ระบบร้านค้าอัตโนมัติ

AUTONOMOUS SHOP SYSTEM

กฤตนัย กัปตพล

กษิดิศ รัตนวิจิตร

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตร์บัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2562

ระบบร้านค้าอัตโนมัติ

นายกฤตนัย กัปตพล 59010027

นายกษิดิศ รัตนวิจิตร 59010056

ผศ.ดร.อำนาจ ขาวเน อาจารย์ที่ปรึกษา

ปีการศึกษา 2562

บทคัดย่อ

โครงงานนี้มีจุดประสงค์เพื่อสร้างอุปกรณ์ตรวจจับสินค้าและทำรายการสินค้าอัตโนมัติเพื่อความสะดวกสบายในร้านค้าสะดวกซื้อ ด้วย Yolov3-Tiny Object Detection โดยผ่าน Darknet Neural Networks เพื่อให้ได้โมเดลที่แม่นยำที่สุดในการตรวจจับสินค้าที่มีลักษณะเฉพาะ โดยการ Train ด้วย Dataset แบบเฉพาะ ข้อดีของงานโครงงานนี้คือการทำงานอย่างรวดเร็วแม่นยำในการตรวจจับสินค้าและสามารถทำรายการสินค้าออกมาชำระเงินได้ตามขั้นตอนในการชำระสินค้าโดยพนักงาน เพิ่มความสะดวกสบายให้กับการใช้บริการร้านสะดวกซื้อและลดปริมาณการใช้แรงงานโดยสิ้นเปลืองและความผิดพลาดที่เกิดขึ้นจากมนุษย์

AUTONOMOUSE SHOP SYSTEM

Krittanai	Kaptapon	59010027
Kasidith	Rattanavijit	59010056
Mr.Amnach	Khawne	Advisor

ABSTRACT

The purpose of this project is to create autonomous device which can detect and classify with Yolov3-Tiny Object Detection and Darknet Neural Networks. In order to decrease operating time and increase precision of the model. The advantage of the device is for convenience of the customer in the store. The device should be operating like cashier to reduce the extravagant use of human resources and human's error. This device seems to be the best solution.

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงไม่ได้หากปราศจากความช่วยเหลือของ อาจารย์ที่ปรึกษา ผศ. ดรอำนาจ ขาวเน ผู้ให้คำปรึกษา คำแนะนำ สนับสนุน และให้ความช่วยเหลือมาตลอดการทำปริญญานิพนธ์ฉบับนี้ ซึ่งทำให้การปฏิบัติการราบรื่นและจนสำเร็จลุล่วงด้วยดี

ขอขอบคุณอาจารย์และบุคลากรต่าง ๆ ในสาขาวิชาวิศวกรรมคอมพิวเตอร์ที่สั่งสอน และให้คำแนะนำต่าง ๆ มาโดยตลอด

ขอขอบคุณรุ่นพี่และเพื่อนหลายๆคนในสาขาวิชาวิศวกรรมศาสตร์ที่ได้แบ่งปันความรู้ คำแนะนำและคำปรึกษาในทุก ๆ ด้าน

ในท้ายสุดนี้ ขอขอบคุณ บิดา มารดา และครอบครัว ที่ได้เลี้ยงดู สั่งสอน และสนับสนุนโอกาสในการศึกษาเสมอมา

กฤตนัย กัปตพล

กษิดิศ รัตนวิจิตร

สารบัญ

หน้า

บทคัดย่อภาษาไทย I

บทคัดย่อภาษาอังกฤษ II

กิตติกรรมประกาศ III

สารบัญ IV

สารบัญ(ต่อ) V

สารบัญ(ต่อ) VI

สารบัญ(ต่อ) VII

สารบัญรูป VIII

สารบัญรูป(ต่อ) IX

สารบัญรูป(ต่อ) X

สารบัญรูป(ต่อ) XI

สารบัญรูป(ต่อ) XII

สารบัญตาราง XIII

บทที่ 1 บทนำ 1

1.1 ความเป็นมาของปัญหา 1

1.2 วัตถุประสงค์ของโครงงาน 1

1.3 ขอบเขตของโครงงาน 2

1.4 เป้าหมายของโครงงาน 2

1.5 ประโยชน์ที่คาดว่าจะได้รับ 2

บทที่ 2 เอกสารและโครงงานที่เกี่ยวข้อง 3

2.1 ทฤษฎีเกี่ยวกับการตรวจจับวัตถุ 3

สารบัญ(ต่อ)

หน้า

2.1.1 หลักการเบื้องต้น 3

2.1.1.1 Neural Network....3

2.1.1.2 Deep Learning....5

2.1.1.3 หลักการ IoU 7

2.1.1.4 Anchor Box....9

2.1.2 เทคนิคการตรวจจับวัตถุ 11

2.1.2.1 Scale-Invariant Feature Transform (SIFT) 11

2.1.2.2 Speed Up Robust Feature (SURF)....12

2.1.2.3 Feature from Accelerated Segment (Fast) 13

2.1.2.4 Fast R-CNN....14

2.1.2.5 YOLO (You Look Only Once)....15

2.2 ทฤษฎีเกี่ยวกับ Software....18

2.1.1 Python....18

2.2.2 PyQT....20

2.2.3 phpMyAdmin....22

2.2.4 Selenium 23

2.2.5 Matplotlib....23

2.3 ทฤษฎีเกี่ยวกับ Hardware....24

2.3.1 Nvidia Jetson Nano 24

2.3.2 7-inch LCD Touch Screen Display....25

สารบัญ(ต่อ)

หน้า

2.3.3 Camera Pi v2 25
บทที่ 3 การออกแบบและพัฒนา 27
3.1 ขั้นตอนการดำเนินโครงงาน 27
3.2 การออกแบบ Application 28
3.3 Sequence Diagram 32
3.4 Modules Design 36
3.4.1 Module Admin 37
3.4.2 Module Customer 56
3.5 UI Design 70
3.5.1 Customer Application 70
3.5.2 Admin Application 78
3.6 การออกแบบ Database 83
3.7 ภาพรวมระบบ 84
3.8 การดำเนินการวิเคราะห์ระบุตัวตนสินค้า 85
3.9 ขั้นตอนการฝึกอบรม 85
3.10 การจัดทำเครื่องชำระสินค้าอัตโนมัติ 88
บทที่ 4 การทดลอง 89
4.1 การทดลองที่ 1 89
4.2 การทดลองที่ 2 90
4.3 การทดลองที่ 3 91

สารบัญ(ต่อ)

หน้า

4.4 การทดลองที่ 4 92
4.5 สรุปผลการทดลอง 92
บทที่ 5 บทสรุปและข้อเสนอแนะ 93
5.1 บทสรุป 93
5.2 ปัญหาและอุปสรรค 93
5.3 Sequence Diagram 94
5.4 Modules Design 94
บรรณานุกรม 96

สารบัญรูป

รูป **หน้า**

2.1 Neural Network....3

2.2 Diagram ตัวอย่างของ Node....5

2.3 รูปแบบ Network....6

2.4 ตัวอย่างการคำนวณจาก output ของ node ก่อนหน้า....6

2.5 กล่องขอบเขตจริงและทำนาย....7

2.6 การคำนวณ IoU....8

2.7 anchor box ที่เรียงต่อกัน....9

2.8 anchor box ที่ถูกแมปในรูปแบบ 2 กล่อง....10

2.9 anchor box ที่ได้ถูกเลือกมาใช้งาน....10

2.10 Scale-Invariant Feature Transform....11

2.11 Speeded Up Robust Feature....12

2.12 Feature from accelerated Segment Test (FAST)....13

2.13 Fast R-CNN....14

2.14 การตรวจจับหลายวัตถุ....15

2.15 ตัวอย่างขั้นตอนการทำงาน YOLO v1....16

สารบัญรูป(ต่อ)

รูป **หน้า**

2.16 anchor box....17
2.17 Python logo....18
2.18 คุณลักษณะของ Python....19
2.19 PyQT logo....20
2.20 ตัวอย่างการจกแต่งแอพพลิเคชั่นโดย QT designer....21
2.21 phpMyAdmin logo....22
2.22 Selenium logo....23
2.23 matplotlib logo....23
2.24 Nvidia Jetson Nano....24
2.25 7-inch LCD Touch Screen....25
2.26 Raspberry Pi Camera v2 8m pixels....25
3.1 Use case ของระบบ....28
3.2 ลำดับการซื้อขายของ Customer....32
3.3 ลำดับการเพิ่ม/ลบสินค้า....33
3.4 ลำดับการแก้ไขข้อมูลสินค้าในฐานข้อมูล....34
3.5 ลำดับการแสดงข้อมูลสถิติการซื้อขาย....35
3.6 โครงสร้างของโปรแกรม....36
3.7 Modules ต่างๆภายใน Admin.py....38
3.8 Modules ต่างๆภายใน Admin.py....38

สารบัญรูป(ต่อ)

รูป **หน้า**

3.9 Flowchart StoreUI 39
3.10 Flowchart addTableStore 40
3.11 Flowchart LoadDataStore 41
3.12 Flowchart Myconverter 42
3.13 Flowchart EditUi 43
3.14 Flowchart addTableEdit 44
3.15 Flowchart UpdateEdit 45
3.16 Flowchart LoadDataEdit 46
3.17 Flowchart AnRUi 47
3.18 Flowchart addTableAnR 48
3.19 Flowchart addNum 49
3.20 Flowchart removeNum 50
3.21 Flowchart LoadDataAnR 51
3.22 Flowchart StatUi 52
3.23 Flowchart addTableStat 53
3.24 Flowchart updateplot 54
3.25 Flowchart LoadDataStat 55
3.26 Modules ต่างๆภายใน Customer.py 56
3.27 Flowchart homeUi 57
3.28 Flowchart warnUi 58

สารบัญรูป(ต่อ)

รูป **หน้า**

3.29 Flowchart loadUi 59

3.30 Flowchart scan 60

3.31 Flowchart listUi 61

3.32 Flowchart paymentUi 62

3.33 Flowchart loadGenQRUi 63

3.34 Flowchart genqr 64

3.35 Flowchart text_qr 65

3.36 Flowchart qrui 66

3.37 Flowchart qrui2 67

3.38 Flowchart paySucUi 68

3.39 Flowchart paySucUi2 69

3.40 หน้า HOME 70

3.41 หน้า Warning 71

3.42 หน้า Loading 1 72

3.43 หน้า List 73

3.44 หน้า payment 74

3.45 หน้า Loading 2 75

3.46 หน้า QRUI 76

3.47 หน้า pay_Successful 77

3.48 หน้า home 78

สารบัญรูป(ต่อ)

รูป **หน้า**

3.49 หน้า Store....79

3.50 หน้า EDIT....80

3.51 หน้า AddAndRemove....81

3.52 หน้า STAT....82

3.53 Entity Relationship Diagram....83

3.54 System Diagram....84

3.55 Performance on the COCO Dataset....85

3.56 ตัวอย่างการ Label image....86

3.57 ตัวอย่าง format ใน file train.text....86

3.58 ตัวอย่างการฝึกอบรมด้วย Darknet บน COLAB....87

3.59 ตัวอย่างการตรวจจับวัตถุจากภาพ....87

3.60 โมเดลอุปกรณ์....88

สารบัญตาราง

ตาราง **หน้า**

ตาราง 3.1 ขั้นตอนการดำเนินโครงงาน 27
ตาราง 3.2 Use case ดูรายการสินค้า.....29
ตาราง 3.3 Use case สแกน QR CODE เพื่อชำระยอดเงิน.....29
ตาราง 3.4 Use case ตรวจสอบยอดเงินที่ได้รับ 29
ตาราง 3.5 Use case จัดการฐานข้อมูล 30
ตาราง 3.6 Use case ปรับเปลี่ยนรายละเอียดสินค้า.....30
ตาราง 3.7 Use case เพิ่มสินค้า.....30
ตาราง 3.8 Use case ลบสินค้า.....31
ตาราง 3.9 Use case ดูสถิติประวัติการซื้อของสินค้า.....31
ตาราง 3.10 Data Dictionary 83
ตาราง 3.11 รายการวัสดุ อุปกรณ์ที่ใช้ในการจัดทำเครื่องตรวจจับสินค้า.....88
ตาราง 4.1 การทดลองที่ 1 89
ตาราง 4.2 การทดลองที่ 2 90
ตาราง 4.3 การทดลองที่ 3 91
ตาราง 4.4 การทดลองที่ 4 92

บทที่ 1

บทนำ

1.1 ความเป็นมาของปัญหา

ในปัจจุบันร้านสะดวกซื้อมีมากมายสามารถพบเห็นได้ทั่วไปในชุมชนต่างๆ และก็กลายเป็นธุรกิจที่รุ่งเรืองไปในประเทศ เมื่อพูดถึงในด้านทางธุระกิจร้านค้าปลีก องค์ประกอบอย่างนึงก็คือพนักงาน

ซึ่งอาจเกิดข้อบกพร่องในบางครั้งและอาจเกิดขึ้นได้เนื่องจากการทำงานของมนุษย์นั้นมีขีดจำกัด ไม่ว่าจะเป็นการทำรายการผิดพลาดที่อาจจะทำให้บริษัทเสียผลประโยชน์หรือจะเป็นการทำงานได้อย่างไม่เต็มประสิทธิภาพเท่าที่ควรเช่นทำงานเชื่องช้า ทำให้เสียเวลาของลูกค้า หรือจะเป็นการทำงานที่ไม่เหมาะสมกับเงินค่าจ้าง เช่นในช่วงกลางวันจะมีผู้ใช้บริการมากกว่าในตอนกลางคืน

จากบทความข้างต้นจะแสดงให้เห็นถึงปัญหาเล็กน้อยที่เมื่อนำมาทบกับจำนวนสาขาที่มีจำนวนมากในด้านของธุระกิจก็จะกลายเป็นปัญหาที่ไม่ควรมองข้าม เราจึงได้ออกแบบอุปกรณ์ที่สามารถทำงานได้ดั่งพนักงานแคชเชียร์ในอุดมคติโดยเน้นด้านความถูกต้องและเวลาในการทำงานของอุปกรณ์ โดยอุปกรณ์สามารถตรวจจับสินค้าได้ถูกต้องมากกว่า 80 เปอร์เซ็นต์ อุปกรณ์สามารถทำงานได้ตลอด 24 ชั่วโมง อุปกรณ์สามารถเรียนรู้ผลิตภัณฑ์ใหม่ๆได้ สามารถทำรายการสินค้า ง่ายต่อการใช้งานและลูกค้าสามารถชำระเงินได้ด้วยตัวลูกค้าเอง

1.2 วัตถุประสงค์ของโครงงาน

1) เพิ่มความสะดวกสบายในการใช้บริการร้านสะดวกซื้อ
2) ลดอัตราข้อผิดพลาดในขั้นตอนการชำระเงิน
3) ลดการใช้แรงงานมนุษย์โดยไม่จำเป็น
4) ทำให้เจ้าของร้านสะดวกซื้อได้ผลสรุปของกำไรในแต่ละช่วง
5) ทำให้เจ้าของร้านสะดวกซื้อทราบว่าสินค้าไหนขายดี

1.3 ขอบเขตของโครงงาน

1) ผลิตภัณฑ์ในร้านสะดวกซื้อจำนวน 15 ชิ้น
2) ผลิตภัณฑ์ต้องเป็นผลิตภัณฑ์ที่มีทั่วไปตามร้านสะดวกซื้อ

1.4 เป้าหมายของโครงงาน

1) ระบบสามารถทำงานได้ตลอด 24 ชั่วโมง
2) ระบบสามารถเก็บภาพและประมวลผลระบุตัวตนสินค้าได้
3) ระบบสามารถคำนวณค่าใช้จ่ายสินค้าที่ถูกระบุได้
4) สามารถคำนวณกำไรในช่วงเวลาได้

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1) เพิ่มความสะดวกสบายในการใช้บริการร้านสะดวกซื้อ
2) ลดอัตราข้อผิดพลาดในขั้นตอนการชำระเงินในร้านค้าสะดวกซื้อ
3) ลดค่าใช้จ่ายในการว่าจ้างพนักงาน
4) เพิ่มความมั่นใจในการตัดสินใจทางการตลาดของเจ้าของร้าน

บทที่ 2

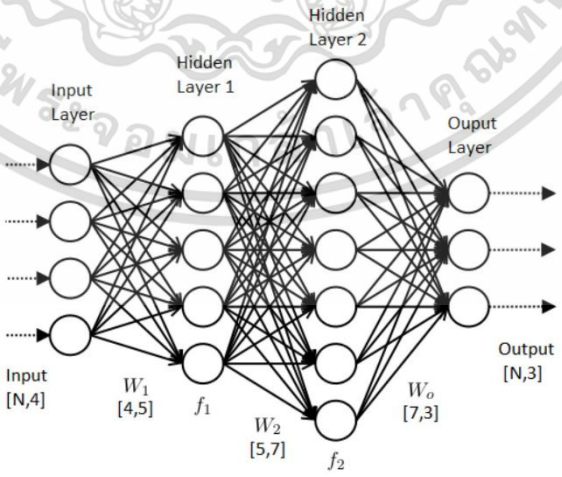
เอกสารและโครงงานที่เกี่ยวข้อง

2.1 ทฤษฎีเกี่ยวกับการตรวจจับวัตถุ

การตรวจจับวัตถุเป็นเทคโนโลยีที่เกี่ยวข้องกับ Computer vision ที่เกี่ยวข้องกับการตรวจจับอินสแตนซ์ของวัตถุหรือคลาสต่างๆ (เช่นมนุษย์อาคารหรือยานพาหนะ) ในวิดีโอและรูปภาพดิจิทัล การตรวจจับวัตถุได้รับการพิสูจน์แล้วว่าเป็นโมดูลที่โดดเด่นสำหรับการใช้งานที่สำคัญมากมายเช่นการเฝ้าระวังวิดีโอการขับขี่แบบอิสระการตรวจจับใบหน้าและอื่น ๆ ในบทนี้เราจะหารือเกี่ยวกับหลักการและเทคนิคที่นิยมใช้กันอย่างแพร่หลายพร้อมกับ library และกรอบที่ใช้สำหรับการใช้เทคนิค บทนี้นำเสนอเทคนิคที่มีอยู่ในสาขา Computer Vision ซึ่งเป็นข้อมูลอ้างอิงสำหรับผู้ใช้เพื่อเลือกเทคนิคที่เหมาะสมพร้อมกับกรอบงานที่เหมาะสมสำหรับการนำไปใช้

2.1.1 หลักการเบื้องต้น

2.1.1.1 Neural Network



รูป 2.1 Neural Network

Neural Network เป็นเทคโนโลยีสาขาหนึ่งของ Artificial Intelligence เป็นแนวคิดที่กำเนิดมานานตั้งแต่ปี 1943 คือการพัฒนาเครือข่ายของ Algorithm ให้ทำงานในรูปแบบเดียวกับการทำงานของเครือข่ายระบบประสาทของมนุษย์ ซึ่งคุณสมบัติดั่งกล่าวของมนุษย์มีด้วยกัน 4 ลักษณะ ดังนี้

1. สามารถระบุตัวตนของสิ่งของได้โดยใช้เวลาช่วงสั้นๆ

มนุษย์สามารถเชื่อมโยงข้อมูลต่าง ๆ ในสมองเข้าด้วยกันและสรุปผลได้อย่างรวดเร็วและถูกต้อง เช่น การเห็นเงาของบุคคลและสามารถเข้าใจว่าเป็นบุคคล

