ระบบค้นหาสินค้าลดราคาด้วยวิธีการเข้าถึงข้อมูลหน้าเว็บไซต์
Find items for sale with web scrapping

ประหยัด เลวัน และ ธงชัย เที่ยงธรรม
สาขาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยรามคำแหง
Email: prayatl@gmail.com, mandarinkb@gmail.com

บทคัดย่อ

แอปพลิเคชันค้นหาสินค้าลดราคา จัดทำเพื่อเป็นเครื่องมือช่วยค้นหาสินค้าลดราคาจากเว็บไซต์ต่างๆ ซึ่งใช้วิธีการเข้าถึงข้อมูลหน้าเว็บไซต์ (web scrapping) โดยได้แบ่งการทำงานออกเป็น 2 ส่วน คือ ส่วนที่หนึ่ง ระบบเข้าถึงข้อมูลหน้าเว็บไซต์ต่าง ซึ่งระบบนี้จะทำการดึงข้อมูลหน้าเว็บไซต์ต่างๆที่ต้องการ จากนั้นทำการจัดเก็บข้อมูลลงใน NoSQL Database ที่มีชื่อว่า Elasticsearch ส่วนที่สอง เป็นส่วนแสดงผล โดยแสดงผ่านแอปพลิเคชันค้นหาสินค้าลดราคา บนระบบปฏิบัติการแอนดรอยด์ การพัฒนาระบบค้นหาสินค้าลดราคาด้วยวิธีการเข้าถึงข้อมูลเว็บไซต์ มีเครื่องมือที่ช่วยพัฒนาได้แก่ โปรแกรม Virtual Studio Code โปรแกรม Android Studio โปรแกรม Postman โปรแกรมภาษาจาวา (Java) โปรแกรม Redis โปรแกรม Elasticsearch และโปรแกรม Docker ผลการทดสอบการดำเนินงานของระบบพบว่า ค่าเฉลี่ยผลความครบถ้วนของข้อมูลที่ทำการสแครบปิ้งมาจากเว็บไซต์ร้านค้าออนไลน์ขนาดใหญ่ ร้อยละ 99.95 และเมื่อเพิ่มจำนวนบอทจาก 3 ไปเป็น 13 บอทเวลาประมวลผลลดลงถึงประมาณ 10 เท่า (170.24/17.21) ในส่วนแอปพลิเคชันค้นหาสินค้าลดราคา ผลการทดสอบเมนูค้นหาพบว่าการค้นพบสินค้าตามที่ต้องการคิดเป็นค่าเฉลี่ย 92.83 % ผลสรุปด้านประสิทธิภาพของแอปพลิเคชันประกอบด้วยเนื้อหา 10 หัวข้อ โดยได้ให้ผู้ใช้และผู้พัฒนาระบบทดลองใช้งานผลปรากฏว่า ผู้ใช้มีความพึงพอใจอยู่ในระดับ “มาก” ($\bar{X}$ = 4.20) และผู้พัฒนาระบบมีความพึงพอใจในมุมมองการออกแบบฟังก์ชันการใช้งานอยู่ในระดับ “มาก” ($\bar{X}$ = 4.20) และผลการทดสอบเซิร์ฟเวอร์สามารถรองรับการใช้งาน 270 ผู้ใช้งานต่อวินาที

คำสำคัญ: ระบบปฏิบัติการแอนดรอยด์ แอปพลิเคชัน web scrapping Elasticsearch

บทนำ

ปัจจุบันการซื้อสินค้าออนไลน์เริ่มมากขึ้น เพราะความก้าวหน้าทางเทคโนโลยีสมาร์ทโฟน และในช่วงสถานะการณ์โควิด การที่ออกไปเลือกซื้อสินค้าตามที่ต่างๆทำให้เกิดการสุ่มเสี่ยงต่อการติดโรค ผู้คนจึงหันมาซื้อสินค้าออนไลน์มากขึ้น ซึ่งการซื้อสินค้าแต่ละครั้ง ผู้สั่งซื้อต้องค้นหาเลือกซื้อสินค้าตามที่ต้องการ อาจจะดูจากราคา โปรโมชั่น คุณภาพสินค้าหรือความน่าเชื่อถือของร้านจากเว็บไซต์ผู้ให้บริการ บางครั้งอาจจะต้องค้นหาจากเว็บไซต์หลายเว็บไซต์เพื่อเปรียบเทียบราคา ให้ตรงกับความต้องการมากที่สุด

