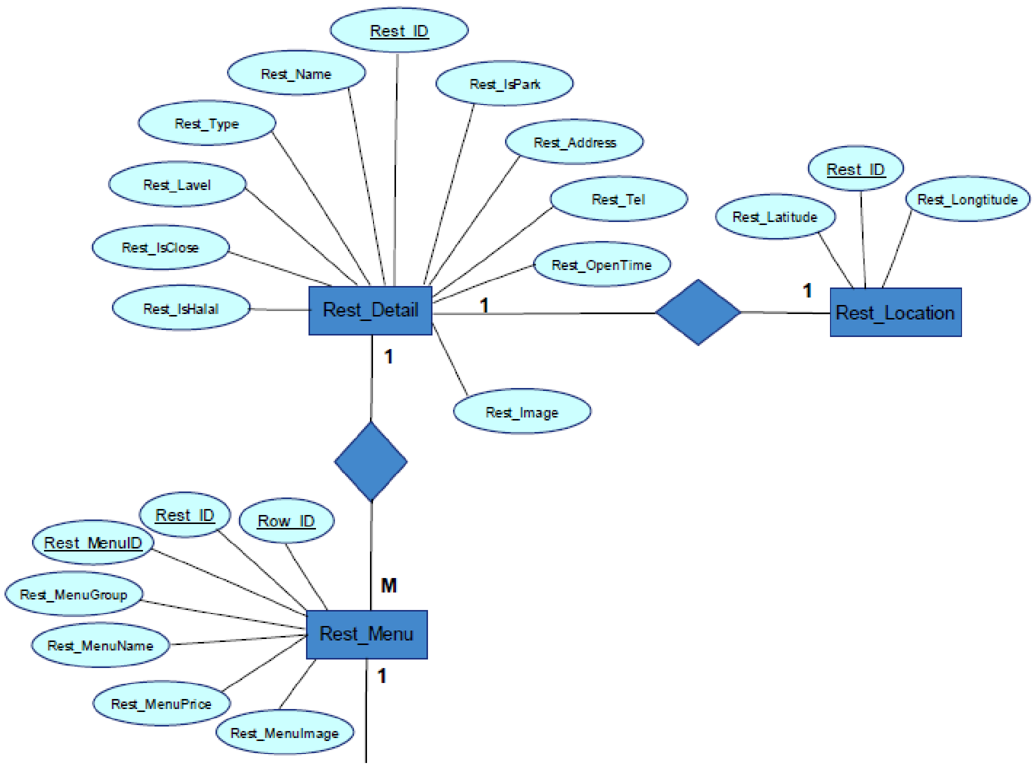
แผนภาพแสดงความสัมพันธ์ จะแสดงให้เห็นถึงความสัมพันธ์ของข้อมูลที่มีต่อกันระบบที่เกี่ยวข้องกัน ดังรูปที่ 3-22 ถึง 3-24



รูปที่ 3-22 E-R Diagram ความสัมพันธ์ของตารางผู้ใช้งาน



รูปที่ 3-23 E-R Diagram ความสัมพันธ์ของตารางรายการอาหารตั้งต้น



รูปที่ 3-24 E-R Diagram ความสัมพันธ์ของตารางร้านอาหาร

ตารางพจนานุกรมข้อมูล จะเป็นข้อมูลรายละเอียดจาก E-R Diagram ที่ได้ออกแบบไว้สามารถกำหนดรายละเอียดของแต่ละตารางในรูปแบบโครงสร้างของฐานข้อมูล ดังแสดงในตารางที่ 3-20 ถึง ตารางที่ 3-31

ตารางที่ 3-20 รายการอาหาร

Attribute	Type	size	Description	Null	Key
Menu_ID	INT	12	รหัสรายการอาหาร	No	PK
Menu_Name	VARCHAR	255	ชื่อรายการอาหาร	No	
Menu_Group	INT	12	กล่มรายการอาหาร	Yes	
Group_Image	VARCHAR		รูปรายการอาหาร	Yes	

ตารางที่ 3-21 ส่วนประกอบเนื้อสัตว์ของรายการอาหาร

Attribute	Type	size	Description	Null	Key
Menu_ID	INT	12	รหัสรายการอาหาร	No	PK
Menu_Chicken	DECIMAL	(12,2)	รายการอาหารที่มีส่วนประกอบของไก่ 0-5 โดยที่ 0:ไม่มี 5:มีเป็นหลัก	No	
Menu_Pork	DECIMAL	(12,2)	รายการอาหารที่มีส่วนประกอบของหมู 0-5 โดยที่ 0:ไม่มี 5:มีเป็นหลัก	No	
Menu_Beef	DECIMAL	(12,2)	รายการอาหารที่มีส่วนประกอบของเนื้อวัว 0-5 โดยที่ 0:ไม่มี 5:มีเป็นหลัก	No	
Menu_Fish	DECIMAL	(12,2)	รายการอาหารที่มีส่วนประกอบของปลา 0-5 โดยที่ 0:ไม่มี 5:มีเป็นหลัก	No	
Menu_Seafood	DECIMAL	(12,2)	รายการอาหารที่มีส่วนประกอบของอาหารทะเล 0-5 โดยที่ 0:ไม่มี 5:มีเป็นหลัก	No	
Menu_WeightM	DECIMAL	(12,2)	ค่าน้ำหนักของส่วนประกอบเนื้อสัตว์มีค่าระหว่าง 0-1	No	

ตารางที่ 3-22 ส่วนประกอบผักของรายการอาหาร

Attribute	Type	size	Description	Null	Key
Menu_ID	INT	12	รหัสรายการอาหาร	No	PK
Menu_MenuTomato	DECIMAL	(12,2)	รายการอาหารที่มีส่วนประกอบของมะเขือเทศ 0-5 โดยที่ 0:ไม่มี 5:มีเป็นหลัก	No	
Menu_MenuMust	DECIMAL	(12,2)	รายการอาหารที่มีส่วนประกอบของผักกาด 0-5 โดยที่ 0:ไม่มี 5:มีเป็นหลัก	No	
Menu_MenuGinger	DECIMAL	(12,2)	รายการอาหารที่มีส่วนประกอบของขิง 0-5 โดยที่ 0:ไม่มี 5:มีเป็นหลัก	No	
Menu_MenuBamboo	DECIMAL	(12,2)	รายการอาหารที่มีส่วนประกอบของหน่อไม้ 0-5 โดยที่ 0:ไม่มี 5:มีเป็นหลัก	No	
Menu_MenuBotle	DECIMAL	(12,2)	รายการอาหารที่มีส่วนประกอบของน้ำเต้า 0-5 โดยที่ 0:ไม่มี 5:มีเป็นหลัก	No	
Menu_MenuPump	DECIMAL	(12,2)	รายการอาหารที่มีส่วนประกอบของฟักทอง 0-5 โดยที่ 0:ไม่มี 5:มีเป็นหลัก	No	
Menu_MenuBruss	DECIMAL	(12,2)	รายการอาหารที่มีส่วนประกอบของกะหล่ำปลี 0-5 โดยที่ 0:ไม่มี 5:มีเป็นหลัก	No	
Menu_MenuCorn	DECIMAL	(12,2)	รายการอาหารที่มีส่วนประกอบของข้าวโพด 0-5 โดยที่ 0:ไม่มี 5:มีเป็นหลัก	No	
Menu_MenuAspara	DECIMAL	(12,2)	รายการอาหารที่มีส่วนประกอบของหน่อไม้ฝรั่ง 0-5 โดยที่ 0:ไม่มี 5:มีเป็นหลัก	No	
Menu_MenuPea	DECIMAL	(12,2)	รายการอาหารที่มีส่วนประกอบของถั่วลันเตา 0-5 โดยที่ 0:ไม่มี 5:มีเป็นหลัก	No	
Menu_MenuMush	DECIMAL	(12,2)	รายการอาหารที่มีส่วนประกอบของเห็ด	No	

Attribute	Type	size	Description	Null	Key
			0-5 โดยที่ 0:ไม่มี 5:มีเป็นหลัก		
Menu_MenuBeam	DECIMAL	(12,2)	รายการอาหารที่มีส่วนประกอบของถั่ว 0-5 โดยที่ 0:ไม่มี 5:มีเป็นหลัก	No	
Menu_MenuCucium	DECIMAL	(12,2)	รายการอาหารที่มีส่วนประกอบของแตงกวา 0-5 โดยที่ 0:ไม่มี 5:มีเป็นหลัก	No	
Menu_MenuLonebeam	DECIMAL	(12,2)	รายการอาหารที่มีส่วนประกอบของถั่วฝักยาว 0-5 โดยที่ 0:ไม่มี 5:มีเป็นหลัก	No	
Menu_MenuCarrot	DECIMAL	(12,2)	รายการอาหารที่มีส่วนประกอบของแครอท 0-5 โดยที่ 0:ไม่มี 5:มีเป็นหลัก	No	
Menu_MenuTurnip	DECIMAL	(12,2)	รายการอาหารที่มีส่วนประกอบของหัวผักกาด 0-5 โดยที่ 0:ไม่มี 5:มีเป็นหลัก	No	
Menu_MenuKale	DECIMAL	(12,2)	รายการอาหารที่มีส่วนประกอบของคะน้า 0-5 โดยที่ 0:ไม่มี 5:มีเป็นหลัก	No	
Menu_MenuWaterspinach	DECIMAL	(12,2)	รายการอาหารที่มีส่วนประกอบของผักบุ้ง 0-5 โดยที่ 0:ไม่มี 5:มีเป็นหลัก	No	
Menu_MenuSalad	DECIMAL	(12,2)	รายการอาหารที่มีส่วนประกอบของผักสลัด 0-5 โดยที่ 0:ไม่มี 5:มีเป็นหลัก	No	
Menu_MenuWeightV	DECIMAL	(12,2)	ค่าน้ำหนักของส่วนประกอบของผักมีค่าระหว่าง 0-1	No	

ตารางที่ 3-23 ส่วนประกอบรสชาติของรายการอาหาร

Attribute	Type	size	Description	Null	Key
Menu_ID	INT	12	รหัสรายการอาหาร	No	PK
Menu_Hot	DECIMAL	(12,2)	รายการอาหารที่มีรสชาติเผ็ด 0-5 โดยที่ 0:ไม่มี 5:มีเป็นหลัก	No	
Menu_Sweets	DECIMAL	(12,2)	รายการอาหารที่มีรสชาติหวาน 0-5 โดยที่ 0:ไม่มี 5:มีเป็นหลัก	No	
Menu_Salt	DECIMAL	(12,2)	รายการอาหารที่มีรสชาติเค็ม 0-5 โดยที่ 0:ไม่มี 5:มีเป็นหลัก	No	
Menu_Sour	DECIMAL	(12,2)	รายการอาหารที่มีรสชาติเปรี้ยว 0-5 โดยที่ 0:ไม่มี 5:มีเป็นหลัก	No	
Menu_WeightR	DECIMAL	(12,2)	ค่าน้ำหนักของรสชาติมีค่าระหว่าง 0-1	No	

ตารางที่ 3-24 ร้านอาหาร

Attribute	Type	size	Description	Null	Key
Rest_ID	INT	12	รหัสร้านอาหาร	No	PK
Rest_Name	VARCHAR	100	ชื่อร้านอาหาร	No	
Rest_Type	INT	3	ประเภทร้านอาหาร	No	
Rest_Lavel	INT	3	ระดับดาวของร้านอาหาร	No	
Rest_IsClose	INT	3	ลักษณะร้าน 0: แบบเปิดไม่มีแอร์ 1: แบบปิดมีแอร์	No	
Rest_IsHalal	INT	3	ฮาลาล 0: ไม่มีฮาลาล 1: มีฮาลาล	No	
Rest_IsPark	INT	3	ลานจอดรถ 0: ไม่มีลานจอดรถ 1: มีลาดจอดรถ	No	
Rest_Address	VARCHAR	255	ที่อยู่ของร้านอาหาร	Yes	
Rest_Tel	VARCHAR	20	เบอร์ติดต่อร้านอาหาร	Yes	
Rest_OpenTime	VARCHAR	100	เวลาเปิดให้บริการ	Yes	
Rest_Image	IMAGE		รูปประจำตัวร้านอาหาร	Yes	
Rest_Latitude	DECIMAL	(12,4)	ละติจูดของร้านอาหาร	Yes	
Rest_Longtitude	DECIMAL	(12,4)	ลองติจูดของร้านอาหาร	Yes	

ตารางที่ 3-25 รายการอาหาร

Attribute	Type	size	Description	Null	Key
Row_ID	INT	12	ลำดับ	No	PK
Rest_detail_ID	INT	12	รหัสร้านอาหาร	No	FK
Rest_MenuGroup	VARCHAR	100	กลุ่มของรายการอาหาร	No	
Rest_MenuID	INT	12	รหัสรายการอาหาร	No	FK
Rest_MenuName	VARCHAR	100	ชื่อรายการอาหาร	No	
Rest_MenuPrice	DECIMAL	(12,2)	ราคาอาหาร	No	
Rest_MenuImage	IMAGE		รูปรายการอาหาร		

ตารางที่ 3-26 สมาชิก

Attribute	Type	size	Description	Null	Key
User_ID	INT	12	รหัสสมาชิก	No	PK
User_Name	VARCHAR	50	ชื่อสมาชิก	No	
User_Email	VARCHAR	50	Emailสมาชิก	No	
User_Password	VARCHAR	30	Password	No	
User_Lavel	INT	3	ประเภทสมาชิก 0: Admin 1: สมาชิกทั่วไป	No	
User_Image	IMAGE		รูปประจำตัวสมาชิก	No	
User_Gender	VARCHAR	10	เพศ M:ชาย F:หญิง :ไม่ระบุ	No	

ตารางที่ 3-27 การกำหนดความต้องการส่วนประกอบเนื้อสัตว์ของสมาชิก

Attribute	Type	size	Description	Null	Key
User_ID	INT	12	รหัสสมาชิก	No	PK
User_MenuChicken	DECIMAL	(12,2)	ค่าความถี่หรือความชอบรับประทานไก่ 0-5 โดยที่ 0:ไม่ชอบเลย 5:ชอบมากที่สุด	No	
User_MenuPork	DECIMAL	(12,2)	ค่าความถี่หรือความชอบรับประทานหมู 0-5 โดยที่ 0:ไม่ชอบเลย 5:ชอบมากที่สุด	No	
User_MenuBeef	DECIMAL	(12,2)	ค่าความถี่หรือความชอบรับประทานเนื้อวัว 0-5 โดยที่ 0:ไม่ชอบเลย 5:ชอบมากที่สุด	No	
User_MenuFish	DECIMAL	(12,2)	ค่าความถี่หรือความชอบรับประทานปลา 0-5 โดยที่ 0:ไม่ชอบเลย 5:ชอบมากที่สุด	No	
User_MenuSeafood	DECIMAL	(12,2)	ค่าความถี่หรือความชอบรับประทานอาหารทะเล 0-5 โดยที่ 0:ไม่ชอบเลย 5:ชอบมากที่สุด	No	
User_MenuWeightM	DECIMAL	(12,2)	ค่าน้ำหนักของเนื้อที่ชอบรับประทานมีค่าระหว่าง 0-1	No	

ตารางที่ 3-28 การกำหนดความต้องการส่วนประกอบผักของสมาชิก

Attribute	Type	size	Description	Null	Key
User_ID	INT	12	รหัสสมาชิก	No	PK
User_MenuTomato	DECIMAL	(12,2)	ค่าความถี่หรือความชอบรับประทานมะเขือเทศ 0-5 โดยที่ 0:ไม่ชอบเลย 5:ชอบมากที่สุด	No	
User_MenuMust	DECIMAL	(12,2)	ค่าความถี่หรือความชอบรับประทานผักกาด 0-5 โดยที่ 0:ไม่ชอบเลย 5:ชอบมากที่สุด	No	
User_MenuGinger	DECIMAL	(12,2)	ค่าความถี่หรือความชอบรับประทานขิง 0-5 โดยที่ 0:ไม่ชอบเลย 5:ชอบมากที่สุด	No	

Attribute	Type	size	Description	Null	Key
User_MenuBamboo	DECIMAL	(12,2)	ค่าความถี่หรือความชอบรับประทานหน่อไม้ 0-5 โดยที่ 0:ไม่ชอบเลย 5:ชอบมากที่สุด	No	
User_MenuBotle	DECIMAL	(12,2)	ค่าความถี่หรือความชอบรับประทานน้ำเต้า 0-5 โดยที่ 0:ไม่ชอบเลย 5:ชอบมากที่สุด	No	
User_MenuPump	DECIMAL	(12,2)	ค่าความถี่หรือความชอบรับประทานฟักทอง 0-5 โดยที่ 0:ไม่ชอบเลย 5:ชอบมากที่สุด	No	
User_MenuBruss	DECIMAL	(12,2)	ค่าความถี่หรือความชอบรับประทานกะหล่ำปลี 0-5 โดยที่ 0:ไม่ชอบเลย 5:ชอบมากที่สุด	No	
User_MenuCorn	DECIMAL	(12,2)	ค่าความถี่หรือความชอบรับประทานข้าวโพด 0-5 โดยที่ 0:ไม่ชอบเลย 5:ชอบมากที่สุด	No	
User_MenuAspara	DECIMAL	(12,2)	ค่าความถี่หรือความชอบรับประทานหน่อไม้ฝรั่ง 0-5 โดยที่ 0:ไม่ชอบเลย 5:ชอบมากที่สุด	No	
User_MenuPea	DECIMAL	(12,2)	ค่าความถี่หรือความชอบรับประทานถั่วลันเตา 0-5 โดยที่ 0:ไม่ชอบเลย 5:ชอบมากที่สุด	No	
User_MenuMush	DECIMAL	(12,2)	ค่าความถี่หรือความชอบรับประทานเห็ด 0-5 โดยที่ 0:ไม่ชอบเลย 5:ชอบมากที่สุด	No	
User_MenuBeam	DECIMAL	(12,2)	ค่าความถี่หรือความชอบรับประทานถั่ว	No	

Attribute	Type	size	Description	Null	Key
			0-5 โดยที่ 0:ไม่ชอบเลย 5:ชอบมากที่สุด		
User_MenuCucium	DECIMAL	(12,2)	ค่าความถี่หรือความชอบรับประทานแตงกวา 0-5 โดยที่ 0:ไม่ชอบเลย 5:ชอบมากที่สุด	No	
User_MenuLonebeam	DECIMAL	(12,2)	ค่าความถี่หรือความชอบรับประทานถั่วฝักยาว 0-5 โดยที่ 0:ไม่ชอบเลย 5:ชอบมากที่สุด	No	
User_MenuCarrot	DECIMAL	(12,2)	ค่าความถี่หรือความชอบรับประทานแครอท 0-5 โดยที่ 0:ไม่ชอบเลย 5:ชอบมากที่สุด	No	
User_MenuTurnip	DECIMAL	(12,2)	ค่าความถี่หรือความชอบรับประทานหัวผักกาด 0-5 โดยที่ 0:ไม่ชอบเลย 5:ชอบมากที่สุด	No	
User_MenuKale	DECIMAL	(12,2)	ค่าความถี่หรือความชอบรับประทานคะน้า 0-5 โดยที่ 0:ไม่ชอบเลย 5:ชอบมากที่สุด	No	
User_MenuWaterspinach	DECIMAL	(12,2)	ค่าความถี่หรือความชอบรับประทานผักบุ้ง 0-5 โดยที่ 0:ไม่ชอบเลย 5:ชอบมากที่สุด	No	
User_MenuSalad	DECIMAL	(12,2)	ค่าความถี่หรือความชอบรับประทานผักสลัด 0-5 โดยที่ 0:ไม่ชอบเลย 5:ชอบมากที่สุด	No	
User_MenuWeightV	DECIMAL	(12,2)	ค่าน้ำหนักของผักที่ชอบรับประทานมีค่าระหว่าง 0-1	No	

ตารางที่ 3-29 การกำหนดความต้องการรสชาติของสมาชิก

Attribute	Type	size	Description	Null	Key
User_ID	INT	12	รหัสสมาชิก	No	PK
User_MenuHot	DECIMAL	(12,2)	ค่าความถี่หรือความชอบรับประทานรสเผ็ด 0-5 โดยที่ 0:ไม่ชอบเลย 5:ชอบมากที่สุด	No	
User_MenuSweets	DECIMAL	(12,2)	ค่าความถี่หรือความชอบรับประทานรสหวาน 0-5 โดยที่ 0:ไม่ชอบเลย 5:ชอบมากที่สุด	No	
User_MenuSalt	DECIMAL	(12,2)	ค่าความถี่หรือความชอบรับประทานรสเค็ม 0-5 โดยที่ 0:ไม่ชอบเลย 5:ชอบมากที่สุด	No	
User_MenuSour	DECIMAL	(12,2)	ค่าความถี่หรือความชอบรับประทานรสเปรี้ยว 0-5 โดยที่ 0:ไม่ชอบเลย 5:ชอบมากที่สุด	No	
User_MenuWeightR	DECIMAL	(12,2)	ค่าน้ำหนักของรสชาติที่ชอบรับประทานมีค่าระหว่าง 0-1	No	

ตารางที่ 3-30 การกำหนดความต้องการด้านร้านอาหารของสมาชิก

Attribute	Type	size	Description	Null	Key
User_ID	INT	12	รหัสสมาชิก	No	PK
User_Rest_Type	INT	3	ประเภทร้านอาหาร	No	
User_Rest_Lavel	INT	3	ระดับดาวของร้านอาหาร	No	
User_Rest_IsClose	INT	3	ลักษณะร้าน 0: แบบเปิดไม่มีแอร์ 1: แบบปิดมีแอร์	No	
User_Rest_IsHalal	INT	3	ฮาลาล 0: ไม่มีฮาลาล 1: มีฮาลาล	No	
User_Rest_IsPark	INT	3	ลานจอดรถ 0: ไม่มีลานจอดรถ 1: มีลาดจอดรถ	No	
User_Rest_Distance	INT	4	ระยะทาง กิโลเมตร	No	

ตารางที่ 3-31 ประเภทร้านอาหารของสมาชิก

Attribute	Type	size	Description	Null	Key
User_ID	INT	12	รหัสสมาชิก	No	PK
User_Menu_Group	INT	3	ประเภทรายการอาหาร 1.แกง-ต้ม 2.ผัด-ทอด 3. ตุ๋น-นึ่ง	No	
User_Menu_Price	INT	3	ราคา 1:น้อยกว่า100 2:100-199 3:200-299 4:300-399 5:400 ขึ้นไป	No	

3.6 การพัฒนาระบบ

ในขั้นตอนการพัฒนาระบบ ผู้วิจัยได้กำหนดกระบวนการทำงานหลักของระบบไว้ 3 ส่วน คือ 1) การพัฒนาเว็บแอปพลิเคชัน (Back-End) 2) การพัฒนาแอปพลิเคชันบนสมาร์ทโฟน โดยมีรายละเอียด ดังนี้

3.6.1 การพัฒนาเว็บแอปพลิเคชัน(Back-End)

ผู้วิจัยได้พัฒนาเว็บแอปพลิเคชันสำหรับบันทึกข้อมูลรายการอาหารตั้งต้น และบันทึกข้อมูลร้านอาหารด้วยภาษา HTML5 โดยใช้เครื่องมือในการพัฒนา ได้แก่ October CMS ซึ่งสามารถนำมาใช้ในการพัฒนาได้โดยไม่เสียค่าใช้จ่าย และง่ายในการพัฒนาระบบได้อย่างรวดเร็ว

ผู้วิจัยได้พัฒนาเว็บเซอร์วิส (RestFul Web Service) สำหรับประมวลผลการทำงานฟังก์ชันต่างๆ ของระบบแนะนำร้านอาหารอัตโนมัติบนสมาร์ทโฟนโดยใช้ข้อมูลเชิงตำแหน่งและรายการอาหาร ด้วยภาษา PHP ประกอบด้วยส่วนการทำงานหลักๆ ดังตารางที่ 3-22

ตารางที่ 3-22 ฟังก์ชันการทำงานของ Restful Webserivce

ลำดับ	API Name	URL	อธิบาย
1	Recommend Menu	/api/findmenu	แนะนำรายการอาหาร
2	Recoomend Restaurant	/api/findrestaurant	แนะนำร้านอาหาร
3	Home	/api/home	เรียกข้อมูลหน้าหลัก
4	Restaurant	/api/restaurant	เรียกข้อมูลร้านอาหาร
5	Find Restaurant by id	/api/restaurant/byids	ค้นหาร้านอาหารโดยใช้ข้อมูลรหัสร้านอาหาร
6	Find Restaurant by idMenu	/api/restaurant/:id/menus	ค้นหาร้านอาหารโดยใช้รายการอาหาร
7	Find Menu by id	/api/menu/:id	ค้นหารายการอาหารโดยใช้รหัสของรายการอาหาร
8	Restype	/api/resttypes	เรียกข้อมูลประเภทร้านอาหาร
9	User Login	/api/user/login	เข้าสู่ระบบ,ออกจากระบบ
10	User Register	/api/user/register	สมัครสมาชิก

ลำดับ	API Name	URL	อธิบาย
11	User Set Profile	/api/user/update-profile	แก้ไขโปรไฟล์
12	User Get Profile	/api/user/profile	ตั้งค่าโปรไฟล์
13	Vote Restaurant	/api/vote/restaurant/:id	โหวดให้คะแนนร้านอาหาร

3.6.2 การพัฒนาแอปพลิเคชันบนสมาร์ทโฟน

ผู้วิจัยได้พัฒนาแอปพลิเคชันบนสมาร์ทโฟน สำหรับแนะนำร้านอาหารอัตโนมัติบนสมาร์ทโฟนโดยใช้ข้อมูลเชิงตำแหน่งและรายการอาหาร ได้ประยุกต์ใช้ระบบแผนที่จาก Google Maps เพื่อแสดงตำแหน่งของร้านอาหารบนแผนที่ ซึ่งได้พัฒนาด้วยซอฟท์แวร์ระบบเปิด (Open Source) ที่ชื่อว่า “Apache Cordova” ซึ่งสามารถนำมาใช้ในการพัฒนาได้โดยไม่เสียค่าใช้จ่าย และสามารถสร้าง (Build) ไปเป็นแอปพลิเคชันให้ทำงานได้ทั้งระบบปฏิบัติการ IOS และ Android

3.7 ทดสอบระบบและการประเมินผล

3.7.1 Unit Testing ตรวจสอบการทำงานของระบบ ซึ่งแบ่งเป็น 2 ระบบ ดังนี้ 1) แอปพลิเคชันบนสมาร์ทโฟน 2) เว็บแอปพลิเคชัน

1) เว็บแอปพลิเคชัน ทดสอบส่วนต่างๆ ของระบบดังต่อไปนี้ ส่วนบันทึกข้อมูลรายการอาหารตั้งต้นและการกำหนดคะแนนความชอบ ส่วนบันทึกประเภทร้านอาหาร ส่วนบันทึกข้อมูลร้านอาหารและรายการอาหารของร้าน เป็นต้น