2. สามารถเข้าใจทักษะหลากหลายและไม่เกี่ยวข้องกันได้

มนุษย์สามารถพัฒนาทักษะหลายๆด้านที่ไม่เกี่ยวข้องกันไปได้พร้อมกันทำให้เกิดความสร้างสรรค์ในการแก้ปัญหา เช่น การที่บุคคลมีความถนัดด้านการวาดภาพและคณิตศาสตร์ ไปได้ด้วยกัน

3. เข้าใจตรรกะและกติกา

มนุษย์สามารถเข้าใจตรรกะและกติกาโดยอย่างยิ่งเมื่อต้องอยู่ร่วมกันทำให้ต้องเคารพกติกาและตรรกะของผู้อื่น

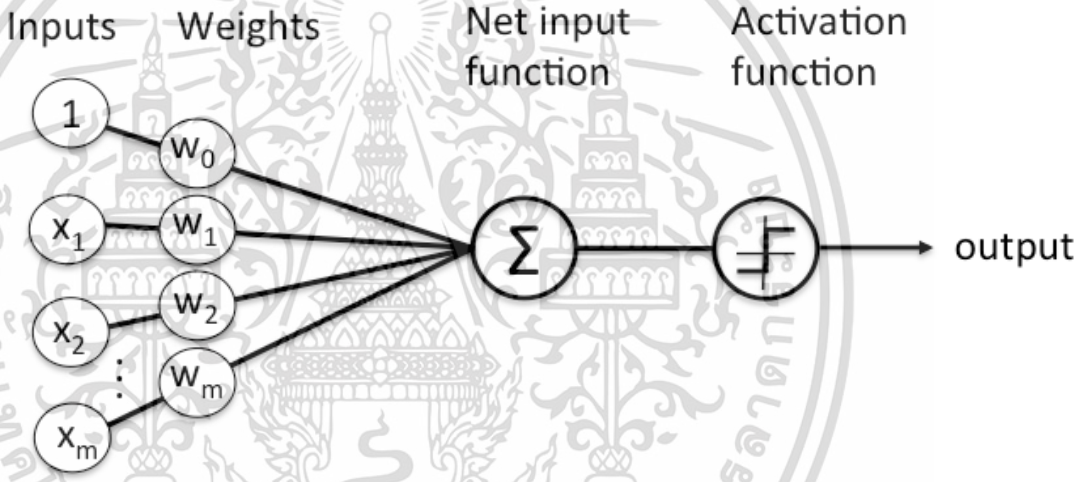
4. เชื่อมโยงหน่วยประมวลผลเพื่อแก้ปัญหา

มนุษย์สามารถเชื่อมโยงการทำงานของระบบประสาทให้ทำงานพร้อมกันเพื่อสังเคราะห์วิธีแก้ปัญหา เช่น การคิดทบทวนของมนุษย์

ทั้งหมดนี้จะแสดงให้เห็นประสิทธิภาพของเครือข่ายระบบประสาทของมนุษย์ที่จะช่วยให้ Algorithm เข้าใจชุดข้อมูลโดยไม่ต้องอาศัยข้อมูลจำนวนมาก สามารถเห็น Pattern ของข้อมูลและการคาดการณ์ข้อมูลบางประเภทอย่างแม่นยำ

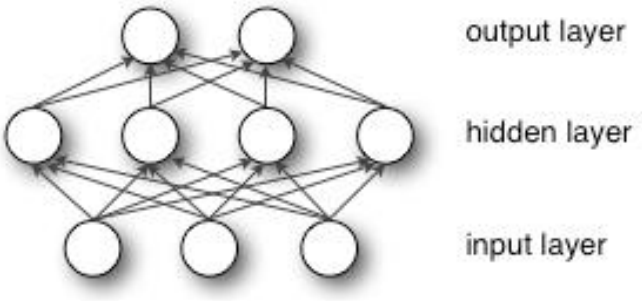
2.1.1.2 Deep Learning

Deep Learning คือชื่อที่เราเรียกการซ้อนทับของ Neural Network หลายๆ ชั้น ชั้นพวกนี้ถูกสร้างจากการรวมตัวของ Nodes หรือจุดที่เกิดการคำนวณและประมวลผล เปรียบเสมือนเซลประสาทของมนุษย์ซึ่งเกิดขึ้นเมื่อมีสิ่งเร้าเพียงพอ Nodes รวมอินพุตจากข้อมูลกับชุดของค่าสัมประสิทธิ์หรือน้ำหนักที่ขยายหรือลดอินพุต กำหนดความสำคัญให้กับอินพุตสำหรับงานที่อัลกอริทึมพยายามเรียนรู้ เช่น อินพุตใดที่มีประโยชน์ที่สุดคือจำแนกข้อมูลโดยไม่มีข้อผิดพลาด ผลิตภัณฑ์อินพุตน้ำหนักเหล่านี้จะถูกรวมเข้าด้วยกัน จากนั้นผลรวมจะถูกส่งผ่านฟังก์ชั่นการเปิดใช้งานของโหนดเพื่อกำหนดว่าควรมีความคืบหน้ามากขึ้นหรือไม่ผ่านเครือข่ายเพื่อให้ได้ผลลัพธ์ที่ดีที่สุด



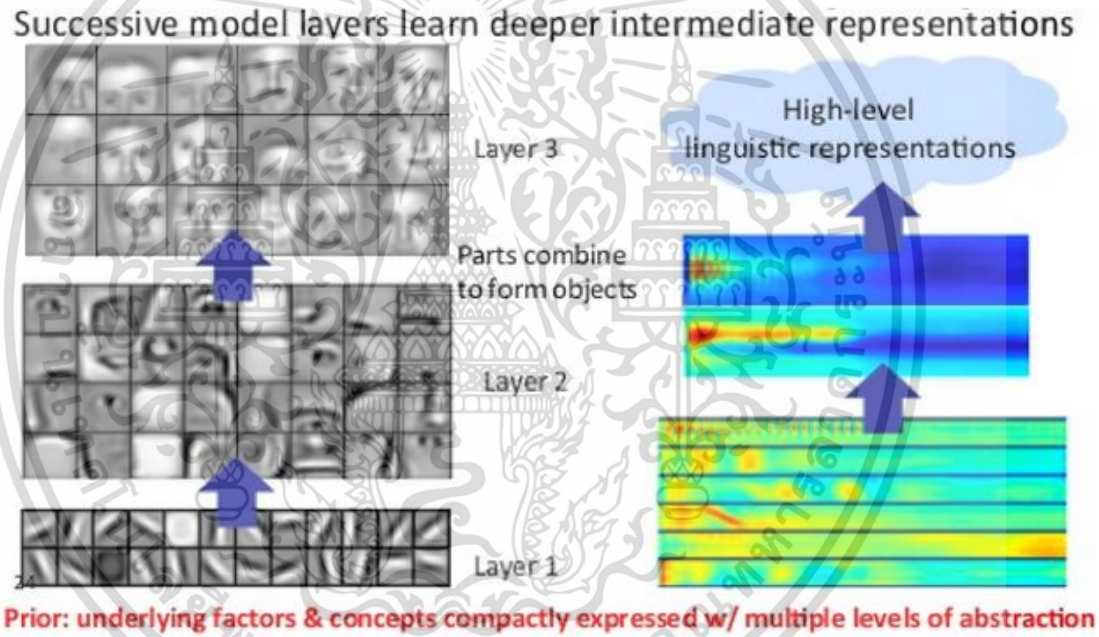
รูปที่ 2.2 Diagram ตัวอย่างของ Node

เลเยอร์ของโหนดเป็นแถวของการทำงานแบบสวิตช์ที่คล้ายกับเซลล์ประสาทที่เปิดหรือปิดเมื่ออินพุตถูกป้อนผ่าน Network เอาต์พุตของเลเยอร์แต่ละอันจะมีอินพุตของเลเยอร์ถัดไปพร้อมกัน โดยเริ่มจากเลเยอร์เริ่มต้นที่รับข้อมูล



รูปที่ 2.3 รูปแบบ Network

Deep Learning Network นั้นมีความแตกต่างจาก Neural Network แบบเลเยอร์เดี่ยวที่ซ่อนอยู่ทั่วไปมากกว่าโดยความลึก นั่นคือจำนวนของเลเยอร์โหนดที่ข้อมูลจะต้องผ่านในกระบวนการหลายขั้นตอนของการจดจำรูปแบบ



รูปที่ 2.4 ตัวอย่างการคำนวณจาก output ของ node ก่อนหน้า

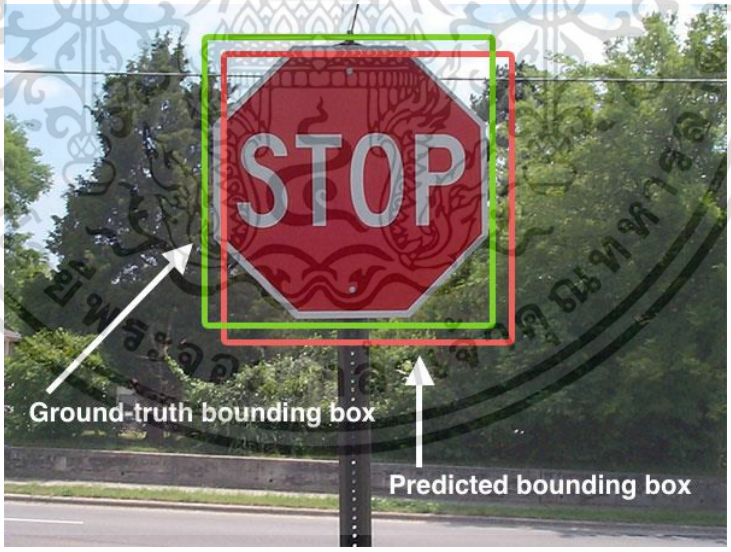
Deep Learning Network นั้นแต่ละชั้นของโหนดจะฝึกอบรมเกี่ยวกับคุณลักษณะที่แตกต่างกันโดยขึ้นอยู่กับผลลัพธ์ของเลเยอร์ก่อนหน้า ยิ่งคุณเข้าไปในโครงข่ายประสาทมากเท่าไหร่คุณสมบัติที่โหนดของคุณจะจดจำก็จะมีความซับซ้อนมากขึ้นเท่านั้น

2.1.1.3 หลักการ IoU

Intersection over Union เป็นตัวชี้วัดการประเมินผลที่ใช้ในการวัดความแม่นยำของการตรวจจับวัตถุบนชุดข้อมูลเฉพาะ เรามักจะเห็นตัวชี้วัดการประเมินนี้ใช้ในการตรวจจับวัตถุเช่น PASCAL VOC ที่ยอดนิยม โดยปกติแล้วคุณจะพบจุดตัดผ่านยูเนี่ยนที่ใช้ในการประเมินประสิทธิภาพของเครื่องตรวจจับวัตถุ HOG + Linear SVM และเครื่องตรวจจับโครงข่ายประสาทเทียม Convolutional (R-CNN, Faster R-CNN, YOLO ฯลฯ); อย่างไรก็ตามโปรดทราบว่าอัลกอริทึมจริงที่ใช้ในการสร้างการคาดการณ์นั้นไม่สำคัญ Intersection over Union เป็นเพียงการวัดผลการประเมิน อัลกอริทึมใด ๆ ที่จัดเตรียมกล่องขอบเขตที่คาดการณ์ไว้เนื่องจากเอาต์พุตสามารถถูกประเมินโดยใช้ IoU

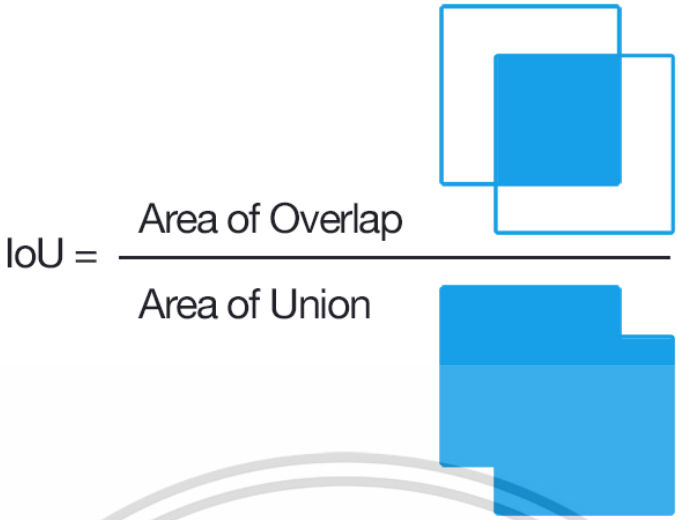
เพื่อเป็นทางการมากขึ้นการใช้ Intersection over Union ในการประเมินการตรวจจับวัตถุที่เราต้องการ:

1. กล่องขอบขอบเขตจริง (เช่นกรอบมือที่มีป้ายกำกับขอบเขตจากชุดทดสอบที่ระบุตำแหน่งที่อยู่ในภาพวัตถุของเรา)
2. กล่องขอบเขตที่ทำนายจากโมเดลของเรา



รูปที่ 2.5 กล่องขอบเขตจริงและทำนาย

ดังรูปที่ 2.5 เราจะเห็นกล่องที่เป็นขอบเขตของวัตถุจริงๆ(กล่องสีเขียว) และกล่องขอบเขตที่การตรวจจับวัตถุตรวจพบได้(กล่องสีแดง)



รูปที่ 2.6 การคำนวณ IoU

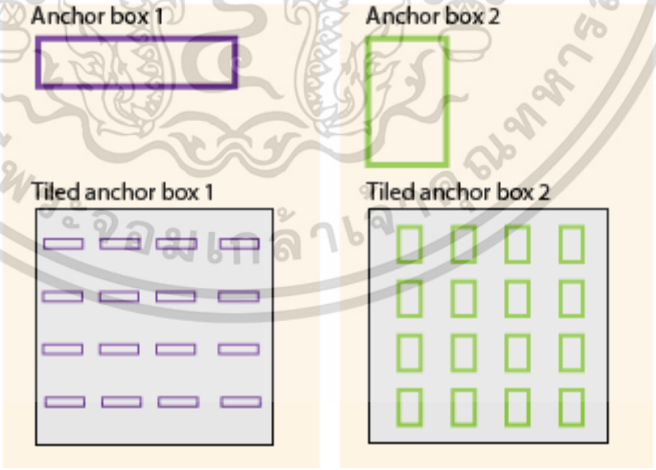
ในตัวเศษที่เราคำนวณพื้นที่ของการทับซ้อนระหว่างกล่องขอบเขตที่คาดการณ์และกล่องขอบเขตความจริงส่วนตัวหารเป็นพื้นที่ของการ Union พื้นที่ที่ล้อมรอบด้วยทั้งกล่องคาดการณ์ที่คาดการณ์และกล่องขอบเขตความจริง

2.1.1.4 Anchor Box

Anchor box เป็นชุดของกล่อง bounding ที่กำหนดไว้ล่วงหน้าซึ่งมีความสูงและความกว้างที่แน่นอน กล่องเหล่านี้ถูกกำหนดให้จับอัตราส่วนและอัตราส่วนของคลาส(วัตถุ)เฉพาะที่คุณต้องการตรวจจับและโดยทั่วไปจะถูกเลือกตามขนาดวัตถุในชุดข้อมูลการฝึกอบรมของคุณ ในระหว่างการตรวจจับกล่องยึดที่กำหนดไว้ล่วงหน้านั้นจะเรียงต่อกันเป็นภาพ เครือข่ายคาดการณ์ความน่าจะเป็นและคุณลักษณะอื่น ๆ เช่นพื้นหลังการตัดกันแบบยูเนี่ยน (IoU) และการออฟเซ็ตสำหรับ anchor box ทุกอัน การทำนายจะใช้ในการปรับแต่ง anchor box แต่ละอัน คุณสามารถกำหนด anchor box แต่ละกล่องสำหรับขนาดวัตถุที่แตกต่างกัน anchor box มีค่าเท่ากับกล่องคาดเดาขอบเขตตอนเริ่มแรก

เครือข่ายไม่ได้คาดการณ์โดยตรงกับขอบเขตกล่อง แต่คาดการณ์ความน่าจะเป็นและการปรับแต่งที่สอดคล้องกับ anchor box แบบเรียงต่อกัน

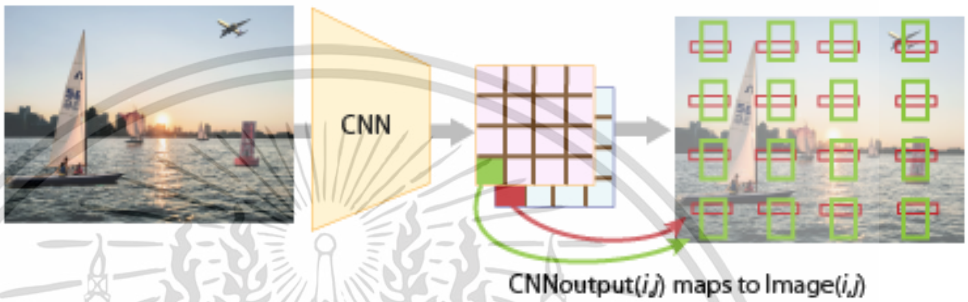
เครือข่ายส่งคืนชุดการทำนายที่ไม่ซ้ำกันสำหรับทุก anchor box ที่กำหนดไว้ แผนผังคุณลักษณะสุดท้ายแสดงการตรวจจับวัตถุสำหรับแต่ละคลาส(วัตถุ) การใช้ anchor box ช่วยให้เครือข่ายสามารถตรวจจับวัตถุหลายชิ้นวัตถุที่มีขนาดต่างกันและวัตถุที่ทับซ้อนกัน



รูปที่ 2.7 anchor box ที่เรียงต่อกัน

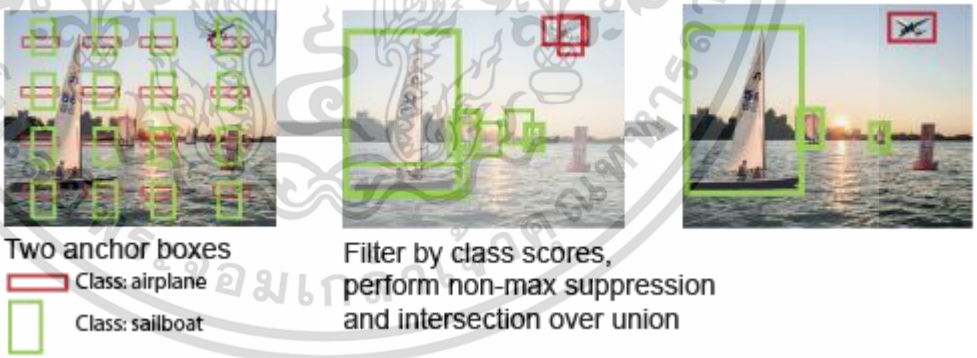
2.1.1.4.1 การทำงานของ anchor box

ตำแหน่งของ anchor box ถูกกำหนดโดยการแมปตำแหน่งที่ตั้งของเอาท์พุทเครือข่ายกลับไปยังอินพุต กระบวนการจะถูกทำซ้ำสำหรับทุกเครือข่ายเอาท์พุท ผลลัพธ์จะสร้างชุดของ anchor box ในภาพทั้งหมด แต่ละ anchor box จะแสดงคำทำนายเฉพาะของคลาส(วัตถุ)ต่างๆ รูปภาพด้านล่างมี anchor box สองกล่องเพื่อคาดการณ์สองครั้งต่อสถานที่บนรูปภาพ