ด้วยเหตุนี้ผู้จัดทำได้พัฒนาแอปพลิเคชันสินค้าลดราคา โดยระบบจะทำการดึงข้อมูลจากหน้าเว็บไซต์และนำสินค้าลดราคามาจัดเก็บไว้ ทำให้ผู้ใช้งานเลือกดูสินค้าที่ลดราคาจากเว็บไซต์ต่างๆที่ทางระบบได้กำหนดไว้ เครื่องมือและเทคโนโลยีที่เกี่ยวข้องในการพัฒนาระบบมีดังต่อไปนี้

1 การทำ web scrapping [1] วิธีการในการดึงข้อมูลจากหน้าเว็บเพจหรือเว็บไซต์ โดยใช้ภาษาโปรแกรมมิ่งเป็นเครื่องมือ ปกติการดึงข้อมูลนั้นมี 2 วิธีหลักๆ คือ ดึงจาก API และทำ web scrapping เมื่อได้ข้อมูลจากเว็บไซต์ที่ต้องการ

ทำการจัดเก็บข้อมูลลงใน NoSQL ของ Elasticsearch

2 Elasticsearch [2] ที่เก็บข้อมูลที่พัฒนาต่อยอดมาจาก Apache Lucene มีจุดเด่นในเรื่องความสามารถของการค้นหา และสรุปข้อมูลขนาดใหญ่ได้อย่างรวดเร็ว โดยวิธีการคิวรี่ของ Elasticsearch มีหลายแบบ แบ่งเป็น 2 แบบหลักๆ คือ

1 Full text queries

2 Term-level queries แบ่งเป็น 11 คิวรี่ ได้แก่ exists query , fuzzy query , ids query , prefix query , range query , regexp query , term query , terms query , terms_set query , type query , wildcard query ในระบบได้เลือกการคิวรี่แบบ regexp query (regular expression) มาใช้งาน

3 Web API เป็นตัวกลางเชื่อมต่อเพื่อให้แอปพลิเคชันสามารถเชื่อมต่อกับเซิร์ฟเวอร์ โดยข้อมูลอยู่ในรูปแบบของ JSON รับส่งข้อมูลผ่านโปรโตคอล HTTP ทางระบบได้ใช้ HTTPS [3] เป็นรุ่นปลอดภัยของโปรโตคอล HTTP ที่ใช้ SSL/TLS โปรโตคอล สำหรับการเข้ารหัสและพิสูจน์ตัวตน HTTPS ถูกระบุโดย RFC 2818 และใช้พอร์ต 443 เป็นค่าเริ่มต้น แทนพอร์ต 80 ของ HTTP

4 Docker เป็นแพลตฟอร์มซอฟต์แวร์ที่ช่วยสร้าง ทดสอบ และติดตั้งแอปพลิเคชันได้อย่างรวดเร็ว Docker จะบรรจุซอฟต์แวร์ลงไปในหน่วยที่เป็นมาตรฐานเรียกว่า “คอนเทนเนอร์” ซึ่งจะมีทุกสิ่งที่ซอฟต์แวร์ต้องใช้ในการเรียกใช้งาน รวมทั้ง ไลบรารี่ เครื่องมือสำหรับระบบ โค้ดและรันไทม์ เมื่อใช้ Docker จะสามารถติดตั้งใช้จริงและปรับขนาดแอปพลิเคชันให้เหมาะสมกับสภาพแวดล้อม [4]