2) แอปพลิเคชันบนสมาร์ทโฟน ทดสอบส่วนต่างๆ ของระบบดังต่อไปนี้ ส่วนสมัครสมาชิก ส่วนตั้งค่าโปรไฟล์ ส่วนแนะนำรายการอาหารและร้านอาหาร ส่วนค้นหาร้านอาหาร เป็นต้น

3.7.2 Integration Testing ทดสอบภาพรวมทั้งหมดของระบบแนะนำรายการอาหารอัตโนมัติบนสมาร์ทโฟนโดยใช้ข้อมูลเชิงตำแหน่งและรายการอาหาร โดยการทดสอบระบบตั้งแต่เริ่มสมัครสมาชิก เข้าสู่ระบบ ตั้งค่าโปรไฟล์ และ แนะนำรายการอาหารและร้านอาหาร ตามลำดับ ระบบสามารถทำงานได้อย่างถูกต้องตามขั้นตอนและกระบวนการหรือไม่ รวมถึงการแสดงผลข้อมูลถูกต้องหรือไม่

3.7.3 System Testing ประเมินคุณภาพระบบแนะนำร้านอาหารอัตโนมัติบนสมาร์ทโฟนโดยใช้ข้อมูลเชิงตำแหน่งและรายการอาหาร โดยผู้เชี่ยวชาญจำนวน 5 ท่าน และผู้ใช้งานทั่วไปจำนวน 30 คน

3.7.4 Acceptance Testing ปรับปรุงแก้ไขระบบแนะนำร้านอาหารอัตโนมัติบนสมาร์ทโฟนโดยใช้ข้อมูลเชิงตำแหน่งและรายการอาหาร ตามคำแนะนำของผู้เชี่ยวชาญและผู้ใช้งานทั่วไปของแบบประเมินที่อยู่ในเอกสารแนบท้าย

บทที่ 4
ผลการวิจัย

ผลการศึกษางานวิจัยในครั้งนี้ได้นำเสนอในรูปแบบรูปภาพและตารางประกอบคำอธิบาย โดยแบ่งผลการศึกษางานวิจัยดังนี้

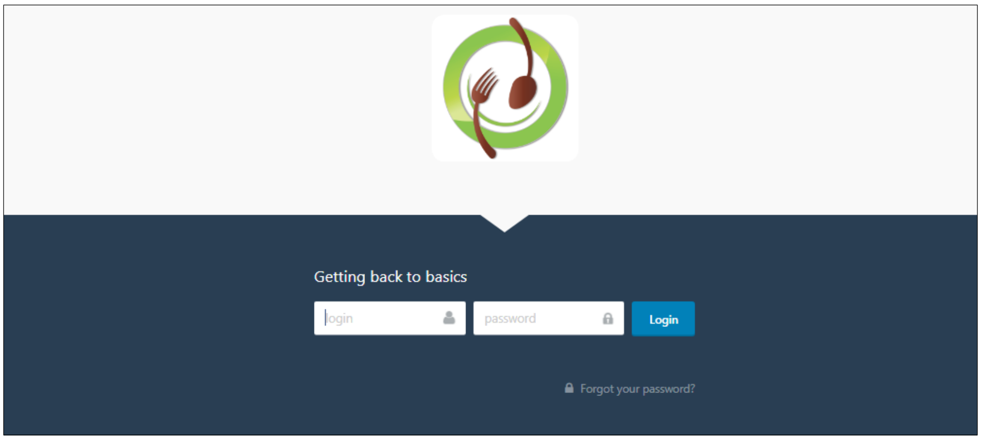
4.1 ผลการพัฒนาระบบแนะนำรายการอาหารอัตโนมัติบนสมาร์ทโฟนโดยใช้ข้อมูลเชิงตำแหน่งและรายการอาหาร

ระบบแนะนำรายการอาหารอัตโนมัติบนสมาร์ทโฟนโดยใช้ข้อมูลเชิงตำแหน่งและรายการอาหารประกอบด้วย 2 ส่วน ได้แก่ 1) เว็บแอปพลิเคชัน 2) แอพลิเคชันบนสมาร์ทโฟน

4.1.1 เว็บแอปพลิเคชัน สร้างขึ้นสำหรับบันทึกข้อมูลรายการอาหารตั้งต้น และการกำหนด คะแนนความชอบให้กับส่วนประกอบต่างๆ ของรายการอาหารโดยผู้ดูแลระบบ และสำหรับบันทึกข้อมูลร้านอาหารและรายการอาหารโดยผู้ประกอบการร้านอาหารหรือผู้บริโภคทั่วไป

4.1.1.1 สำหรับผู้ดูแลระบบ จะประกอบไปด้วย 1) หน้าเข้าสู่ระบบ 2) หน้าหลัก 3) หน้าบันทึกข้อมูลรายการอาหารตั้งต้นและการกำหนดคะแนนความชอบให้กับส่วนประกอบต่างๆ ของรายการอาหาร เป็นต้น 4) หน้าบันทึกข้อมูลประเภทร้านอาหาร

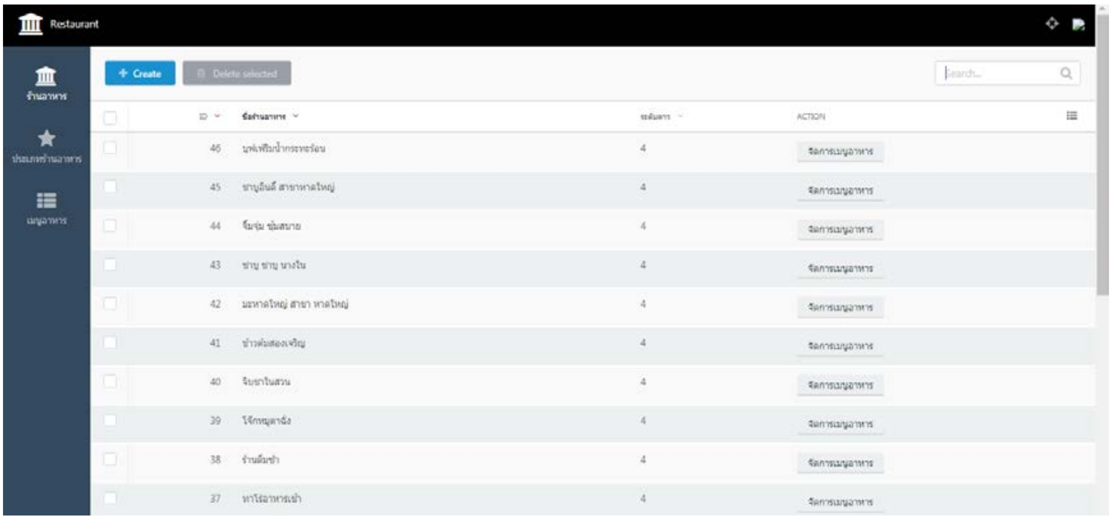
1) หน้าเข้าสู่ระบบ (Log In) เพื่อใช้ในการ Log In เข้าใช้งานระบบ



รูปที่ 4-1 หน้าเข้าสู่ระบบ

จากรูปที่ 4-1 แสดงหน้าเข้าสู่ระบบ โดยประกอบด้วยฟิวล์ข้อมูลที่จำเป็นสำหรับการ Log In เข้าสู่ระบบ ได้แก่ User Name และ Password และปุ่ม LogIn

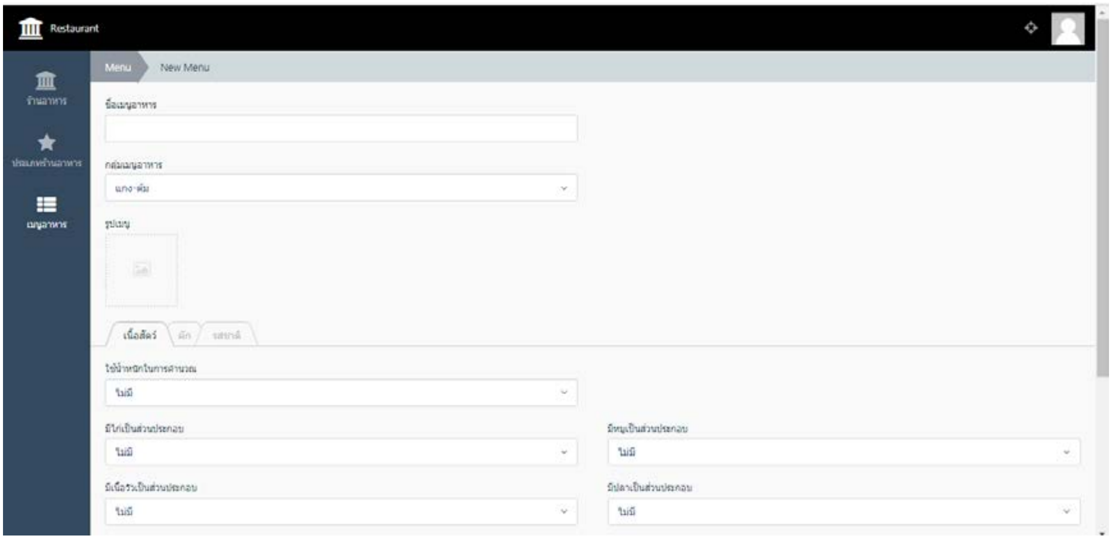
2) หน้าหลัก (Home) เป็นหน้าแรกของโปรแกรมเมื่อผู้ใช้ทำการ Log In เข้าสู่ระบบได้สำเร็จ



รูปที่ 4-2 หน้าหลัก

จากรูปที่ 4-2 ประกอบด้วย แถบเมนูหลัก มี 3 เมนู ได้แก่ เมนูร้านอาหาร เมนูประเภทร้านอาหาร เมนูเมนูอาหาร เมื่อคลิกเลือกเมนูจะแสดงรายการข้อมูลนั้น ๆ

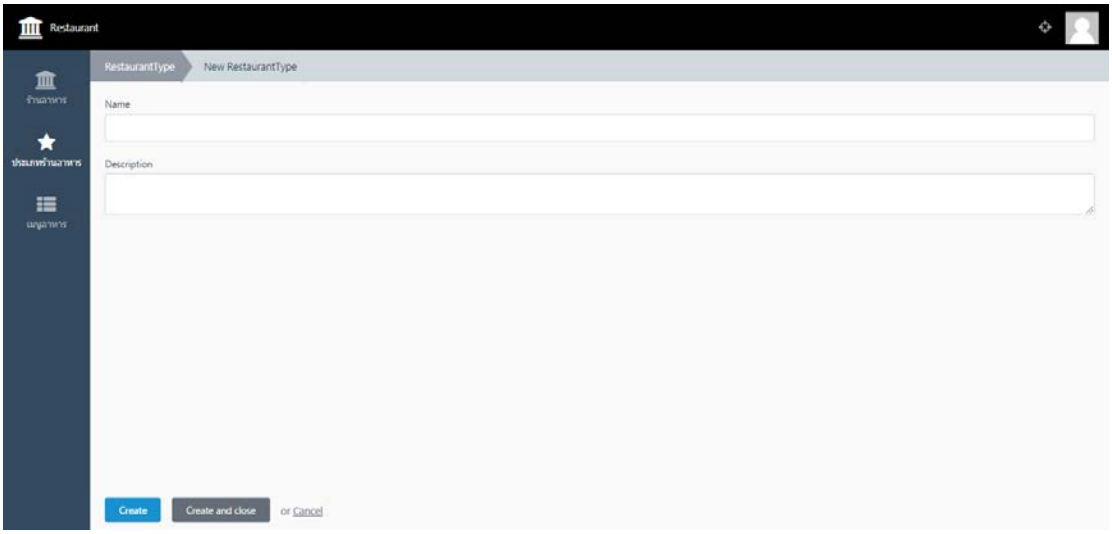
3) หน้าบันทึกข้อมูลรายการอาหารตั้งต้นและการกำหนดคะแนนความชอบให้กับส่วนประกอบต่างๆ ของรายการอาหาร



รูปที่ 4-3 หน้าบันทึกข้อมูลรายการอาหารตั้งต้นและการกำหนดคะแนนความชอบให้กับส่วนประกอบของรายการอาหาร

จากรูปที่ 4-3 ประกอบด้วยฟิวล์ข้อมูลเกี่ยวกับรายละเอียดของรายการอาหาร เช่น ชื่อเมนูอาหาร กลุ่มของเมนูอาหาร รูปเมนูอาหาร การกำหนดค่าคะแนนความชอบและค่าน้ำหนักของส่วนประกอบต่างๆ และปุ่ม Create สำหรับบันทึก

4) หน้าบันทึกข้อมูลประเภทร้านอาหาร

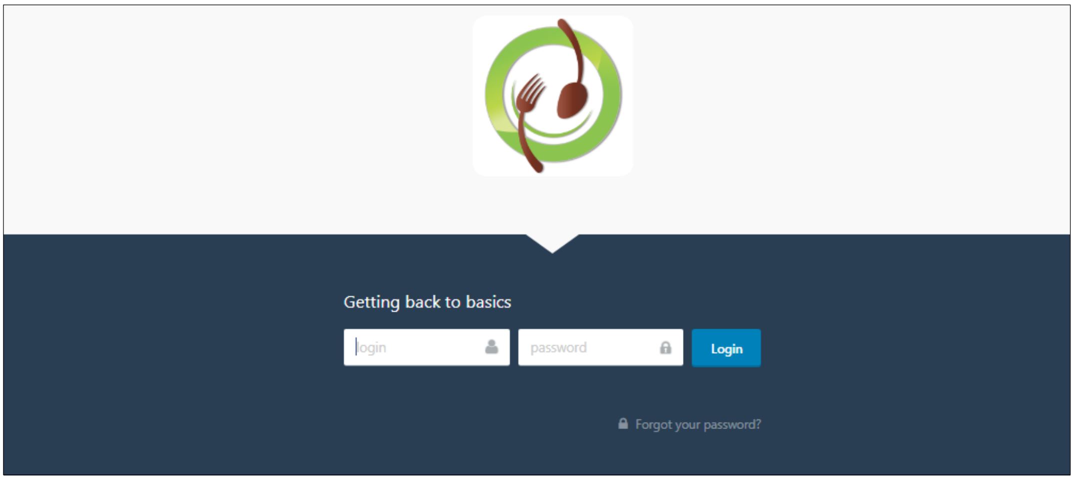


รูปที่ 4-4 หน้าบันทึกข้อมูลประเภทอาหาร

จากรูปที่ 4-4 ประกอบด้วย ฟิวล์ข้อมูล ชื่อประเภทร้านอาหาร และรายละเอียดสำหรับกำหนดประเภทร้านอาหาร และปุ่ม Create สำหรับบันทึก

4.1.1.2 สำหรับผู้ประกอบการร้านอาหารหรือผู้ใช้ทั่วไป จะประกอบไปด้วย 1) หน้าเข้าสู่ระบบ 2) หน้าหลัก 3) หน้าบันทึกข้อมูลร้านอาหารและรายการอาหาร

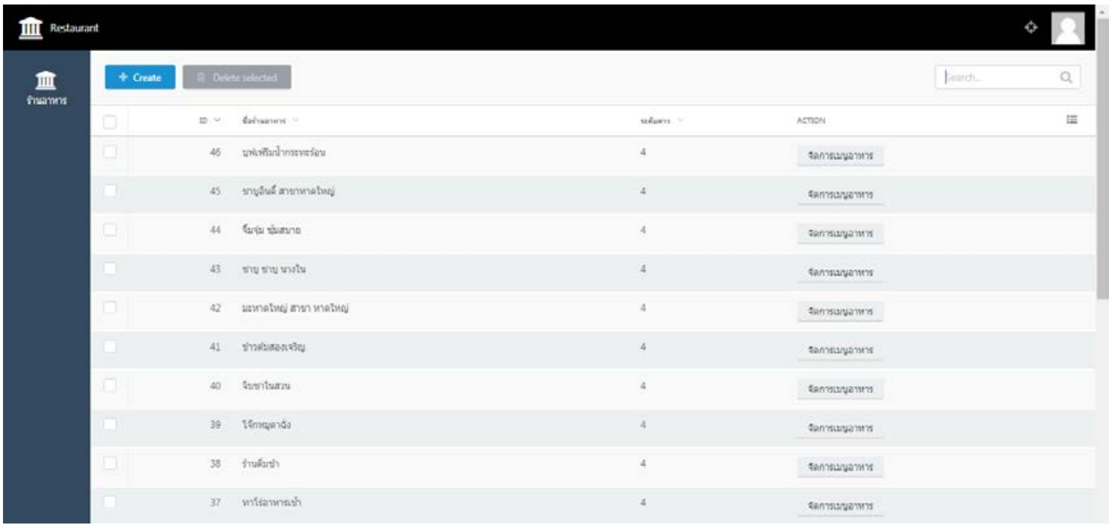
1) หน้าเข้าสู่ระบบ (Log In) เพื่อใช้ในการ Log In เข้าใช้งานระบบ



รูปที่ 4-5 หน้าเข้าสู่ระบบ

จากรูปที่ 4-5 แสดงหน้าเข้าสู่ระบบ โดยประกอบด้วยฟิวล์ข้อมูลที่จำเป็นสำหรับการ Log In เข้าสู่ระบบ ได้แก่ User Name และ Password และปุ่ม LogIn

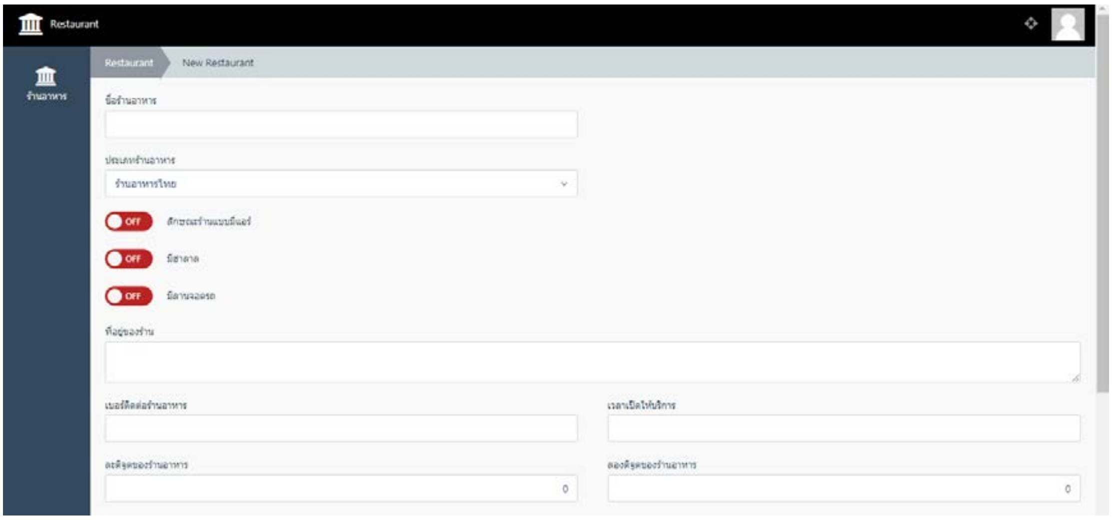
2) หน้าหลัก (Home) เป็นหน้าแรกของโปรแกรมเมื่อผู้ใช้ทำการ Log In เข้าสู่ระบบได้สำเร็จ



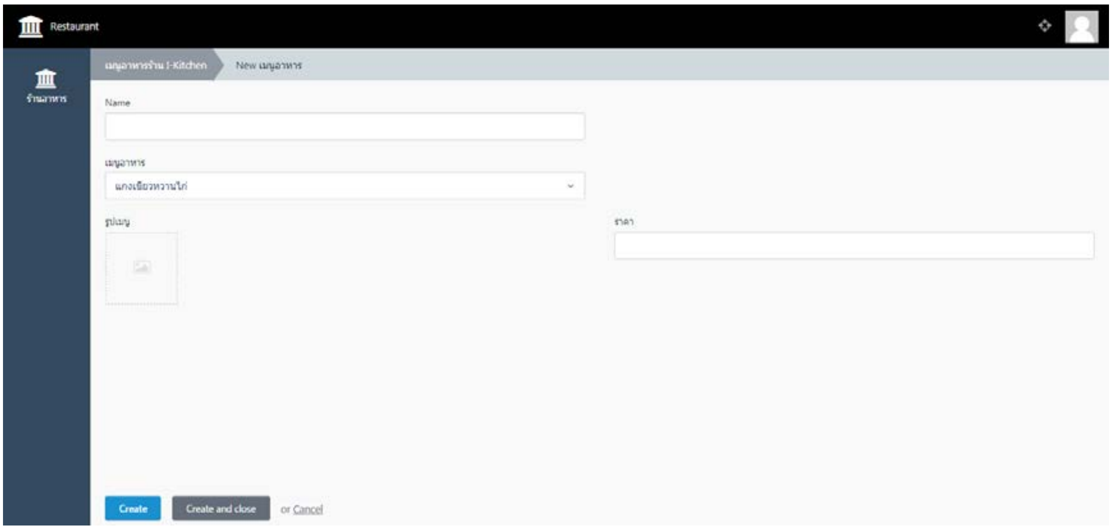
รูปที่ 4-6 หน้าหลัก

จากรูปที่ 4-6 ประกอบด้วย เมนูหลักจำนวน 1 เมนู ได้แก่ เมนูร้านอาหาร ซึ่งมีปุ่ม Create สำหรับบันทึกและแก้ไขข้อมูลร้านอาหาร และปุ่มDelete สำหรับลบข้อมูลร้านอาหาร

3) หน้าบันทึกข้อมูลร้านอาหารและรายการอาหาร



รูปที่ 4-7 หน้าบันทึกข้อมูลร้านอาหาร

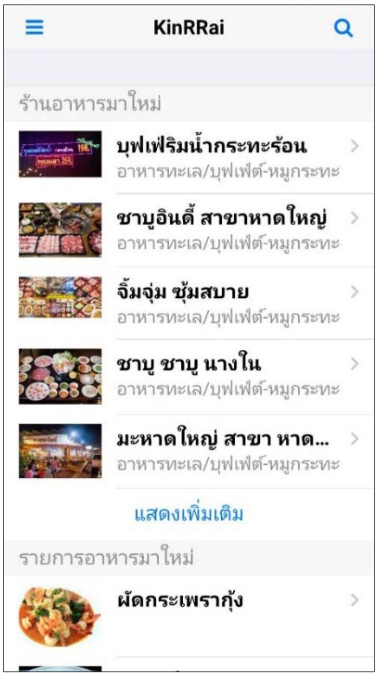


รูปที่ 4-8 หน้าบันทึกข้อมูลรายการอาหาร

จากรูปที่ 4-7 และ รูปที่ 4-8 ประกอบด้วยฟิวล์ข้อมูลที่จำเป็นต้องกรอกเกี่ยวกับรายละเอียดของร้านอาหาร ได้แก่ ชื่อร้านอาหาร รูปของร้านอาหาร ที่อยู่ เบอร์โทรศัพท์ Location ของร้านอาหาร เวลาเปิด-ปิด อื่น ๆ และ ปุ่ม Create สำหรับบันทึกข้อมูล

4.1.2 แอปพลิเคชันบนสมาร์ทโฟน เป็นแอปพลิเคชันในการแนะนำร้านอาหารและรายการอาหาร จะประกอบไปด้วยส่วนต่างๆ ดังนี้ 1) หน้าหลัก (Home) 2) สมัครสมาชิก (Register) 3)หน้าเข้าสู่ระบบ (Log In) 4) หน้าตั้งค่า 5) หน้าแนะนำ 6) หน้าค้นหา

1) หน้าหลัก (Home) ประกอบด้วย 1.1) เมนูหลัก 1.2)ร้านอาหารมาใหม่ 1.3) รายการอาหารมาใหม่ 1.4) ระยะทาง



รูปที่ 4-9 หน้าหลักของแอปพลิเคชัน (Home)

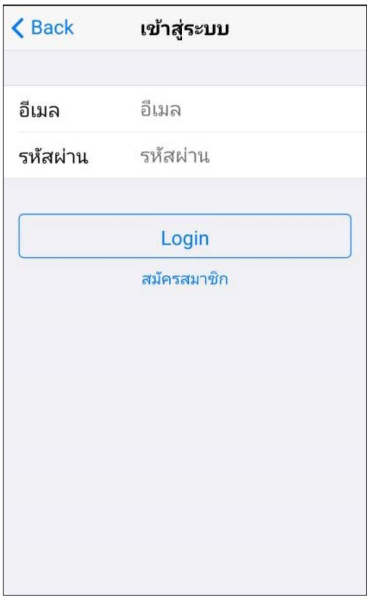
จากรูปที่ 4-9 แสดงหน้าหลักของแอปพลิเคชัน ซึ่งประกอบไปด้วย 1.1) คือเมนูหลัก ได้แก่ เมนูเข้าสู่ระบบ เมนูตั้งค่า เมนูค้นหา เมนูแนะนำ 1.2) ร้านอาหารมาใหม่ในช่วงเวลา1 สัปดาห์ล่าสุด 1.3) รายการอาหารมาใหม่ในช่วงเวลา1สัปดาห์ล่าสุด 1.4) ระยะทางระหว่างตำแหน่งผู้ใช้ ณ ปัจจุบันกับร้านอาหารนั้นๆ

2) หน้าสมัครสมาชิก (Register) เพื่อทำการสมัครสมาชิกก่อนการใช้งาน Applicaiton

รูปที่ 4-10 หน้าสมัครสมาชิก (Register)

จากรูป 4-10 แสดงหน้าสมัครสมาชิก ส่วนของการกรอกข้อมูลต่างๆ ได้แก่ ชื่อ นามสกุล อีเมล รหัสผ่าน และปุ่มสมัครสมาชิก

3) หน้าเข้าสู่ระบบ (Log In)



รูปที่ 4-11 หน้าเข้าสู่ระบบ (Log In)

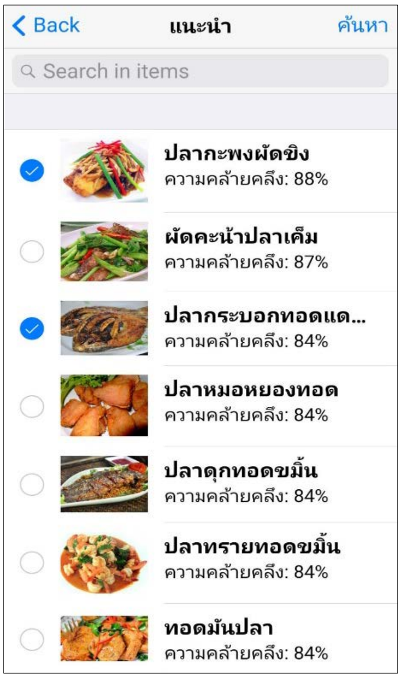
จากรูปที่ 4-11 แสดงหน้าเข้าสู่ระบบ ส่วนของการกรอกข้อมูล อีเมล รหัสผ่าน ปุ่มสำหรับเข้าสู่ระบบ และปุ่มสมัครสมาชิก