รูปที่ 2.8 anchor box ที่ถูกแมปในรูปแบบ 2 กล่อง

Anchor box แต่ละกล่องถูกสร้างกระจายอยู่ทั่วรูปภาพ จำนวนเอาต์พุตเครือข่ายเท่ากับจำนวน anchor box และเครือข่ายจะสร้างการคาดการณ์สำหรับผลลัพธ์ทั้งหมด



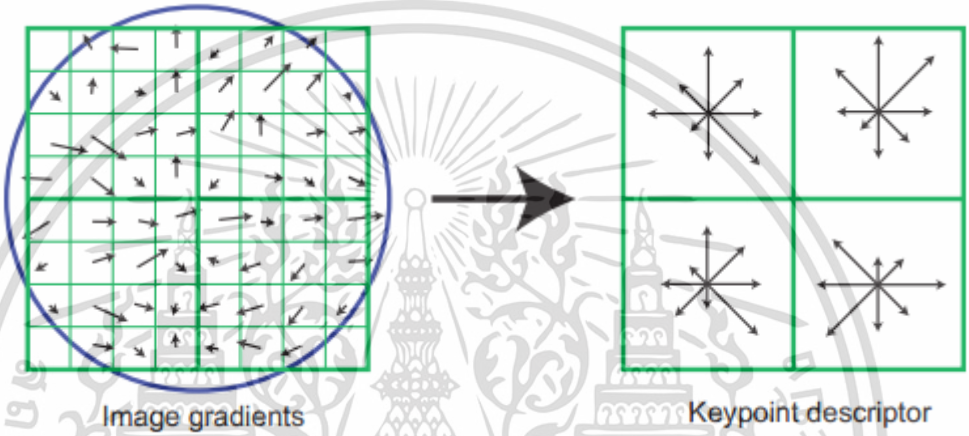
รูปที่ 2.9 anchor box ที่ได้ถูกเลือกมาใช้งาน

ในการสร้างการตรวจจับวัตถุขั้นสุดท้าย anchor box ที่อยู่ในคลาสพื้นหลังจะถูกลบออกและรายการที่เหลือจะถูกกรองด้วยคะแนนความมั่นใจ anchor box ที่มีคะแนนความเชื่อมั่นสูงสุดจะถูกเลือก

2.1.2 เทคนิคการตรวจจับวัตถุ

2.1.2.1 Scale-Invariant Feature Transform (SIFT)

วิธี SIFT สามารถระบุวัตถุได้อย่างมีประสิทธิภาพแม้จะอยู่ในสภาพที่ยุ่งเหยิงและอยู่ภายใต้การบิดเบี้ยวบางส่วน เนื่องจากตัวบอกคุณสมบัติ SIFT นั้นไม่แปรเปลี่ยนตามขนาด การวางแนวและการบิดเบือนเลียนแบบ

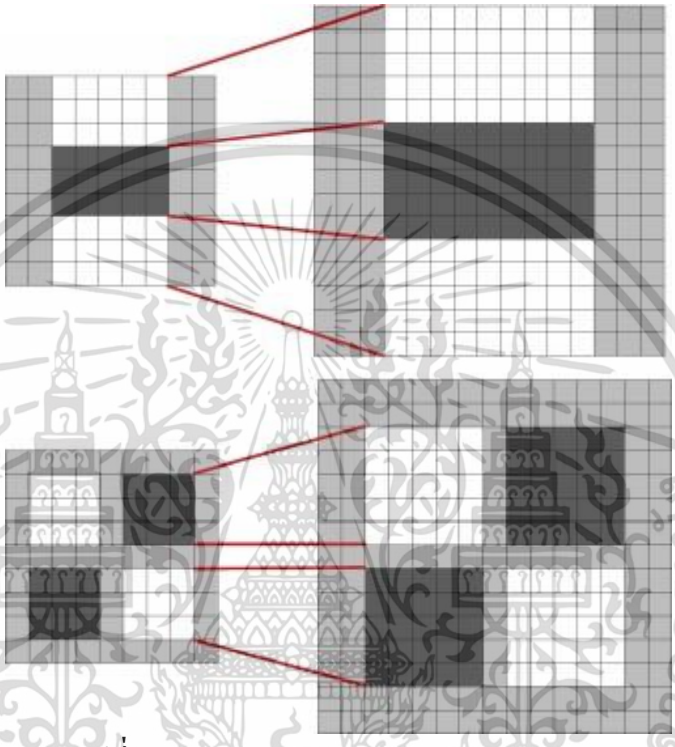


รูปที่ 2.10 Scale-Invariant Feature Transform

พื้นที่จุดสนใจ (จุดสำคัญ) ถูกตรวจพบในตำแหน่งที่โดดเด่นในภาพ ในการโลคัลไลซ์เซชันของ Keypoint ในบรรดา Keypoint จะมีการเลือกจุดสำคัญที่แตกต่างกันโดยการเปรียบเทียบแต่ละพิกเซลในคุณสมบัติที่ตรวจพบกับพิกเซลอยู่ใกล้เคียง

2.1.2.2 Speeded Up Robust Feature (SURF)

อัลกอริธึม SURF มีเทคนิคการตรวจจับคล้ายกับอัลกอริธึม SIFT ความแตกต่างคือ อัลกอริธึม SURF ทำให้การตรวจจับ extrema ในพื้นที่สเกลง่ายขึ้นโดยการสร้างพื้นที่สเกลผ่านการเปลี่ยนแปลงการกระจายแทนที่จะใช้ตัวกรอง Difference of Gaussian (DoG) เพื่อประมาณ Laplacian ของ Gaussian, SURF ใช้การเป็นตัวแทนตัวกรองกล่อง

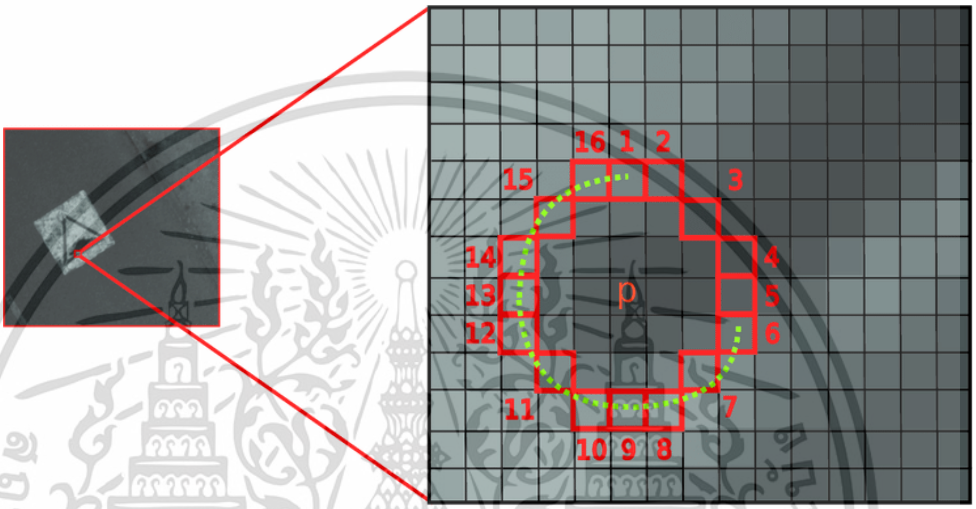


รูปที่ 2.11 Speed Up Robust Feature (SURF)

ตัวอธิบายภาพถูกสร้างขึ้นโดยการวัดการไล่ระดับสีของภาพ อัลกอริธึม SURF ที่ใช้ตัวอธิบายภาพมีความทนทานต่อการแปลงภาพที่แตกต่างกันและการรบกวนในภาพโดยสรุป แม้จะมีเวลาลดลงสำหรับการคำนวณคุณสมบัติและการจับคู่ แต่ก็มีความยุ่งยากในการให้การรับรู้วัตถุแบบเรียลไทม์ในสภาพแวดล้อมระบบฝังตัวที่ จำกัด ทรัพยากร

2.1.2.3 Feature from Accelerated Segment Test (FAST) corner detector

มุมในภาพอินพุตมีคุณสมบัติที่โดดเด่นซึ่งแตกต่างอย่างชัดเจนจากพิกเซลแวดล้อม การตรวจจับและการติดตามมุมในภาพที่เชื่อถือได้นั้นเป็นไปได้แม้ว่าภาพจะมีความผิดปกติทางเรขาคณิต ดังนั้นอัลกอริทึมการรู้จำวัตถุส่วนใหญ่ใช้ข้อมูลมุมเพื่อแยกคุณลักษณะ ในวิธี SIFT มีประสิทธิภาพที่ดี แต่ไม่มีประสิทธิภาพสำหรับการจดจำวัตถุแบบเรียลไทม์ เนื่องจากใช้เวลาในการคำนวณนาน

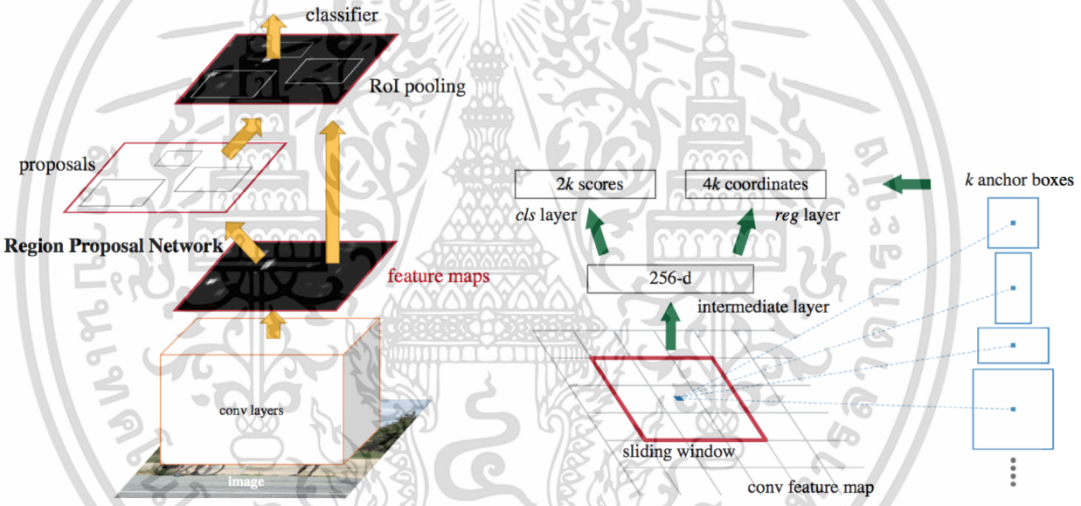


รูปที่ 2.12 Feature from Accelerated Segment Test (FAST)

เครื่องตรวจจับมุมโมเดล FAST เร็วกว่าตัวโมเดลตรวจจับ SIFT 10 เท่าโดยไม่ทำให้ประสิทธิภาพการทำงานลดลง พบมุมโดยตรวจสอบวงกลมสิบหกพิกเซลรอบ ๆ ตัวเลือกมุม ตัวเลือกนี้ถูกตรวจจับว่าเป็นมุมหากความเข้มของจำนวนพิกเซลต่อเนื่องที่กำหนดมีทั้งหมดข้างต้นหรือต่ำกว่าความเข้มของพิกเซลกลางด้วยขีด จำกัด บางอย่าง จุดสนใจที่ดึงมานั้นอยู่ในบริเวณที่มีความคมชัดสูงและโดดเด่นของภาพ

2.1.2.4 Fast R-CNN

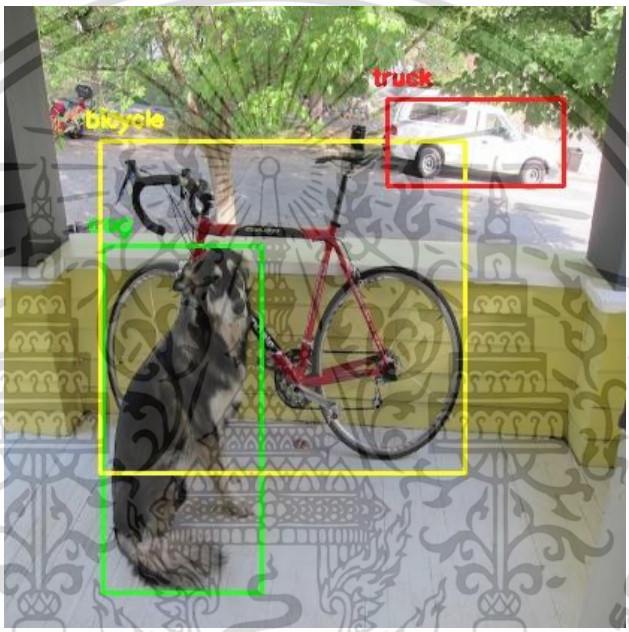
เครือข่าย Fast R-CNN นำภาพทั้งหมดมาเป็นอินพุตและชุดข้อมูลของวัตถุ เครือข่ายแรกประมวลผลภาพทั้งหมดที่มีconvolutional layer และ max pooling หลายอันเพื่อสร้างแผนที่คุณลักษณะของ convolutional จากนั้นสำหรับแต่ละวัตถุจะเป็นตัวแทนของขอบเขตที่สนใจ (RoI) จะแยกเวกเตอร์คุณลักษณะที่มีความยาวคงที่ออกจากแผนที่คุณลักษณะ แต่ละฟีเจอร์ของเวกเตอร์จะถูกป้อนเข้าสู่ลำดับของเลเยอร์ที่เชื่อมต่ออย่างสมบูรณ์ (fully connected) ซึ่งในที่สุดจะแยกออกเป็นเลเยอร์สองเลเยอร์หนึ่งที่สร้างความน่าจะเป็นของซอฟต์แม็กซ์สำหรับคลาส K-object และสองที่สร้างตัวเลขที่มีค่าสำหรับคลาส K-object แต่ละคลาส แต่ละชุดของค่า 4 ค่าจะเข้ารหัสตำแหน่งกล่องขอบที่ปรับปรุงแล้วสำหรับหนึ่งในคลาส K



รูปที่ 2.13 Fast R-CNN

2.1.2.5 YOLO (You Look Only Once)

YOLO เป็น neural network ที่สามารถตรวจจับวัตถุบนรูปภาพได้โดยจะตีกรอบสีเหลี่ยมรอบวัตถุนั้น ๆ และยังสามารถตรวจจับได้หลายวัตถุในเวลาเดียวกัน โดยเราใช้เครือข่ายประสาทเดียวกับภาพเต็ม เครือข่ายนี้แบ่งภาพเป็นช่วงบริเวณและคาดการณ์กล่องขอบเขตและความน่าจะเป็นสำหรับแต่ละบริเวณ กล่องขอบเขตเหล่านี้ถูกถ่วงน้ำหนักโดยความน่าจะเป็นที่คาดการณ์ไว้

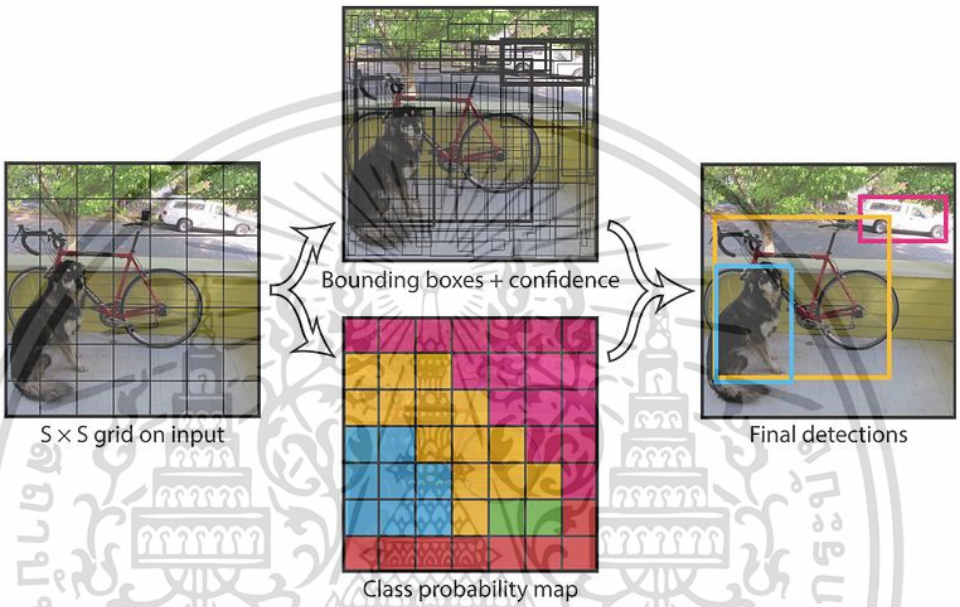


รูป 2.14 การตรวจจับหลายวัตถุ

นวัตกรรมที่โดดเด่นของ YOLO นั้นก็คือ เมื่อพูดถึงการตรวจจับวัตถุในการประมวลผลครั้งเดียวด้วยประสิทธิภาพที่สูงและการใช้เวลาที่รวดเร็ว YOLO สามารถคาดเดาวัตถุได้มากมายด้วยการผ่าน network เพียงครั้งเดียว

2.1.2.5.1 YOLO v1

YOLO (May 2016) โดยแรงบันดาลใจมาจาก GoogleNet มี 24 convolutional layers และ 2 dense layers สำหรับการคาดเดาวัตถุ สถาปัตยกรรมเรียกว่า Darknet (neural network framework)

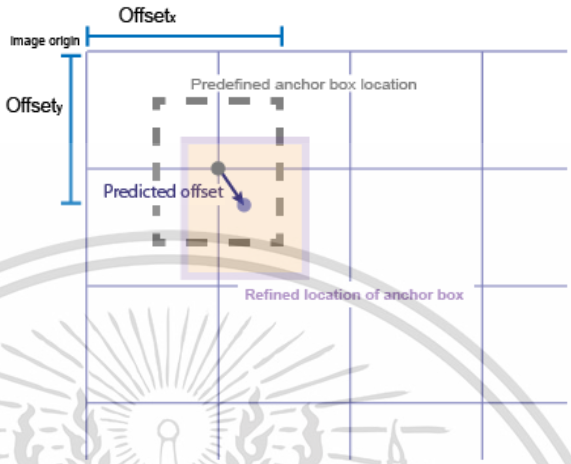


รูป 2.15 ตัวอย่างขั้นตอนการทำงาน YOLO v1

Algorithm ทำงานโดยการแบ่งรูปออกเป็นตาราง สำหรับทุกช่องในตารางจะมีคะแนนความมั่นใจคาดเดาไว้และความเป็นไปได้ของ class (วัตถุ) นั้น ๆ ส่วนคะแนนความมั่นใจนั้นถูกคิดโดยหลักการ IoU(intersection over union), เมตริก

2.1.2.5.2 YOLO v2

YOLO v2 (December 2016) มีการพัฒนาต่อจาก YOLO v1 เนื่องจากจริง ๆ แล้ว YOLO v1 ก็ไม่ได้สามารถตรวจจับวัตถุได้อย่างสมบูรณ์