วิธีการดำเนินงาน

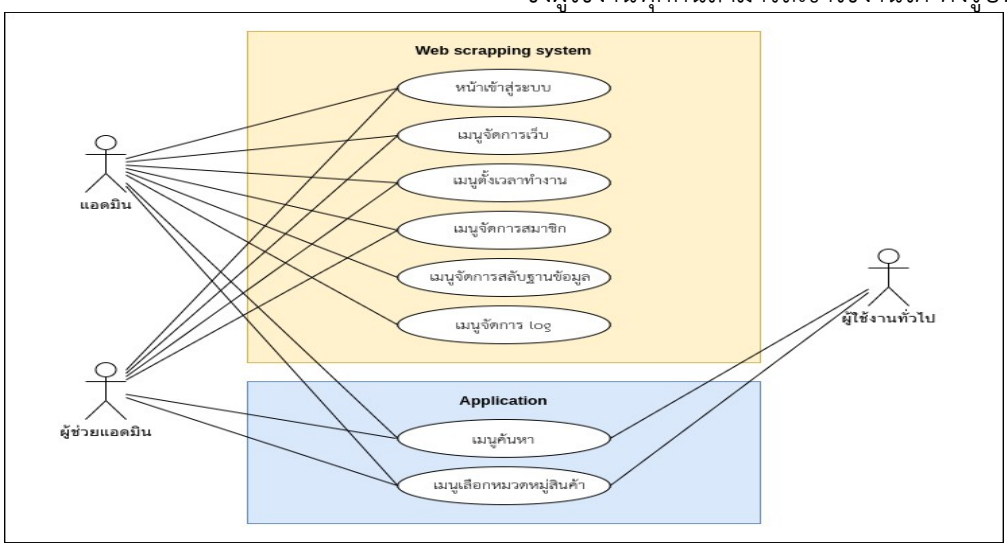
ในระบบได้แบ่งการทำงานเป็น 2 ส่วน คือ

1 หน้าเว็บไซต์ที่ใช้ควบคุมบอท

2 แอปพลิเคชันค้นหาสินค้าลดราคา

ระบบงานหน้าเว็บไซต์ที่ใช้ควบคุมบอท ในขั้นตอนแรกจะทำการเก็บข้อมูลไว้ในฐานข้อมูล ระบบจะมีคุณสมบัติ แสดง เพิ่ม ลบ แก้ไข ข้อมูลในระบบ เมนูหน้าเว็บไซต์ที่ใช้ควบคุมบอทมี 6 เมนู ได้แก่ หน้าเข้าสู่ระบบ เมนูจัดการเว็บ เมนูตั้งเวลาทำงาน เมนูจัดการสมาชิก เมนูจัดการสลับฐานข้อมูล เมนูจัดการ log ซึ่งผู้ใช้งานที่สามารถใช้เมนูดังกล่าวได้ มี 2 ผู้ใช้งาน คือ แอดมิน และผู้ช่วยแอดมิน ดังรูปที่ 1

แอปพลิเคชันค้นหาสินค้าลดราคา มีรายละเอียด 2 เมนู คือ เมนูค้นหา และเมนูเลือกหมวดหมู่สินค้า ซึ่งผู้ใช้งานทุกคนสามารถเข้าใช้งานได้ ดังรูปที่ 1



รูปที่ 1 Use Case Diagram ระบบค้นหาสินค้าลดราคา

พัฒนาบอท(Bot) เพื่อดึงข้อมูลในเว็บไซต์

บอท ในที่นี้หมายถึง โปรแกรมที่ทำงานเพื่อดึงข้อมูลหน้าเว็บไซต์ โดยจะทำตามเงื่อนไขที่ได้กำหนดไว้ภายในโปรแกรม

เพื่อให้การดึงหน้าเว็บไซต์ทำงานได้เร็วมากขึ้น ได้แบ่งการทำงานออกเป็นส่วนย่อยและพัฒนาบอทเป็น 3 บอท ได้แก่