4) หน้าตั้งค่า

รูปที่ 4-12 หน้าตั้งค่า

จากรูปที่ 4-12 แสดงหน้าตั้งค่าโปรไฟล์ของผู้ใช้ สำหรับใช้ในการประมวลผลของเมนูแนะนำ ผู้ใช้สามารถกรอกข้อมูลความต้องการร้านอาหารเช่น ระดับดาว ระยะทาง ลักษณะของร้านมีแอร์หรือไม่ มีลานจอดรถหรือไม่ มีฮาลาลหรือไม่ รวมทั้งกำหนดคะแนนความชอบส่วนประกอบ เนื้อสัตว์ ผัก และรสชาติของรายการอาหารที่ต้องการ เป็นต้น

5) หน้าแนะนำ

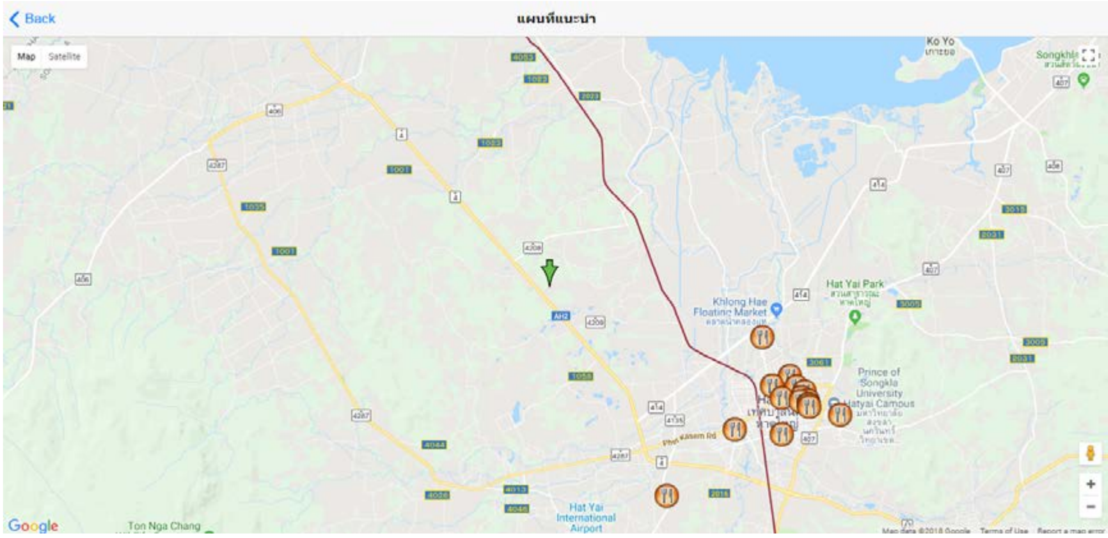


รูปที่ 4-13 หน้าแนะนำรายการอาหาร

รูปที่ 4-14 หน้าแนะนำร้านอาหาร

จากรูปที่ 4-13 รายการอาหารที่แนะนำ ได้แก่ รายการอาหารที่มีค่าความคล้ายคลึงกับความต้องการของผู้ใช้ที่ได้กำหนดไว้ในหน้าตั้งค่า เรียงจากรายการอาหารที่มีค่าความคล้ายคลึงมากไปหาน้อยตามลำดับ ผู้ใช้สามารถคลิกเลือกรายการอาหารที่ต้องการได้มากกว่า 1 รายการ เพื่อ

ค้นหาร้านอาหารที่มีค่าความคล้ายคลึงกับโปรไฟล์ที่ได้กำหนดไว้ในหน้าตั้งค่าและมีรายการอาหารดังกล่าวเหล่านั้น เรียงจากร้านที่มีค่าความคล้ายคลึงมากไปหาน้อยตามลำดับ ดังรูปที่ 4-14



รูปที่ 4-15 หน้าแสดงร้านอาหารบนแผนที่

จากรูปที่4-15 แสดงที่ตั้งของร้านอาหารบนแผนที่ ผู้ใช้สามารถคลิกเพื่อดูข้อมูลเบื้องต้นของแต่ละร้านได้

6) หน้าค้นหา

รูปที่ 4-16 หน้าค้นหา

รูปที่ 4-17 หน้าค้นหา

รูปที่ 4-16 แสดงหน้าค้นหาร้านอาหาร โดยเลือกประเภทร้านอาหารที่ต้องการค้นหา จากนั้นค้นหาด้วยชื่อร้าน จะแสดงร้านอาหารที่ได้จากการค้นหาดังรูปที่ 4-17

4.2 ผลการคำนวณค่าความคล้ายคลึงและผลการแนะนำรายการอาหาร

ตารางที่ 4-1 ความต้องการของผู้ใช้ ตัวอย่างที่ 1

ตัวอย่างที่ 1	อยากทานมากที่สุด ค่าความชอบ=5	อยากทานมาก ค่าความชอบ=4	อยากทานปานกลาง ค่าความชอบ=3	อยากทานน้อย ค่าความชอบ=2	อยากทานน้อยมาก ค่าความชอบ=1
เนื้อสัตว์	กุ้ง	-	-	-	-
ผัก	สะตอ	-	-	-	-
รสชาติ	เผ็ด	-	-	-	-

จากตารางที่ 4-1 ตัวอย่างที่ 1 ผู้ใช้กำหนดค่าความชอบเนื้อสัตว์ ผัก และรสชาติ ระดับ 5 เพียงระดับเดียว

ตารางที่ 4-2 ผลการเปรียบเทียบการคำนวณค่าความคล้ายคลึงของเมนูอาหารจากการคำนวณและการพัฒนาระบบ เรียงตามลำดับจากมากไปหาน้อย จำนวน 5 รายการ

ลำดับ	ชื่อเมนู	ผลลัพธ์ ค่าความคล้ายคลึงจากการคำนวณ (0-1)	ผลลัพธ์ ค่าความคล้ายคลึงจากระบบ (0-100 %)
1	สะตอผัดเครื่องแกงกุ้ง	1.00	100%

ลำดับ	ชื่อเมนู	ผลลัพธ์ ค่าความคล้ายคลึงจากการคำนวณ (0-1)	ผลลัพธ์ ค่าความคล้ายคลึงจากระบบ (0-100 %)
2	ผัดเผ็ดกุ้งสะตอ	1.00	100%
3	ฉู่ฉี่กุ้ง	0.50	50%
4	แกงคั่วหน่อไม้ดองสะตอกุ้ง	0.45	45%
5	สะตอผัดกะปิกุ้ง	0.33	33%

จากตารางที่ 4-2 ผลการพัฒนาระบบนั้นได้ค่าความคล้ายคลึงที่ตรงกับผลการคำนวณ 100%

ตารางที่ 4-3 ความต้องการของผู้ใช้ ตัวอย่างที่ 2

ตัวอย่างที่ 2	อยากทานมากที่สุด ค่าความชอบ=5	อยากทานมาก ค่าความชอบ=4	อยากทานปานกลาง ค่าความชอบ=3	อยากทานน้อย ค่าความชอบ=2	อยากทานน้อยมาก ค่าความชอบ=1
เนื้อสัตว์	กุ้ง	ปลา	-	-	-
ผัก	บล็อกเคอรี่	ขิง	-	-	-
รสชาติ	เค็ม	เผ็ด	-	-	-

จากตารางที่ 4-3 ตัวอย่างที่ 2 ผู้ใช้กำหนดค่าความชอบเนื้อสัตว์ ผัก และรสชาติ ระดับ 5 และระดับ 4

ตารางที่ 4-4 ผลการเปรียบเทียบการคำนวณค่าความคล้ายคลึงของเมนูอาหารจากการคำนวณและการพัฒนาระบบ เรียงตามลำดับจากมากไปหาน้อย จำนวน 5 รายการ

ลำดับ	**ชื่อเมนู**	**ผลลัพธ์ ค่าความคล้ายคลึงจากการคำนวณ (0-1)**	**ผลลัพธ์ ค่าความคล้ายคลึงจากระบบ (0-100 %)**
1	บล็อกเคอรี่ผัดกุ้ง	0.25	25%
2	คั่วกลิ้งไตปลา	0.23	24%
3	กุ้งทอดกระเทียม	0.23	23%
4	ทอดมันกุ้ง	0.23	23%
5	ปลากระพงผัดขิง	0.22	22%

จากตารางที่ 4-4 ผลการพัฒนาระบบนั้นได้ค่าความคล้ายคลึงที่ตรงกับผลการคำนวณ 100%

ตารางที่ 4-5 ความต้องการของผู้ใช้ ตัวอย่างที่ 3

<u>ตัวอย่างที่ 3</u>	**อยากทานมากที่สุด ค่าความชอบ=5**	**อยากทานมาก ค่าความชอบ=4**	**อยากทานปานกลาง ค่าความชอบ=3**	**อยากทานน้อย ค่าความชอบ=2**	**อยากทานน้อยมาก ค่าความชอบ=1**
เนื้อสัตว์	กุ้ง	ปลา	หมู	ไข่	เนื้อ
ผัก	บล็อกเคอรี่	ขิง	เห็ด	ข้าวโพดอ่อน	มะเขือเทศ
รสชาติ	เปรี้ยว	เผ็ด	เค็ม	หวาน	-

จากตารางที่ 4-5 ตัวอย่างที่ 3 ผู้ใช้กำหนดค่าความชอบเนื้อสัตว์ ผัก และรสชาติ ทุกระดับ

ตารางที่ 4-6 ผลการเปรียบเทียบการคำนวณค่าความคล้ายคลึงของเมนูอาหารจากการคำนวณและการพัฒนาระบบ เรียงตามลำดับจากมากไปหาน้อย จำนวน 5 รายการ

ลำดับ	**ชื่อเมนู**	**ผลลัพธ์ ค่าความคล้ายคลึงจากการคำนวณ (0-1)**	**ผลลัพธ์ ค่าความคล้ายคลึงจากระบบ (0-100 %)**
1	ต้มยำกุ้งน้ำข้น	0.14	14%
2	ต้มยำกุ้งมะพร้าวอ่อน	0.14	14%

ลำดับ	ชื่อเมนู	ผลลัพธ์ ค่าความคล้ายคลึงจากการคำนวณ (0-1)	ผลลัพธ์ ค่าความคล้ายคลึงจากระบบ (0-100 %)
3	กุ้งต้มยำมะม่วงเบา	0.14	14%
4	กุ้งผัดน้ำพริกเผา	0.14	14%
5	บล็อกเคอรี่ผัดกุ้ง	0.14	14%

จากตารางที่ 4-6 ผลการพัฒนาระบบนั้นได้ค่าความคล้ายคลึงที่ตรงกับผลการคำนวณ 100%

ตารางที่ 4-7 ความต้องการของผู้ใช้ ตัวอย่างที่ 4

ตัวอย่างที่ 4	**อยากทานมากที่สุด ค่าความชอบ=5**	**อยากทานมาก ค่าความชอบ=4**	**อยากทานปานกลาง ค่าความชอบ=3**	**อยากทานน้อย ค่าความชอบ=2**	**อยากทานน้อยมาก ค่าความชอบ=1**
เนื้อสัตว์	อาหารทะเล, ไข่	-	-	-	-
ผัก	หัวผักกาด, มะระ	-	-	-	-
รสชาติ	เผ็ด, เค็ม	-	-	-	-

จากตารางที่ 4-7 ตัวอย่างที่ 4 ผู้ใช้กำหนดค่าความชอบเนื้อสัตว์ ผัก และรสชาติ ระดับ 5 จำนวนหลาย Item

ตารางที่ 4-8 ผลการเปรียบเทียบการคำนวณค่าความคล้ายคลึงของเมนูอาหารจากการคำนวณและการพัฒนาระบบ เรียงตามลำดับจากมากไปหาน้อย จำนวน 5 รายการ

ลำดับ	**ชื่อเมนู**	**ผลลัพธ์ ค่าความคล้ายคลึงจากการคำนวณ (0-1)**	**ผลลัพธ์ ค่าความคล้ายคลึงจากระบบ (0-100 %)**
1	มะระผัดไข่	0.22	22%
2	ซีฟู๊ดผัดฉ่า	0.19	19%
3	ต้มข่าซีฟู๊ดหัวปลี	0.18	18%
4	ไข่เจียว	0.18	18%
5	ไข่เจียวต้มยำ	0.18	18%

จากตารางที่ 4-8 ผลการพัฒนาระบบนั้นได้ค่าความคล้ายคลึงที่ตรงกับผลการคำนวณ 100%

ตารางที่ 4-9 ความต้องการของผู้ใช้ ตัวอย่างที่ 5

<u>ตัวอย่างที่ 5</u>	**อยากทานมากที่สุด ค่าความชอบ=5**	**อยากทานมาก ค่าความชอบ=4**	**อยากทานปานกลาง ค่าความชอบ=3**	**อยากทานน้อย ค่าความชอบ=2**	**อยากทานน้อยมาก ค่าความชอบ=1**
เนื้อสัตว์	ปู	-	-	-	-
ผัก	-	-	-	-	-
รสชาติ	-	-	-	-	-

จากตารางที่ 4-9 ตัวอย่างที่ 5 ผู้ใช้กำหนดค่าความชอบเนื้อสัตว์ ระดับ 5 เพียงระดับเดียว

ตารางที่ 4-10 ผลการเปรียบเทียบการคำนวณค่าความคล้ายคลึงของเมนูอาหารจากการคำนวณและการพัฒนาระบบ เรียงตามลำดับจากมากไปหาน้อย จำนวน 5 รายการ

ลำดับ	**ชื่อเมนู**	**ผลลัพธ์ ค่าความคล้ายคลึงจากการคำนวณ (0-1)**	**ผลลัพธ์ ค่าความคล้ายคลึงจากระบบ (0-100 %)**
1	ปูม้าผัดพริกไทยดำ	1.00	100%
2	ปูนิ่มกระเพรากรอบ	1.00	100%

ลำดับ	ชื่อเมนู	ผลลัพธ์ ค่าความคล้ายคลึงจากการคำนวณ (0-1)	ผลลัพธ์ ค่าความคล้ายคลึงจากระบบ (0-100 %)
3	แกงส้มปูม้า	1.00	100%
4	ปูม้าผัดผงกะหรี่	1.00	100%
5	ปูม้าผัดต้นหอม	1.00	100%

จากตารางที่ 4-10 ผลการพัฒนาระบบนั้นได้ค่าความคล้ายคลึงที่ตรงกับผลการคำนวณ 100%

ตารางที่ 4-11 ความต้องการของผู้ใช้ ตัวอย่างที่ 6

ตัวอย่างที่ 6	**อยากทานมากที่สุด ค่าความชอบ=5**	**อยากทานมาก ค่าความชอบ=4**	**อยากทานปานกลาง ค่าความชอบ=3**	**อยากทานน้อย ค่าความชอบ=2**	**อยากทานน้อยมาก ค่าความชอบ=1**
เนื้อสัตว์	-	-	-	-	-
ผัก	คะน้า	-	-	-	-
รสชาติ	-	-	-	-	-

จากตารางที่ 4-11 ตัวอย่างที่ 6 ผู้ใช้กำหนดค่าความชอบผัก ระดับ 5 เพียงระดับเดียว

ตารางที่ 4-12 ผลการเปรียบเทียบการคำนวณค่าความคล้ายคลึงของเมนูอาหารจากการคำนวณและการพัฒนาระบบ เรียงตามลำดับจากมากไปหาน้อย จำนวน 5 รายการ

ลำดับ	ชื่อเมนู	ผลลัพธ์ ค่าความคล้ายคลึงจากการคำนวณ (0-1)	ผลลัพธ์ ค่าความคล้ายคลึงจากระบบ (0-100 %)
1	คะน้าไก่กรอบ	1.00	100%
2	ผัดคะน้าหมูกรอบ	1.00	100%
3	ผัดคะน้าปลาเค็ม	1.00	100%
4	ข้าวผัดเครื่องแกง	0.63	63%
5	ข้าวผัดหมู	0.56	56%

จากตารางที่ 4-12 ผลการพัฒนาระบบนั้นได้ค่าความคล้ายคลึงที่ตรงกับผลการคำนวณ 100%

ตารางที่ 4-13 ความต้องการของผู้ใช้ ตัวอย่างที่ 7

<u>ตัวอย่างที่ 7</u>	**อยากทานมากที่สุด ค่าความชอบ=5**	**อยากทานมาก ค่าความชอบ=4**	**อยากทานปานกลาง ค่าความชอบ=3**	**อยากทานน้อย ค่าความชอบ=2**	**อยากทานน้อยมาก ค่าความชอบ=1**
เนื้อสัตว์	-	-	-	-	-
ผัก	-	-	-	-	-
รสชาติ	หวาน	-	-	-	-

จากตารางที่ 4-13 ตัวอย่างที่ 7 ผู้ใช้กำหนดค่าความชอบรสชาติ ระดับ 5 เพียงระดับเดียว

ตารางที่ 4-14 ผลการเปรียบเทียบการคำนวณค่าความคล้ายคลึงของเมนูอาหารจากการคำนวณและการพัฒนาระบบ เรียงตามลำดับจากมากไปหาน้อย จำนวน 5 รายการ

ลำดับ	**ชื่อเมนู**	**ผลลัพธ์ ค่าความคล้ายคลึงจากการคำนวณ (0-1)**	**ผลลัพธ์ ค่าความคล้ายคลึงจากระบบ (0-100 %)**
1	ปลากระพงซอสหวาน	1.00	10%
2	ต้มข่าไก่หัวปลี	0.50	50%

ลำดับ	ชื่อเมนู	ผลลัพธ์ ค่าความคล้ายคลึงจากการคำนวณ (0-1)	ผลลัพธ์ ค่าความคล้ายคลึงจากระบบ (0-100 %)
3	หมึกไข่ผัดผงกะหรี่	0.50	50%
4	ปูม้าผัดผงกะหรี่	0.50	50%
5	กุ้งผัดผงกะหรี่	0.50	50%

จากตารางที่ 4-14 ผลการพัฒนาระบบนั้นได้ค่าความคล้ายคลึงที่ตรงกับผลการคำนวณ 100%

ตารางที่ 4-15 ผลการแนะนำเมนูอาหารจากระบบ ตัวอย่างที่ 1-ตัวอย่างที่ 7

<table>
<tr><th>ตัวอย่างที่ 1</th><th>ตัวอย่างที่ 2</th><th>ตัวอย่างที่ 3</th></tr>
<tr><td>


</td><td>


</td><td>


</td></tr>
</table>

<table>
<tr><th>ตัวอย่างที่ 4</th><th>ตัวอย่างที่ 5</th><th>ตัวอย่างที่ 6</th></tr>
<tr>
<td>


</td>
<td>


</td>
<td>


</td>
</tr>
</table>

ตัวอย่างที่ 7


จากตารางที่ 4-15 ผลการแนะนำรายการอาหารจากการตั้งค่าความต้องการของผู้ใช้ตาม ตัวอย่างที่ 1 ถึง ตัวอย่างที่ 7 นั้น ผลการคำนวณค่าความคล้ายคลึงจากระบบตรงกับการคำนวณค่าความคล้ายคลึง 100%

4.3 ผลการประเมินประสิทธิภาพการทำงานของระบบ

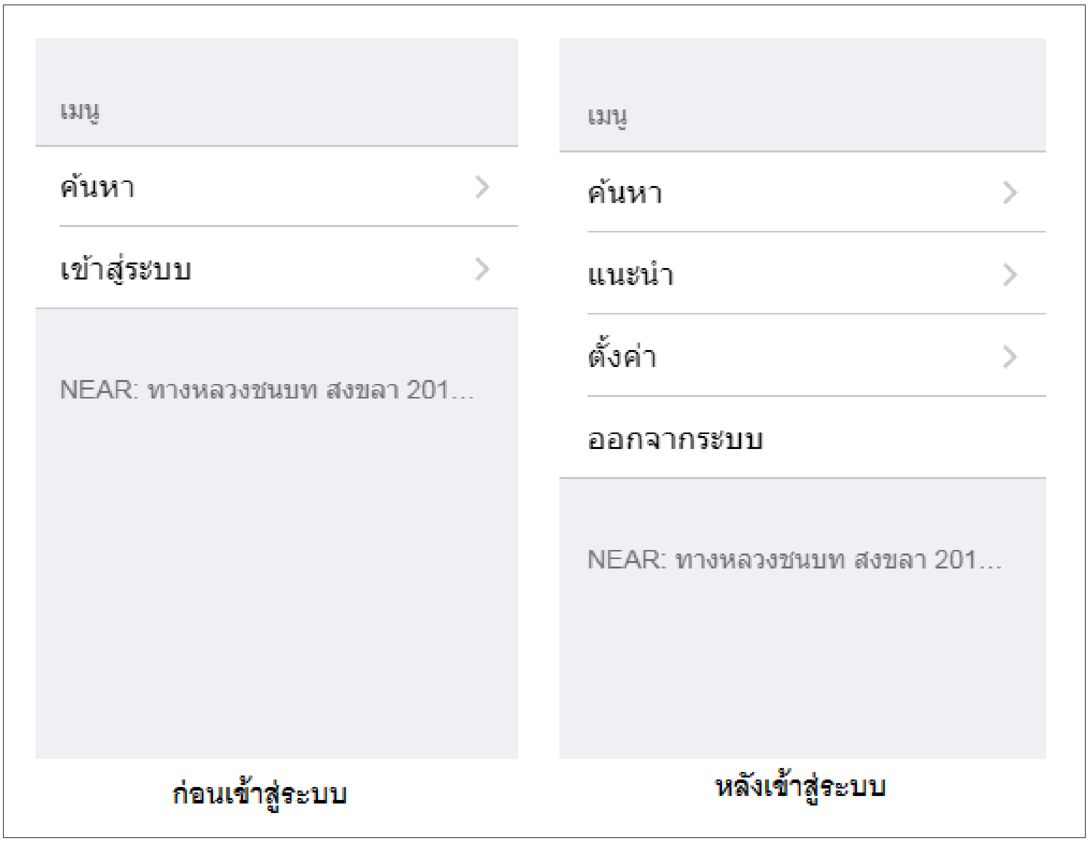
ผลการทดสอบระบบประกอบด้วย 1) ผลการทดสอบสมัครสมาชิก 2) ผลการทดสอบเข้าสู่ระบบ 3) ผลการทดสอบตั้งค่าโปรไฟล์ 4) ผลการทดสอบแนะนำรายการอาหารและร้านอาหาร 5) ผลการทดสอบการค้นหาร้านอาหาร ได้ผลการทดสอบดังนี้

1) ผลการทดสอบสมัครสมาชิก ผลที่ได้จากกการทดสอบสมัครสมาชิกนั้น ได้ทำการสมัครสมาชิกจำนวน 10 ผู้ใช้งาน ผลปรากฏว่าระบบสามารถบันทึกข้อมูลการสมัครสมาชิกได้อย่างครบถ้วนและถูกต้องดังรูปที่ 4-18

15	test10	test10@test10.com	$2y$10$kAhWDq.iDlLgO/euQcbpFu5zAKWnHVRM3MyEN8vbsei...
14	test9	test9@test9.com	$2y$10$EO1j541SeiRrQI7LO6I37uBAReXFrT/Mt20TR7yta1n...
13	test8	test8@test8.com	$2y$10$Xa/vBKvwou7LHzcqRhwzXuwmxgx5AtN30oW6ikh3f9B...
12	test7	test7@test7.com	$2y$10$uvCMmei0lvvePPAdLy2RXe8E0NQj4dcqV1UPCESmvlk...
11	test6	test6@test6.com	$2y$10$9nA7TzlpjyAv21C39uUvm./PXAKcnBlsYLPsVdL204H...
10	test	test5@test5.com	$2y$10$tngmTjoQOrzlazX83k/yA.nzEmnuFqfKU1dotK3Etfx...
9	test4	test4@test4.com	$2y$10$D3ore6gZn.wzWuUns4lCdOxMQy4jPR6JnXZJR8FgK84...
8	test3	test3@test3.com	$2y$10$x6R2rSbSEJSgpt/H5n2N3.TILAEdE3OK.NoorFqcxB1...