รูป 2.16 anchor box

YOLO v2 ได้มีการนำ anchor boxes กล่องที่ถูกกำหนดขนาดให้มีอัตราส่วนเท่ากับวัตถุที่ตรวจจับใน dataset

Anchor boxes จึงมีประสิทธิภาพมากกว่ากล่องขอบเขตที่ถูกใช้ใน YOLO v1 เพื่อเพิ่มประสิทธิภาพในการตรวจจับวัตถุที่มีขนาดเล็ก

2.1.2.3 YOLO v3

YOLO v3 (April 2018) มีการพัฒนาเพิ่มขึ้นเล็กน้อย กล่องขอบเขตถูกคาดเดาในสเกลที่แตกต่างจากเดิม และมีการขยาย network (Darknet) เพิ่มเป็น 53 convolutional layers

2.2 ทฤษฎีเกี่ยวกับ Software

โดย Application ในส่วนของ Customer จะทำการบันทึกภาพจากกล้องจากกล้อง Camera Pi v2 และทำการตรวจจับภาพวัตถุผ่าน darknet โดยใช้ Model ที่ได้จากการทดลอง เพื่อสร้างรายการสินค้าที่จะต้องทำการชำระเงิน และสร้างรูปภาพ QR Code จากยอดเงินที่ลูกค้าต้องการชำระ โดยจะต้องแสกน QR Code ผ่าน iBanking เพื่อชำระผ่านระบบ PromptPay เมื่อดำเนินการเสร็จสิ้นจะทำการอัพเดทไปยังฐานข้อมูล โดยด้าน Hardware ที่เราต้องใช้จะต้องมี กล้องสำหรับตรวจจับ และ CPU สำหรับการประมวลผล

ส่วนทางด้าน Software เราจะทำการใช้ภาษา Python เนื่องจากมี library open source จำนวนมาก เพื่อช่วยอำนวยความสะดวกในการเขียนโปรแกรม

2.2.1 Python

รูป 2.17 Python logo

Python คือภาษาโปรแกรมมิ่งอย่างนึงที่ไม่ยึดติดกับเพลตฟอร์ม นั่นหมายความว่า Python สามารถดำเนินการได้บนระบบปฏิบัติการต่าง ๆ ได้ไม่ว่าจะเป็น Unix, Linux, Windows NT, Windows 2000, Windows XP หรือ FreeBSD ที่สำคัญยังเป็น Open Source ที่เปิดให้นักพัฒนาเข้าถึงได้สะดวกไม่มีค่าใช้จ่าย

Python เป็นภาษาการเขียนโปรแกรมระดับสูงตีความและวัตถุประสงค์ทั่วไป สร้างโดย Guido van Rossum และเปิดตัวครั้งแรกในปีพ. ศ. 2534 ปรัชญาการออกแบบของ Python เน้นการอ่านรหัสด้วยการใช้ช่องว่างที่มีความสำคัญ การสร้างภาษาและแนวทางเชิงวัตถุมี

วัตถุประสงค์เพื่อช่วยโปรแกรมเมอร์เขียนโค้ดที่ชัดเจนและสมเหตุสมผลสำหรับโครงการขนาดเล็กและขนาดใหญ่

รูป 2.18 คุณลักษณะของ Python

นอกจากนั้น Python ยังเป็นภาษาที่ถูกออกแบบมาให้ใกล้เคียงภาษาของมนุษย์ทำให้สามารถเข้าใจง่ายไม่ซับซ้อนเมื่อเทียบกับภาษาโปรแกรมมิ่งอื่น ๆ และมี library มาตรฐานต่าง ๆ ที่เป็น Open Source ที่สามารถพบได้ใน https://www.python.org

2.2.2 PyQt

รูป 2.19 PyQT logo

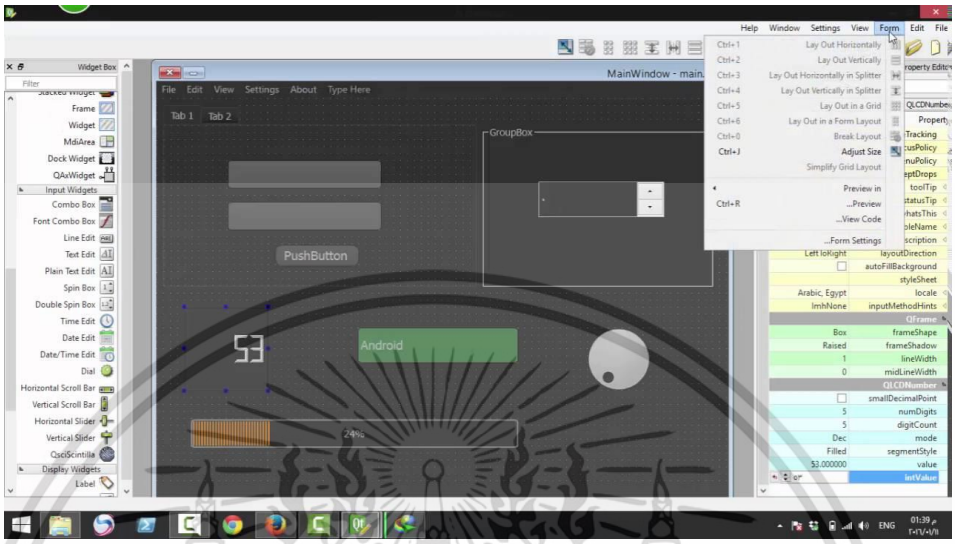
PyQt คือกลุ่มของ Python v2 และ v3 หรือ GUI widgets toolkit ที่จำเป็นต่อการใช้งาน Qt แอพพลิเคชั่นเฟรมเวิร์คและดำเนินการบนแพลตฟอร์มทั้งหมดที่ Qt ได้รองรับนั่นคือ Windows, OS X, Linux, iOS และ Android

PyQt เป็นหนึ่งใน Python ที่ได้รับความนิยมมากที่สุดสำหรับ Qt cross-platform c ++ framework PyQt พัฒนาโดย Riverbank Computing Limited

Qt ได้รับการพัฒนาโดยเป็นส่วนหนึ่งของโครงการ Qt PyQt สำหรับ Qt 4 และ Qt 5 PyQt มีการแจกจ่ายภายใต้ตัวเลือกของสิทธิ์การใช้งาน: GPL เวอร์ชัน 3 หรือใบอนุญาตการค้า

PyQt มีให้บริการในสองรุ่น PyQt4 ซึ่งจะสร้างต่อจาก Qt 4.x และ 5.x และ PyQt5 ซึ่งจะสร้างได้เฉพาะกับ 5.x ทั้งสองรุ่นสามารถสร้างขึ้นสำหรับ Python 2 และ 3 PyQt มีมากกว่า 620 คลาสที่ครอบคลุมส่วนติดต่อผู้ใช้แบบกราฟิกการจัดการ XML การสื่อสารเครือข่ายฐานข้อมูล SQL การท่องเว็บและเทคโนโลยีอื่น ๆ ที่มีอยู่ใน Qt

QtPy เป็น Abstraction Layer หรือการซ่อนการทำงานของระบบย่อยทำให้ไม่เกิดความสับสนขึ้นในการทำงาน ที่ทำให้สามารถสร้างแอพพลิเคชั่นโดยใช้ API เรียกใช้ไปยัง PyQt หรือ PySide



รูป 2.20 ตัวอย่างการตกแต่งแอพพลิเคชั่นโดย QT designer

2.2.3 phpMyAdmin

รูป 2.21 phpMyadmin logo

phpMyAdmin คือโปรแกรมที่ถูกพัฒนาโดยใช้ภาษา PHP เพื่อใช้ในการบริหารจัดการฐานข้อมูล Mysql แทนการคีย์คำสั่ง เนื่องจากการใช้ฐานข้อมูลที่เป็น MySQL บางครั้งจะมีความยุ่งยากในการใช้งาน ดังนั้นจึงต้องมีเครื่องมือในการจัดการฐานข้อมูล MySQL ขึ้นมาเพื่อให้สามารถจัดการ DBMS ที่เป็น MySQL ได้ง่าย และสะดวกยิ่ง โดย phpMyAdmin ถือเป็นเครื่องมือชนิด หนึ่งที่ใช้ในการจัดการ

phpMyAdmin รองรับการทำงานที่หลากหลายบน MySQL และ MariaDB การดำเนินการที่ใช้บ่อย (การจัดการฐานข้อมูล, ตาราง, คอลัมน์, ความสัมพันธ์, ฯลฯ) สามารถดำเนินการผ่านผู้ใช้ในขณะที่คุณยังสามารถเรียกใช้คำสั่ง SQL ใด ๆ ได้โดยตรง

ความสามารถของ phpMyAdmin:

1) สร้างและลบ Database

2) ใช้ web interface ที่ง่ายต่อการใช้ง่าย

3) รองรับหลาย function ของ MySQL

4) นำไฟล์ข้อมูลจาก CSV และ SQL fileเข้าไปเก็บเป็นข้อมูลในตารางได้

5) ค้นหาข้อมูลแบบ globally หรือ subset ได้

2.2.4 Selenium

รูป 2.22 Selenium logo

Selenium เป็น open source frameworks สำหรับการทดสอบ การใช้งานเว็บแอพพลิเคชั่น Selenium มีเครื่องมือการเล่นสำหรับการเขียนการทดสอบการทำงานโดยไม่จำเป็นต้องเรียนรู้ภาษาสคริปต์ การทดสอบ(Selenium IDE) นอกจากนี้ยังมีการเขียนโปรแกรมในภาษาที่นิยม ได้แก่ C#, Groovy , Java, PHP, Perl, Python, Ruby และ Scala การทดสอบนั้นสามารถทำงานกับเว็บบราวเซอร์ที่ทันสมัยได้ ซีลีเนียมสามารถทำงานบน Windows, Linux และ Mac OS

2.2.5 Matplotlib

รูป 2.23 matplotlib logo

เป็นไลบรารี่การจำลองภาพของ Python ใน 2 มิติ Matplotlib เป็นไลบรารี่สำหรับการสร้างภาพข้อมูลหลายแพลตฟอร์มที่สร้างขึ้นจากอาร์เรย์ NumPy และออกแบบมาเพื่อทำงานกับกอง SciPy ที่กว้างขึ้น มันถูกนำมาใช้โดย John Hunter ในปี 2002 ประโยชน์ที่ยิ่งใหญ่ที่สุดของการสร้างภาพข้อมูลคือมันช่วยให้เราสามารถเข้าถึงข้อมูลจำนวนมหาศาลในภาพที่ย่อยได้ง่าย

2.3 ทฤษฎีเกี่ยวกับ Hardware

ในส่วนของ Hardware เราจะใช้ Nvidia Jetson Nano ทำงานเสมือน Central Processor Unit(CPU) ใช้บริหารจัดการระบบและประมวลผลของอุปกรณ์ ใช้ Camera Pi v2 เพื่อเก็บอินพุตก่อนประมวล และ LCD touchscreen ใช้เพื่อสื่อสารกับผู้ใช้

2.3.1 Nvidia Jetson Nano

Nvidia Jetson Nano เป็นคอมพิวเตอร์ขนาดเล็กที่มีประสิทธิภาพสามารถรับพลังงานได้จาก micro-USB และมาพร้อมกับตัวรับส่ง I/O เพิ่มเติมตั้งแต่ GPIO จนถึง CSI ซึ่งช่วยให้สามารถเรียกใช้ Neural Network สำหรับแอปพลิเคชันเช่น การจัดประเภทรูปภาพการตรวจสอบวัตถุการแบ่งกลุ่มและการประมวลผลคำพูด ทั้งหมดในแพลตฟอร์มที่ใช้งานง่าย

รูป 2.24 Nvidia Jetson Nano

นอกจากนั้นยังมีการรองรับจาก NVIDIA JetPack ที่ประกอบด้วย board support package(BSP), Linux OS, NVIDIA CUDA, cuDNN, และ TensorRT library ต่าง ๆ สำหรับ deep learning, การประมวลผลมัลติมีเดียอีกมากมาย

Jetson Nano รองรับเซ็นเซอร์ความละเอียดสูงสามารถประมวลผลเซ็นเซอร์จำนวนมากในแบบคู่ขนานและสามารถเรียกใช้เครือข่ายประสาทที่ทันสมัยหลายแห่ง

ในแต่ละสตรีมเซ็นเซอร์ นอกจากนี้ยังสนับสนุนเฟรมเวิร์ก AI ยอดนิยมมากมายทำให้ผู้พัฒนาสามารถรวมโมเดลและเฟรมเวิร์กที่ต้องการลงในผลิตภัณฑ์ได้อย่างง่ายดาย

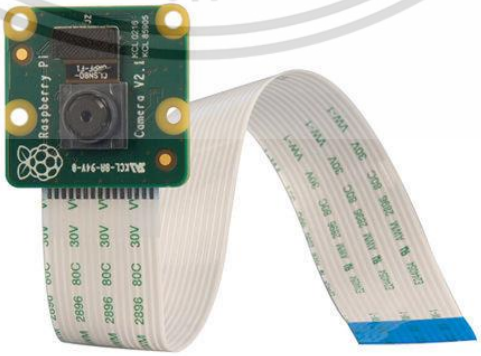
2.3.2 7-inch LCD Touch Screen Display

หน้าจอแสดงผลขนาด 1024*600 pixel แบบจอภาพสัมผัสสำหรับการสื่อสารระหว่างผู้ใช้และอุปกรณ์

รูป 2.25 7 inch LCD Touch Screen

2.3.3 Camera Pi v2

Raspberry Pi Camera Board v2 เป็นเซ็นเซอร์ภาพ Sony IMX219 คุณภาพสูง 8m pixels ที่ออกแบบมาเองสสำหรับ Add-on board สำหรับ Raspberry Pi ที่มีเลนส์โฟกัสคงที่ สามารถถ่ายภาพนิ่งได้ 3280 x 2464 pixels และรองรับวิดีโอ 1080p30, 720p60 และ 640x480p90



รูป 2.26 Raspberry Pi Camera v2 8m pixels

คุณสมบัติ

• ความละเอียดสูง 8 ล้านพิกเซล

• ถ่ายวิดีโอคุณภาพระดับ HD ความคมชัด 1080p, 720p และ 640x480 ด้วยอัตราแสดงผล 30 (1080p), 60 (720p และ 640x480) และ 90 (640x480) เฟรมต่อวินาที

• เซ็นเซอร์รับแสงจาก Sony IMX219PQ CMOS image sensor

• ความละเอียดสูงสุดที่สนับสนุน 3280 x 2464

• อัตราเฟรมสูงสุดในการจับภาพ 30 เฟรมต่อวินาที

• อุณหภูมิในการทำงานสูงสุด +60oC

• อุณหภูมิในการทำงานต่ำสุด -20oC

• ขนาด 23.86 x 25 x 9 มม.

• น้ำหนักเพียง 3 กรัม

• ต่อกับบอร์ด Raspberry Pi ด้วยบัส CSI (Common System Interface) ซึ่งเป็นการเชื่อมต่อแบบ point-to-point บัสนี้พัฒนาโดย Intel ออกแบบมาเพื่อการรับส่งข้อมูลความเร็วสูง 12 ถึง 16 GB/s ด้วยการใช้เทคนิค low-voltage differential signaling

• ใช้กับบอร์ด Raspberry Pi ที่ติดตั้งอิมเมจเวอร์ชั่นล่าสุด

• สายแพ 15 จุดต่อ ยาว 15CM.

บทที่ 3

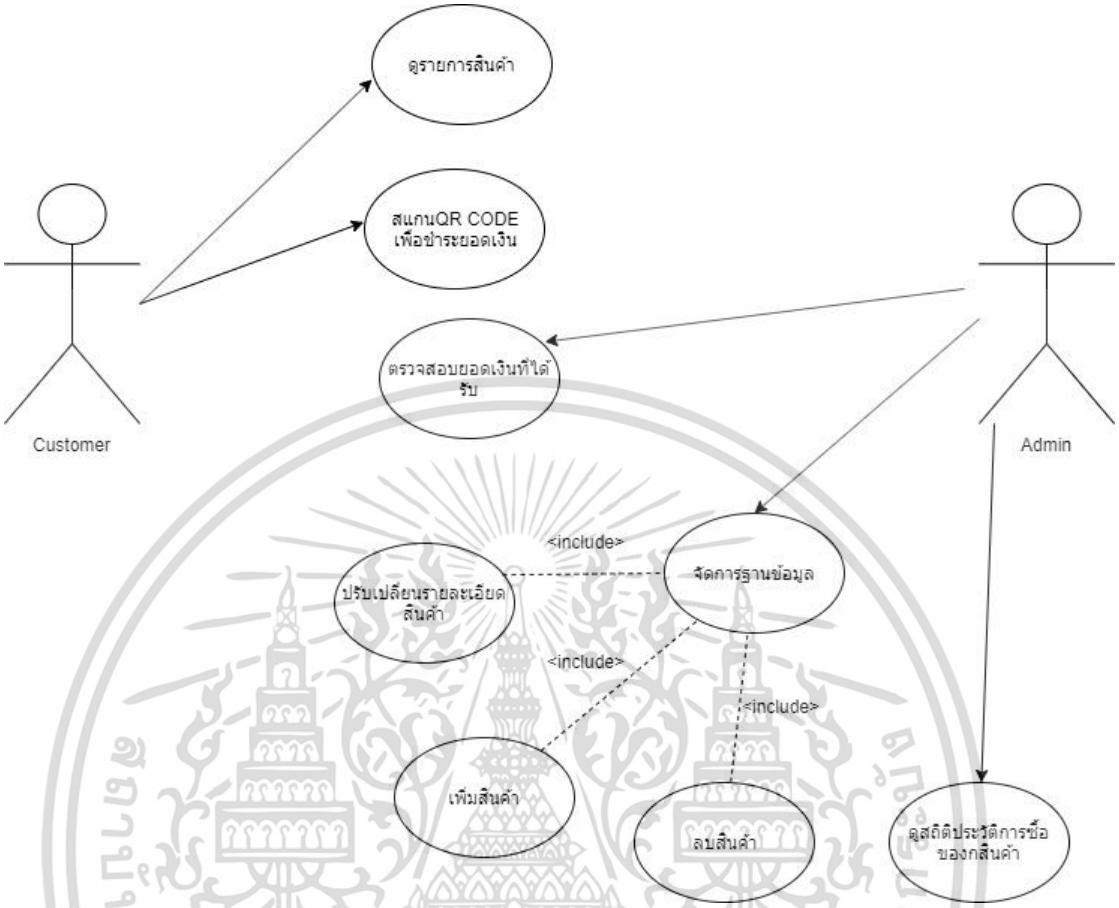
การออกแบบและพัฒนา

3.1 ขั้นตอนการดำเนินโครงงาน

ตาราง 3.1 ขั้นตอนการดำเนินโครงงาน

รายละเอียดการดำเนินการ	พ.ศ. 2562					พ.ศ. 2563		
	ส.ค.	ก.ย.	ต.ค.	พ.ย.	ธ.ค.	ม.ค.	ก.พ.	มี.ค.
1. กำหนดหัวข้อและศึกษาข้อมูล								
2. กำหนดแผนการดำเนินการ								
3. ติดตั้ง Software ที่จำเป็น								
4. ออกแบบโมเดลอุปกรณ์								
5. ติดตั้ง Hardware ที่จำเป็นในการทำ Dataset								
6. จัดทำ Dataset								
7. Label Dataset								
8. ทดสอบ Model								
9. รวบรวมสถิติของ model								
10. ติดตั้ง Hardware เพิ่มเติมสำหรับ UI								
11. จัดทำ Application								
12. จัดทำรายงาน								