1 start bot

2 fetch url bot

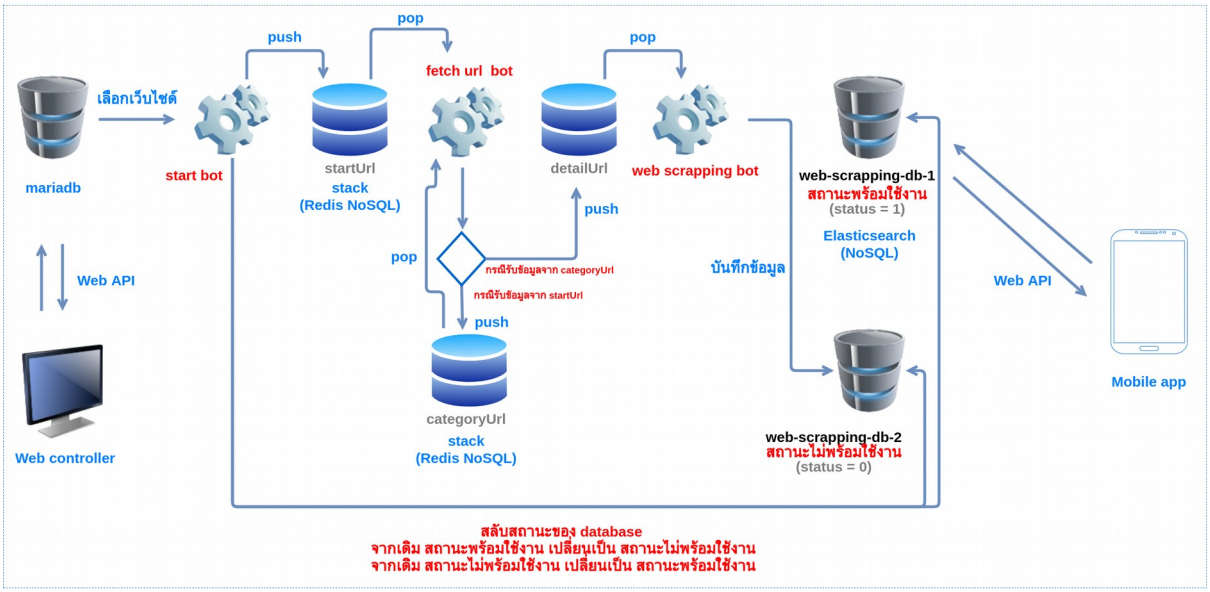
3 web scrapping bot

start bot ทำงานตามเวลาที่ได้ตั้งค่าไว้ในระบบ จากนั้น start bot ทำการเลือกเว็บไซต์ที่ต้องการดึงข้อมูลตามที่ได้ตั้งค่าไว้ในระบบ และทำการจัดเก็บข้อมูลลงใน Redis [5] ที่มีชื่อว่า startUrl เปรียบเสมือนเป็น stack ซึ่งเป็นลักษณะของ temporary database เพื่อส่งต่อการทำงานให้ fetch url bot ทำงานต่อ