รูปที่ 4-18 ผลการทดสอบสมัครสมาชิก

2) ผลการทดสอบเข้าสู่ระบบ หลังจากสมาชิกสามชิกเรียบร้อย ได้ทำการเข้าสู่ระบบโดยใช้อีเมลและรหัสผ่าน ผลปรากฎว่าสามารถเข้าสู่ระบบได้ดังรูปที่ 4-19



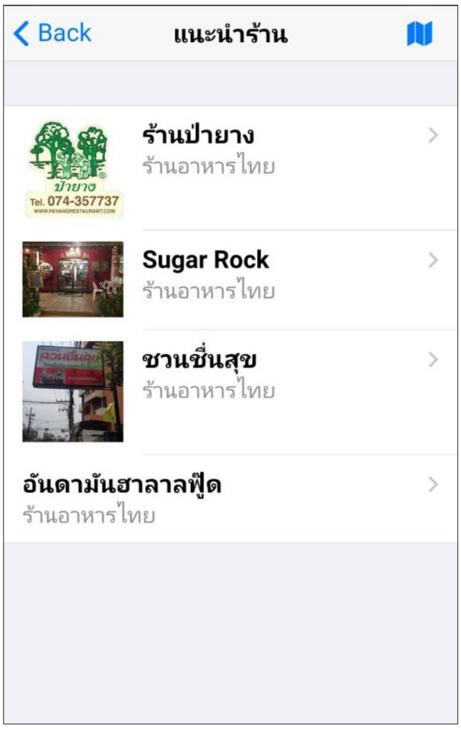
รูปที่ 4-19 ผลการทดสอบเข้าสู่ระบบ

3) ผลการทดสอบตั้งค่าโปรโฟล์ ผลที่ได้จากการตั้งค่าโปรไฟล์ปรากฏว่าสามารถตั้งค่าโปรไฟล์และสามารถเปลี่ยนแปลงข้อมูลได้อย่างมีความถูกต้องดังรูปที่ 4-20

รูปที่ 4-20 ผลการทดสอบตั้งค่าโปรไฟล์

4) ผลการทดสอบแนะนำรายการอาหารและร้านอาหาร จากการทดสอบตั้งค่าโปรไฟล์ดังนี้ 1) ความต้องการด้านร้านอาหาร ระดับดาว 4 ดาว ระยะทาง 5 กิโลเมตร มีแอร์ ไม่มีฮาลาล ไม่มีลานจอดรถ 2) ความต้องการด้านรายการอาหาร เนื้อสัตว์ ได้แก่ อยากทานปลามากที่สุด ผัก ได้แก่อยากทานขิงมากที่สุด รสชาติ ได้แก่ อยากทานรสเค็มมากที่สุด ระบบสามารถแนะนำรายการอาหารที่มีค่าคล้ายคลึงกับความต้องการของผู้ใช้จากมากไปหาน้อยตามลำดับดังรูปที่ 4-21 ร้านอาหารที่มีรายการอาหารที่เลือกและมีค่าความคล้าคลึงจากมากไปหาน้อยดังรูปที่ 4-22

รูปที่ 4-21 ผลการทดสอบแนะนำรายการอาหาร



รูปที่ 4-22 ผลการทดสอบแนะนำร้านอาหาร

ทดสอบเลือกรายการอาหารปลากะพงผัดขิงและผัดคะน้าปลาเค็ม ระบบแนะนำร้านอาหารที่มีรายการอาหารดังกล่าวดังรูปที่ 4-22 ได้แก่ ร้าน Sugar Rock ร้านป่ายาง ครัวปุ๊กลุ๊ค อื่นๆ ตามลำดับ ซึ่งร้าน Sugar Rock มีรายการอาหารทั้ง 2 รายการ และ ระดับ 4 ดาว ระยะทาง 5 กิโลเมตร มีแอร์ ตรงกับความต้องการของผู้ใช้ที่กำหนดไว้ข้างต้น

5) ผลการทดสอบการค้นหาร้านอาหาร ระบบสามารถค้นหาร้านอาหารตามคำค้นได้อย่างถูกต้องและแม่นยำดังรูปที่ 4-23

รูปที่ 4-23 ผลการทดสอบค้นหาร้านอาหาร

4.4 ผลการเก็บข้อมูลร้านอาหารในอำเภอหาดใหญ่ จ.สงขลา

ผู้วิจัยได้ลงพื้นที่เก็บรวมรวมข้อมูลร้านอาหารในอำเภอหาดใหญ่ จ.สงขลา ดังนี้ 1) ข้อมูลทั่วไปของร้านอาหาร ได้แก่ ชื่อร้านอาหาร ที่อยู่ เบอร์โทรศัพท์ วันเวลาเปิดปิด ตำแหน่งที่ตั้ง ประเภทร้านอาหาร ลักษณะของร้านอาหาร เครื่องหมายฮาลาล 2) รายการอาหาร ได้แก่ ชื่อรายการอาหาร ราคา โดยแบ่งตามประเภทของร้านอาหารจำนวน 9 ประเภท ได้แก่ ร้านอาหารไทยจำนวน 25 ร้าน ร้านอาหารเพื่อสุขภาพและอาหารมังสวิรัติ 5 ร้าน ร้านสเต๊ก 5 ร้าน ร้านก๋วยเตี๋ยว 5 ร้าน ร้านอาหารทะเลและบุฟเฟ่ต์-หมูกระทะ 5 ร้าน ร้านขนมจีน 5 ร้าน ร้านชากาแฟและเบเกอรี่-เค้ก 5 ร้าน ร้านไอศกรีมและขนมหวาน 5 ร้าน ร้านอาหารเช้า 5 ร้าน ดังแสดงในตารางที่ 4-15

ตารางที่ 4-16 ผลการเก็บรวมรวมข้อมูลร้านอาหารในอำเภอหาดใหญ่ จ.สงขลา

ลำดับ	ประเภทร้านอาหาร	จำนวน
1	ร้านอาหารไทย	25
2	อาหารเพื่อสุขภาพ/อาหารมังสวิรัติ	5
3	สเต๊ก	5
4	ก๋วยเตี๋ยว	5
5	อาหารทะเล/บุฟเฟ่ต์-หมูกระทะ	5
6	ขนมจีน	5
7	ชา/กาแฟ-เบเกอรี่/เค้ก	5
8	ไอศกรีม/ขนมหวาน	5
9	อาหารเช้า	5
รวมทั้งหมด 65 ร้าน		

ผู้วิจัยได้ทำการรวมรวมรายการอาหารไทย ซึ่งกำหนดเป็นรายการอาหารตั้งต้นในสำหรับระบบแนะนำรายการอาหารจากการเก็บข้อมูลรายการอาหารทั้งหมดในร้านอาหารของอำเภอหาดใหญ่ จ.สงขลา จำนวน 25 ร้าน ทั้งหมด 160 รายการ ดังแสดงในรูปที่ 4-24

14	แกงส้มดอกขจรกุ้ง	162	ผัดกระเพรากุ้ง
13	แกงคั่วหน่อไม้ดองสะตอหมู	161	แพนงปลา
12	แกงคั่วหน่อไม้ดองสะตอกุ้ง	160	แพนงไก่
11	แกงส้มปลามะระอ่อน	159	ผัดพริกทะเลรวมมิตร
10	แกงส้มปลายอดมะพร้าว	158	ผัดพริกไก่
9	แกงส้มกุ้งชะอมไข่	157	ผัดพริกหมู
8	แกงหมูใบชะมวง	156	ข้าวผัดไก่
7	แกงเผ็ดหมูตุ๋น	155	ข้าวผัดหมู
6	แกงคั่วใบชะพลูปู	154	เครื่องแกงทะเลรวมิตร
5	แกงคั่วใบชะพลูกุ้ง	153	แกงจืดเต้าหู้ไก่สับ
4	แกงเขียวหวานซีฟู้ด	152	ต้มซุปเนื้อซี่โครง
3	แกงเขียวหวานกุ้ง	151	ต้มแซ่บทะเล
2	แกงเขียวหวานหมู	150	ต้มแซ่บเนื้อ

Displayed records: 141-160 of 161 « ‹ 8 ›

รูปที่ 4-24 ผลการเก็บรวมรวมรายการอาหาร

บทที่ 5
สรุปผลวิจัยและข้อเสนอแนะ

5.1 สรุปผลการวิจัย

งานวิจัยนี้ได้นำเสนอการพัฒนาระบบแนะนำร้านอาหารอัตโนมัติบนสมาร์ทโฟนโดยใช้ข้อมูลเชิงตำแหน่งและรายการอาหาร โดยแบ่งการพัฒนาออกเป็น 2 ส่วน คือ 1) การพัฒนาขั้นตอนวิธีการกรองข้อมูลสำหรับแนะนำรายการอาหาร 2) การพัฒนาแอปพลิเคชันแนะนำร้านอาหารอัตโนมัติบนสมาร์ทโฟนโดยใช้ข้อมูลเชิงตำแหน่งและรายการอาหาร โดยสามารถสรุปผลของงานในแต่ละส่วนได้ ดังนี้

5.1.1 การพัฒนาขั้นตอนวิธีการกรองข้อมูลสำหรับแนะนำรายการอาหาร

การพัฒนาขั้นตอนวิธีการกรองข้อมูลสำหรับแนะนำรายการอาหารและร้านอาหาร ผู้วิจัยใช้ Content-Based Filtering ในการกรองข้อมูล โดยให้ความสำคัญของข้อมูลรายการอาหาร และหาค่าความคล้ายคลึงของรายการอาหารที่มีอยู่กับความต้องการในรายการอาหารของผู้ใช้ และประยุกต์ใช้ระบบแนะนำที่ใช้ Multicriteria Rating ซึ่งพิจารณาคะแนนความชอบส่วนประกอบของรายการอาหารได้แก่ เนื้อสัตว์ ผัก และรสชาติ แทนการพิจารณาความชอบในรายการอาหารเพียงค่าเดียว ทำให้สามารถค้นหารายการอาหารที่มีความใกล้เคียงกับความชอบของผู้ใช้มากยิ่งขึ้น การหาค่าความคล้ายคลึง (Similarity Measure) ผู้วิจัยเลือกใช้วิธี K-NN (K-nearest Neighbor) ในการหาสมาชิกที่มีความใกล้เคียง เนื่องจากเป็นวิธีที่นิยมใช้มากที่สุด และเลือกใช้วิธีการคำนวณระยะห่างระหว่างข้อมูลด้วย Standard Euclidean Distance ข้อดีของการคำนวณวิธีนี้คือ การแนะนำผลลัพธ์ได้มีความถูกต้องสูง รองรับข้อมูลจำนวนมากและสามารถเรียนรู้ได้เร็ว

5.1.2 การพัฒนาแอปพลิเคชันแนะนำร้านอาหารอัตโนมัติบนสมาร์ทโฟนโดยใช้ข้อมูลเชิงตำแหน่งและรายการอาหาร

การแอปพลิเคชันแนะนำร้านอาหารบนสมาร์ทโฟนโดยใช้ข้อมูลเชิงตำแหน่งและรายการอาหาร ทำให้ผู้ใช้สามารถค้นหารายการอาหารที่ต้องการและร้านอาหารที่อยู่บริเวณใกล้เคียงได้อย่างสะดวกและรวดเร็วผ่านสมาร์ทโฟนที่มีใช้อย่างแพร่หลายในปัจจุบัน ผู้วิจัยได้พัฒนาระบบแนะนำร้านอาหารบนสมาร์ทโฟนโดยใช้ข้อมูลเชิงตำแหน่งและรายการอาหารบนโครงสร้างของ Mobile Application Cross Platform ซึ่งสามารถใช้งานได้กับคอมพิวเตอร์ส่วนบุคคลและสมาร์ท

โฟนทั้งระบบปฏิบัติการ Android และ iOS โดยใช้การพัฒนาระบบเพียงครั้งเดียว ส่งผลให้ลดขั้นตอนและระยะเวลาในการพัฒนาสำหรับนักพัฒนาระบบได้เป็นอย่างดี และเลือกใช้ RestFul Protocol ในการรับส่งข้อมูลและหว่างเว็บเซอร์วิส และสมาร์ทโฟน ซึ่งสามารถเรียกใช้ได้ง่ายเพียงแค่ใช้ XML และ HTTP เท่านั้น

จากภาพรวมของการพัฒนางานวิจัยนี้ สามารถสรุปได้ว่า ระบบแนะนำร้านอาหารอัตโนมัติบนสมาร์ทโฟนโดยใช้ข้อมูลเชิงตำแหน่งและรายการอาหาร โดยใช้เทคนิคการกรองแบบอิงเนื้อหา (Content-based FIltering) สามารถแนะนำรายการอาหารที่มีความคล้ายคลึงกับความต้องการของผู้ใช้โดยระบุส่วนประกอบของรายการอาหารและรสชาติได้ ซึ่งแตกต่างจากแอปพลิเคชันโดยทั่วไปที่มีใช้ในปัจจุบัน ส่งผลให้ผู้ใช้สามารถค้นหาร้านอาหารที่มีลักษณะตรงกับความต้องการและมีรายการอาหารที่ต้องการรับประทานได้มากยิ่งขึ้น สามารถเข้าถึงร้านอาหารได้สะดวกและรวดเร็วมากยิ่งขึ้น การให้คะแนนความชอบของผู้ใช้แบบ Multicriteria Rating เป็นการคำนึงถึงคุณสมบัติของข้อมูลที่ผู้ใช้ใช้พิจารณาให้คะแนนแต่ละรายการอาหารทำให้การค้นหารายการอาหารที่มีความใกล้เคียงกันมีความคลาดเคลื่อนน้อยลง ระบบมีการบริการและจัดเก็บข้อมูลที่อยู่บนคลาวด์ (Cloud) สามารถลดการประมวลผลบนเครื่อง Terminal ตอบสนองความต้องการของผู้ใช้ได้ทุกที่ทุกเวลา

5.2 ข้อจำกัดของการวิจัย

การวิจัยในครั้งนี้ มีการเก็บข้อมูลเฉพาะร้านอาหารในอำเภอหาดใหญ่ จ.สงขลา เท่านั้น ทำให้สามารถแนะนำรายการอาหารได้ในจำนวนจำกัด หากเก็บข้อมูลในพื้นที่ที่กว้างยิ่งขึ้นจะทำให้มีรายการอาหารที่หลากหลายและปริมาณมากขึ้น

5.3 ข้อเสนอแนะ

ผู้วิจัยได้นำเสนอแนวทางในการศึกษาและพัฒนางานวิจัยในอนาคต โดยแบ่งออกเป็น 2 ส่วน ดังนี้

5.3.1 การประยุกต์ใช้งานวิจัย

1) ผลการวิจัยเกี่ยวกับแนวทางการออกแบบและพัฒนาระบบแนะนำรายการอาหารอัตโนมัติบนสมาร์ทโฟนโดยใช้ข้อมูลเชิงตำแหน่ง สามารถนำผลที่ได้เป็นแนวทางไปประยุกต์ใช้เพื่อพัฒนาระบบแนะนำร้านอาหารบนบนสมาร์ทโฟนให้มีความเหมาะสมมากยิ่งขึ้น

2) สามารถนำแนวทางในการพัฒนาระบบไปประยุกต์ใช้กับข้อมูลอื่นๆ หรือผลิตภัณฑ์อื่นๆ ตัวอย่างเช่น ภาพยนตร์ หนังสือ เป็นต้น

5.3.2 การพัฒนางานวิจัยในอนาคต

ผู้วิจัยได้นำเสนอแนวทางการการพัฒนางานวิจัยในอนาคต เพื่อต่อยอดจากงานวิจัยครั้งนี้ โดยมีรายละเอียด ดังนี้

1) ควรพิจารณาปัจจัยอื่นๆ ของร้านอาหารเพิ่มขึ้น นอกเหนือจากงานวิจัยนี้ เพื่อใช้ในการกำหนดคะแนนความชอบและการแนะนำให้ตรงกับความต้องการของผู้ใช้มากยิ่งขึ้น

2) หากมีการแยกส่วนประกอบของรายการอาหารและการกำหนดคะแนนความชอบได้ละเอียดมากยิ่งขึ้นก็สามารถแนะนำรายการอาหารได้ตรงกับความต้องการของผู้ใช้มากขึ้น

3) ในการนำเสนอร้านอาหารที่แนะนำ ในร้านอาหารที่ได้จากการแนะนำควรแสดงรายการอาหารที่ค้นหาให้ผู้ใช้ได้เห็นผลลัพธ์ชัดเจน ว่ารายการอาหารดังกล่าวมีอยู่ในร้านอาหารอาหารนั้นๆ โดยไม่ต้องคลิกเข้าไปดูในข้อมูลรายการอาหารของร้าน

4) ควรนำคะแนนความชอบในการให้จำนวนดาวจากผู้ใช้แก่ร้านอาหารไปเป็นปัจจัยในการคำนวณสำหรับแนะนำร้านอาหารเพื่อความเหมาะสมมากยิ่งขึ้น

5) ควรเพิ่มฟังก์ชันการค้นหาเส้นทางไปยังร้านอาหาร จะทำให้ระบบมีความน่าสนใจมากยิ่งขึ้น

บรรณานุกรม

ทัศนวรรณ แก้วใส และ สุพจน์ นิตย์สุวัฒน์. 2552. ระบบแนะนำภาพยนตร์ด้วยเทคนิคตัวกรองเชิงร่วมมือร่วมกับวิธีเคมีน. วิทยาศาสตร์มหาบัณฑิต, ภาควิชาเทคโนโลยีสารสนเทศ คณะเทคโนโลยีสารสนเทศ, มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ.

ปฐมพงษ์ ฉับพลัน และ ฐิมาพร เพชรแก้ว. 2553. การประยุกต์ใช้ Google Maps API ในการเฝ้าระวังปัญหายาเสพติด. วิทยาศาสตร์มหาบัณฑิต, ภาควิชาการจัดการเทคโนโลยีสารสนเทศ สำนักวิชาสารสนเทศศาสตร์, มหาวิทยาลัยวลัยลักษณ์.

เปรมฤดี ผลชอบ และ สุพจน์ นิตย์สุวัฒน์. 2553. ระบบให้คำแนะนำในการเลือกซื้อคอมพิวเตอร์โน๊ตบุ๊คด้วยเทคนิค. วิทยาศาสตรมหาบัณฑิต, ภาควิชาเทคโนโลยีสารสนเทศ คณะเทคโนโลยีสารสนเทศ, มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ.

พงษ์ศักดิ์ มิขุนทอง และ สุรศักดิ์ เจริญจันทร์. 2540. เชฟอาหารไทย. พิมพ์ครั้งที่ 1. กรุงเทพฯ : สำนักพิมพ์แม่บ้าน

วลัยนุช สกุลนุ้ย และ ทองพูล หีบไธสง. 2552. การพัฒนาระบบแนะนำหนังสือในธุรกิจพาณิชย์อิเล็กทรอนิกส์โดยใช้การกรองแบบอิงเนื้อหา กรณีศึกษา บริษัท สำนักพิมพ์วังอักษร จำกัด. วิทยาศาสตรมหาบัณฑิต, ภาควิชาเทคโนโลยีสารสนเทศ คณะเทคโนโลยีสารสนเทศ, มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ.

วรชัญ์ เฉลิมพรพงศ์. 2555. รูปแบบค่าความพึงพอใจเพื่อระบบแนะนำในอนาคต. วิทยาศาสตรมหาบัณฑิต, สาขาวิชาคณิตศาสตร์ ภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์, จฬุาลงกรณ์มหาวิทยาลัย.

ศูนย์สารสนเทศ กลุ่มสารสนเทศข้อมูล. (2018). Web Service คืออะไร. กรมประมง กระทรวงเกษตรและสหกรณ์. https://www.fisheries.go.th/it/fishdealer/webService.php (สืบค้นเมื่อ 21 มีนาคม 2557).

สำนักคอมพิวเตอร์ มหาวิทยาลัยทักษิณ. (2018).CMS คืออะไร. มหาวิทยาลัยทักษิณ. http://www2.tsu.ac.th/org/tsucms/page_detial_menu.php?idm=2&mid=356 (สืบค้นเมื่อ 22 มีนาคม 2557).

สุวรรณา ชัยชนะ. 2540. อาหารไทย ครบทุกภาค. พิมพ์ครั้งที่ 1. กรุงเทพฯ : สำนักพิมพ์แม่บ้าน

9’M webmaster@sourcecode.in.th. (2556). รู้จักกับแอนดรอยด์. http://www.sourcecode.in.th/articles.php?id=71 (สืบค้นเมื่อ 15 พฤศจิกายน 2556).

Butter CMS. (2018). Heroku CMS. Butter CMS. https://buttercms.com/heroku-cms/ (accessed 26 May 2018).

Digital Ocean. (2018). Digital Ocean. https://www.digitalocean.com/ (accessed 26 May 2018).

Ashutosh Bhargave, Niranjan Jadhav, Apurva Joshi, Prachi Oke and Prof. Mr. S. R Lahane. 2013. Digital Ordering System for Restaurant Using Android. International Journal of Scientific and Research Publications 2013, Volume 3, Issue 4 (April) : . http://www.ijsrp.org/research-paper-0413/ijsrp-p1605.pdf.

Chaiyohosting.2018. การเตรียมความพร้อมก่อนพัฒนา IOS Application. chaiyo hosting. http://www.chaiyohosting.com/ios-application/ (สืบค้นเมื่อ 20 มีนาคม 2557).

Cymiz. 2555. Cloud Server คืออะไรมีหลักการทำงานอย่างไร. Cymiz, http://www.jongjarern.com/insurances/index.php?topic=1712.0 (สืบค้นเมื่อ 20 มีนาคม 2557).

Doug Tidwell , Jame Snell and Pavel Kulohenko. (2000). Programming Web Service with SOAP, O’Reilly.

G.Admavicius and Y. Kown. (2007). New Recommendation Techniques for Multicriteria Rating Systems, IEEE Intelligence Systems vol.22, No. 3 : pp.48-55.

Giuseppe Di Fabbrizio and Narendra Gupta. (2010). Have2eat: a Restaurant Finder with Review Summarization for Mobile Phones, Coling 2010 : Demonstation Volume, page17-20

Google Play (2557). Yelp. Google Play. https://play.google.com/store/apps/details?id=com.yelp.android&hl=thhttp://www.ondemandworld.com/WhereToEatFree/ (accessed 26 March 2014)

KFC. (2018). KFC Mobile App. KFC. https://kfc.pl/en/mobile-app/ (accessed 4 Fabruary 2557).

Keerthi S. Shetty and Sanjay Singh. (2009). Cloud Based Application Development for Accessing Restaurant Information on Mobile Device using LBS. Institute of Technology Manipal University, Manipal-576104 India.

NloveCooking. (2012). อาหารไทย. NloveCooking. https://nlovecooking.com/ (accessed 30 March 2014)

OnDemandWold. (2014). Where To Eat. OnDemandWold Mobile and Web Software. http://www.ondemandworld.com/WhereToEatFree/ (accessed 26 March 2014).

Pimpler, E. (2015). Anatomy of a Hybrid Mobile GIS Application. Geospatial Training Services. http://www.geospatialtraining.com/anatomy-of-a-hybrid-mobile-gis-application/ (accessed 20 February 2015).

Redda, Y. (2012). Cross platform Mobile Applications Development. Master Thesis in Department of Computer and Information Science, Norwegian University of Science and Technology.

Software Engineering ITSchool Mae Fah Luang University. (2552). SOAP & REST Concepts (Online). Mae Fah Luang University. http://soapui.wordpress.com/2009/05/13/ (accessed 6 November 2013).

Spoorthy Reddy Parne. (2010). Android Application of Restaurant Finder. B.Tech, Jawaharlal Nehru Technological University, India.

Thaiware (2556). Kinraidee. Thaiware. http://software.thaiware.com/565-kinraidee_%E0%B8%81%E0%B8%B4%E0%B8%99%E0%B9%84%E0%B8%A3%E0%B8%94%E0%B8%B5.html (accessed 25 March 2014).

Vivek Ranjan, Nikesh Masiwal and Nikita Verma. (2013). e-Restaurant: Online Restaurant Management System for Android. International Conference & Workshop On Advance Computing.

WDS PhoneTablet. (2555). เปรียบเทียบ Windows Phone vs Android vs iOS ต่างกันอย่างไร. WDS PhoneTablet. http://www.wdsphonetablet.com/windows-phone-vs-android-vs-ios/ (สืบค้นเมื่อ 19 มีนาคม 2557).

Wongnai (2018). Wongnai Application. Wongnai. https://www.wongnai.com/about (accessed 4 Fabruary 2014).

Xamarin Inc. (2015). The Xamarin Story. Xamarin. https://xamarin.com/about (accessed 19 March 2015).

You Geeng Neng, Chang Ching Jung and You Hsiao Chen. (2012). On the Design of a LBS-based App for Searching Vegetarian Restaurant. Department of Multimedia Design National Taichung University of Science and Technology, Taichung, TAIWAN.

ZAGAT (2556). ZAGAT. ZAGAT Inc. http://www.zagat.com/ (accessed 25 March 2014).

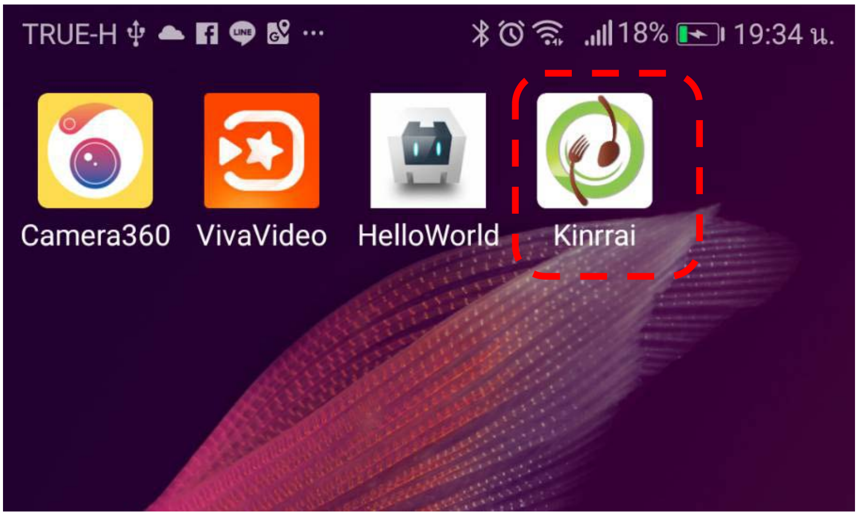
ภาคผนวก ก

คู่มือการใช้งานระบบแนะนำร้านอาหารอัตโนมัติบนสมาร์ทโฟนโดยใช้ข้อมูลเชิงตำแหน่งและรายการอาหาร

คู่มือใช้งานสำหรับผู้ใช้ทั่วไป

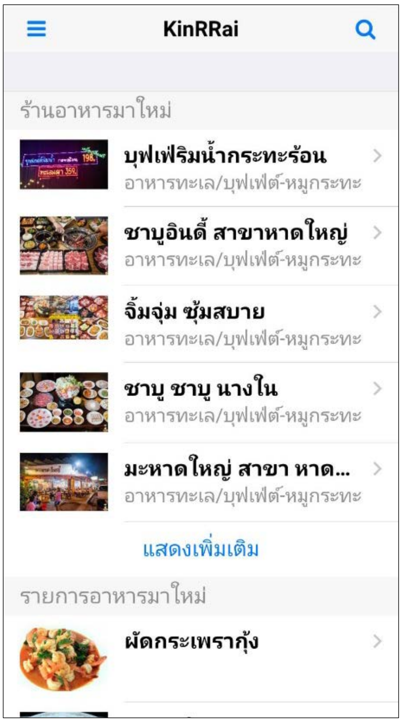
1. การสมัครสมาชิกและ Log In เข้าสู่แอปพลิเคชัน

   1.1 เปิดแอปพลิเคชัน ชื่อ “Kinrrai” ดังรูปที่ ก-1



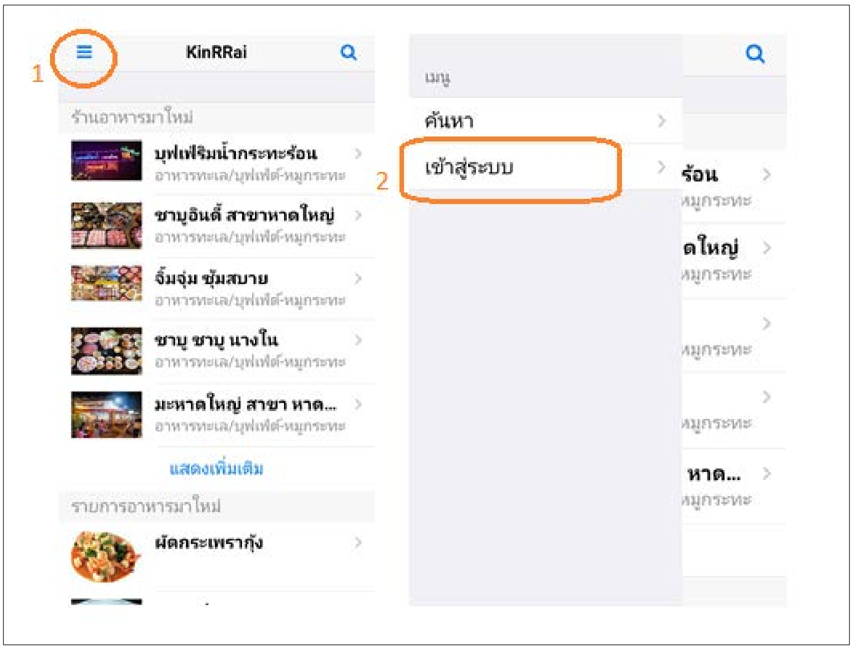
รูปที่ ก-1 ไอคอนแอปพลิเคชัน Kinrrai

   1.2 เมื่อเข้าสู่แอปพลิเคชันจะปรากฏหน้าจอหน้าหลักของแอปพลิเคชัน ดังรูปที่ ก-2



รูปที่ ก-2 หน้าหลัก

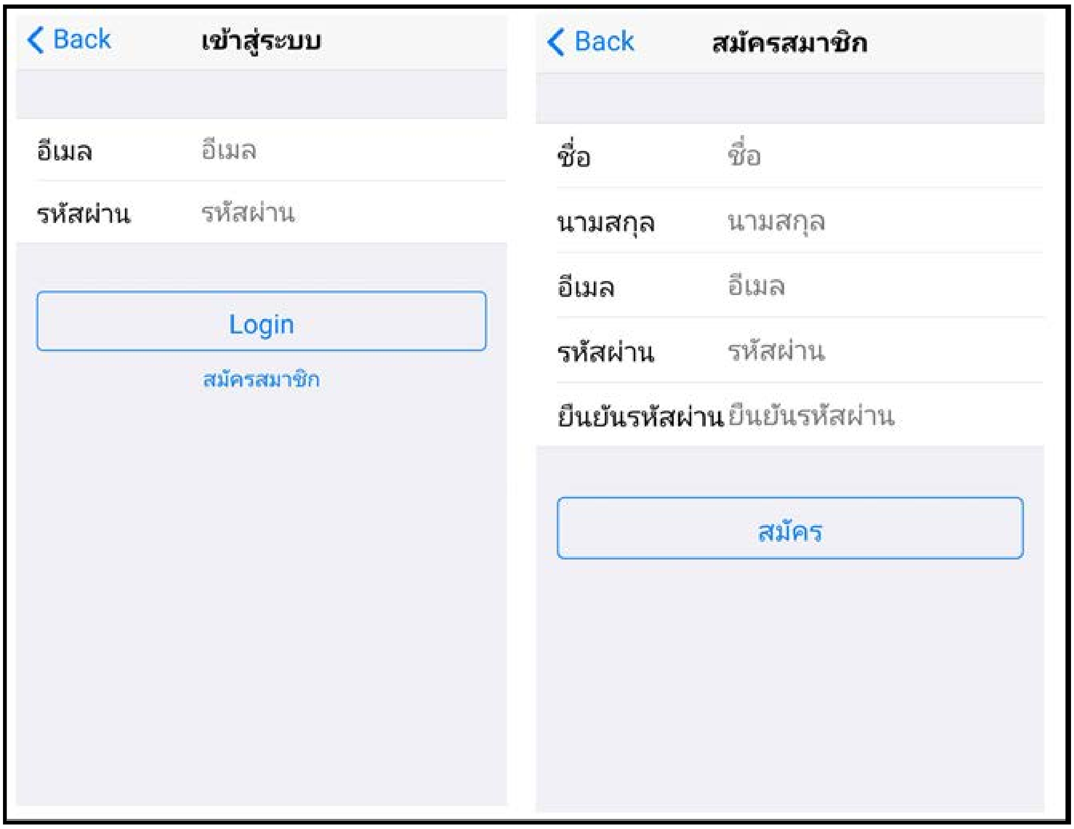
1.3 คลิกแถบเมนู เลือกเมนู “เข้าสู่ระบบ” ดังรูปที่ ก-3