3.2 การออกแบบ Application



รูป 3.1 Use case ของระบบ

โดย Customer จะเข้าใช้งานโปรแกรมในส่วน Customer เมื่อนำสินค้าลงในช่องวางสินค้าแล้ว Customer สามารถที่จะดูรายการสินค้ากับยอดเงินที่ต้องการชำระได้และเมื่อดำเนินรายการต่อลูกค้าจะสามารถเห็น QR Code เพื่อใช้ iBanking แสกน QR Code ได้

ในส่วนของ Admin จะเข้าใช้งานโปรแกรมในส่วน Admin โดยจะสามารถตรวจสอบสินค้าในคลังเพิ่มสินค้า ลดสินค้า แก้ไขราคาขายต่างๆได้ใน Store menu และสามารถดูค่าสถิติกำไรยอดขายทั้งหมดได้ใน Stat menu

ตาราง 3.2 Use case ดูรายการสินค้า

ID	Use case
Title	ดูรายการสินค้า
Description	เครื่องทำการแสดงรายการสินค้าที่ตรวจพบ
Actor(s)	Customer
Main Flow	เป็นส่วนที่ให้Customer เช็คว่าสินค้าทีต้องชำระนั้นถูกต้องหรือไม่
Exceptional Flow	เมื่อกดยกเลิกหรือไม่กดยืนยันภายในเวลา15วินาทีจะกลับสู่หน้า HOME

ตาราง 3.3 Use case สแกน QR CODE เพื่อชำระยอดเงิน

ID	Use case
Title	สแกน QR CODE เพื่อชำระยอดเงิน
Description	ผู้ใช้สามารถชำระเงินด้วย QR CODE ที่เครื่องสร้างขึ้น
Actor(s)	Customer
Main Flow	เครื่องจะทำการ Generate QR CODE เพื่อให้ Customer ชำระเงิน
Exceptional Flow	เมื่อกดยกเลิกหรือไม่กดยืนยันภายในเวลา60วินาทีจะกลับสู่หน้า HOME

ตาราง 3.4 Use case ตรวจสอบยอดเงินที่ได้รับ

ID	Use case
Title	ตรวจสอบยอดเงินที่ได้รับ
Description	ผู้ใช้สามารถตรวจสอบยอดเงินที่ได้รับ
Actor(s)	Admin
Main Flow	สามารถเข้าไปตรวจสอบรายการที่ทำสำเร็จในแอพพลิเคชั่นAdmin
Exceptional Flow	

ตาราง 3.5 Use case จัดการฐานข้อมูล

ID	Use case
Title	จัดการฐานข้อมูล
Description	ผู้ใช้สามารถตรวจสอบจัดการฐานข้อมูล
Actor(s)	Admin
Main Flow	Admin สามารถเพิ่มสินค้าหรือนำสินค้าออกจากคลังได้
Exceptional Flow	

ตาราง 3.6 Use case ปรับเปลี่ยนรายละเอียดสินค้า

ID	Use case
Title	ปรับเปลี่ยนรายละเอียดสินค้า
Description	ผู้ใช้สามารถปรับเปลี่ยนข้อมูลต่าง ๆ ของสินค้า
Actor(s)	Admin
Main Flow	Admin สามารถเข้าไปแก้ไขรายละเอียดราคาสินค้าได้ เช่น ราคาขาย และราคาต้นทุน
Exceptional Flow	

ตาราง 3.7 Use case เพิ่มสินค้า

ID	Use case
Title	เพิ่มสินค้า
Description	ผู้ใช้สามารถเข้าไปแก้ไขสินค้าที่จะแสดงเพิ่ม
Actor(s)	Admin
Main Flow	Admin สามารถเข้าไปแก้ไขสินค้าที่จะแสดงเพิ่มได้แต่สินค้านั้นต้องผ่านการเทรนมาก่อน
Exceptional Flow	

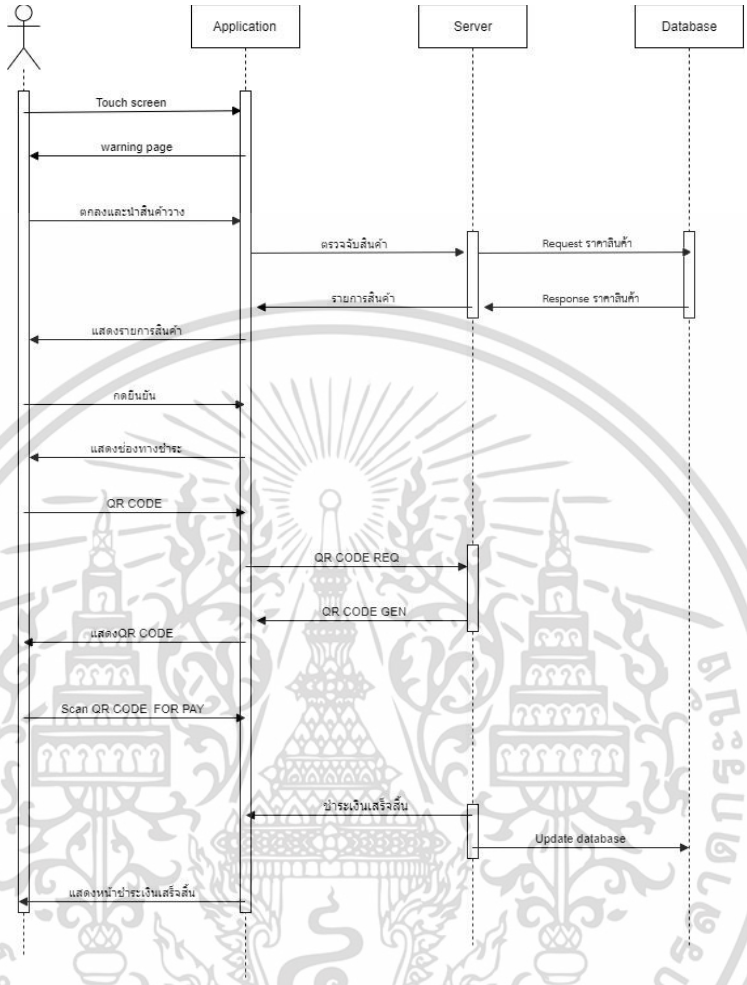
ตาราง 3.8 Use case ลบสินค้า

ID	Use case
Title	ลบสินค้า
Description	ผู้ใช้สามารถเข้าไปลบสินค้านั้น ๆ ออกจากรายการสินค้า
Actor(s)	Admin
Main Flow	Admin สามารถเข้าไปแก้ไขให้สินค้าชนิดนั้นไม่สามารถคิดเงินได้
Exceptional Flow	

ตาราง 3.9 Use case ดูสถิติประวัติการซื้อของสินค้า

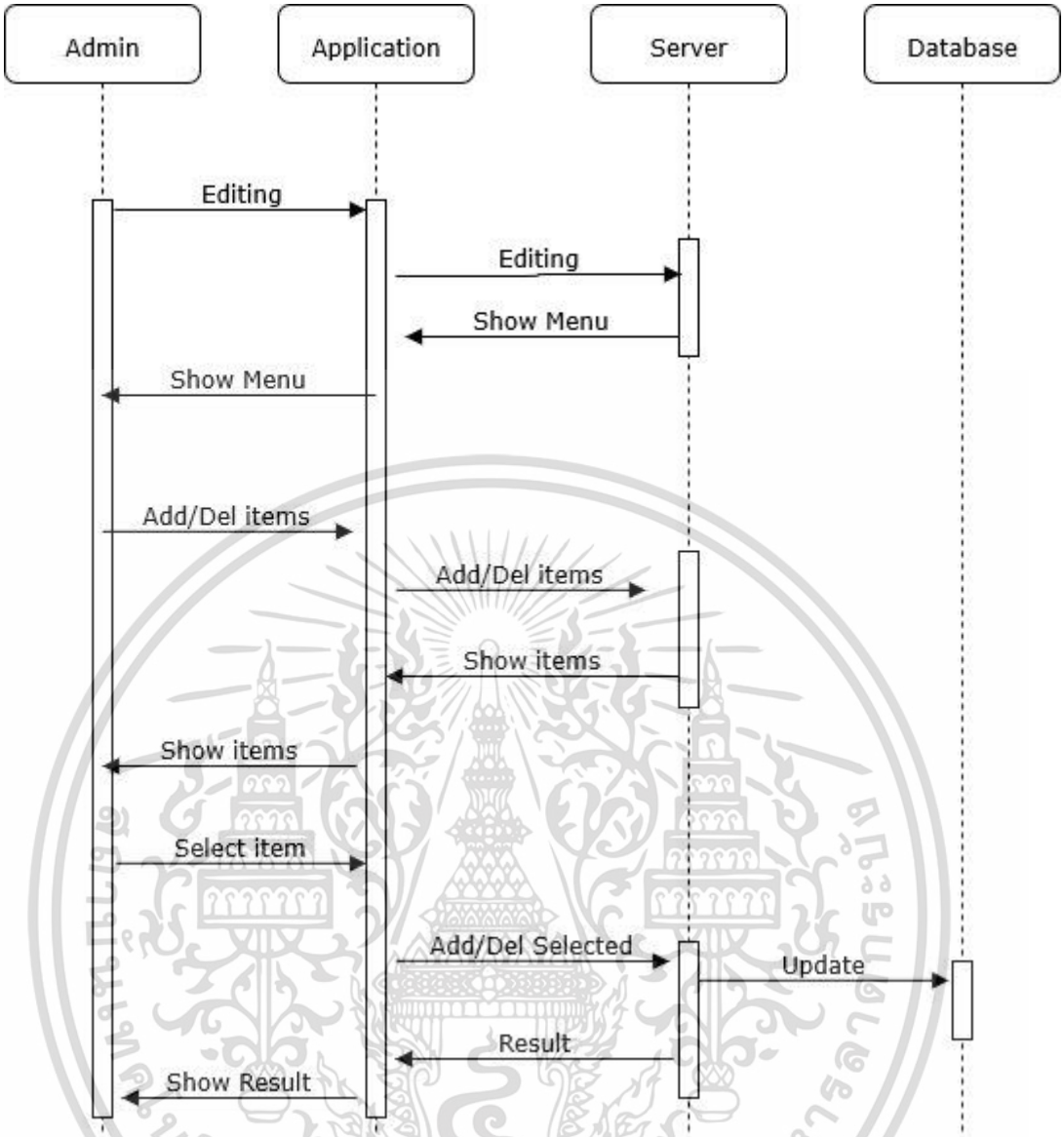
ID	Use case
Title	ดูสถิติประวัติการซื้อของสินค้า
Description	ผู้ใช้สามารถดูประวัติการซื้อขายสินค้าได้
Actor(s)	Admin
Main Flow	Admin สามารถเข้าไปตรวจสอบสถิติต่าง ๆ ของแต่ละช่วงเวลา ว่าเดือนไหนสินค้าตัวไหนมียอดขายดี ช่วงเวลาไหนมีกำไรหรือขาดทุน
Exceptional Flow	

3.3 Sequence Diagram



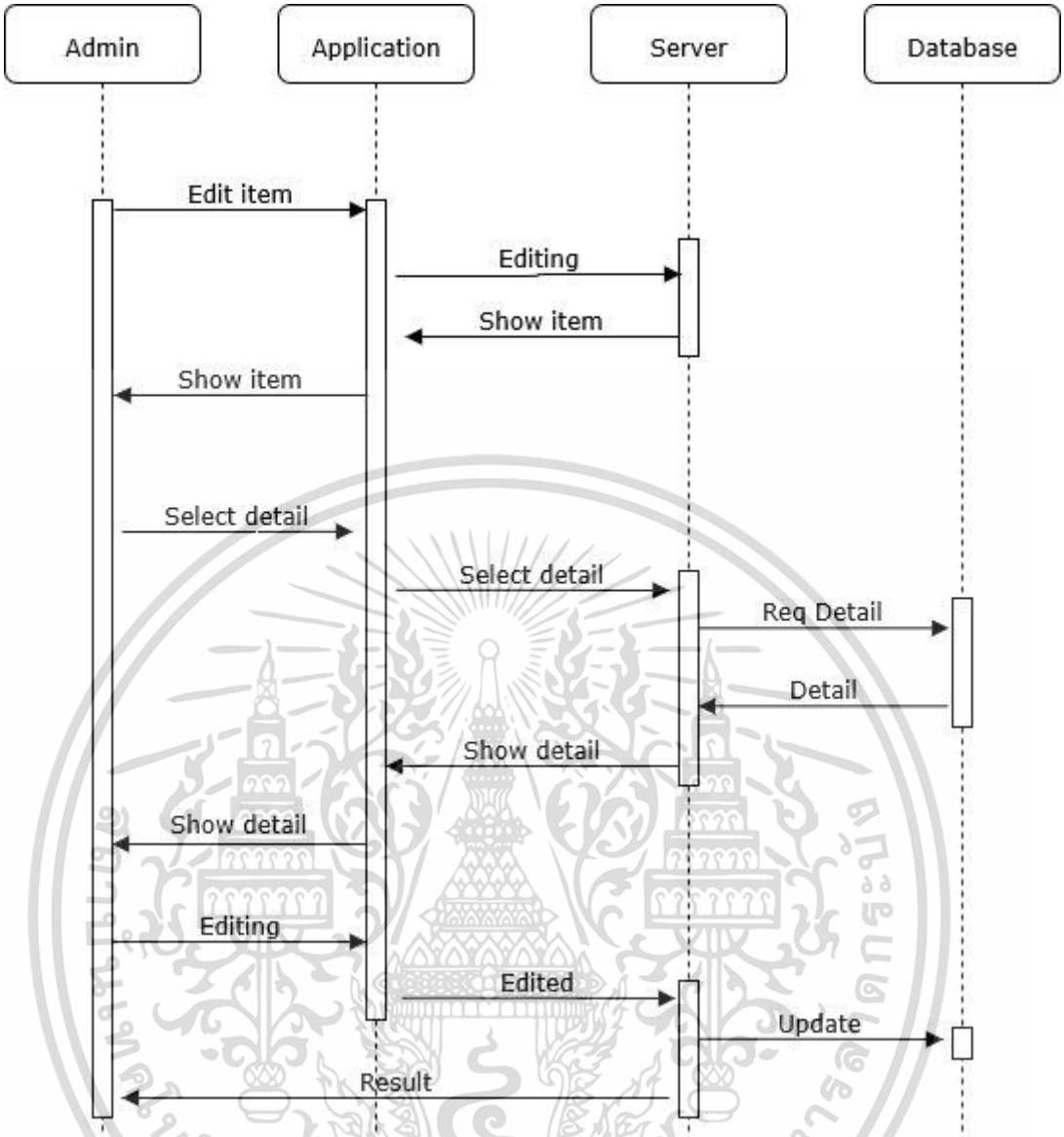
รูป 3.2 ลำดับการซื้อขายของ Customer

การซื้อขายของลูกค้าจะเริ่มต้นจากการสัมผัสหน้าจอของอุปกรณ์จากนั้นอุปกรณ์จะแสดงข้อความแจ้งเตือนขั้นตอนการใช้งานและข้อควรระวังให้ผู้ใช้ทราบผู้ใช้ต้องนำสินค้าที่ต้องการชำระเงินมาวางไว้ตามข้อกำหนดและกดยืนยันอุปกรณ์จะแสดงช่องทางการชำระเงินให้ผู้ใช้ทราบและหลังจากนั้นระบบจะส่ง QR code ให้อุปกรณ์และอุปกรณ์จะแสดง QR code ขึ้นมาให้ผู้ใช้ Scan หลังจากนั้นเมื่อผู้ใช้ดำเนินการต่อจะ Update รายการไปยัง Database



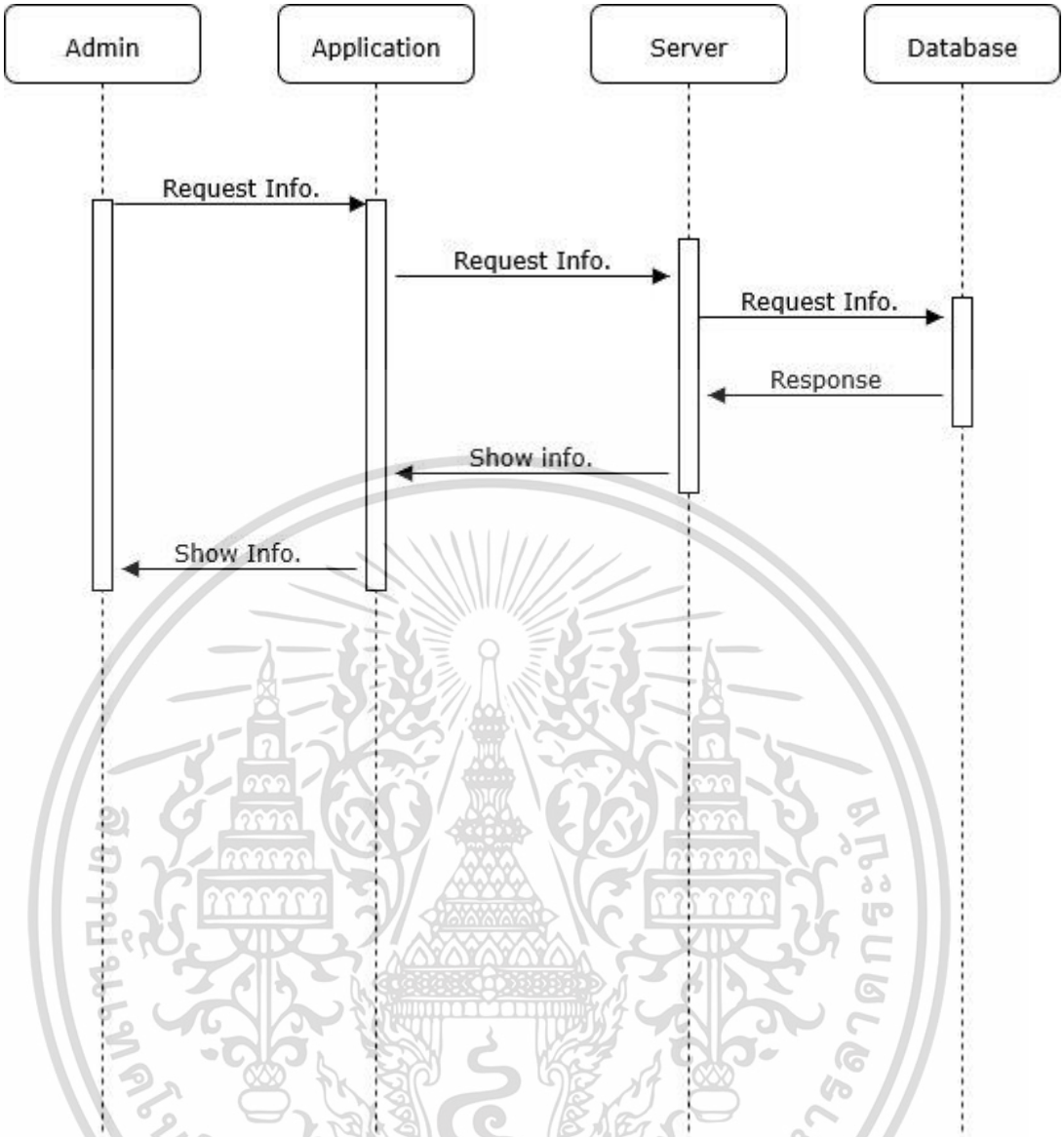
รูป 3.3 ลำดับการเพิ่ม/ลบสินค้า

เมื่อ Admin ต้องการแก้ไขเพิ่มหรือลบสินค้าออกจากระบบสามารถทำได้โดยการเลือกตัวเลือกการแก้ไขปรับแต่งบนอุปกรณ์และหลังจากนั้นอุปกรณ์จะส่งคำร้องให้เซิร์ฟเวอร์เพื่อแสดงรายการสินค้าให้ Admin เพิ่มหรือลบในระบบเมื่อทำรายการเสร็จจะอัพเดทความเปลี่ยนแปลงในฐานข้อมูลและแจ้งผลการเปลี่ยนแปลงบนหน้าอุปกรณ์