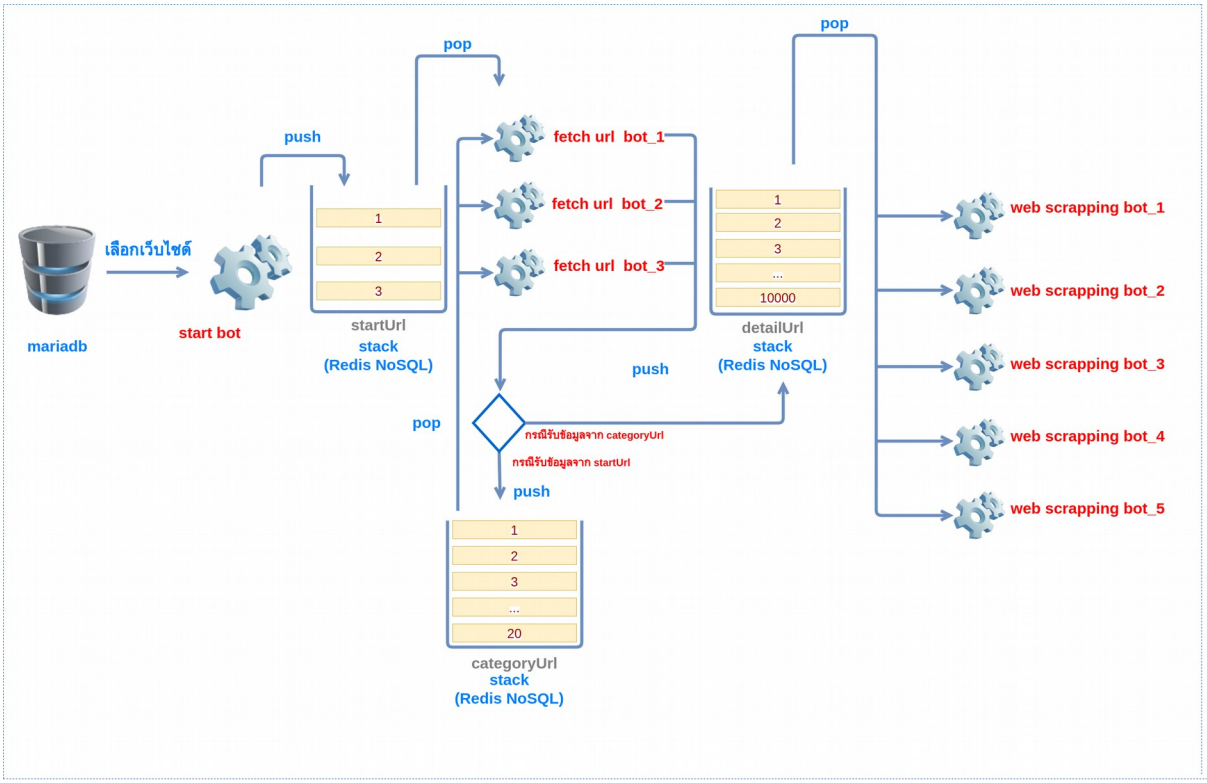
fetch url bot นำข้อมูลที่ได้จาก startUrl และทำงานตามเงื่อนไข จากนั้นลำดับสุดท้ายจะทำการจัดเก็บข้อมูลที่ได้ลงใน Redis ที่ทีชื่อว่า detailUrl เพื่อให้ web scrapping bot ทำงานต่อ

web scrapping bot นำข้อมูลที่ได้จาก detailUrl เรียบร้อยแล้วทำการดึงข้อมูลในเว็บไซต์ จากนั้นนำข้อมูลที่ได้จัดเก็บลง Elasticsearch database ดังรูปที่ 2

จะเห็นได้ว่าการทำงานดังกล่าวเป็นลักษณะแบบส่งต่อการทำงานเป็นช่วง โดยไม่จำเป็นต้องให้บอทตัวใดตัวหนึ่งทำงานจนเสร็จ ถ้ามีข้อมูลใน Redis database บอทตัวถัดไปสามารถทำงานต่อได้เลย เป็นลักษณะของ pipeline และเมื่อเกิดปัญหาคอขวดคือข้อมูลใน Redis มีจำนวนมากจนบอททำงานไม่ทัน ก็สามารถเพิ่มจำนวนบอทในแต่ละช่วง เพื่อแก้ปัญหาดังกล่าวได้ ดังรูปที่ 3



รูปที่ 2 ขั้นตอนการทำงานของบอทเพื่อดึงข้อมูลหน้าเว็บไซต์



รูปที่ 3 แสดงการเพิ่มจำนวนบอท

ผลการศึกษาและอภิปรายผล

ผลการพัฒนา Bot เพื่อดึงข้อมูลหน้าเว็บไซต์

1 ทดสอบความครบถ้วนของข้อมูล ได้ทำการทดสอบดึงข้อมูลหน้าเว็บไซต์ เทสโก้ โลตัส ผลปรากฏว่าความครบถ้วนของข้อมูลอยู่ในระดับ 99.95% ดังตารางที่ 1