รูปที่ ก-3 เมนูเข้าสู่ระบบ

1.4 ป้อนข้อมูล อีเมล และ รหัสผ่าน เพื่อเข้าสู่ระบบ

1.4.1 กรณียังไม่สมัครสมาชิก คลิกปุ่มสมัครสมาชิก กรอกข้อมูล ชื่อ นามสกุล อีเมล รหัสผ่าน แล้วคลิกปุ่มสมัคร ดังรูปที่ ก-4



รูปที่ ก-4 หน้าเข้าสู่ระบบและหน้าสมัครสมาชิก

2. การตั้งค่า Profile สำหรับใช้เมนูแนะนำ (จะเห็นเมนูนี้เมื่อสมัครสมาชิกและ Log In เข้าสู่ระบบแล้วเท่านั้น)

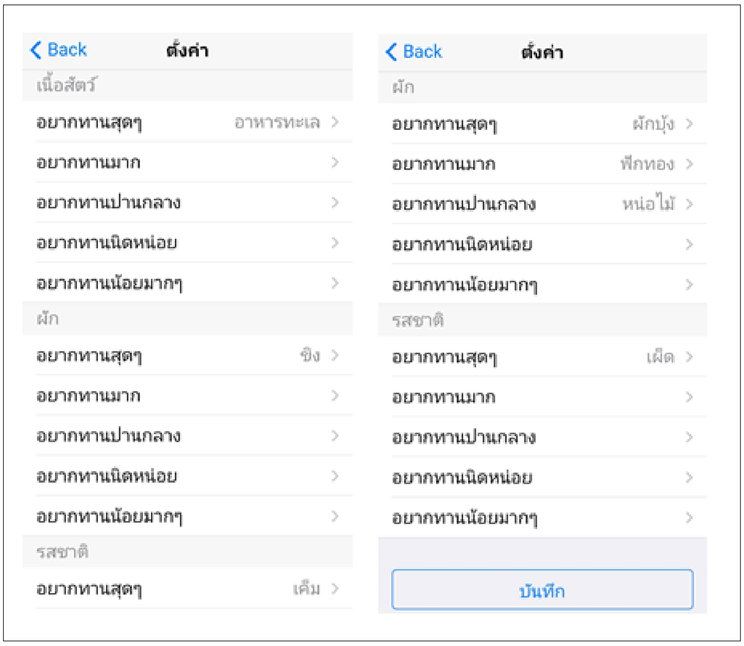
   2.1 คลิกเลือกเมนู “ตั้งค่า” ดังรูปที่ ก-5

รูปที่ ก-5 เมนูตั้งค่า

2.2 กรอกข้อมูลเบื้องต้น (เป็นการกำหนดเงื่อนไขของร้านอาหารที่ต้องการให้ระบบแนะนำ) ดังรูปที่ ก-6

รูปที่ ก-6 หน้าตั้งค่า

2.3 กรอกข้อมูลความต้องการด้านรายการอาหาร (เป็นการกำหนดค่าคะแนนความชอบส่วนประกอบในรายการอาหาร เพื่อใช้ในการแนะนำรายการอาหารที่ตรงกับความต้องการของผู้ใช้) ซึ่งประกอบด้วย 3 ส่วน ได้แก่ เนื้อสัตว์ ผัก และรสชาติ ดังรูปที่ ก-7

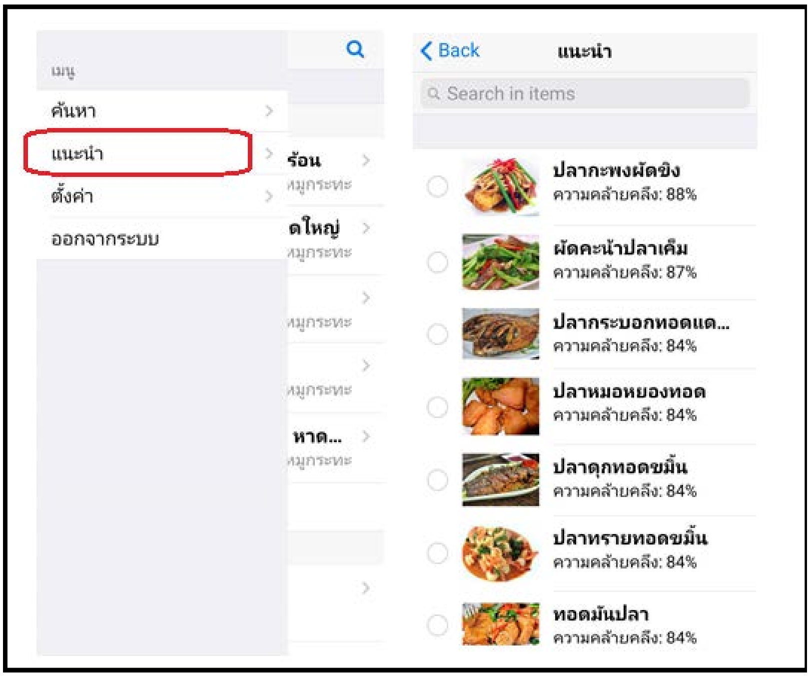


รูปที่ ก-7 หน้ากำหนดคะแนนความชอส่วนประกอบของรายการอาหาร

ผู้ใช้สามารถกำหนดค่าคะแนนความชอบในแต่ละส่วนเพียงแค่ระดับเดียวหรือหลายระดับได้

3. การใช้เมนูแนะนำ (จะเห็นเมนูนี้เมื่อสมัครสมาชิกและ Log In เข้าสู่ระบบแล้วเท่านั้น)

3.1 คลิกเลือกเมนู “แนะนำ” จะแสดงรายการรายการอาหารที่ระบบประมวลผลแล้วมีค่าความคล้ายคลึงกับความต้องการของผู้ใช้จากมากไปหาน้อยเรียงตามลำดับ ดังรูปที่ ก-8



รูปที่ ก-8 แสดงรายการอาหารแนะนำ

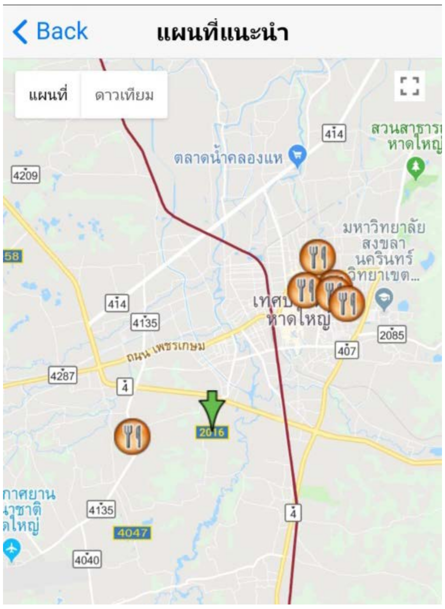
3.2 ผู้ใช้สามารถคลิกเลือกหน้ารายการรายการอาหารเพื่อ ค้นหาร้านอาหารที่มีรายการอาหารดังกล่าวและร้านมีความคล้ายคลึงกับความต้องการของผู้ใช้ที่ได้กำหนดไว้ในเมนูตั้งค่า Profile ส่วนของข้อมูลเบื้องต้น ดังรูปที่ ก-9

รูปที่ ก-9 การคลิกเลือกรายการอาหาร

3.3 แสดงร้านอาหารที่มีรายการอาหารดังกล่าวและมีความคล้ายคลึงกับความต้องการของผู้ใช้เรียงจากมากไปหาน้อยตามลำดับ ดังรูปที่ ก-10

รูปที่ ก-10 ร้านอาหารที่แนะนำ

3.4 คลิกแผนที่เพื่อดูตำแหน่งร้านอาหารทั้งหมดบนแผนที่ ดังรูปที่ ก-11

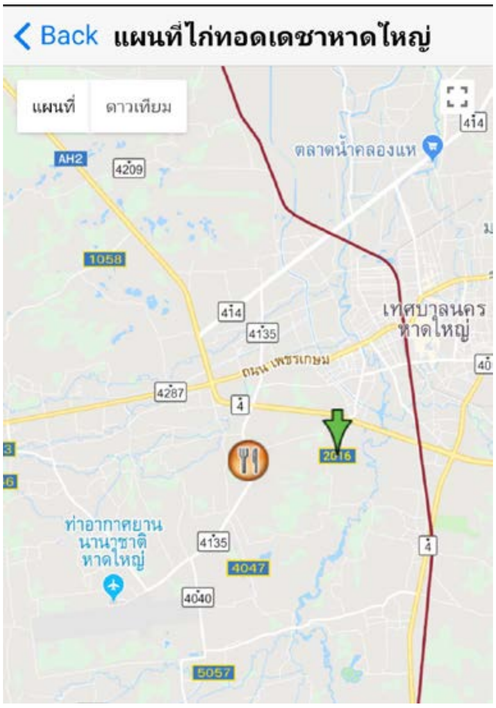


รูปที่ ก-11 ร้านอาหารทั้งหมดบนแผนที่

3.5 คลิกเพื่อดูข้อมูลรายละเอียดของร้านอาหารและรายการอาหาร ดังรูปที่ ก-12

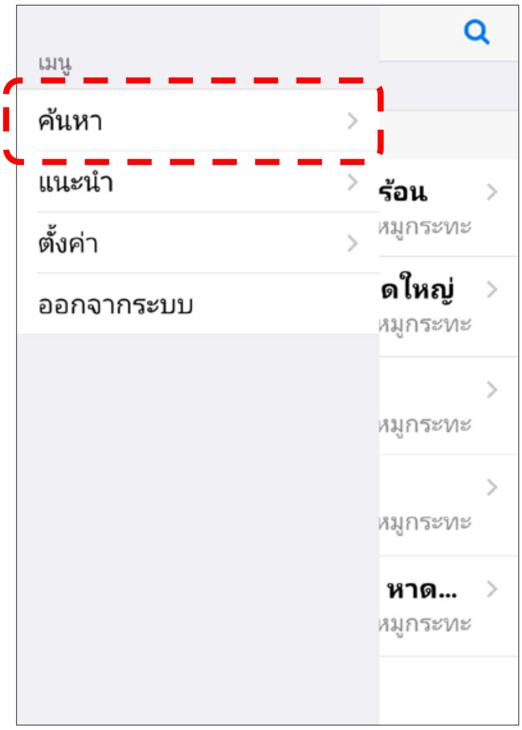
รูปที่ ก-12 รายละเอียดของร้านอาหาร

3.5.1 คลิกเพื่อดูตำแหน่งที่ตั้งของร้านอาหารบนแผนที่ ดังรูปที่ ก-13



รูปที่ ก-13 ตำแหน่งร้านอาหารบนแผนที่

4. การใช้เมนูค้นหาร้านอาหาร คลิกเลือกเมนู “ค้นหา” ดังรูปที่ ก-14



รูปที่ ก-14 แสดงเมนูค้นหา

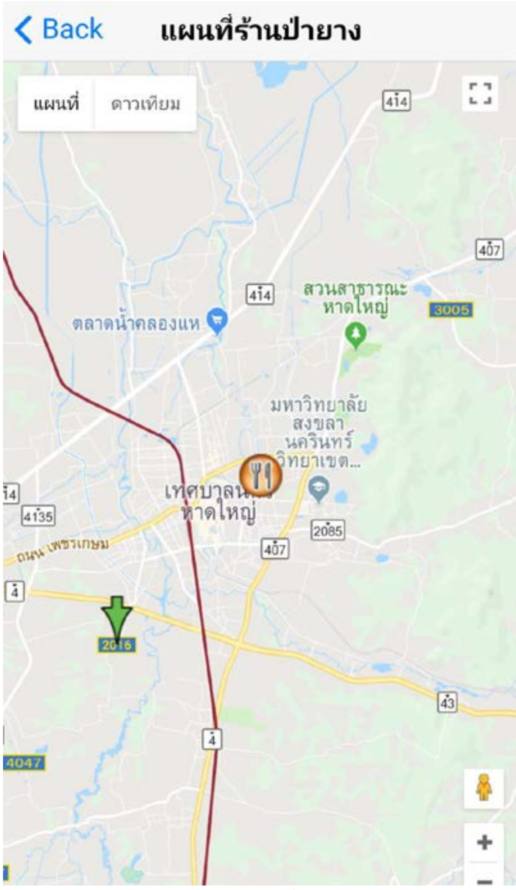
4.1 แสดงรายการประเภทของร้านอาหารที่ต้องการค้นหา ดังรูปที่ ก-15

รูปที่ ก-15 รายการประเภทร้านอาหารที่ต้องการค้นหา

4.2 คลิกเลือกประเภทร้านอาหารที่ต้องการค้นหา กรอกชื่อของร้านอาหารในช่องค้นหา ดังรูปที่ ก-16

รูปที่ ก-16 ตัวอย่างการค้นหาร้านอาหาร

4.3 แสดงตำแหน่งที่ตั้งของร้านอาหารบนแผนที่ ดังรูปที่ ก-17

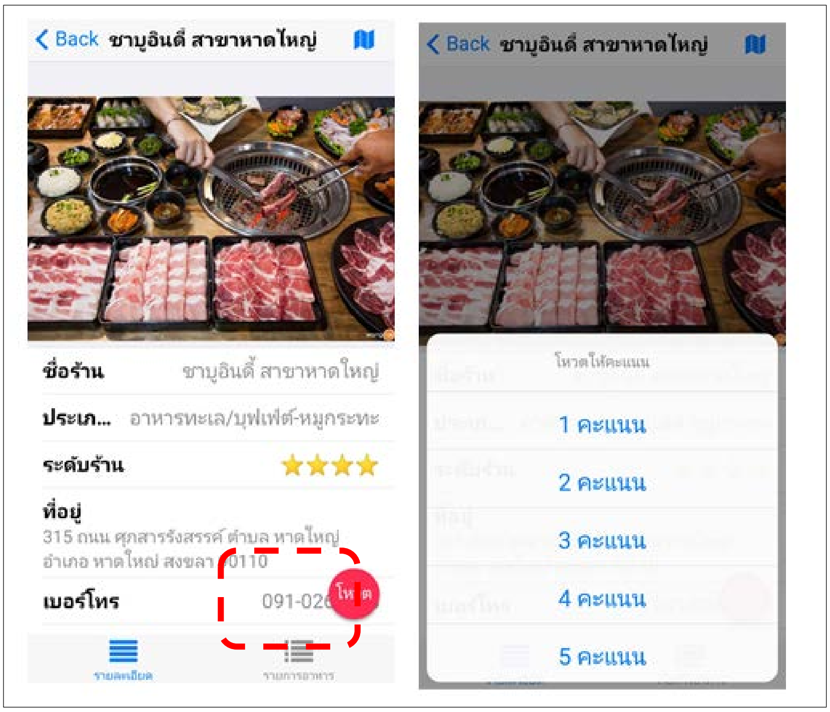


รูปที่ ก-17 ตำแหน่งที่ตั้งของร้านอาหาร

4.5 คลิกเพื่อดูข้อมูลรายละเอียดของร้านอาหารและรายการอาหาร ดังรูปที่ ก-18

รูปที่ ก-18 รายละเอียดของร้านอาหาร

5. โหวดให้คะแนนร้านอาหาร คลิกเข้าไปในรายละเอียดของร้านอาหาร คลิกปุ่มโหวด โหวดให้คะแนนกับร้านในช่วง 1-5 คะแนน ดังรูปที่ ก-19

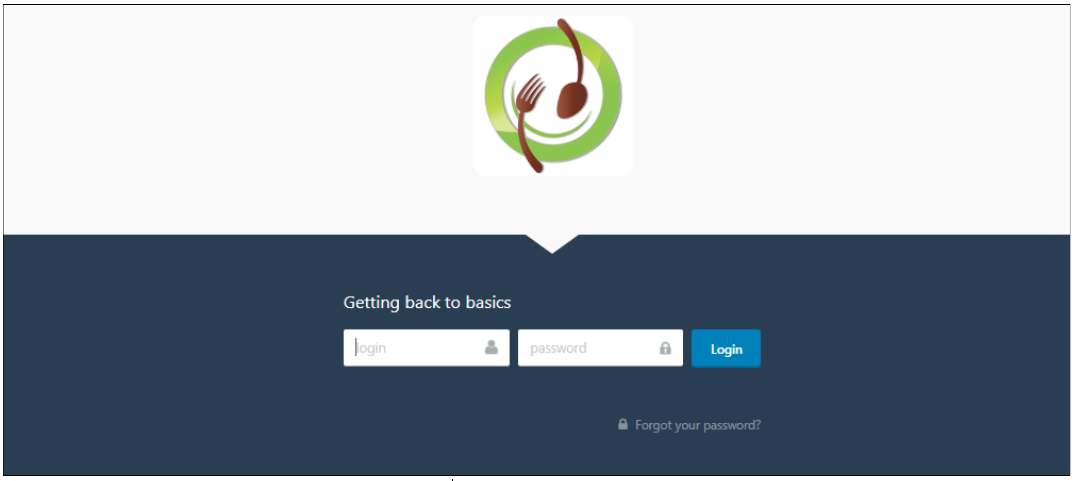


รูปที่ ก-19 โหวดให้คะแนนร้านอาหาร

คู่มือใช้งานสำหรับร้านอาหาร

1. การ Log In เข้าสู่ระบบ

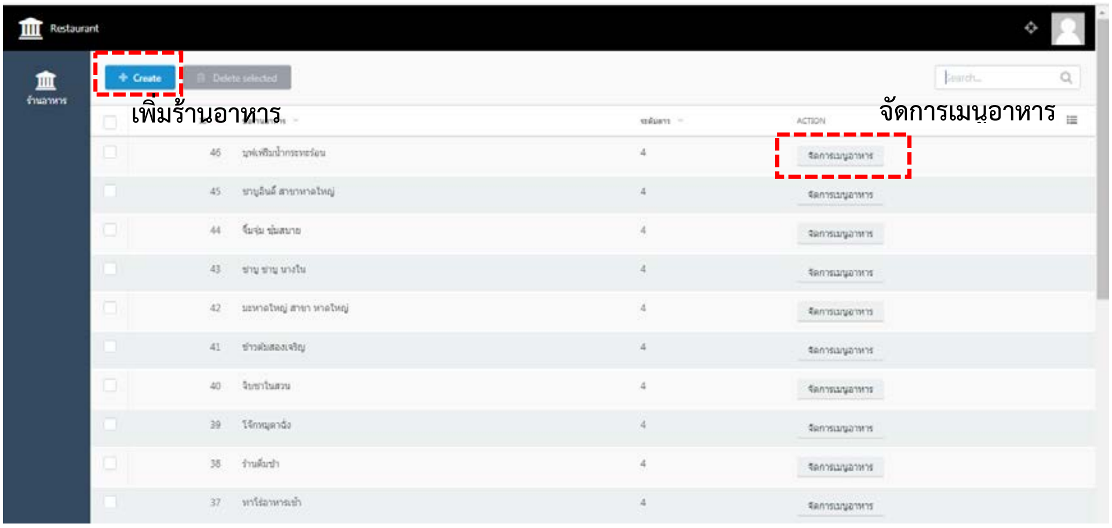
เปิดเว็บเบราเซอร์ (Web Browser) แล้วกรอก URL ชื่อ **http://kinrrai.kritlab.com/backend** จะปรากฏหน้าจอดังรูปที่ ก-20 กรอกข้อมูล UserName และรหัสผ่าน



รูปที่ ก-20 หน้าเข้าสู่ระบบ

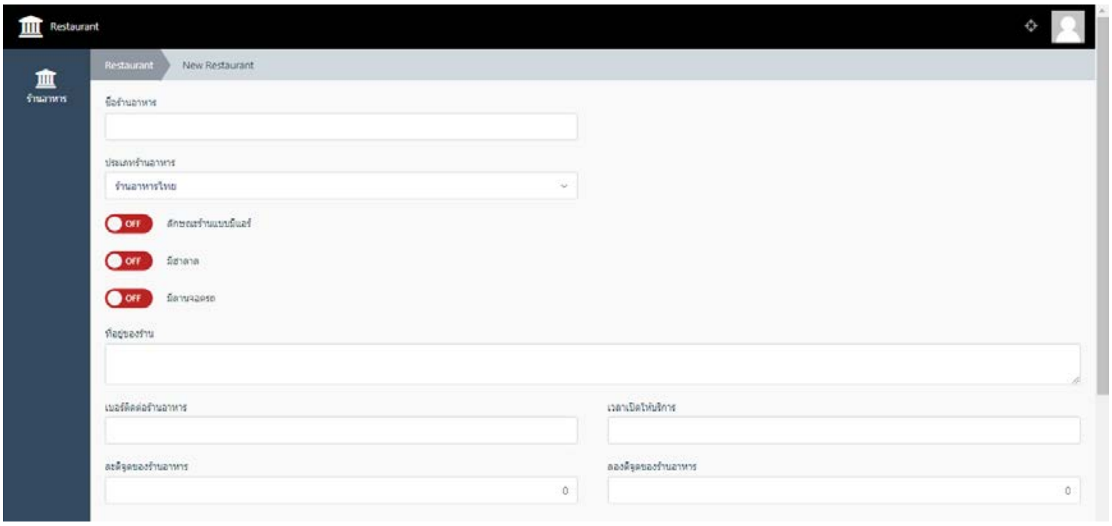
2. การเพิ่ม ลบ แก้ไข ข้อมูลร้านอาหาร

2.1 คลิกเลือกเมนู “ร้านอาหาร” จะปรากฏหน้าจอ ดังรูปที่ ก-21



รูปที่ ก-21 หน้าร้านอาหาร

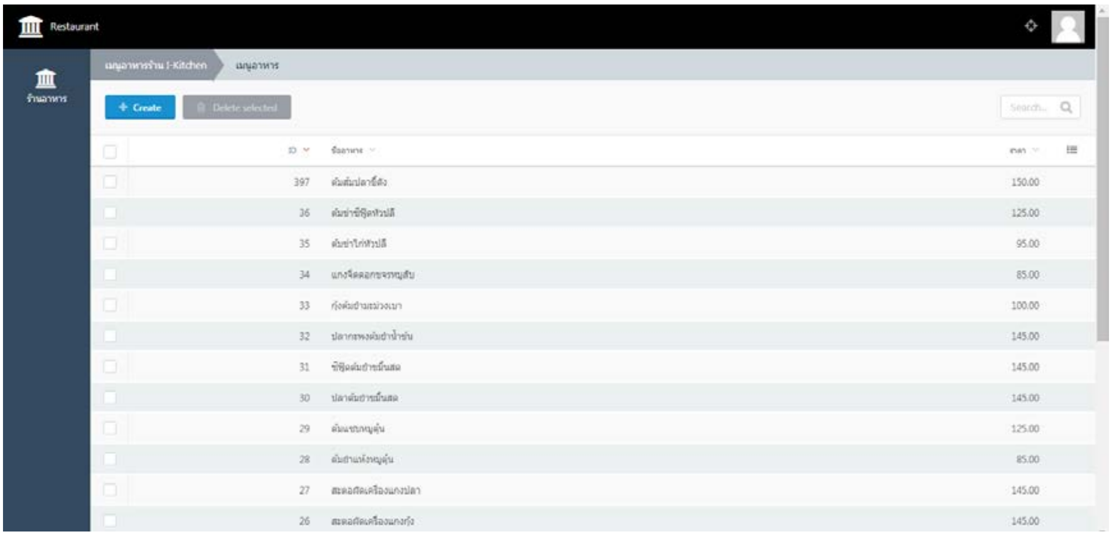
2.1 คลิกปุ่ม Create เพื่อเพิ่มข้อมูลร้านอาหาร ดังรูปที่ ก-22