รูป 3.4 ลำดับการแก้ไขข้อมูลสินค้าในฐานข้อมูล

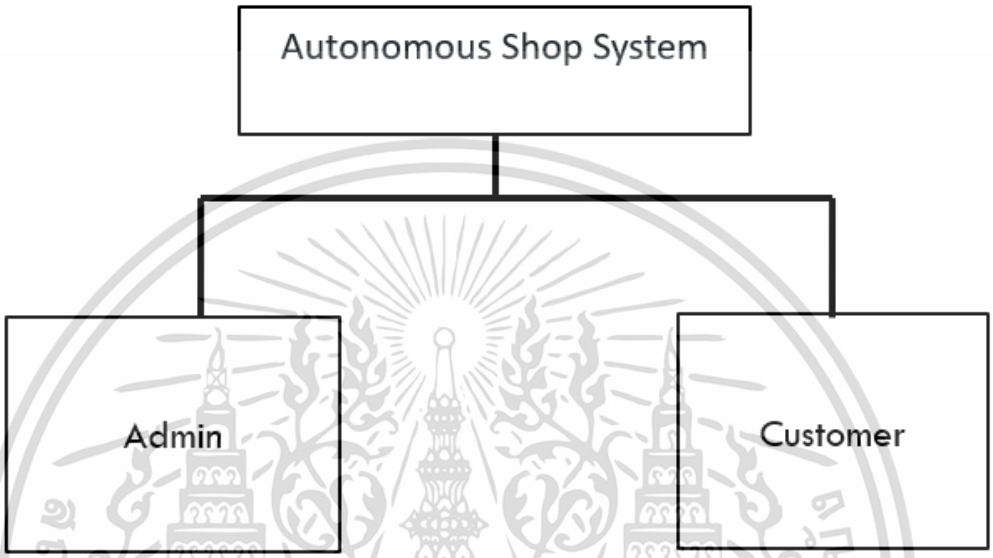
Admin สามารถแก้ไขข้อมูลสินค้าที่อยู่ในระบบได้ด้วยหน้าต่างแก้ไขสินค้าบนหน้าจออุปกรณ์ อุปกรณ์จะแสดงรายการสินค้าในระบบให้ผู้ใช้ได้เลือกที่จะแก้ไขหลังจากนั้นอุปกรณ์จะเรียกขอข้อมูลจากฐานข้อมูลและแสดงให้ผู้ใช้ได้แก้ไขข้อมูลและจะอัพเดทความเปลี่ยนแปลงไปยังฐานข้อมูลหลังจากนั้นจะส่งผลลัพธ์ออกทางหน้าจออุปกรณ์



รูป 3.5 ลำดับการแสดงข้อมูลสถิติการซื้อขาย

Admin สามารถเรียกดูข้อมูลสถิติการซื้อขายจากแอพพลิเคชั่น Admin โดยจะติดต่อฐานข้อมูล และฐานข้อมูลจะส่งข้อมูลสถิติมาแสดงบนแอพพลิเคชั่นตามลำดับ

3.4 Modules Design



รูป 3.6 โครงสร้างของโปรแกรม

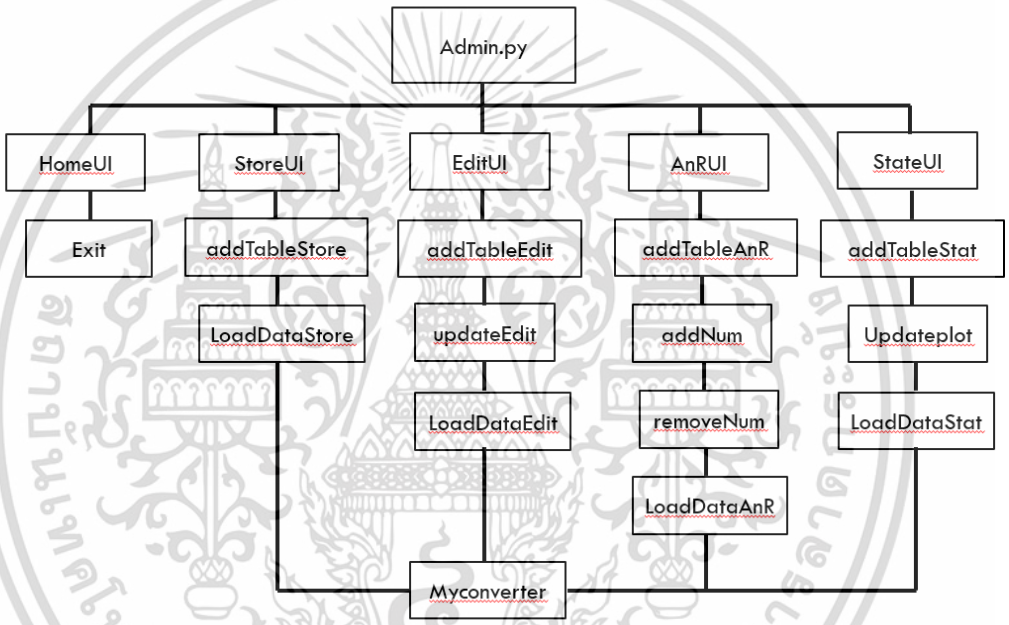
โดยโปรแกรมAutonomous Shop System จะแบ่งเป็น 2 ส่วน โดยมีส่วนโดยจะแบ่งเป็น 2 ส่วน โดยมีส่วน Admin และ ส่วน Customer

สำหรับโปรแกรมส่วน Admin นั้นมีไว้สำหรับดูยอดขาย ปรับราคาสินค้าในคลัง แก้ไขจำนวนสินค้าในคลัง

ดูสินค้าในคลัง ตรวจสอบค่าสถิติต่างๆ ซึ่งติดตั้งบน Windows สำหรับเครื่องของ Admin

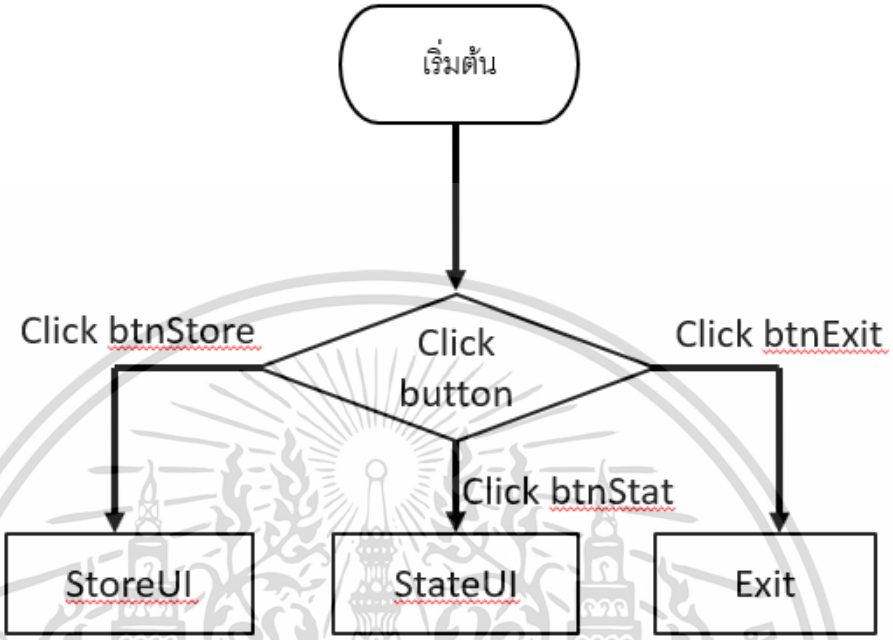
สำหรับโปรแกรมส่วน Customer นั้นมีไว้สำหรับชำระสินค้าอัตโนมัติสำหรับลูกค้าในร้านสะดวกซื้อซึ่งติดตั้งบน เครื่อง Nvidia Jetson Nano บนระบบปฏิบัติการ Linux ติดตั้งไว้ในตู้ชำระสินค้าอัตโนมัติ

3.4.1 Module Admin



รูป 3.7 Modules ต่างๆภายใน Admin.py

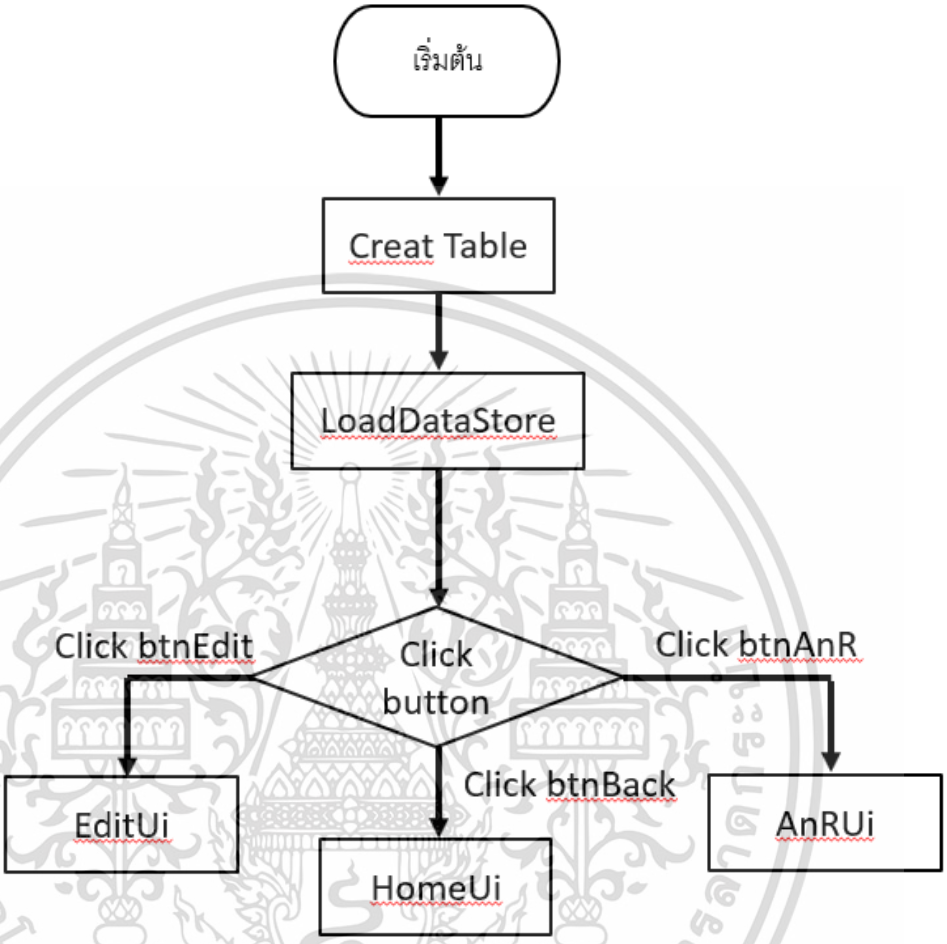
3.4.1.1 Module HomeUi



รูป 3.8 Modules ต่างๆภายใน Admin.py

Module HomeUi ทำหน้าที่เป็นหน้าเริ่มต้นของโปรแกรมโดยจะมีหน้าที่สำหรับ connect สู่หน้า StoreUi และหน้า StatUi และ Module exit เพื่อออกจากโปรแกรม

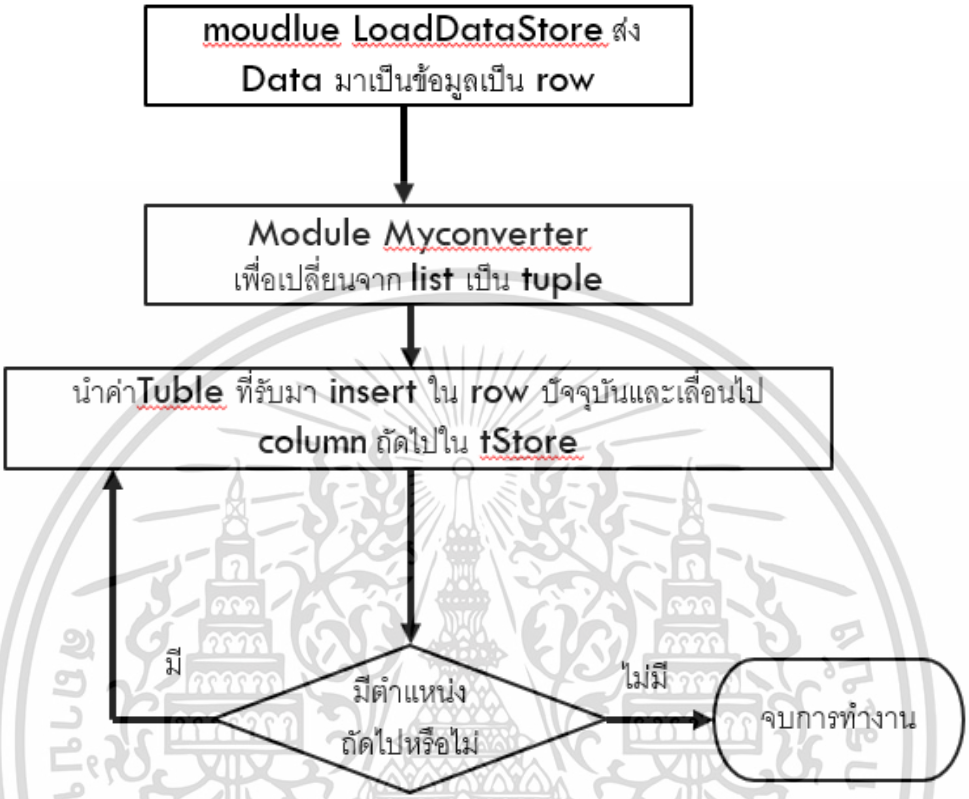
3.4.1.2 Module StoreUI



รูป 3.9 Flowchart StoreUI

Module StoreUi ทำหน้าที่สร้างตาราง (ID,Name,Cost,Sale,Store) เพื่อเรียกใช้งาน Module LoadDataStore เพื่อดึงข้อมูลแสดงลงในตารางและมีอีกหน้าที่คือเป็นเส้นทาง connect เข้าสู่หน้า EditUi เมื่อคลิกที่ปุ่ม btnEdit , เข้าสู่หน้า AnRUi เมื่อคลิกที่ปุ่ม btnAnR และ HomeUi เมื่อคลิกที่ปุ่ม btnBack

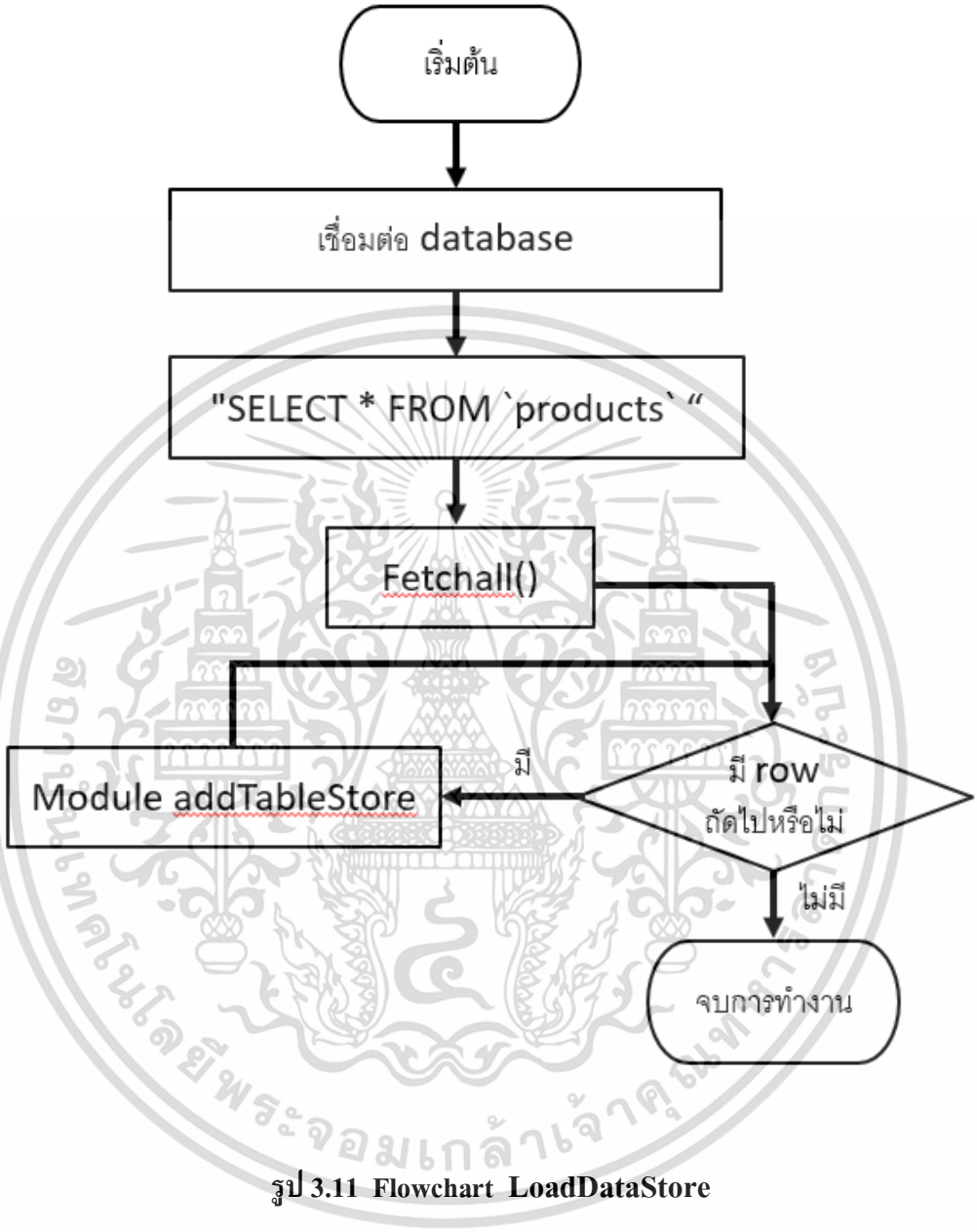
3.4.1.3 Module addTableStore



รูป 3.10 Flowchart addTableStore

Module addTableStore มีหน้าที่ในการรับค่าจาก Module LoadDataStore ที่ผ่านการเรียกใช้งาน Module Myconverter ส่ง tuple มา Insert ลงในตาราง tStore โดยมี input เป็น Data