2 ทดสอบเพิ่มจำนวนบอท เพื่อดูระยะเวลาการดึงข้อมูลทั้ง 3 เว็บไซต์ คือเทสโก้ โลตัส แม็คโคร บิ๊กซี ผลปรากฏว่าเมื่อเพิ่มจำนวนบอท เป็น 4 บอท เวลาการทำงานลดเกินครึ่ง จากนั้นเพิ่มจำนวนบอทไปตามลำดับ เวลาการทำงานก็ลดลงเรื่อยๆ ดังตารางที่ 2 และรูปที่ 4

ตารางที่ 1 ผลการทดสอบความครบถ้วนของข้อมูล

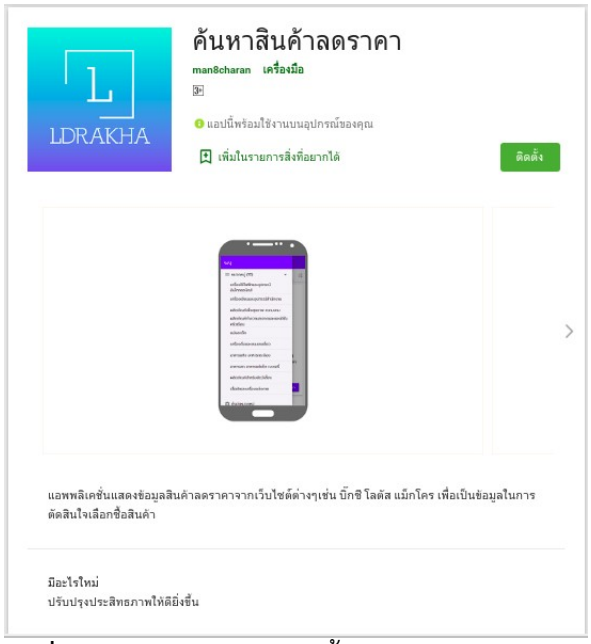
ลำดับ	วันที่	ข้อมูลสินค้าทั้งหมด	ข้อมูลสินค้าที่ได้	เปอร์เซ็นต์
1	10/05/64	4235	4230	99.88
2	10/05/64	4235	4235	100
3	10/05/64	4235	4235	100
4	10/05/64	4235	4235	100
5	10/05/64	4235	4223	99.71
6	10/05/64	4235	4235	100
7	10/05/64	4235	4234	99.98
8	10/05/64	4235	4235	100
9	10/05/64	4235	4232	99.93
10	10/05/64	4235	4235	100
			ค่าเฉลี่ย	**99.95**

ตารางที่ 2 ผลการทดสอบเพิ่มจำนวนบอท

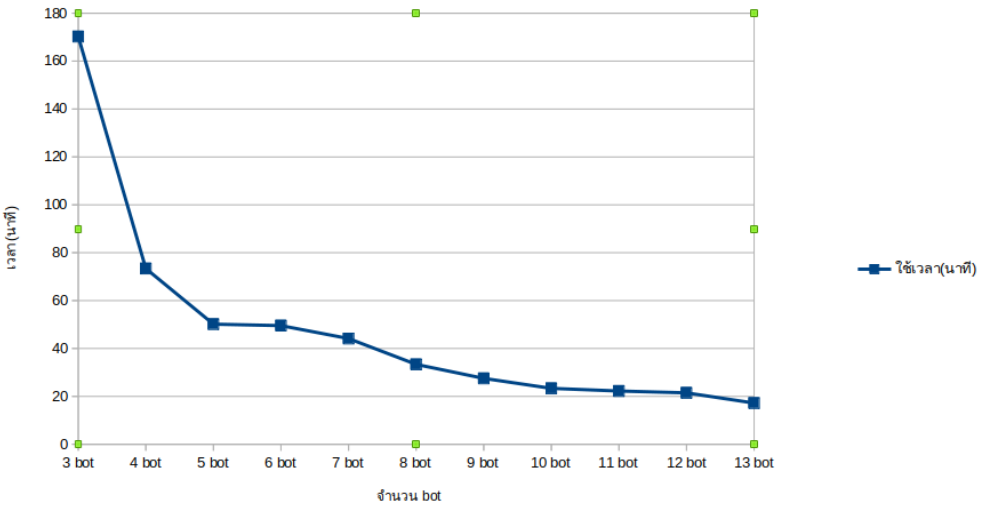
ลำดับ	วันที่	จำนวนบอท	ใช้เวลา (นาที)
1	11/03/64	3 Bot	170.24
2	11/03/64	4 Bot	73.38
3	11/03/64	5 Bot	50.16
4	11/03/64	6 Bot	49.55
5	11/03/64	7 Bot	44.14
6	11/03/64	8 Bot	33.36
7	11/03/64	9 Bot	27.51
8	11/03/64	10 Bot	23.34
9	11/03/64	11 Bot	22.24
10	11/03/64	12 Bot	21.47
11	11/03/64	13 Bot	17.21