รูปที่ ก-22 หน้าเพิ่มข้อมูลร้านอาหาร

3. การเพิม ลบ แก้ไข ข้อมูลรายการอาหาร

3.1 คลิกปุ่ม จัดการรายการอาหาร เพื่อเพิ่มรายการรายการอาหาร ดังรูปที่ ก-23

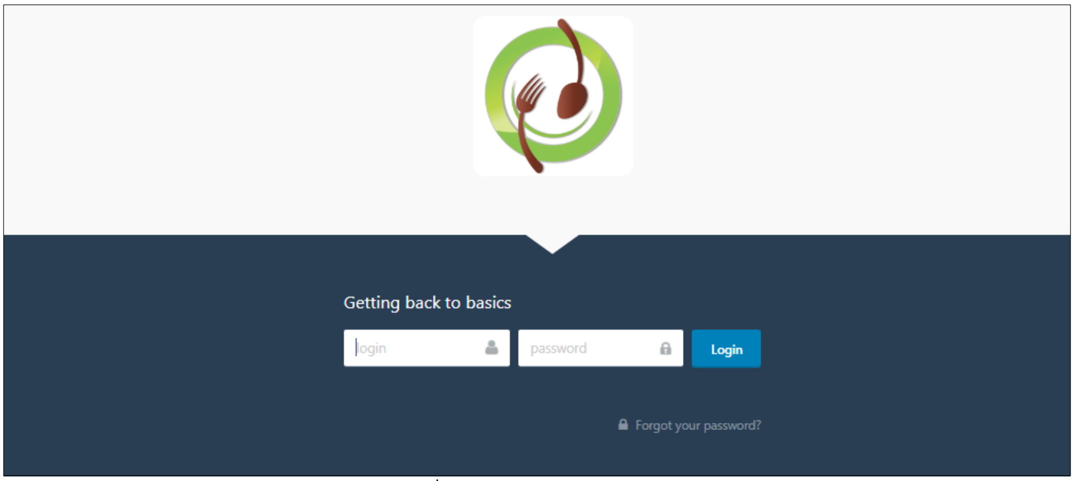


รูปที่ ก-23 หน้าเพิ่มข้อมูลรายการอาหาร

คู่มือใช้งานสำหรับ Admin

1. การเพิ่ม ลบ แก้ไข ข้อมูลรายการอาหารตั้งต้น

1.1 เปิดเว็บเบราเซอร์ (Web Browser) แล้วกรอก URL ชื่อ **http://kinrrai.kritlab.com/backend** จะปรากฏหน้าจอดังรูปที่ กรอกข้อมูล UserName และรหัสผ่าน ดังรูปที่ ก-24

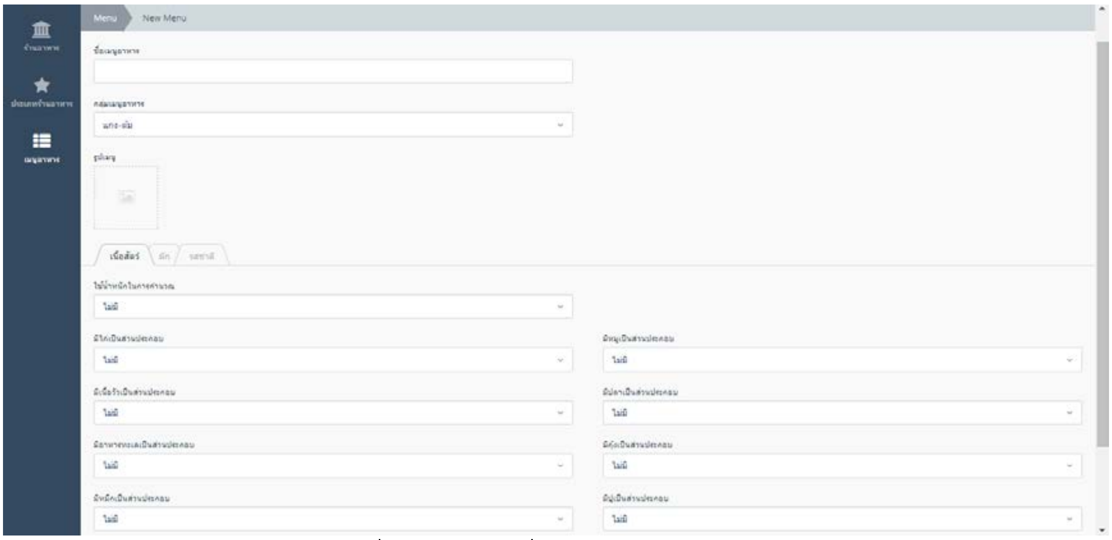


รูปที่ ก-24 หน้าเข้าสู่ระบบ

1.1 คลิกเมนู “รายการอาหาร” จะปรากฏรายการอาหารทั้งหมด ดังรูปที่ ก-25

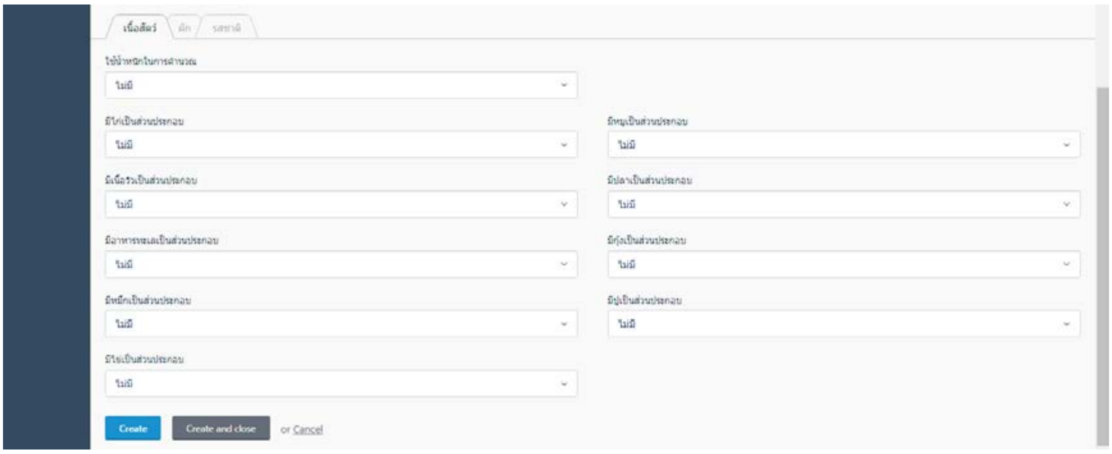
รูปที่ ก-25 หน้ารายการอาหาร

1.2 คลิกปุ่ม “Create” เพื่อเพิ่มข้อมูลรายการอาหาร จะปรากฎหน้าจอดังรูปที่ ก-26



รูปที่ ก-26 หน้าเพิ่มข้อมูลรายการอาหาร

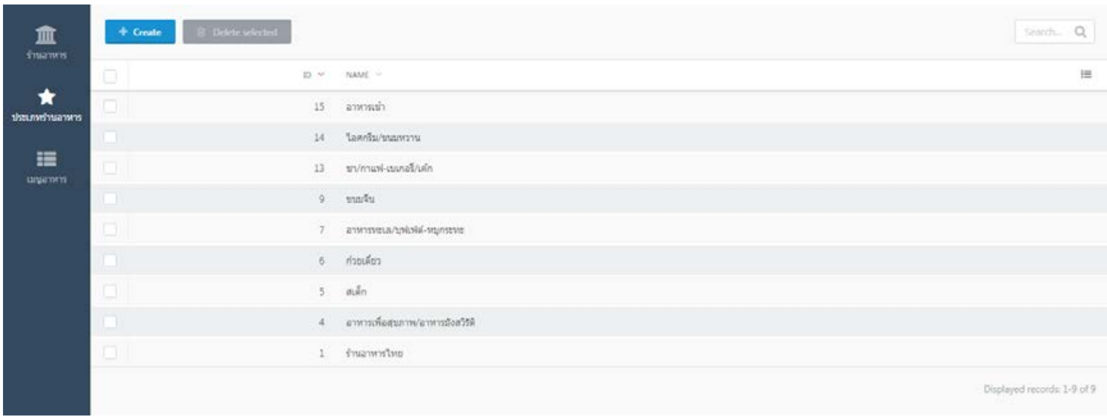
2. การกำหนดค่าคะแนนความชอบส่วนประกอบของรายการอาหารและการกำหนดค่าน้ำหนัก ดังรูปที่ ก-27



รูปที่ ก-27 หน้ากำหนดคะแนนความชอบและค่าน้ำหนักให้กับรายการอาหาร

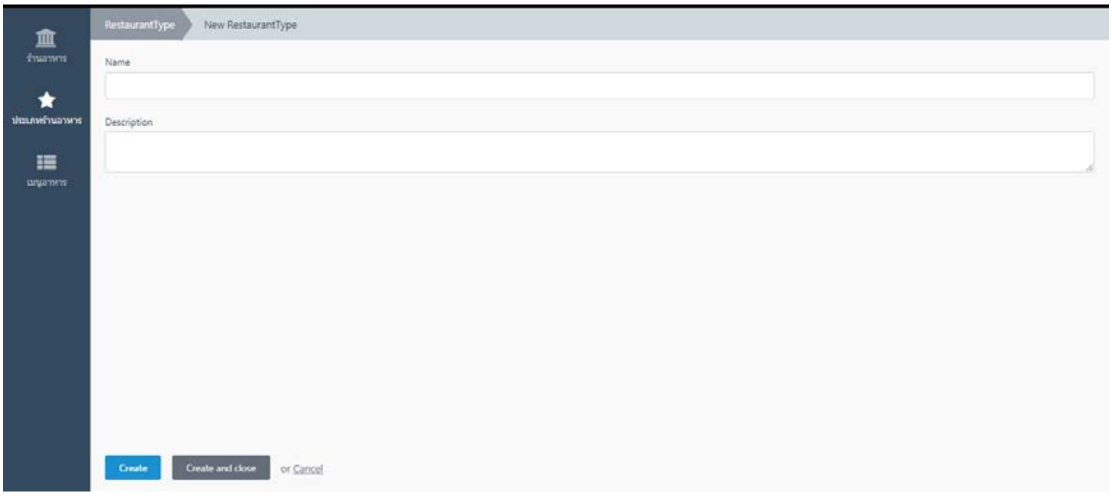
3. การเพิ่ม ลบ แก้ไข ข้อมูลประเภทร้านอาหาร

3.1 คลิกเมนู “ประเภทร้านอาหาร” จะปรากฏประเภทร้านอาหารทั้งหมดดังรูปที่ ก-28



รูปที่ ก-28 หน้าประเภทร้านอาหาร

3.2 คลิกปุ่ม “Create” เพื่อเพิ่มข้อมูลประเภทร้านอาหาร จะปรากฏหน้าจอดังรูปที่ ก-29



รูปที่ ก-29 หน้าเพิ่มข้อมูลประเภทร้านอาหาร

ภาคผนวก ข

ตัวอย่างชุดคำสั่งระบบแนะนำร้านอาหารอัตโนมัติบนสมาร์ทโฟนโดยใช้ข้อมูลเชิงตำแหน่งและรายการอาหาร

ตัวอย่างชุดคำสั่ง API

ลำดับ	API Name	URL
1	Recommend Menu	/api/findmenu
2	Recoomend Restaurant	/api/findrestaurant
3	Home	/api/home
4	Restaurant	/api/restaurant
5	Find Restaurant by id	/api/restaurant/byids
6	User Register	/api/user/register
7	User Set Profile	/api/user/update-profile
8	Vote Restaurant	/api/vote/restaurant/:id

1. Recommend Menu **คำสั่งแนะนำรายการอาหาร**

```
use Kritlab\Kinrrai\Models\Menu;
use RainLab\User\Models\User;
function onStart(){
        $user = Auth::findUserByLogin(input('login'));
        if($user === null){
                return response()->json([
                    'status' => 'fail',
                    'message' => 'User not found'
                ]);
        }
        $menu = Menu::all();
        $meatVar = [
                'meat_chicken',
                'meat_pork',
                'meat_beef',
                'meat_fish',
```

```
        'meat_seafood',
        'meat_squid',
        'meat_prawn',
        'meat_shellfish',
        'meat_crab',
        'meat_egg',
];
$vegetableVar = [
        'vegetable_tomato',
        'vegetable_must',
        'vegetable_ginger',
        'vegetable_bamboo',
        'vegetable_botle',
        'vegetable_pump',
        'vegetable_bruss',
        'vegetable_corn',
        'vegetable_aspara',
        'vegetable_pea',
        'vegetable_mush',
        'vegetable_bean',
        'vegetable_cucumber',
        'vegetable_longbean',
        'vegetable_carrot',
        'vegetable_turnip',
        'vegetable_kale',
        'vegetable_waterspinach',
        'vegetable_salad',
        'vegetable_eggplant',
        'vegetable_chapru',
```

```
        'vegetable_chamoung',
        'vegetable_chaaom',
        'vegetable_coconut',
        'vegetable_mara',
        'vegetable_kajorn',
        'vegetable_banana',
        'vegetable_cashew',
        'vegetable_sator',
        'vegetable_taro',
        'vegetable_wan',
        'vegetable_tumleng',
        'vegetable_boub',
        'vegetable_leang',
        'vegetable_broccoli',
];
$tasteVar = [
        'taste_hot',
        'taste_sweet',
        'taste_salt',
        'taste_sour',
];
$userMeat = [];
foreach ($meatVar as $meat){
        $userMeat[$meat] = ($user->$meat / 5);
}

$userVegetable = [];
foreach ($vegetableVar as $vegetable){
        $userVegetable[$vegetable] = ($user->$vegetable / 5);
```

```
}
$userTaste = [];
foreach ($tasteVar as $taste){
        $userTaste[$taste] = ($user->$taste / 5);
}
$grandTotalMenu = [];
$menu->each(function($item)
use($meatVar,$vegetableVar,$tasteVar,$userMeat,$userVegetable,$userTaste,&$grand
TotalMenu){

        $menuMeat = [];
        $totalMeat = 0;
        $menuVegetable = [];
        $totalVegetable = 0;
        $menuTaste = [];
        $totalTaste = 0;
        foreach ($meatVar as $meat){
                $menuMeat[$meat] = abs((($item->$meat / 5) * ($item-
>meat_is_use_weight/5)) - $userMeat[$meat]);
        }
        foreach ($vegetableVar as $vegetable){
                $menuVegetable[$vegetable] = abs((($item->$vegetable
/ 5) * ($item->vegetable_is_use_weight/5)) - $userVegetable[$vegetable]);
        }
        foreach ($tasteVar as $taste){
                $menuTaste[$taste] = abs((($item->$taste / 5) * ($item-
>taste_is_use_weight/5)) - $userTaste[$taste]);
        }
        $totalMeat = array_sum($menuMeat)/count($menuMeat);
```

```
                    $totalVegetable =
array_sum($menuVegetable)/count($menuVegetable);
                    $totalTaste = array_sum($menuTaste)/count($menuTaste);
                    $grandTotal = 1/(1 + $totalMeat + $totalVegetable +
$totalTaste);
                    $data = $item->toArray();
                    $data['grand_total'] = $grandTotal;
                    $grandTotalMenu[] = $data;
            });
            $grandTotalMenu = collect($grandTotalMenu);

            $sorted = $grandTotalMenu->sortByDesc(function ($item, $key) {
              return $item['grand_total'];
            })->values()->all();
            return response()->json([
              'status' => 'success',
              'data' => [
                    'menus' => $sorted
              ]
            ]);
      }
```

2. Recoomend Restaurant คำสั่งแนะนำร้านอาหาร

```
      use Kritlab\Kinrrai\Models\Restaurant;
      use Kritlab\Kinrrai\Models\RestaurantMenu;
      use RainLab\User\Models\User;
      use Location\Coordinate;
      use Location\Distance\Vincenty;
```

```
function onStart(){
        try{
                $user = Auth::findUserByLogin(input('login'));
        } catch (Exception $ex) {
            return response()->json([
                    'status' => 'fail',
                    'message' => $ex->getMessage()
                ]);
        }
        $menuIds = input('menu_ids');
        $menuIds = json_decode($menuIds, true);
        $currentLocation = input('current_location');
        $results = [];
        $restaurantIds = RestaurantMenu::distinct()->select('restaurant_id')-
>whereIn('menu_id', $menuIds)->lists('restaurant_id');

        $restaurants = Restaurant::whereIn('id', $restaurantIds)->get()-
>each(function($item) use($user, $menuIds, &$results){
                $total = 0;
                if($user->level === $item->level){
                        $total++;
                }
                if($user->is_close === $item->is_close){
                        $total++;
                }
                if($user->is_park === $item->is_park){
                        $total++;
                }
                if($user->is_halal === $item->is_halal){
```

```
                        $total++;
                    }
                    if($user->distance <= $item->distance){
                        $total++;
                    }
                    $item->menus->each(function($item) use($menuIds, &$total){
                        if(in_array($item->menu_id, $menuIds)){
                            $total++;
                        }
                    });
                    $data = $item->toArray();
                    $data['total'] = $total;
                    $results[] = $data;
                });

                $results = collect($results);
                $sorted = $results->sortByDesc(function ($item, $key) {
                    return $item['total'];
                })->values()->all();
                $sorted = collect($sorted)->map(function($restaurant)
use($currentLocation){
                    $currentLocation = json_decode($currentLocation, true);
                    $latitude = array_get($currentLocation, 'lat', 0);
                    $longtitude = array_get($currentLocation, 'lng', 0);
                    $distance = null;
                    if($latitude > 0 && $longtitude > 0 && $restaurant['latitude'] > 0
&& $restaurant['longtitude'] > 0){
                        $coordinate1 = new Coordinate($latitude, $longtitude);
```

```
                    $coordinate2 = new Coordinate($restaurant['latitude'],
$restaurant['longtitude']);
                    $calculator = new Vincenty();

                    $distance = $calculator->getDistance($coordinate1,
$coordinate2);
                }
                $restaurant = $restaurant;
                $restaurant['distance'] = $distance ? floor($distance/1000).'กม.' :
null;
                return $restaurant;
            });
        return response()->json([
                'status' => 'success',
                'data' => [
                    'restaurants' => $sorted
                ]
            ]);
        }
```

3.Home **คำสั่งแสดงผลหน้าหลัก**

```
use Kritlab\Kinrrai\Models\Restaurant;
    use Kritlab\Kinrrai\Models\Menu;
    use Location\Coordinate;
    use Location\Distance\Vincenty;
    use Carbon\Carbon;

    function onStart(){
        $currentLocation = input('current_location');
```

```
$restaurantModel = Restaurant::with('restaurantType')-
>where('created_at', '>', Carbon::now()->subDays(7))->orderBy('created_at', 'desc');
		if($restaurantModel->count() < 5){
			$restaurantModel = Restaurant::with('restaurantType')-
>orderBy('created_at', 'desc')->limit(5);
		}

		$restaurants = $restaurantModel->get()->map(function($restaurant)
use($currentLocation){
			$currentLocation = json_decode($currentLocation, true);
			$latitude = array_get($currentLocation, 'lat', 0);
			$longtitude = array_get($currentLocation, 'lng', 0);
			$distance = null;
			if($latitude > 0 && $longtitude > 0 && $restaurant->latitude > 0
&& $restaurant->longtitude > 0){
				$coordinate1 = new Coordinate($latitude, $longtitude);
				$coordinate2 = new Coordinate($restaurant->latitude,
$restaurant->longtitude);
				$calculator = new Vincenty();

				$distance = $calculator->getDistance($coordinate1,
$coordinate2);
			}
			$restaurant = $restaurant->toArray();
			$restaurant['distance'] = $distance ? floor($distance/1000).'กม.' :
null;
			return $restaurant;
		});
```

```
                $menuModel = Menu::where('created_at', '>', Carbon::now()-
>subDays(7))->orderBy('created_at', 'desc');
                if($menuModel->count() < 5){
                        $menuModel = Menu::limit(5)->orderBy('created_at', 'desc')-
>limit(5);
                }

                $menus = $menuModel->get();
                return response()->json([
                   'status' => 'success',
                   'data' => [
                           'restaurants' => $restaurants,
                           'menus' => $menus
                   ]
                ]);
        }
```

4. Restaurant คำสั่งแสดงผลหน้ารายละเอียดของร้านอาหาร

```
use Kritlab\Kinrrai\Models\Restaurant;
    use Location\Coordinate;
    use Location\Distance\Vincenty;

    function onStart(){

            $restaurants = Restaurant::orderBy('name', 'desc')->get()-
>map(function($restaurant){
                    $latitude = input('latitude');
                    $longtitude = input('longtitude');
                    $distance = null;
```

```
            if($latitude && $longtitude){
                    $coordinate1 = new Coordinate($latitude, $longtitude);
                    $coordinate2 = new Coordinate($restaurant->latitude,
$restaurant->longtitude);
                    $calculator = new Vincenty();

                    $distance = $calculator->getDistance($coordinate1,
$coordinate2);
            }
            return [
                    'name' => $restaurant->name,
                    'level' => $restaurant->level,
                    'is_close' => $restaurant->is_close,
                    'is_halal' => $restaurant->is_halal,
                    'is_park' => $restaurant->is_park,
                    'address' => $restaurant->address,
                    'tel' => $restaurant->tel,
                    'open_time' => $restaurant->open_time,
                    'image' => $restaurant->image,
                    'latitude' => $restaurant->latitude,
                    'longtitude' => $restaurant->longtitude,
                    'distance' => floor($distance/1000),
                    'image_url' => $restaurant->image_url,
                    'created_at' => $restaurant->created_at-
>toDateTimeString(),
                    'created_at' => $restaurant->created_at-
>toDateTimeString(),
                    'rest_type_id' => $restaurant->restaurantType->id,
```

```
                    'rest_type_name' => $restaurant->restaurantType-
>name
                ];
            });

            return response()->json([
                'status' => 'success',
                'data' => [
                    'restaurants' => $restaurants
                ]
            ]);
    }
```

5. Find Restaurant by idMenu คำสั่งค้นหาร้านอาหารด้วยรายการอาหาร

```
            use Kritlab\Kinrrai\Models\Restaurant;
            use Location\Coordinate;
            use Location\Distance\Vincenty;

            function onStart(){

                $restaurants = Restaurant::orderBy('name', 'desc')->get()-
    >map(function($restaurant){
                        $latitude = input('latitude');
                        $longtitude = input('longtitude');
                        $distance = null;
                        if($latitude && $longtitude){
                            $coordinate1 = new Coordinate($latitude,
    $longtitude);
                            $coordinate2 = new Coordinate($restaurant-
    >latitude, $restaurant->longtitude);
```

```
                });

                        $calculator = new Vincenty();

                        $distance = $calculator-
>getDistance($coordinate1, $coordinate2);
                    }
                    return [
                        'name' => $restaurant->name,
                        'level' => $restaurant->level,
                        'is_close' => $restaurant->is_close,
                        'is_halal' => $restaurant->is_halal,
                        'is_park' => $restaurant->is_park,
                        'address' => $restaurant->address,
                        'tel' => $restaurant->tel,
                        'open_time' => $restaurant->open_time,
                        'image' => $restaurant->image,
                        'latitude' => $restaurant->latitude,
                        'longtitude' => $restaurant->longtitude,
                        'distance' => floor($distance/1000),
                        'image_url' => $restaurant->image_url,
                        'created_at' => $restaurant->created_at-
>toDateTimeString(),
                        'created_at' => $restaurant->created_at-
>toDateTimeString(),
                        'rest_type_id' => $restaurant->restaurantType-
>id,
                        'rest_type_name' => $restaurant-
>restaurantType->name
                    ];
                });
```

```
                return response()->json([
                    'status' => 'success',
                    'data' => [
                            'restaurants' => $restaurants
                    ]
                ]);
        }
```

6. User Register คำสั่งสมัครสมาชิก

```
function onStart(){
        try{
                $user = Auth::register([
                        'name' => input('name'),
                        'surname' => input('surname'),
                    'email' => input('email'),
                    'password' => input('password'),
                    'password_confirmation' => input('password_confirmation'),
                ], true);
        } catch (Exception $ex) {
      return response()->json([
                    'status' => 'fail',
                    'message' => $ex->getMessage()
                ]);
    }
        return response()->json([
            'status' => 'success',
            'data' => [
                    'user' => $user
            ]
```

```
        ]);
    }
```

7. User Set Profile คำสั่งตั้งค่าโปรไฟล์

```
        function onStart(){
            try{
                $user = Auth::findUserByLogin(input('email'));
                $user->name = input('name');
                $user->surname = input('surname');
                $user->level = input('level');
                $user->distance = input('distance');
                $user->is_close = input('is_close');
                $user->is_halal = input('is_halal');
                $user->is_park = input('is_park');

                $meatVar = [
                    'meat_chicken',
                    'meat_pork',
                    'meat_beef',
                    'meat_fish',
                    'meat_seafood',
                    'meat_squid',
                    'meat_prawn',
                    'meat_shellfish',
                    'meat_crab',
                    'meat_egg',
                ];
                foreach($meatVar as $mv){
                    $user->$mv = 0;
                }
```

```
$meat5 = input('meat5');
foreach($meat5 as $m5){
        $user->{'meat_'.$m5} = 5;
}
$meat4 = input('meat4');
foreach($meat4 as $m4){
        $user->{'meat_'.$m4} = 4;
}
$meat3 = input('meat3');
foreach($meat3 as $m3){
        $user->{'meat_'.$m3} = 3;
}
$meat2 = input('meat2');
foreach($meat2 as $m2){
        $user->{'meat_'.$m2} = 2;
}
$meat1 = input('meat1');
foreach($meat1 as $m1){
        $user->{'meat_'.$m1} = 1;
}

$vegetableVar = [
        'vegetable_tomato',
        'vegetable_must',
        'vegetable_ginger',
        'vegetable_bamboo',
        'vegetable_botle',
        'vegetable_pump',
```

'vegetable_bruss',
'vegetable_corn',
'vegetable_aspara',
'vegetable_pea',
'vegetable_mush',
'vegetable_bean',
'vegetable_cucumber',
'vegetable_longbean',
'vegetable_carrot',
'vegetable_turnip',
'vegetable_kale',
'vegetable_waterspinach',
'vegetable_salad',
'vegetable_eggplant',
'vegetable_chapru',
'vegetable_chamoung',
'vegetable_chaaom',
'vegetable_coconut',
'vegetable_mara',
'vegetable_kajorn',
'vegetable_banana',
'vegetable_cashew',
'vegetable_sator',
'vegetable_taro',
'vegetable_wan',
'vegetable_tumleng',
'vegetable_boub',
'vegetable_leang',
'vegetable_broccoli',

```
];
foreach($vegetableVar as $vv){
	$user->$vv = 0;
}

$vegetable5 = input('vegetable5');
foreach($vegetable5 as $v5){
	$user->{'vegetable_'.$v5} = 5;
}
$vegetable4 = input('vegetable4');
foreach($vegetable4 as $v4){
	$user->{'vegetable_'.$v4} = 4;
}
$vegetable3 = input('vegetable3');
foreach($vegetable3 as $v3){
	$user->{'vegetable_'.$v3} = 3;
}
$vegetable2 = input('vegetable2');
foreach($vegetable2 as $v2){
	$user->{'vegetable_'.$v2} = 2;
}
$vegetable1 = input('vegetable1');
foreach($vegetable1 as $v1){
	$user->{'vegetable_'.$v1} = 1;
}

$tasteVar = [
	'taste_hot',
	'taste_sweet',
```

```
        'taste_salt',
        'taste_sour',
];
foreach($tasteVar as $tv){
        $user->$tv = 0;
}

$taste5 = input('taste5');
foreach($taste5 as $t5){
        $user->{'taste_'.$t5} = 5;
}
$taste4 = input('taste4');
foreach($taste4 as $t4){
        $user->{'taste_'.$t4} = 4;
}
$taste3 = input('taste3');
foreach($taste3 as $t3){
        $user->{'taste_'.$t3} = 3;
}
$taste2 = input('taste2');
foreach($taste2 as $t2){
        $user->{'taste_'.$t2} = 2;
}
$taste1 = input('taste1');
foreach($taste1 as $t1){
        $user->{'taste_'.$t1} = 1;
}
```

```
                    if(input('password', null) !== null &&
input('password_confirmation', null) !== null){
                            if(input('password') !==
input('password_confirmation')){
                                    return response()->json([
                                        'status' => 'fail',
                                        'message' => 'Password not match'
                                    ]);
                            }
                            $user->password = input('password');
                    }

                    $user->save();

                    if (strlen(post('password'))) {
                Auth::login($user->reload(), true);
            }

              } catch (Exception $ex) {
            return response()->json([
                        'status' => 'fail',
                        'message' => $ex->getMessage()
                    ]);
        }
              return response()->json([
                  'status' => 'success',
                  'data' => [
                        'user' => $user
                  ]
```

```
            ]);
        }
```