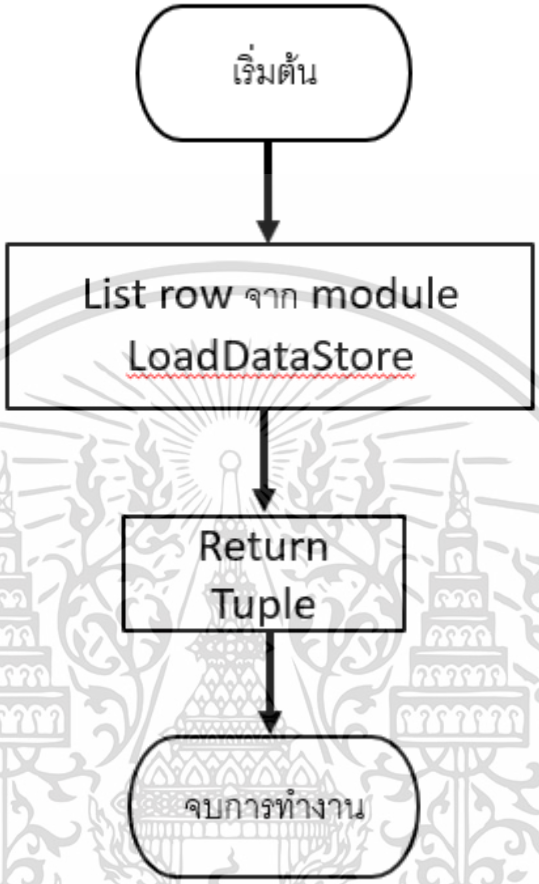
3.4.1.4 Module LoadDataStore



รูป 3.11 Flowchart LoadDataStore

Module LoadDataStore เชื่อมต่อกับ Database ด้วย pymysql ต่อมาใช้คำสั่ง Fetchall ของ Query "SELECT * FROM `products`" และเรียกใช้งาน Module addTableStore เพื่อ Insert data ลงในตาราง tStore

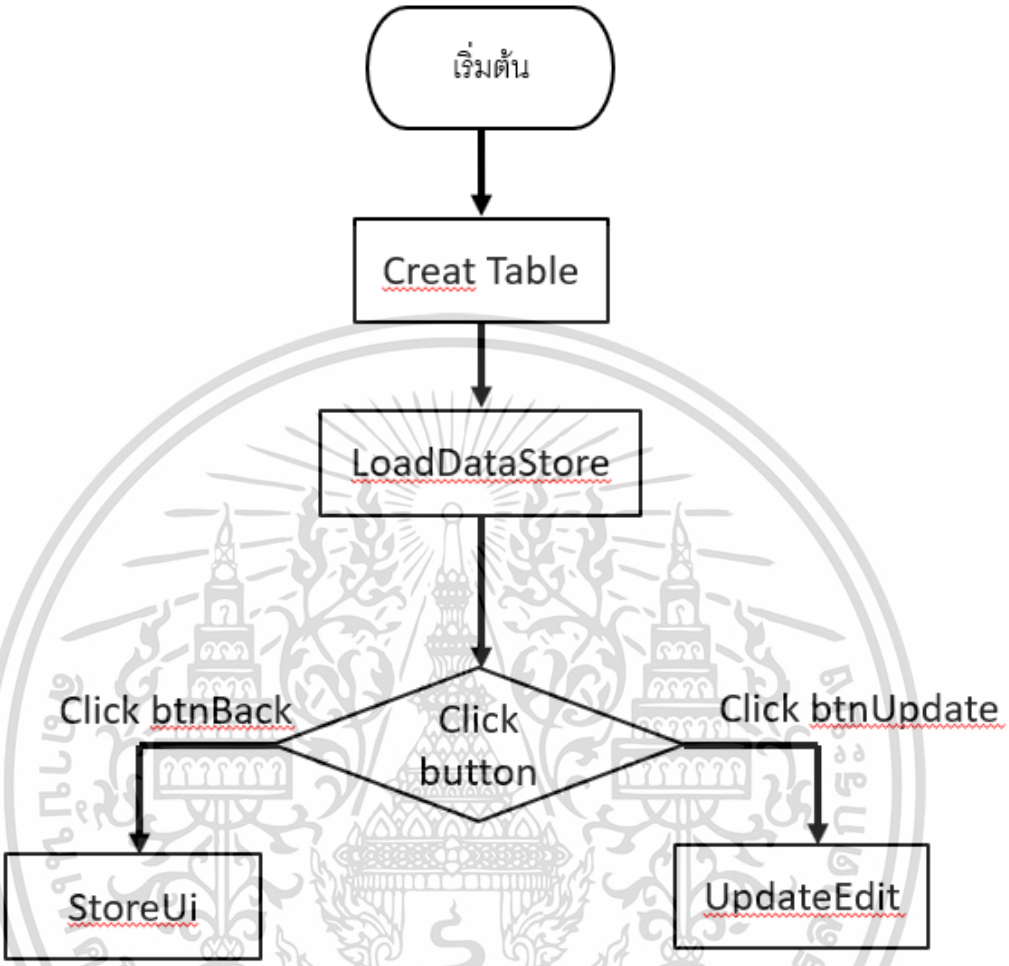
3.4.1.5 Module Myconverter



รูป 3.12 Flowchart Myconverter

Module Myconverter มีหน้าที่รับค่า row ที่เป็น list มาจาก Module LoadDataStore มา map เป็น tuple เพื่อความปลอดภัยของ Data จะไม่ทำให้เกิดข้อผิดพลาดอย่างเช่นถ้าเกิดมี input เป็น “"os.system ('rm -rf /')” เป็นต้น

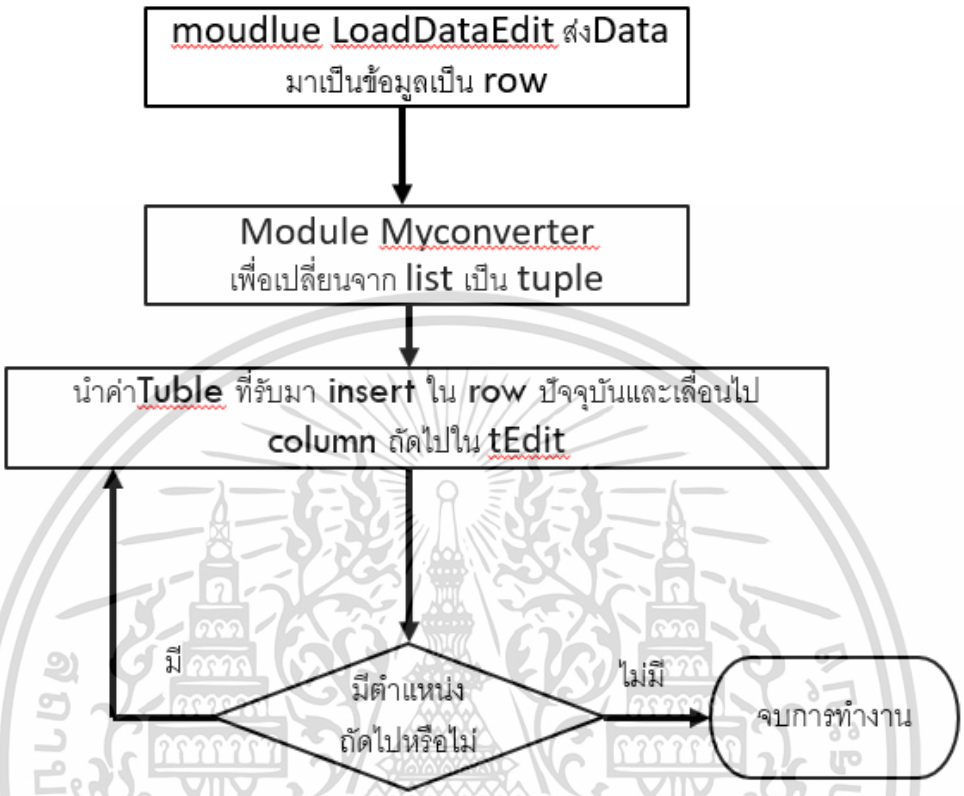
3.4.1.6 Module EditUi



รูป 3.13 Flowchart EditUi

Module EditUi ทำหน้าที่สร้างตาราง (Name,Cost,Sale) เพื่อเรียกใช้งาน Module LoadDataEdit เพื่อดึงข้อมูลแสดงลงในตารางและมีอีกหน้าที่คือเป็นเส้นทาง connect เข้าสู่หน้า StoreUi เมื่อคลิกที่ปุ่ม btnBack และทำการเรียก Module UpdateEdit เมื่อคลิกที่ btnUpdate

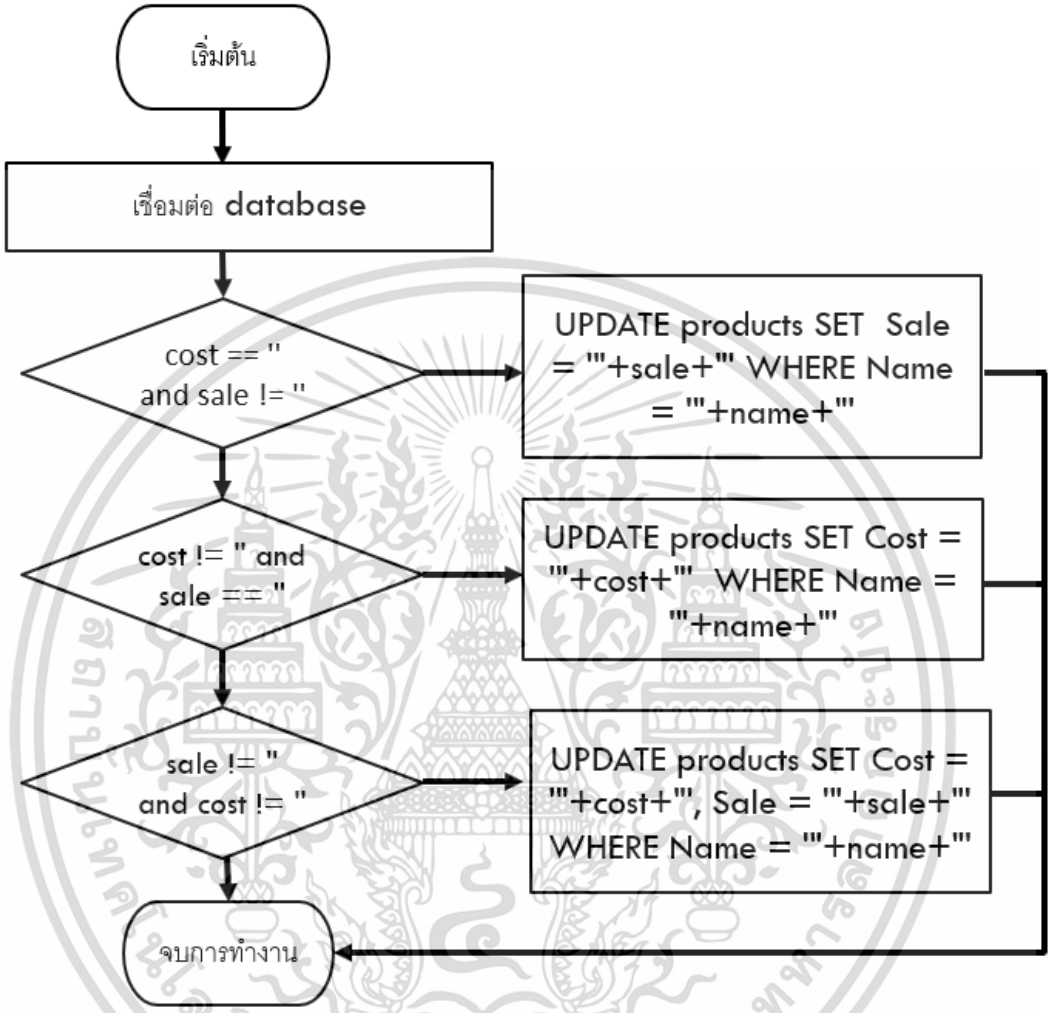
3.4.1.7 Module addTableEidt



รูป 3.14 Flowchart addTableEdit

Module addTableEidt มีหน้าที่ในการรับค่าจาก Module LoadDataEdit ที่ผ่านการเรียกใช้งาน Module Myconverter ส่ง tuple มา Insert ลงในตาราง tEdit โดยมี input เป็น Data

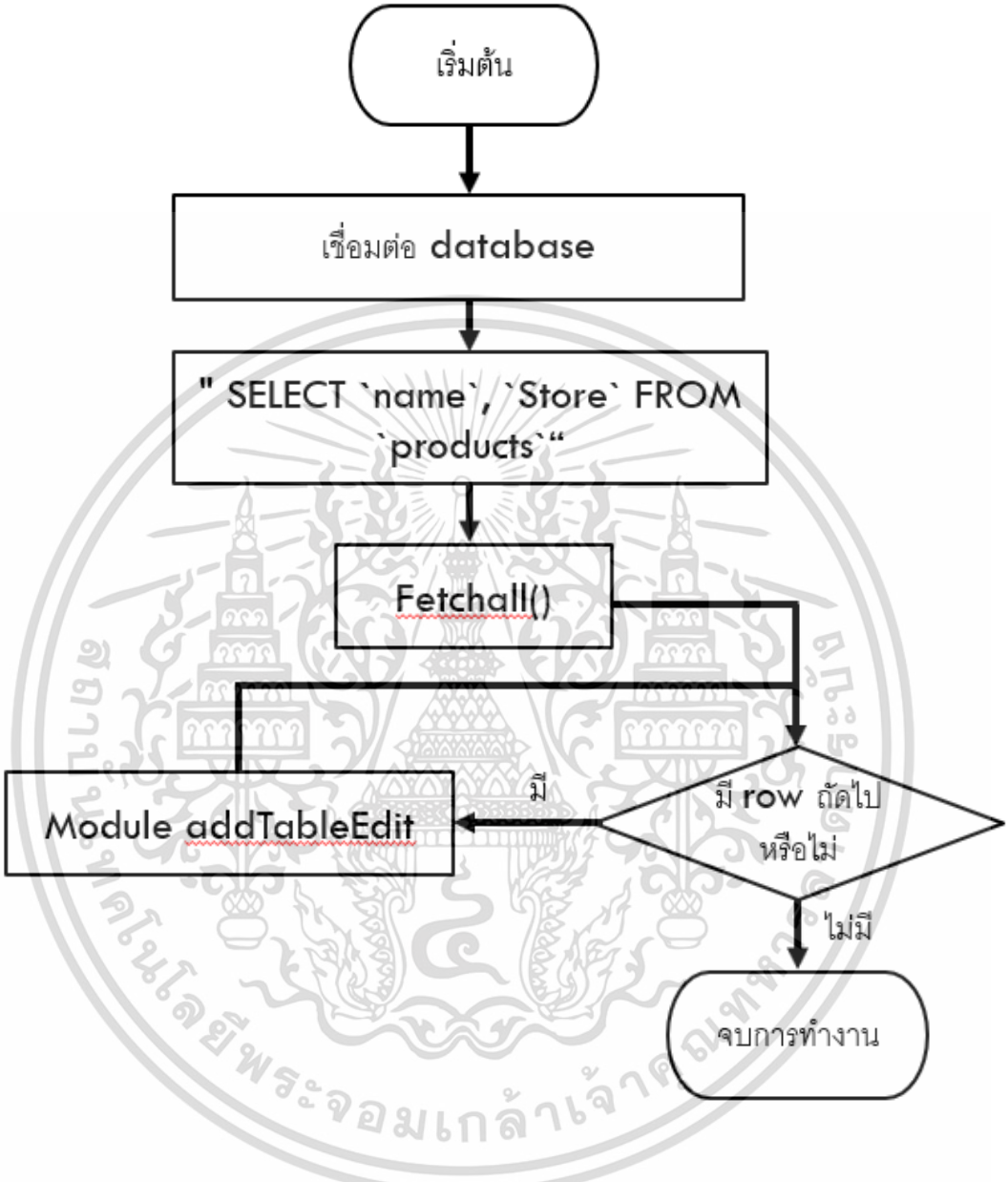
3.4.1.8 Module UpdateEdit



รูป 3.15 Flowchart UpdateEdit

Module UpdateEdit มีหน้าที่เมื่อเลือกname และ กรอกค่าลงในช่องที่ต้องการเปลี่ยนแปลงแล้ว คลิกที่ปุ่ม btnUpdate จะทำการ Update ค่าต่างๆลงใน ฐานข้อมูล

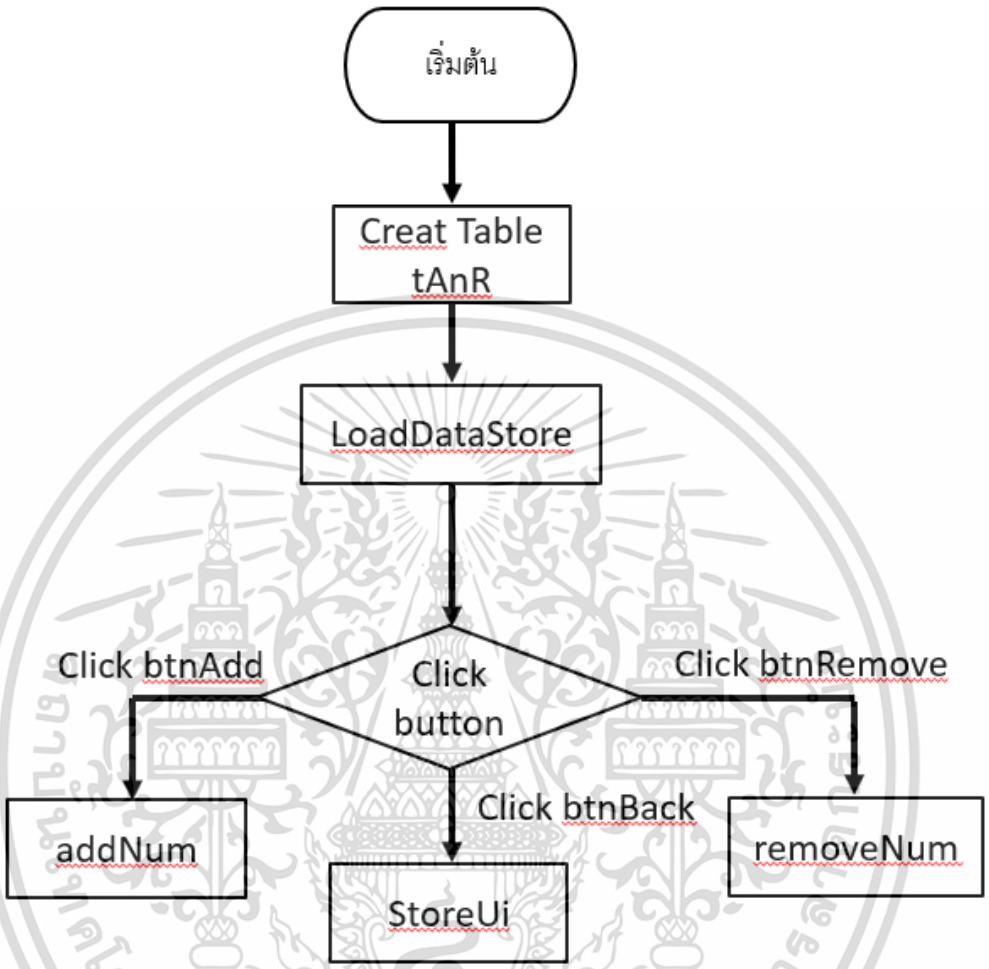
3.4.1.9 Module LoadDataEdit



รูป 3.16 Flowchart LoadDataEdit

Module LoadDataEdit เชื่อมต่อกับ Database ด้วย pymysql ต่อมาใช้คำสั่ง Fetchall ของ Query "SELECT `name`, `cost`, `sale` FROM `products` " และเรียกใช้งาน Module addTableEdit เพื่อ Insert data ลงในตาราง tEdit