ข้อมูลที่ได้แสดงผ่านแอปพลิเคชันค้นหาสินค้าลดราคา

การติดตั้งแอปพลิเคชันค้นหาสินค้าลดราคาสามารถดาวน์โหลดได้จาก Play Store โดยทำการค้นคำว่า “ค้นหาสินค้าลดราคา” จากนั้นกดค้นหาจะพบแอปพลิเคชัน ค้นหาสินค้าลดราคาจากนั้นกดติดตั้ง ดังรูปที่ 5



รูปที่ 5 หน้าจอแสดงการติดตั้งของแอปพลิเคชัน



รูปที่ 4 กราฟแสดงการใช้เวลาทำงานของบอท

แอปพลิเคชันสินค้าลดราคา ในเมนูค้นหาได้ใช้เครื่องมือของ Elasticsearch ช่วยค้นหา โดยเลือกใช้คิวรี่แบบ regexp query (Regular Expression) [6] ซึ่งเป็นการกำหนดรูปแบบหรือกลุ่มคำ เพื่อเอาไว้ใช้ค้นหาข้อความต่างๆตามที่ต้องการ

ทดสอบ 100 คำค้น แยกเป็น 10 หมวดหมู่ หมวดหมู่ละ 10 คำค้น ผลลัพธ์ที่ได้

1. ไม่พบข้อมูล 17 คำค้น
2. พบข้อมูล 83 คำค้น คิดเป็นค่าเฉลี่ย 92.83 % ของการค้นพบสินค้าตามที่ต้องการ

ตารางที่ 3 ผลการทดสอบคำค้น (ข้อมูลวันที่ 23/07/64)

ลำดับ	หมวดหมู่สินค้า	จำนวนเปอร์เซ็นต์ค้นพบสินค้าตามที่ต้องการ
1	เครื่องใช้ไฟฟ้าและอุปกรณ์อิเล็กทรอนิกส์	100.00
2	เครื่องเขียนและอุปกรณ์สำนักงาน	96.20
3	ผลิตภัณฑ์เพื่อสุขภาพความงาม	100.00
4	ผลิตภัณฑ์ทำความสะอาดและของใช้ในครัวเรือน	95.41
5	แม่และเด็ก	100.00
6	เครื่องดื่มและขนมขบเคี้ยว	99.64
7	อาหารแห้ง อาหารกระป๋อง	99.07
8	อาหารสด อาหารแช่แข็ง เบเกอรี่	68.03
9	ผลิตภัณฑ์สำหรับสัตว์เลี้ยง	100.00
10	เสื้อผ้าและเครื่องแต่งกาย	70.00

ผลการทดสอบเซิร์ฟเวอร์เพื่อดูการรองรับการใช้งานจากผู้ใช้งาน

ปัจจุบันใช้งาน 1 เซิร์ฟเวอร์ (2 cpu ram 4 gb ssd 80 gb) รองรับการใช้งานได้ 270 ผู้ใช้งานต่อวินาที

ตารางที่ 4 ผลการทดสอบ load testing

ลำดับ	จำนวน user ต่อ 1 วินาที	ใช้ cpu server (%)	error (%)	error message
1	20	20.03	0.00	
2	40	28.57	0.00	
3	60	39.18	0.00	
4	80	40.82	0.00	
5	100	47.24	0.00	
6	120	50.51	0.00	
7	140	53.81	0.00	
8	160	57.22	0.00	
9	180	64.4	0.00	
10	200	72.28	0.00	
11	270	77.16	0.00	
12	271	79.8	7.01	Connection timed out