8. Vote Restaurant คำสั่งโหวดให้คะแนนร้านอาหาร

```
use Kritlab\Kinrrai\Models\RestaurantRating;
function onStart(){
        try{
                $user = Auth::findUserByLogin(input('login'));
        } catch (Exception $ex) {
      return response()->json([
                   'status' => 'fail',
                   'message' => $ex->getMessage()
                ]);
   }
        try{
                $restaurantRating = new RestaurantRating;
                $restaurantRating->user_id = $user->id;
                $restaurantRating->restaurant_id = $this->param('id');
                $restaurantRating->star = input('star');
                $restaurantRating->save();
        } catch (Exception $ex) {
      return response()->json([
                   'status' => 'fail',
                   'message' => $ex->getMessage()
                ]);
   }
        return response()->json([
           'status' => 'success',
```

```
            'data' => [
                'rating' => $restaurantRating
            ]
        ]);
}
```

ภาคผนวก ค

ตัวอย่างแบบสอบถามเพื่อประเมินความพึงพอใจในการใช้งาน
ระบบแนะนำร้านอาหารอัตโนมัติบนสมาร์ทโฟนโดยใช้ข้อมูลเชิงตำแหน่งและรายการอาหาร

ตัวอย่าง แบบสอบถามเพื่อประเมินความพึงพอใจในการใช้งานระบบแนะนำร้านอาหารอัตโนมัติบนสมาร์ทโฟนโดยใช้ข้อมูลเชิงตำแหน่งและรายการอาหาร

แบบสอบถามเพื่อประเมินความพึงพอใจในการใช้งาน
ระบบแนะนำร้านอาหารอัตโนมัติบนมือถือโดยใช้ข้อมูลเชิงตำแหน่งและเมนูอาหาร

คำชี้แจง

แบบสอบถามความคิดเห็นนี้ เป็นการสอบถามข้อมูลความคิดเห็นของผู้เชี่ยวชาญด้านคอมพิวเตอร์และ ผู้ใช้ทั่วไปภายหลังที่ได้ทดลองใช้โปรแกรมที่พัฒนาขึ้น แบ่งการประเมินผลออกเป็น 4 ด้านดังนี้

1.1 ด้าน Functional Requirement Test การประเมินผลความถูกต้อง และประสิทธิภาพของระบบว่าตรงตามความต้องการของผู้ใช้ระบบมากน้อยเพียงใด

1.2 ด้าน Functional Test เป็นการประเมินความถูกต้องและประสิทธิภาพในการทำงานของระบบว่าสามารถทำงานได้ตามฟังก์ชันงานขอระบบมากน้อยเพียงใด

1.3 ด้าน Usability Test เป็นการประเมินลักษณะการออกแบบระบบว่ามีความง่ายต่อการใช้งานมากน้อยเพียงใด

1.4 ด้าน Security Test เป็นการประเมินระบบในด้านการรักษาความปลอดภัยของข้อมูลในระบบว่ามีความปลอดภัยมากน้อยเพียงใด

ในการตอบแบบสอบถามนี้ ขอความกรุณาให้ท่านดำเนินการดังนี้

ทำเครื่องหมาย ✓ ลงในช่องในแบบสอบถามที่ตรงกับระดับความคิดเห็นของท่านมากที่สุด โดยตัวเลขของระดับประสิทธิภาพต่อแบบประเมินแต่ละด้านมีความหมายดังนี้

5 หมายถึง มีความพึงพอใจในระดับดีมาก

4 หมายถึง มีความพึงพอใจในระดับดี

3 หมายถึง มีความพึงพอใจในระดับปานกลาง

2 หมายถึง มีความพึงพอใจในระดับน้อย

1 หมายถึง มีความพึงพอใจในระดับน้อยมากๆ

ตัวอย่างการประเมิน

รายการประเมิน	ระดับความพึงพอใจ				
	5	4	3	2	1
ความง่ายต่อการใช้งานของระบบ		✓			

แบบสอบถาม

ส่วนที่ 1 สถานภาพของผู้ตอบแบบสอบถาม

- ◯ ผู้ดูแลระบบ
- ◯ นักศึกษา สาขา...
- ◯ อาจารย์
- ◯ ผู้เชี่ยวชาญด้านคอมพิวเตอร์

ส่วนที่ 2 การประเมินผล

2.1 การประเมินระบบด้าน Functional Requirement Test เป็นการประเมินผลความถูกต้องและประสิทธิภาพของระบบว่าตรงตามความต้องการของผู้ใช้ระบบมากน้อยเพียงใด

รายการประเมิน	ระดับความพึงพอใจ				
	5	4	3	2	1
1.ความสามารถของระบบในด้านการจัดการข้อมูลผู้ใช้					
2.ความสามารถของระบบในด้านการแนะนำเมนูอาหาร					
3.ความสามารถของระบบในด้านการแนะนำร้านอาหาร					
4.ความสามารถของระบบในด้านการค้นหาร้านอาหาร					
5.ความสามารถของระบบในด้านการเพิ่ม แก้ไขข้อมูล โปรไฟล์					
6.ความสามารถของระบบในด้านการแสดงตำแหน่งที่ตั้งของร้านอาหารบนแผนที่					

2.2 ด้าน Functional Test เป็นการประเมินความถูกต้องและประสิทธิภาพในการทำงานของระบบว่าสามารถทำงานได้ตามฟังก์ชันงานขอระบบมากน้อยเพียงใด

รายการประเมิน	ระดับความพึงพอใจ				
	5	4	3	2	1
1.ความถูกต้องในการจัดเก็บข้อมูลนำเข้า					
2.ความถูกต้องในการค้นหาข้อมูล					
3.ความถูกต้องในการปรับปรุงแก้ไขข้อมูล					
4.ความถูกต้องในการลบข้อมูล					
5.ความถูกต้องของผลลัพธ์ที่ได้จาการประมวลผล					
6.เมนูอาหารที่แนะนำมีความคล้ายคลึงกับความต้องการของผู้ใช้					
7.ร้านอาหารที่แนะนำมีความคล้ายคลึงกับความต้องการของผู้ใช้					
8.ความครอบคลุมของโปรแกรมที่พัฒนากับระบบจริง					
9.การป้องกันข้อผิดพลาดที่อาจจะเกิดขึ้น					

2.3 ด้าน Usability Test เป็นการประเมินลักษณะการออกแบบระบบว่ามีความง่ายต่อการใช้งานมากน้อยเพียงใด

รายการประเมิน	ระดับความพึงพอใจ				
	5	4	3	2	1
1.ความง่ายต่อการใช้งานระบบ					
2.ความเหมาะสมในการเลือกใช้ขนาดตัวอักษรและจอภาพ					
3.ความเหมาะสมในการใช้สีตัวอักษรและรูปภาพ					
4.ความเหมาะสมในการใช้ข้อความ สัญลักษณ์หรือรูปภาพในการอธิบายสื่อความหมาย					
5.ความเป็นมาตรฐานเดียวกันในการออกแบบหน้าจอภาพ					
6.ความเหมาะสมในการปฏิสัมพันธ์โต้ตอบกับผู้ใช้					
7.ความเหมาะสมในการวางตำแหน่งส่วนประกอบของจอภาพ					
8.คำศัพท์ที่ใช้มีความคุ้นเคยและปฏิบัติตามได้โดยง่าย					

2.4 ด้าน Security Test เป็นการประเมินระบบในด้านการรักษาความปลอดภัยของข้อมูลในระบบว่ามีมากน้อยเพียงใด

รายการประเมิน	ระดับความพึงพอใจ				
	5	4	3	2	1
1.การกำหนดรหัสผู้ใช้และรหัสผ่านในการตรวจสอบผู้เข้าใช้ระบบ					
2.การตรวจสอบสิทธิ์ก่อนการใช้งานของผู้ใช้ระบบในระดับต่าง ๆ					
3.การควบคุมให้ใช้งานตามสิทธิ์ของผู้ใช้ได้อย่างถูกต้อง					

ข้อเสนอแนะ

....

....

....

....

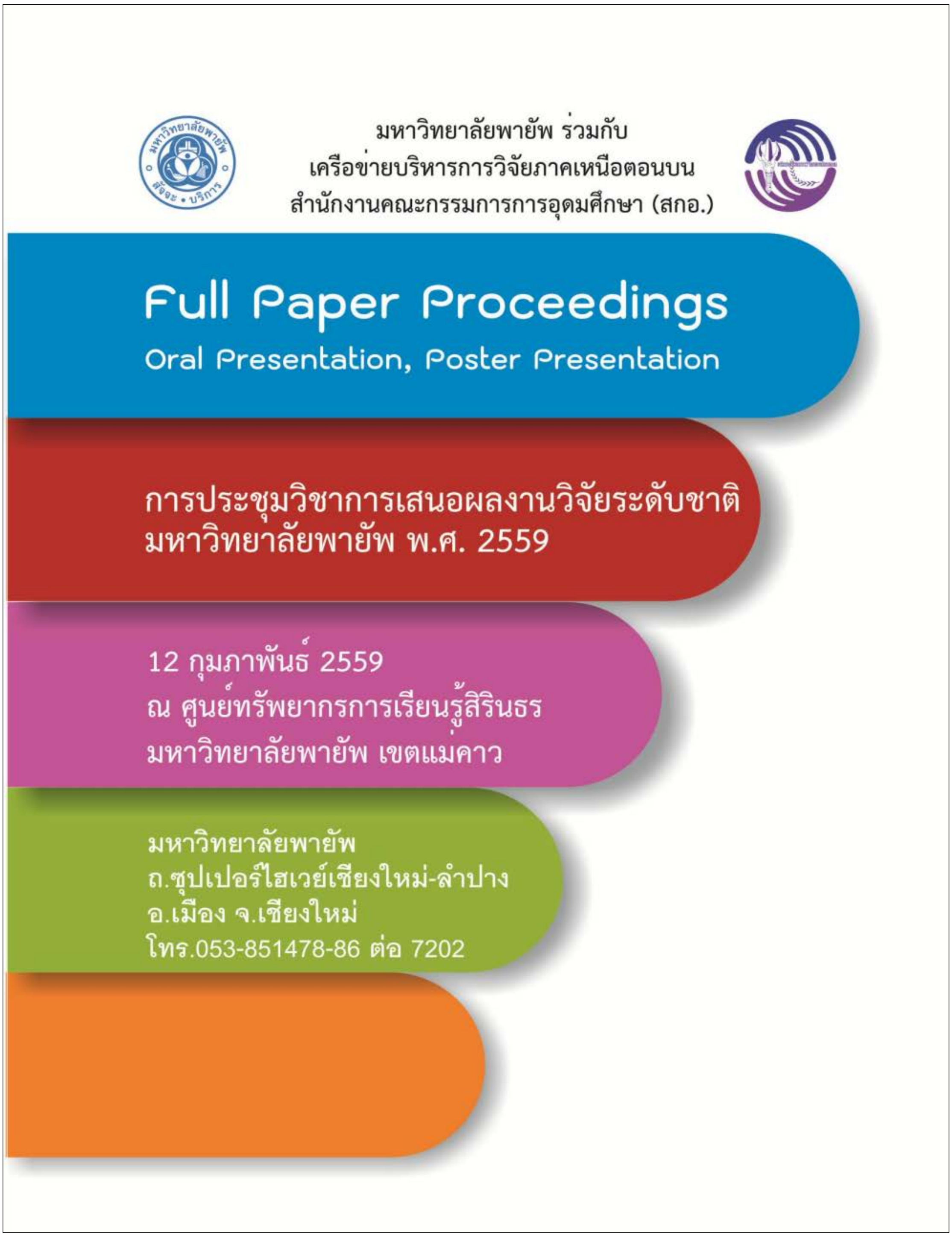
....

ขอขอบพระคุณเป็นอย่างสูงในการให้ข้อมูลที่เป็นจริงจากท่าน

ภาคผนวก ง
บทความวิจัยที่นำเสนอและได้รับการตีพิมพ์

บทความงานวิจัยเรื่องที่1

ได้นำเสนอและตีพิมพ์บทความการประชุมวิชาการและเสนอผลงานวิจัยระดับชาติมหาวิทยาลัยพายัพ พ.ศ.2559



การออกแบบต้นแบบระบบแนะนำร้านอาหารอัตโนมัติบนมือถือโดยใช้ข้อมูลเชิงตำแหน่งและเมนูอาหาร

Prototype Design for Restaurant Recommender System on Mobile using Location Based Service and Menu

ปาลิตา แสงศิริ[1] ทวีศักดิ์ เรืองพีระกุล[2] และนิคม สุวรรณวร[2]
Palita Sangsiri[1] Taweesak Reangperakun[2] and Nikom Suvanvorn[2]
[1]โครงการจัดการศึกษาพิเศษ หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาการจัดการเทคโนโลยีสารสนเทศ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยสงขลานครินทร์
[2] ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยสงขลานครินทร์

บทคัดย่อ

ปัจจุบันผู้บริโภคมีตัวเลือกในการเลือกร้านอาหารเพื่อรับประทานมากขึ้น การค้นหาและตัดสินใจเลือกร้านอาหารหรือเมนูที่ถูกใจก็ทำได้ยากมากยิ่งขึ้น ส่งผลให้ผู้บริโภคไม่สะดวกในการเข้าถึงข้อมูลของร้านอาหาร เมนูอาหาร ส่วนประกอบของเมนูอาหาร และรสชาติของอาหารที่ถูกใจ อีกทั้งไม่สะดวกในการค้นหาร้านอาหารที่อยู่บริเวณใกล้เคียงได้ ทางผู้วิจัยจึงมีแนวคิดในการออกแบบระบบที่จะนำมาแก้ไขปัญหาดังกล่าวข้างต้น ได้แก่ ระบบแนะนำ (Recommender System) สำหรับแนะนำเมนูอาหารตามความต้องการของผู้บริโภค (User Profile) เช่น ผู้บริโภคชอบรับประทานอาหารที่มี ไก่ และผักคะน้าเป็นส่วนประกอบ หรือเป็นเมนูที่มีรสชาติเผ็ด จากเมนูอาหารที่มีอยู่ (Item Profile) และตามตำแหน่งของร้านอาหาร และลักษณะของร้านอาหารที่ผู้บริโภคต้องการ (Restaurant Profile) โดยใช้เทคนิคการกรองแบบอิงเนื้อหา (Content-based Filtering) ซึ่งมีความสะดวกในการใช้งานผ่านอุปกรณ์สื่อสารเคลื่อนที่แบบพกพา โดยใช้การบริการเชิงตำแหน่ง Google Maps API ค้นหาพิกัดของร้านอาหารที่มีเมนูอาหารตรงตามความต้องการของผู้บริโภค และมีการบริการและจัดเก็บข้อมูลที่อยู่บนคลาวด์ (Cloud) เพื่อลดการประมวลผลบนเครื่อง Terminal ตอบสนองความต้องการของผู้ใช้บริการได้ทุกที่ทุกเวลา ลดความยุ่งยากในการเข้าถึงข้อมูลร้านอาหาร งานวิจัยชิ้นนี้จึงนำเสนอในส่วนของการวิเคราะห์และออกแบบสถาปัตยกรรมระบบ และส่วนติดต่อกับผู้ใช้ ของระบบแนะนำร้านอาหารอัตโนมัติบนมือถือโดยใช้ข้อมูลเชิงตำแหน่งและเมนูอาหาร

คำสำคัญ: ระบบแนะนำ ร้านอาหาร ข้อมูลเชิงตำแหน่ง

ABSTRACT

Nowadays restaurant has become various places. Finding and making a decision for food consumption are also difficulty to find a favorites restaurant, menu and ingredient. Including to finding a nearly restaurant also affect to make a decision for food consumption. The researchers had an idea to design the system to solving the problems that mention above. Recommender System is the program that recommends the menus for consumers such as finding the ingredient or the taste of food from menus, position of restaurant and restaurant profile. This research use content-based filtering techniques; Google Maps API led to more convenient. This system can be served and data storage on cloud led to decrease evaluation on terminal system. The Google Maps API service can be used all the time and reduced the complicate of restaurant searching. This research aim to analyze and design the architecture system and graphical user interface of the automatically restaurants recommends system on mobile phone by using spatial data and menus.

Keywords : Recommender System, Restaurant, Google Maps API

1 ที่อยู่ที่สามารถติดต่อได้ : 131/2 ม.1 ต.เกาะใหญ่ อ.กระแสสินธุ์ จ.สงขลา 90270 อีเมล์ : puyphay_red@hotmail.com
[2] ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยสงขลานครินทร์

บทนำ

ปัจจุบันธุรกิจร้านอาหารประเภทต่างๆ ได้เติบโตและเพิ่มจำนวนขึ้นอย่างรวดเร็ว เพื่อตอบสนองความต้องการของผู้บริโภค และการแข่งขันทางธุรกิจ ทำให้ผู้บริโภคมีตัวเลือกในการเลือกร้านอาหารเพื่อรับประทานมากขึ้น การค้นหาและตัดสินใจเลือกร้านอาหารหรือเมนูที่ถูกใจก็ทำได้ยากมากยิ่งขึ้น ผู้บริโภคโดยส่วนใหญ่จะค้นหาร้านอาหารและตัดสินใจเลือกอาหารเพื่อรับประทานโดยอ้างอิงจากประสบการณ์ที่เคยรับประทานที่ผ่านมา จากคำแนะนำของเพื่อนๆ หรือการประชาสัมพันธ์ต่างๆ ดังนั้นหากผู้บริโภคตัดสินใจเลือกร้านอาหารหรือเมนูอาหารโดยใช้ข้อมูลทั่วไปเท่านั้น เช่น ชื่อร้านอาหารประเภทอาหาร เป็นต้น ซึ่งการใช้ข้อมูลทั่วไปเหล่านั้น ไม่สามารถเข้าถึงข้อมูลที่แน่ชัดได้ อาทิเช่น เมนูอาหาร ส่วนประกอบของแต่ละเมนูอาหาร รสชาติของอาหาร ส่งผลต่อการเลือกร้านอาหารที่มีเมนูอาหาร หรือรสชาติที่ผู้บริโภคต้องการได้ ข้อจำกัดต่างๆเหล่านี้มีผลในการตัดสินใจเลือกร้านอาหารเพื่อรับประทาน ทำให้ไม่สามารถตอบสนองความต้องการที่แท้จริงได้

เทคโนโลยีที่นำมาใช้ในร้านอาหารปัจจุบันยังไม่สนับสนุนหรือแก้ปัญหาดังกล่าว หากพิจารณาถึงระบบบริหารจัดการร้านอาหาร ส่วนใหญ่เป็นระบบที่พัฒนาและใช้งานเฉพาะเจาะจงสำหรับร้านใดร้านหนึ่งเท่านั้น ทำให้มีระบบร้านอาหารบนเว็บ และระบบร้านอาหารบนอุปกรณ์สื่อสารเคลื่อนที่แบบพกพา ได้รับการพัฒนาและเป็นที่นิยมสำหรับร้านอาหารมากขึ้น รองรับฟังก์ชันการทำงานได้หลากหลาย เช่น ค้นหาร้านอาหารจากสถานที่ตั้งของร้าน ดูเมนูอาหาร ดูข้อมูลทั่วไปเกี่ยวกับร้าน เป็นต้น ตัวอย่างร้านอาหารบนมือถือ เช่น KFC Thailand ซึ่งเป็นระบบร้าน KFC ที่พัฒนาให้รองรับการใช้งานบนอุปกรณ์สื่อสารเคลื่อนที่แบบพกพา [1] และ Wongnai ระบบค้นหาร้านอาหาร มีฟังก์ชันสำหรับค้นหาร้านอาหารแสดงบนแผนที่ Google Maps แสดงความคิดเห็นและสามารถจัดระดับความนิยมให้กับร้านนั้นๆได้ [2] แต่ยังไม่สามารถแนะนำร้านอาหาร ดูรายการอาหาร ส่วนประกอบหรือรสชาติของอาหารแต่ละเมนูและราคาได้

ผู้วิจัยจึงมีแนวคิดในการออกแบบระบบที่จะนำมาแก้ไขปัญหาดังกล่าวข้างต้น ได้แก่ ระบบแนะนำ (Recommender System) ที่มีการแนะนำผู้บริโภคตามข้อกำหนดดังต่อไปนี้

- แนะนำเมนูอาหารตามความต้องการของผู้บริโภค (User Profile) โดยระบบจะดำเนินการค้นหาและเปรียบเทียบข้อมูลความต้องการของผู้บริโภคในฐานข้อมูล กับเมนูอาหารในร้านอาหารต่างๆ เพื่อจัดทำรายการอาหารและร้านอาหารแนะนำให้กับผู้บริโภคโดยอัตโนมัติ
- เมนูอาหาร(Item Profile) ที่ระบุชื่อรายการอาหาร ส่วนประกอบของแต่ละรายการอาหาร รสชาติของอาหาร รวมทั้งรูปภาพประกอบ
- ตำแหน่งของร้านอาหาร และลักษณะของร้านอาหารที่ผู้บริโภคต้องการ (Restaurant Profile) โดยใช้เทคนิคการกรองแบบอิงเนื้อหา (Content-based Filtering)

ระบบดังกล่าวส่งเสริมให้ผู้ใช้งานได้รับความสะดวกในการใช้งานผ่านอุปกรณ์สื่อสารเคลื่อนที่แบบพกพา โดยใช้การบริการเชิงตำแหน่ง Google Maps API ค้นหาพิกัดของร้านอาหารที่มีเมนูอาหารตรงตามความต้องการของผู้บริโภค ซึ่งมีการบริการและจัดเก็บข้อมูลที่อยู่บนคลาวด์ (Cloud) เพื่อลดการประมวลผลบนเครื่อง Terminal ตอบสนองความต้องการของผู้ใช้บริการได้ทุกที่ทุกเวลา ลดความยุ่งยากในการเข้าถึงข้อมูลร้านอาหาร

วัตถุประสงค์

1. วิเคราะห์และออกแบบระบบแนะนำร้านอาหารอัตโนมัติบนมือถือโดยใช้ข้อมูลเชิงตำแหน่งและเมนูอาหาร
2. วิเคราะห์วิจัยเพื่อหาวิธีการในการกรองข้อมูล (Content-based Filtering Technique) ที่เหมาะสมสำหรับการนำเสนอระบบแนะนำร้านอาหาร

วิธีดำเนินการวิจัย

ทฤษฎีที่เกี่ยวข้อง

- **ระบบแนะนำ (Recommender Systems)**

Recommender Systems[3] ระบบแนะนำข้อมูล ผลิตภัณฑ์ สินค้าให้กับผู้ใช้ระบบโดยอ้างอิงจากสมมุติฐาน การเรียนรู้ข้อมูลความชอบหรือความต้องการ ณ ขณะนั้นของผู้ใช้ ตัวอย่างเช่น ระบบแนะนำในการเลือกซื้อหนังสือ ซีดีเพลง หรือผลิตภัณฑ์อื่นๆ ประกอบด้วย 4 ส่วนหลักดังนี้

1) Item Profile คือข้อมูลพื้นฐานสำหรับใช้ในการประมวลผล เราจะต้องสร้างโปรไฟล์ของแต่ละ Item บันทึกหรือเก็บรวบรวมข้อมูล

2) User Profile คือ เป็นข้อมูลที่ได้จากการป้อนของผู้ใช้ เช่นการให้เรตติ้ง เช่นระดับความนิยม หรือประวัติการใช้งานในอดีต

3) ส่วนของอัลกอริธึมเป็นส่วนที่ใช้ประมวลผลข้อมูลออกมาให้ตรงตามความต้องการของผู้ใช้

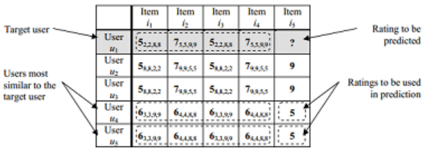
4) ส่วนในการนำเสนอให้แก่ผู้ใช้

- **Content-based Filtering[3]**

เป็นวิธีของการทำ Information Filtering การทำงานของเทคนิควิธีนี้คือ จะให้ความสำคัญต่อเนื้อหาของข้อมูลเป็นสำคัญเพื่อค้นหาข้อมูลที่ผู้ใช้แต่ละคนสนใจ ซึ่งวิธีนี้จะทำการคำนวณหาค่าความคล้ายคลึงกัน (Similarity Measure) ระหว่างข้อมูลหรือสินค้าที่ระบบมีอยู่กับความต้องการในตัวข้อมูลหรือสินค้าที่ผู้ใช้ต้องการสำหรับการให้เรตติ้งต่อข้อมูลยังไม่ทั่วถึงหรือยังไม่ได้ให้เรตติ้งแก่ข้อมูลนั้นๆจะไม่ส่งผลกระทบต่อเทคนิควิธีนี้

1) ระบบแนะนำที่ใช้ Multicriteria Rating [4]

ระบบแนะนำในปัจจุบันนั้นทำการค้นหากลุ่มผู้ใช้ที่มีความชอบใกล้เคียงกัน โดยพิจารณาจากคะแนนความชอบเพียงค่าเดียวที่ผู้ใช้ได้ให้คะแนนต่อไอเท็มนั้นๆ โดยไม่ได้คำนึงถึงคุณสมบัติที่ผู้ใช้ใช้ในการพิจารณาให้คะแนน ทำให้การค้นหาผู้ที่มีความชอบใกล้เคียงกัน (Neighbor) ของระบบมีความคลาดเคลื่อนและไม่สามารถค้นหาผู้ที่มีความชอบใกล้เคียงกัน (Neighbor) ที่ตรงกับความต้องการของผู้ใช้ที่แท้จริงได้



รูปที่1 : แสดงการทำนายค่า Rating**แบบ** Multicriteria**[4]**

จากรูปที่ 2 แสดงให้เห็นว่าผู้ใช้ U1 นั้นมีคะแนนความชอบแต่ละคุณสมบัติของ i1 ถึง i4 แตกต่างจากผู้ใช้ U2 และ U3 แต่มีคะแนนความชอบของแต่ละคุณสมบัติของ i1 ถึง i4 ใกล้เคียงกับผู้ใช้ U4 และ U5 มากกว่า ดังนั้นคะแนนความชอบที่ระบบควรแนะนำจึงควรเป็น 5 ตามผู้ใช้ U4 และ U5 ซึ่งเรียกวิธีการนี้ว่า Multicriteria Rating

- **การหาค่าความคล้ายคลึง (Similarity measure)**

K-NN (K-nearest Neighbor) [3] คือวิธีการหาสมาชิกที่มีคลามคล้ายคลึงกันมากที่สุด โดยการหาค่าระยะห่างระหว่างข้อมูล (Standard Euclidean Distance) เป็นวิธีการหาค่าความคล้ายคลึงที่ง่ายและนิยมใช้มากที่สุด มีฟังก์ชันการใช้งานดังนี้

X= (x1, x2, x3,...,xn)

โดย xi หมายถึงค่าข้อมูลในแอททริบิวต์ที่ I

การคำนวณระยะทาง d(X,Y) คือระยะห่างระหว่างข้อมูล X และ Y โดยการวัดระยะทางของ Standard Euclidean Distance กับการให้น้ำหนักหรือการระดับให้ความสำคัญกับแอททริบิวต์แต่ละตัวซึ่งเรียกว่า Distance Weighted เพื่อให้เหมาะกับการคำนวณหา ค่าความคล้ายคลึงกัน ดังแสดงในสมการ (1) และ (2)

$$d(X,Y)=\sqrt{(c-w_1)(x_1-y_1)^2+(c-w_2)(x_2-y_2)^2+...+(c-w_n)(x_n-y_n)^2} \quad (1)$$

$$d(X,Y)=\sqrt{\sum_{i=1}^{n}(C-W_i)(X_i-Y_i)^2} \quad (2)$$

โดยที่ กำหนดค่าคงที่ C = 6

X คือ เรคคอร์ดของข้อกำหนดของ User Profile ที่ผู้ใช้ต้องการ

Y คือ เรคคอร์ดของข้อกำหนดของ ItemProfile ในระบบ

W คือ ค่าของน้ำหนักของข้อมูลแต่ละแอททริบิวต์

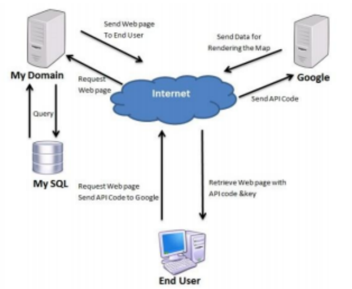
X_i คือ ค่าข้อมูลในแอททริบิวต์ที่ i ของเรคอร์ด X

Y_i คือ ค่าข้อมูลในแอททริบิวต์ที่ i ของเรคอร์ด Y

W_i คือ ค่าน้ำหนักของข้อมูลในแอททริบิวต์ที่ i

- **Google Maps API**

Google Maps API[5] เป็นส่วนที่ใช้ในการเชื่อมต่อกับ Google เพื่อเรียกใช้ข้อมูลทางภูมิศาสตร์ด้วยภาษาJavaScript และรับส่งข้อมูลในรูปแบบของภาษาเอกซ์เอ็มแอล (XML)



รูปที่ 2 : แสดงการทำงานกับ Google Maps API

ที่มา : ปฐมพงษ์ ฉับพลัน และฐิมาพร เพชรแก้ว. (2553).“**การประยุกต์ใช้ Google Maps API ในการเฝ้าระวังปัญหายาเสพติด**”, สำนักวิชาสารสนเทศศาสตร์ มหาวิทยาลัยวลัยลักษณ์.