สรุปผล

ผลการทดสอบจากผู้ใช้งาน 10 คน โดยแยกเป็นผู้ใช้งานทั่วไป 5 คน และผู้พัฒนาระบบ 5 คน ผลปรากฏว่า ผู้ใช้มีความพึงพอใจอยู่ในระดับ “มาก” ($\overline{X}$ = 4.20) ส่วนผู้พัฒนาระบบมีความพึงพอใจในมุมมองการออกแบบและฟังก์ชันการใช้งานอยู่ในระดับ “มาก” ($\overline{X}$ = 4.20)

ตารางที่ 5 ผลจากการประเมินจากผู้ใช้งานแอปพลิเคชัน

แบบประเมิน	ค่าเฉลี่ย	ค่าเบี่ยงเบน
ผู้ใช้งานทั่วไป	4.20	0.59
ผู้พัฒนาระบบ	4.20	0.56

กิตติกรรมประกาศ

การจัดทำโครงงานฉบับนี้สำเร็จได้ด้วยดีต้องกราบขอบพระคุณอาจารย์ที่ปรึกษา อาจารย์ประหยัด เลวัน ที่ได้ให้คำแนะนำ ตรวจสอบโครงงานเพื่อปรับปรุงแก้ไขข้อบกพร่องต่างๆ และขอขอบพระคุณคณาจารย์ทุกท่าน ที่ได้ประสิทธิ์ประสาทวิชาความรู้ เพื่อนำความรู้ที่ได้มาจัดทำโครงงานจนสำเร็จลุล่วงไปด้วยดี ผู้จัดทำจึงขอขอบพระคุณมา ณ โอกาสนี้

เอกสารอ้างอิง

1. STACKPYTHON. (4 ธันวาคม 2563). [ออนไลน์] **Web Scraping**. สืบค้นจาก https://stackpython.co/tutorial/web-scraping-python-beautifulsoup-requests (วันที่สืบค้น 26 กรกฎาคม 2564)

2. AWS. (2564). [ออนไลน์] **ElasticSearch.** สืบค้นจาก https://aws.amazon.com/th/elasticsearch-service/the-elk-stack/what-is-elasticsearch/ (วันที่สืบค้น 26 กรกฎาคม 2564)

3. SSL.com. (13 กรกฎาคม 2563). [ออนไลน์] **HTTPS คืออะไร.** สืบค้นจาก https://www.ssl.com/th/%E0%B8%84%E0%B8%B3%E0%B8%96%E0%B8%B2%E0%B8%A1%E0%B8%97%E0%B8%B5%E0%B9%88%E0%B8%9E%E0%B8%9A%E0%B8%9A%E0%B9%88%E0%B8%AD%E0%B8%A2/https-%E0%B8%84%E0%B8%B7%E0%B8%AD%E0%B8%AD%E0%B8%B0%E0%B9%84%E0%B8%A3/ (วันที่สืบค้น 26 กรกฎาคม 2564)

4. AWS. (2564). [ออนไลน์] Docker คืออะไร. สืบค้นจาก https://aws.amazon.com/th/docker/ (วันที่สืบค้น 26 กรกฎาคม 2564

5. Softmelt. (2554). [ออนไลน์] **Redis คืออะไร ?.** สืบค้นจาก https://www.softmelt.com/article.php?id=564 (วันที่สืบค้น 26 กรกฎาคม 2564)

6. Thirawat T. (26 กรกฎาคม 2560). [ออนไลน์] **Regular Expressions คืออะไร ?**. สืบค้นจาก https://medium.com/@_trw/regular-expressions-%E0%B8%84%E0%B8%B7%E0%B8%AD%E0%B8%AD%E0%B8%B0%E0%B9%84%E0%B8%A3-2fab4a91ea34 (วันที่สืบค้น 26 กรกฎาคม 2564)