- Cloud Server[6]

กลุ่มคอมพิวเตอร์จำลองบนก้อนเมฆที่ให้บริการโดยใช้ทรัพยากร ซอฟต์แวร์ และข้อมูลร่วมกัน เป็นการทำงานร่วมกันของเซิร์ฟเวอร์จำนวนมาก แม้เซิร์ฟเวอร์ตัวใดเสียไปจะไม่ส่งผลกระทบต่อการใช้งาน เพราะระบบจะทำการสวิทช์การใช้งานไปยังตัวอื่นอัตโนมัติ ซึ่งแตกต่างจากเว็บโฮสติ้งโดยทั่วไปเมื่ออุปกรณ์ตัวใดเสียก็จะส่งผลกระทบต่อการใช้งาน

งานวิจัยที่เกี่ยวข้อง

- เปรมฤดี ผลชอบ และ สุพจน์ นิตย์สุวัฒน์ [3] ระบบให้คำแนะนำในการเลือกซื้อคอมพิวเตอร์โน๊ตบุ้ด้วยเทคนิค Content-based Filtering งานวิจัยนี้ได้พัฒนาระบบและนำเสนอการนำเทคนิค Content-based Filtering ม า ประยุกต์ใช้ในการแนะนำการเลือกซื้อคอมพิวเตอร์โน๊ตบุ้คเพื่อทำนายความคล้ายคลึงกันของคอมพิวเตอร์โน๊ตบุ๊คจากคุณลักษณะพื้นฐานของคอมพิวเตอร์โน๊ตบุ๊คที่มีอยู่ในระบบกับความต้องการของผู้ใช้ ผลจากการวิจัยระบบจะแนะนำคอมพิวเตอร์โน้ตบุ๊คที่ใกล้เคียงกับความต้องการของผู้ใช้ ผลการประเมินความพึงพอใจของผู้ใช้อยู่ในระดับดีจากงานวิจัยชิ้นนี้สามารถนำหลักการที่ใช้ในการหาผู้ที่มีความชอบใกล้เคียงกัน (Neighbor) ไปประยุกต์ใช้เพื่อให้มีความเหมาะสมกับระบบแนะนำเมนูอาหารให้ตรงกับความต้องการของผู้ใช้มากที่สุดได้
- Keerthi S. Shetty and Sanjay Singh [7] **Cloud Base Application Development for Accessing Restaurant Information on Mobile Device using LBS**ได้ศึกษาวิจัย พัฒนาแอพลิเคชัน สำหรับร้านอาหารบนคราวด์ (Cloud) โดยใช้ Location Base Service ผู้ใช้สามารถค้นหาร้านอาหารโดยใช้การป้อนข้อมูล Location แล้วทำการลงทะเบียนเพื่อเข้าถึงข้อมูลของร้านอาหารนั้นๆพัฒนาบนระบบปฏิบัติการ Android บนโทรศัพท์มือถือ ด้วยภาษาจาวาสคริป ผลของการวิจัย ผู้ใช้สามารถเข้าถึงบริการต่างได้โดยตรงเมื่อไหร่ก็ได้ตามความต้องการการประมวลผลทำได้เร็วและมีการนำความสามารถของ Cloud Computing มาประยุกต์ใช้ ซึ่งมีการจัดเก็บข้อมูลที่มีความปลอดภัยและเชื่อถือจากงานวิจัยนี้สามารถนำ Location Base Service ไปประยุกต์ใช้ในการดึงพิกัดมาเพื่อแสดงตำแหน่งของร้านอาหารที่ได้จาการแนะนำของระบบบนมือถือและประยุกต์ใช้ Cloud Computing สำหรับจัดเก็บข้อมูลที่มีความปลอดภัยมากยิ่งขึ้นและรองรับแพลตฟอร์มที่หลากหลาย
- Giuseppe Di Fabbrizio and Narendra Gupta [8] Have2eat: a Restaurant Finder with Review Summarization for Mobile PhonesHave2eat ซึ่งเป็นแอพลิเคชันบนโทรศัพท์มือถือที่ได้รับความนิยมสำหรับไอโฟนและระบบปฏิบัติการ Android ช่วยให้ผู้ใช้สามารถค้นหาและเข้าถึงร้านอาหารที่อยู่บริเวณใกล้ๆ โดยจะแสดงร้านอาหารทั้งหมดที่อยู่รอบๆบริเวณนั้นผลของการวิจัย แอพลิเคชันบนมือถือสามารถค้นหาร้านอาหารได้ และสามารถจัดเรตติ้งร้านอาหารได้จากการรีวิวของลูกค้า ระบบสามารถค้นหาและแสดงข้อมูลของร้านอาหาร ผู้ใช้สามารถแสดงความคิดเห็นและใช้เทคนิคในการจัดเรตติ้งได้จากความคิดเห็นของลูกค้า สามารถนำไปประยุกต์ใช้ให้เกิดประโยชน์มากยิ่งขึ้น

จากงานวิจัยข้างต้นนี้ เทคนิค Content-based Filtering จะให้ความสำคัญต่อเนื้อหาของข้อมูลเป็นสำคัญเพื่อค้นหาข้อมูลที่ผู้ใช้แต่ละคนสนใจ ซึ่งวิธีนี้จะทำการคำนวณหาค่าความคล้ายคลึงกัน (Similarity Measure) ระหว่างข้อมูลหรือสินค้าที่ระบบมีอยู่กับความต้องการในตัวข้อมูลหรือสินค้าที่ผู้ใช้ต้องการ เป็นวิธีที่นิยมและเหมาะสมสำหรับงานวิจัยชิ้นนี้ ข้อดีของวิธีดังกล่าวนี้ การให้เรตติ้งต่อข้อมูลยังไม่ทั่วถึงหรือยังไม่ได้ให้เรตติ้งแก่ข้อมูลนั้นๆจะไม่ส่งผลกระทบต่อเทคนิควิธีนี้ และการประยุกต์ใช้ Multicriteria Rating กับงานวิจัยนี้จะสามารถแนะนำได้ถูกต้องมากยิ่งขึ้น

ขอบเขตงานวิจัย

- ขอบเขตการพัฒนาระบบ

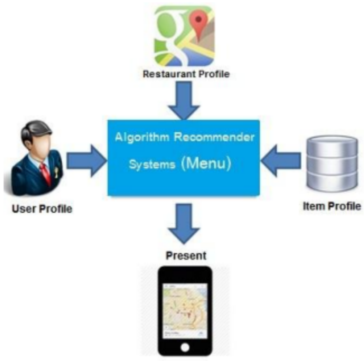
การพัฒนาระบบแนะนำเมนูอาหารอัตโนมัติบนมือถือโดยใช้ข้อมูลเชิงตำแหน่ง ระบบประกอบด้วย 2ส่วนหลัก ได้แก่ Restaurant Mobile Application และ Restaurant Web Service

- วิเคราะห์การกรองข้อมูล

งานวิจัยชิ้นนี้ใช้วิธีการ Content-based Filtering และ Multicriteria Rating ในการแนะนำร้านอาหาร

โครงสร้างระบบส่วนแนะนำเมนูอาหารจะประกอบด้วย 3 ส่วนหลัก ได้แก่

1) User Profile คือ ความต้องการของผู้บริโภคด้านเมนูอาหารและด้านร้านอาหาร
2) Item Profile คือชิ้นข้อมูลเมนูอาหารที่มีในระบบ ประเภทเมนู ส่วนประกอบ รสชาติ ราคา เป็นต้น
3) Restaurant Profile คือ ข้อมูลรายละเอียดร้านอาหาร เช่น ระยะทาง ระดับดาว บรรยากาศ ลานจอดรถ



รูปที่ 3 : โครงสร้างระบบส่วนแนะนำเมนูอาหาร

- กระบวนการคำนวณหาค่าความคล้ายคลึง

1) ผู้ใช้ตั้งค่า User Profile โดยการเลือกประเภทอาหารและกำหนดค่าความชอบในรายละเอียดย่อยของ Item ต่างๆ โดยกำหนดค่าน้ำหนัก 0-5 ยกตัวอย่างดังนี้ ผู้ใช้เลือกประเภทอาหารได้แก่ ข้าวแกง ชอบกินไก่และซีฟู๊ดมากที่สุดให้ค่าน้ำหนักเท่ากัน
2) กำหนดให้ค่าน้ำหนักอยู่ในรูป 0-1 (หารด้วย 5)
3) จะนำค่าน้ำหนักความสำคัญมาคูณแต่ละ Item โดย

- ระบบคำนวณค่าน้ำหนักความสำคัญแต่ละ Item ให้อัตโนมัติหากผู้ใช้ไม่ได้กำหนด
- ผู้ใช้กำหนดและจัดลำดับค่าน้ำหนักความสำคัญให้กับแต่ละ Item ตามความต้องการ
- ผู้ใช้สามารถที่จะกำหนดให้ไม่นำ Item ใดๆ มาใช้คำนวณได้เช่นกัน

ตัวอย่างเช่น Item ส่วนประกอบ (เนื้อ) ค่าน้ำหนักความสำคัญเท่ากับ 1 Item ส่วนประกอบ (ผัก) ค่าน้ำหนักความสำคัญเท่ากับ 0.7 เป็นต้น

4) คำนวณหาค่า Standard Euclidean Distance ระหว่างแต่ละ Item ของ User กับแต่ละ Item ของเมนูอาหาร ตามสูตรดังนี้

X_iคือ Item ที่ i ของ User และY_iคือ Item ที่ i ของ เมนูอาหาร เช่น Item ที่ 1 คือ ไก่

$$d(X,Y) = \frac{\sum_{i=0}^{n} | X_i - Y_i |}{\mathrm{n}}$$

ค่าที่ได้จะอยู่ในช่วง 0-1

5) คำนวณหาค่าความคล้ายคลึง (Similarity Measure) ตามสูตรดังนี้

$$Sv = \frac{1}{1 + \mathrm{d(X,Y)}}$$

ค่าที่ได้จะอยู่ในช่วง 0-1 ค่ายิ่งมากยิ่งมีความคล้ายคลึงมาก

6) แสดงเมนูอาหารที่มีค่าความคล้ายคลึงมากที่สุดเรียงตามลำดับ 10 เมนู ผู้บริโภคสามารถเลือกเมนูอาหารที่ต้องการ เพื่อค้นหาร้านอาหาร โดยสามารถเลือกเมนูที่ต้องการได้มากกว่า1 เมนู เพื่อค้นหาร้านอาหารที่มีเมนูทั้งหมดที่เลือก

ตารางที่ 1 ผลลัพธ์ของการหาค่าความคล้ายคลึงของเมนูอาหาร

ลำดับ	ชื่อเมนู	ค่าความคล้ายคลึง	เลือกเมนูที่ชอบ
1	แกงไก่หน่อไม้	0.80	/
2	ไก่ผัดพริกแกงหน่อไม้	0.75	
3	ต้มไก่หน่อไม้	0.70	/
4	ไก่ผัดพริกหยวก	0.50	/
...	...	...	

7) คำนวณหาความคล้ายคลึงของร้านอาหาร Restaurant Profile กับ User Profile ที่มีเมนูอาหารทั้งหมดที่ผู้ใช้เลือก วิธีการทำเช่นเดียวกันกับ Item Profile

ตารางที่ 2 ผลลัพธ์การหาค่าความคล้ายคลึงของร้านอาหาร

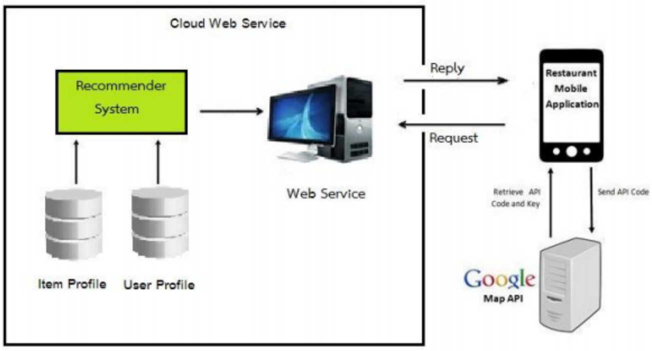
ชื่อร้าน	ค่าความคล้ายคลึง	เมนูแกงไก่หน่อไม้	ต้มไก่หน่อไม้	ไก่ผัดพริกหยวก
ร้านที่1	$S_v = 0.8$	/		
ร้านที่2	$S_v = 0.8$	/	/	/
ร้านที่3	$S_v = 0.75$		/	/
ร้านที่4	$S_v = 0.7$		/	

ร้านที่5 $S_v = 0.5$ / / /

ผลการดำเนินงาน

ออกแบบสถาปัตยกรรมของระบบแนะนำร้านอาหารอัตโนมัติบนมือถือโดยใช้ข้อมูลเชิงตำแหน่งและเมนูอาหาร

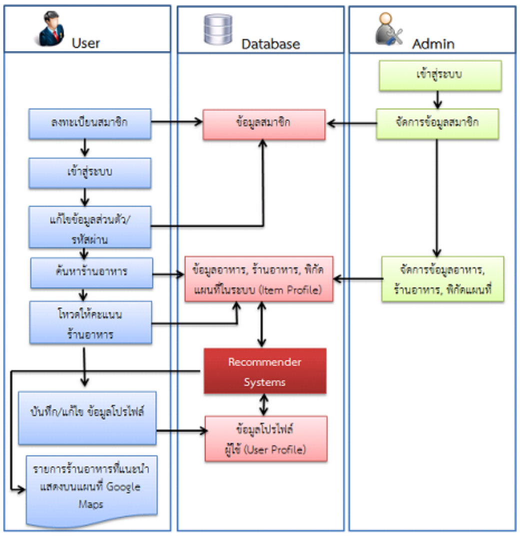
การทำงานโดยรวมของระบบ เริ่มจากการนำข้อมูลเมนูอาหารและส่วนประกอบของแต่ละเมนูอาหาร รวมทั้งข้อมูลของร้านอาหาร เข้าสู่ระบบฐานข้อมูลของ Web Service ที่ให้บริการ ผู้บริโภคสามารถกำหนดความต้องการ (User Profile)ของตนเอง เกี่ยวกับความชอบในเมนู รสชาติ หรือส่วนประกอบของอาหาร ลักษณะของร้านอาหารบน Mobile Application หลังจากนั้นระบบจะทำการวิเคราะห์เพื่อหาค่าความคล้ายคลึงระหว่าง User Profile กับ Item Profile และนำเสนอเมนูและร้านอาหารที่คล้ายคลึงกับความต้องการของผู้บริโภคมากที่สุดไปยัง Mobile Application



รูปที่ 4 :สถาปัตยกรรมของระบบแนะนำร้านอาหารอัตโนมัติบนมือถือโดยใช้ข้อมูลเชิงตำแหน่งและเมนูอาหาร

แนวคิดการทำงานของระบบ

การทำงานของระบบแนะนำร้านอาหารอัตโนมัติบนมือถือโดยใช้ข้อมูลเชิงตำแหน่งและเมนูอาหารนั้น เป็นการนำเข้าข้อมูลของเมนูอาหาร ส่วนประกอบอาหารแต่ละเมนูอาหาร ข้อมูลทั่วไปของร้านอาหาร รวมถึงตำแหน่งละติจูด ลองติจูด ของร้านอาหารต่างๆ ลักษณะการแนะนำของระบบ ยกตัวอย่างเช่น ผู้บริโภคชอบรับประทานอาหารที่มี กุ้ง, ปลาหมึก, ถั่วฝักยาว เป็นส่วนประกอบ และเป็นเมนูที่มีมีรสชาติเผ็ด ระยะทางไม่เกิน 10 กิโลเมตรจากตำแหน่งของผู้บริโภค ณ ขณะนั้น ระบบจะทำการวิเคราะห์หาค่าความคล้ายคลึงของเมนูอาหารที่ผู้บริโภคกำหนดกับเมนูในระบบฐานข้อมูล โดยใช้เทคนิคการกรองแบบอิงเนื้อหา (Content-based Filtering) และใช้การบริการเชิงตำแหน่ง Google Maps API ค้นหาพิกัดของร้านอาหารที่มีเมนูอาหาร ซึ่งมีการบริการและจัดเก็บข้อมูลที่อยู่บนคลาวด์ (Cloud) เพื่อลดการประมวลผลบนเครื่อง Terminal ตอบสนองความต้องการของผู้ใช้บริการได้ทุกที่ทุกเวลา



รูปที่ 5 : แนวคิดการทำงานของระบบแนะนำร้านอาหาร

ออกแบบส่วนติดต่อผู้ใช้

การออกแบบในส่วนติดต่อกับผู้ใช้ ผู้ใช้สามารถกำหนดข้อมูลความต้องการของตนเอง เพื่อให้ระบบแนะนำร้านอาหารและเมนูอาหารที่ตรงกับความต้องการได้ อีกทั้งยังสามารถค้นหาร้านอาหารตามประเภทของร้านอาหาร หรือค้นหาตามสถานที่ตั้งของร้านได้ และยังสามารถเลือกดูในมุมมองของแผนที่ได้

รูปที่ 6 : ส่วนติดต่อกับผู้ใช้

สรุปและอภิปรายผลการวิจัย

งานวิจัยนี้นำเสนอการออกแบบสถาปัตยกรรมและส่วนติดต่อกับผู้ใช้ของระบบแนะนำร้านอาหารอัตโนมัติบนมือถือโดยใช้ข้อมูลเชิงตำแหน่งและเมนูอาหารโดยใช้เทคนิคการกรองแบบอิงเนื้อหา (Content-based Filtering) เป็นแนวทางในการพัฒนาระบบสำหรับใช้งานจริงสำหรับผู้บริโภคเพื่อค้นหาร้านอาหารและเมนูอาหารได้สะดวกรวดเร็วและตรงตามความต้องการมากยิ่งขึ้น อีกทั้งระบบมีการบริการและจัดเก็บข้อมูลอยู่บนคลาวด์ (Cloud) สามารถลดการ

ประมวลผลบนเครื่อง Terminal ตอบสนองความต้องการของผู้ใช้บริการได้ทุกที่ทุกเวลา เหมาะสมสำหรับเจ้าของกิจการร้านอาหารที่ต้องการประชาสัมพันธ์ร้านให้เป็นที่รู้จักมากยิ่งขึ้น

ข้อเสนอแนะ

ระบบนี้หากมีวิธีการแยกแยะส่วนประกอบของเมนูอาหารได้ละเอียดและชัดเจนมากยิ่งขึ้น อีกทั้งหากสามารถกำหนดน้ำหนักความสำคัญแต่ละ Item ได้ละเอียดมากขึ้น จะสามารถแนะนำเมนูอาหารได้ตรงกับความต้องการของผู้ใช้มากขึ้น และวิธีการดังกล่าวนี้สามารถนำไปประยุกต์ใช้กับข้อมูลอื่นๆ หรือผลิตภัณฑ์อื่นๆ ให้ตรงตามความต้องการมากยิ่งขึ้นในอนาคต

เอกสารอ้างอิง

Yum Thailand. (2555). **KFC Thailand (ออนไลน์).** สืบค้นจาก : https://play.google.com/store/apps/details?id=air.yum.th.kfc [4 กุมพาพันธ์ 2557].

Wongnai (ออนไลน์). สืบค้นจาก : http://www.wongnai.com/iphone-app [4กุมพาพันธ์ 2557].

เปรมฤดี ผลชอบ และ สุพจน์ นิตย์สุวัฒน์. (2553). **"ระบบให้คำแนะนำในการเลือกซื้อคอมพิวเตอร์โน๊ตบุ๊คด้วยเทคนิค",** ภาควิชาเทคโนโลยีสารสนเทศ คณะเทคโนโลยีสารสนเทศ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ.

G. Admavicius and Y. Kown. (2007). **New Recommendation Techniques for Multicriteria Rating Systems,** IEEE Intelligence Systems vol.22 No. 3 : pp.48-55.

ปฐมพงษ์ ฉับพลัน และฐิมาพร เพชรแก้ว. (2553).**"การประยุกต์ใช้ Google Maps API ในการเฝ้าระวังปัญหายาเสพติด",** สำนักวิชาสารสนเทศศาสตร์ มหาวิทยาลัยวลัยลักษณ์.

Cymiz. (2555).**Cloud Server คืออะไรมีหลักการทำงานอย่างไร (ออนไลน์).** สืบค้นจาก : http://www.jongjarern.com/insurances/index.php?topic=1712.0 [20 มีนาคม 2557]

Keerthi S. Shetty and Sanjay Singh. (2009). **Cloud Based Application Development for Accessing Restaurant Information on Mobile Device using LBS,** Institute of Technology ManipalUniversity : Manipal-576104 India.

Giuseppe Di Fabbrizio and Narendra Gupta. (2010). **Have2eat: a Restaurant Finder with Review Summarization for Mobile Phones,** Coling 2010 : Demonstation Volume, page17-20

มหาวิทยาลัยพายัพ ร่วมกับเครือข่ายบริหารการวิจัยภาคเหนือตอนบน
สำนักงานคณะกรรมการการอุดมศึกษา(สกอ.)

ขอมอบเกียรติบัตรฉบับนี้ เพื่อแสดงว่าผลงานวิจัย เรื่อง

การออกแบบต้นแบบระบบแนะนำร้านอาหารอัตโนมัติบนมือถือโดยใช้ข้อมูลเชิงตำแหน่งและเมนูอาหาร

โดย ปาลิตา แสงศิริ ทวีศักดิ์ เรืองพีระกุล และนิคม สุวรรณวร

ได้รับพิจารณาเข้าร่วมนำเสนอใน Payap University Research Symposium 2016 ภาคบรรยาย
"การประชุมวิชาการเสนอผลงานวิจัยระดับชาติ มหาวิทยาลัยพายัพ พ.ศ. 2559"

ให้ไว้ ณ วันที่ 12 กุมภาพันธ์ พ.ศ. 2559

(ผู้ช่วยศาสตราจารย์ ดร. สมพันธ์ วงษ์ดี)
อธิการบดีมหาวิทยาลัยพายัพ

(ศาสตราจารย์ ดร. สัญชัย จตุรสิทธา)
ประธานเครือข่ายบริหารการวิจัยภาคเหนือตอนบน

ประวัติผู้เขียน

ชื่อ สกุล นางสาวปาลิตา แสงศิริ

รหัสประจำตัวนักศึกษา 5610121047

วุฒิการศึกษา

วุฒิ	ชื่อสถาบัน	ปีที่สำเร็จการศึกษา
วิศวกรรมศาสตรบัณฑิต (วิศวกรรมคอมพิวเตอร์)	มหาวิทยาลัยวลัยลักษณ์	2551

ทุนการศึกษา

ทุนอุดหนุนการวิจัยเพื่อวิทยานิพนธ์ ปีการศึกษา 2557

ตำแหน่งและสถานที่ทำงาน

พ.ศ. 2551-2552

โปรแกรมเมอร์ บริษัท Up Master solution กรุงเทพมหานคร

พ.ศ. 2553-2554

เจ้าหน้าที่บ่มเพาะวิสาหกิจ มหาวิทยาลัยวลัยลักษณ์

พ.ศ. 2554-2556

ผู้ช่วยสอน สำนักวิชาสารสนเทศศาสตร์ มหาวิทยาลัยวลัยลักษณ์

พ.ศ. 2556-2559

โปรแกรมเมอร์ บริษัท วัฒนาโฮมเซนเตอร์ สงขลา

พ.ศ. 2559-ปัจจุบัน

ซีเนียร์ โปรแกรมเมอร์ บริษัท กรีนรีเวอร์ พาเนล (ประเทศไทย) จำกัด สงขลา

การตีพิมพ์เผยแพร่ผลงาน

ปาลิตา แสงศิริ ทวีศักดิ์ เรืองพีระกุล และนิคม สุวรรณวร “*การออกแบบต้นแบบระบบแนะนำร้านอาหารอัตโนมัติบนมือถือโดยใช้ข้อมูลเชิงตำแหน่งและเมนูอาหาร*” การประชุมวิชาการเสนอผลงานวิจัยระดับชาติ มหาวิทยาลัยพายัพ พ.ศ.2559 วันที่ 12 กุมภาพันธ์ 2559 มหาวิทยาลัยพายัพ. จังหวัดเชียงใหม่