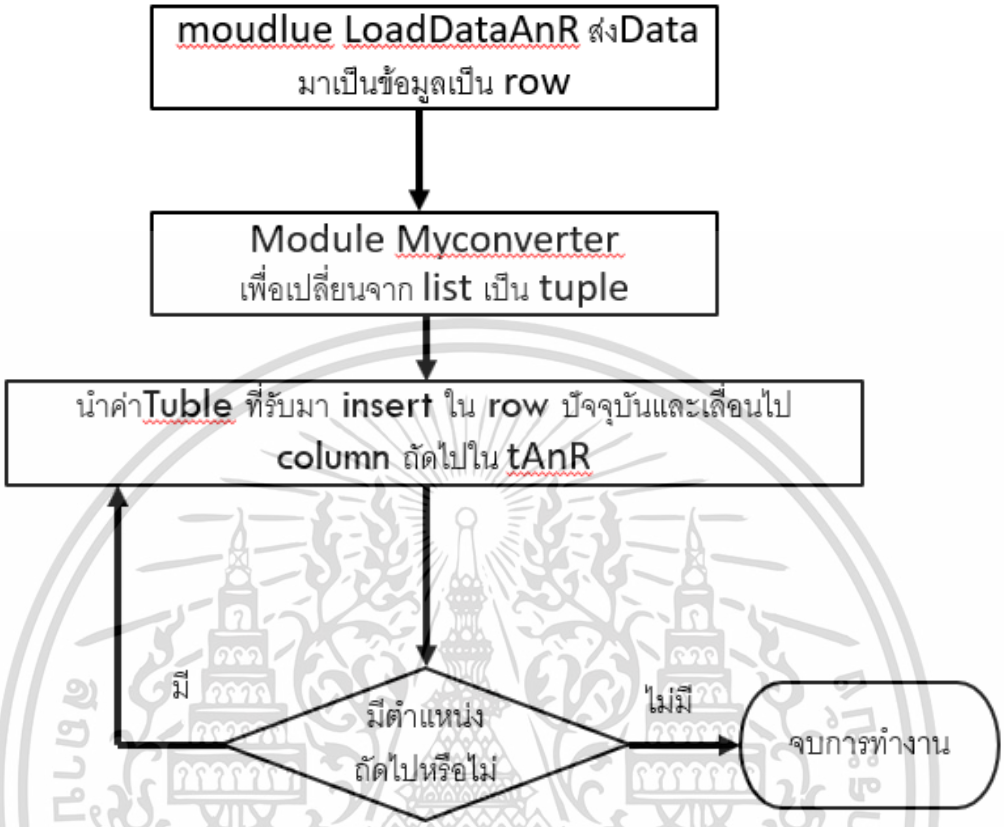
3.4.1.10 Module AnRUi



รูป 3.17 Flowchart AnRUi

Module AnRUi ทำหน้าที่สร้างตาราง (Name,Store) เพื่อเรียกใช้งาน Module LoadDataAnR เพื่อดึงข้อมูลแสดงลงในตารางและมีอีกหน้าที่คือเป็นเส้นทาง connect เข้าสู่หน้า StoreUi เมื่อคลิกที่ปุ่ม btnBack, ทำการเรียก Module addNum เมื่อคลิกที่ปุ่ม btnAdd และทำการเรียก Module removenum เมื่อคลิกที่ปุ่ม btnRemove

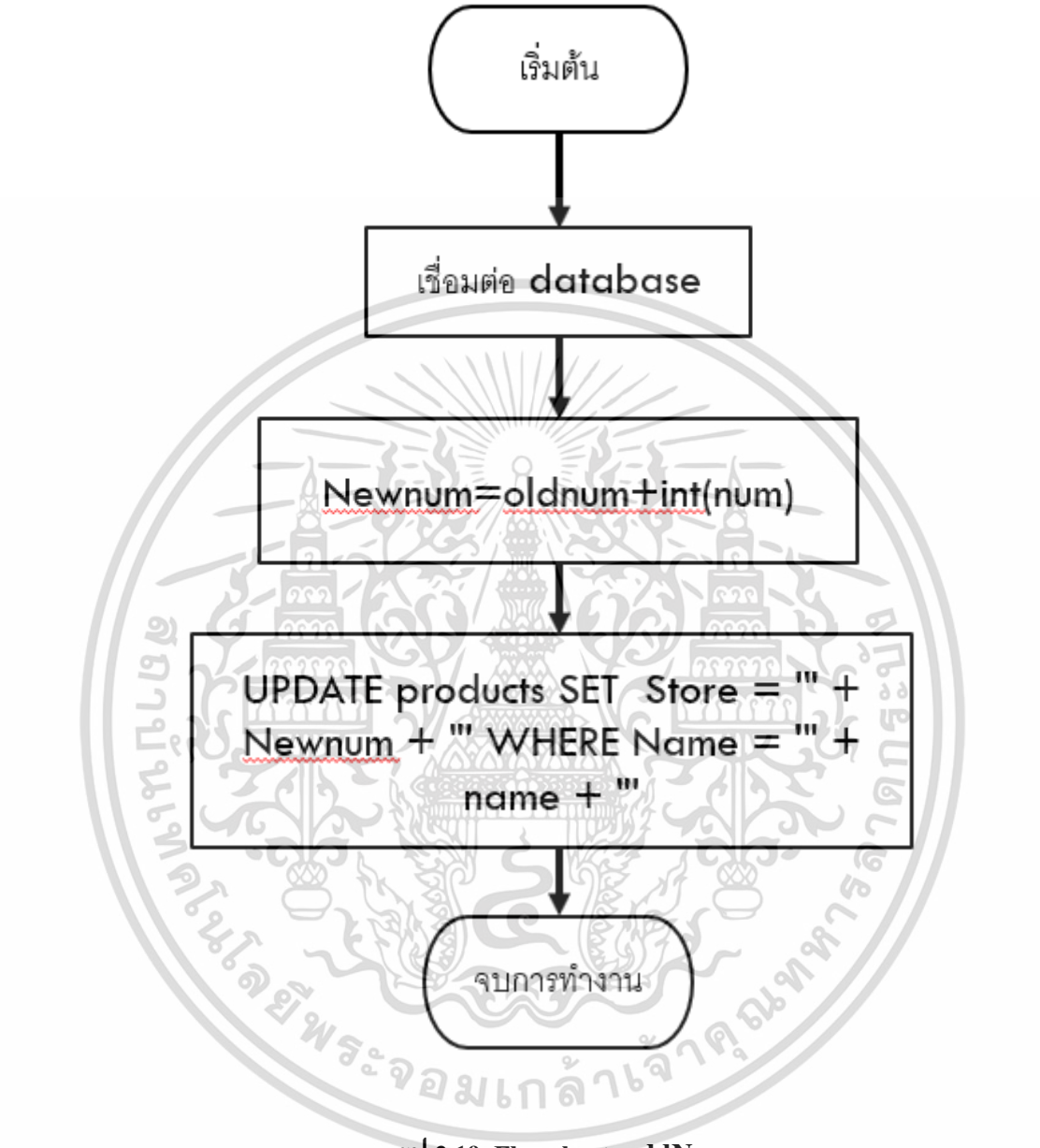
3.4.1.11 Module addTableAnR



รูป 3.18 Flowchart addTableAnR

Module addTableAnR มีหน้าที่ในการรับค่าจาก Module LoadDataAnR ที่ผ่านการเรียกใช้งาน Module Myconverter ส่ง tuple มา Insert ลงในตาราง tAnR โดยมี input เป็น Data

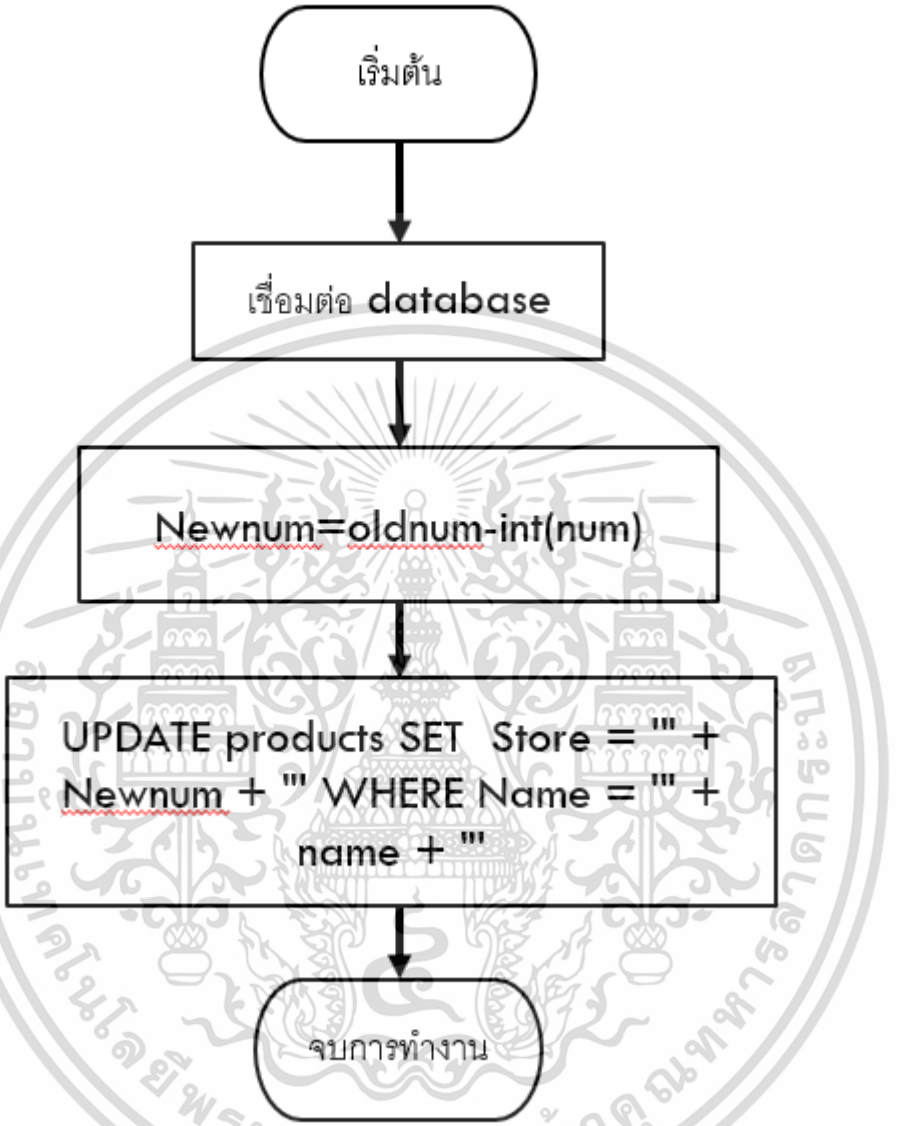
3.4.1.12 Module addNum



รูป 3.19 Flowchart addNum

Module addNum มีหน้าที่เมื่อเลือกname และ จำนวนสินค้าที่ต้องการเพิ่มแล้ว คลิกที่ปุ่ม btnAdd จะทำการ Update ค่าต่างๆลงใน ฐานข้อมูลโดยการทำการหาค่าเดิมและบวกกับจำนวนที่ต้องการจะเพิ่มแล้วเปลี่ยนแปลง

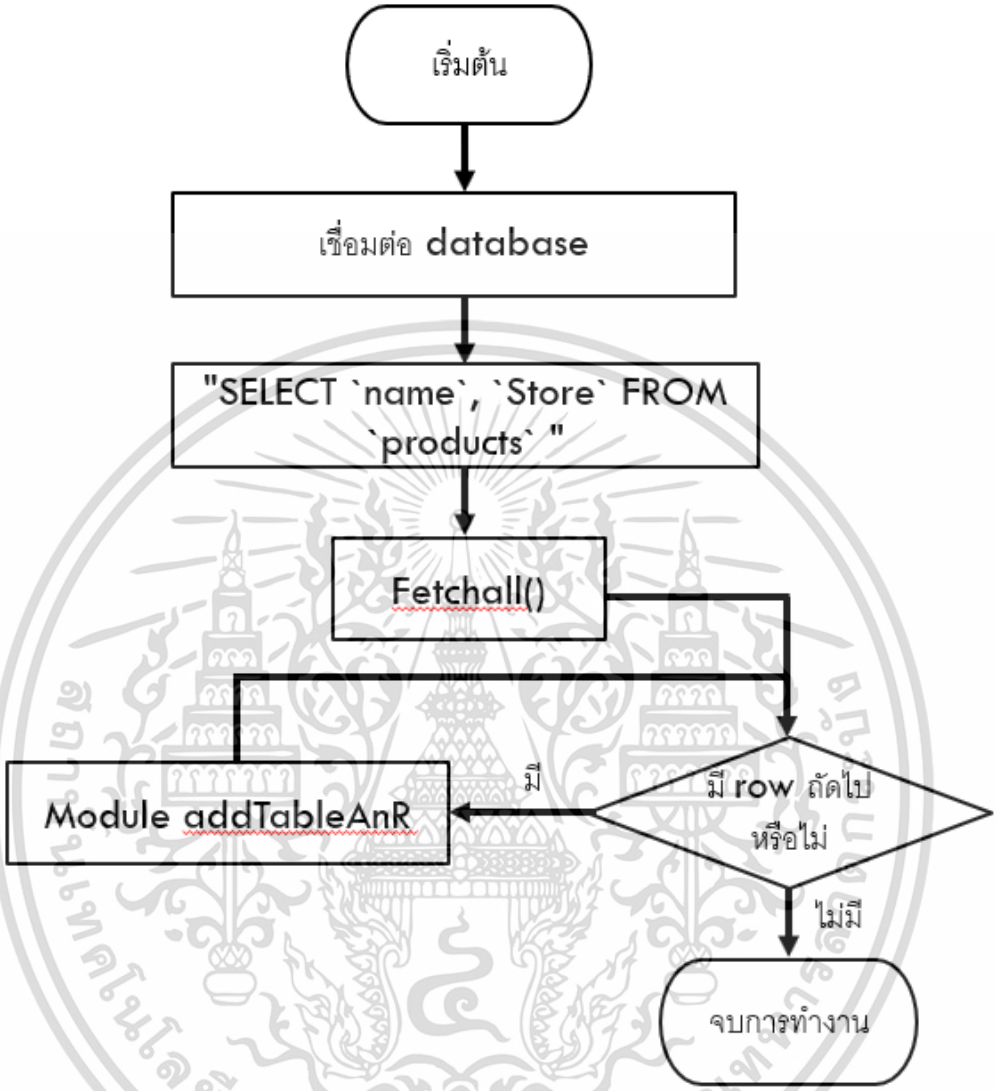
3.4.1.13 Module removeNum



รูป 3.20 Flowchart removeNum

Module removeNum มีหน้าที่เมื่อเลือก name และ จำนวนสินค้าที่ต้องการลดแล้ว คลิกที่ปุ่ม btnRemove จะทำการ Update ค่าต่างๆลงใน ฐานข้อมูลโดยการทำการหาค่าเดิมและลบกับจำนวนที่ต้องการจะลดแล้วเปลี่ยนแปลง

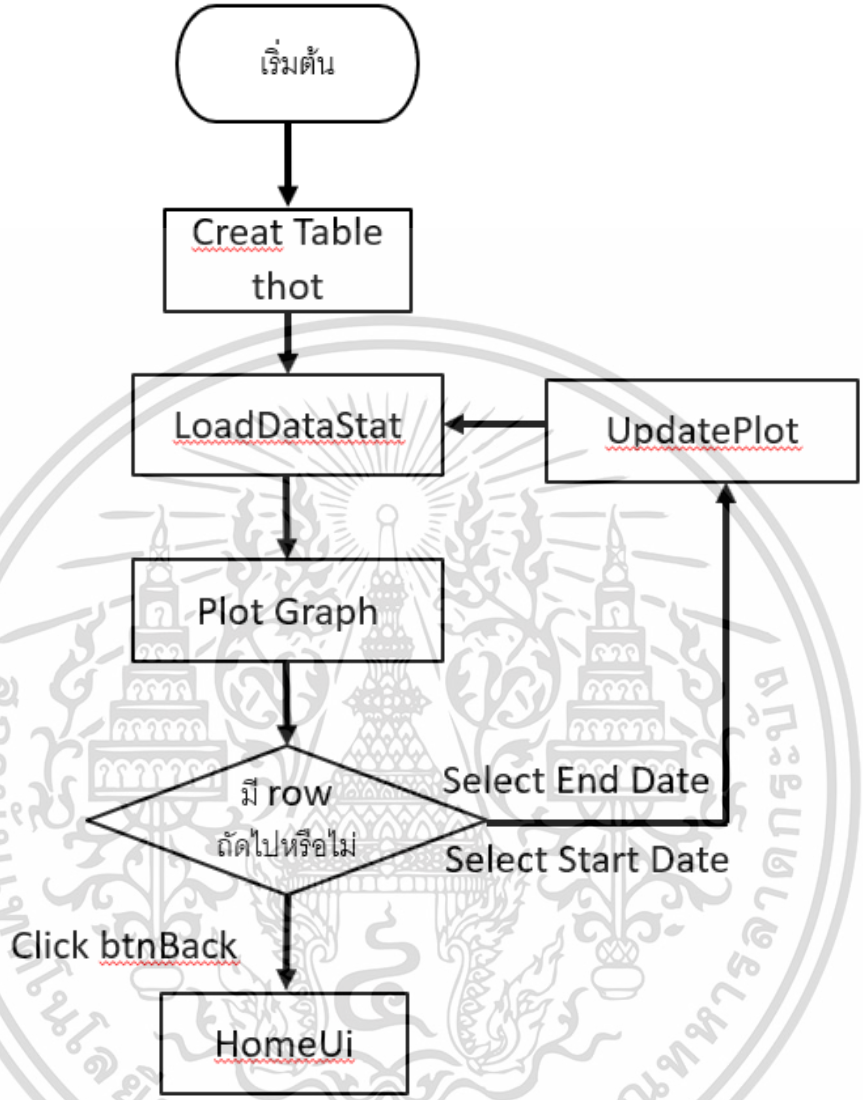
3.4.1.14 Module LoadDataAnR



รูป 3.21 Flowchart LoadDataAnR

Module LoadDataAnR เชื่อมต่อกับ Database ด้วย pymysql ต่อมาใช้คำสั่ง Fetchall ของ Query "SELECT `name`, `Store` FROM `products` "และเรียกใช้งาน Module addTableAnR เพื่อ Insert data ลงในตาราง tAnR

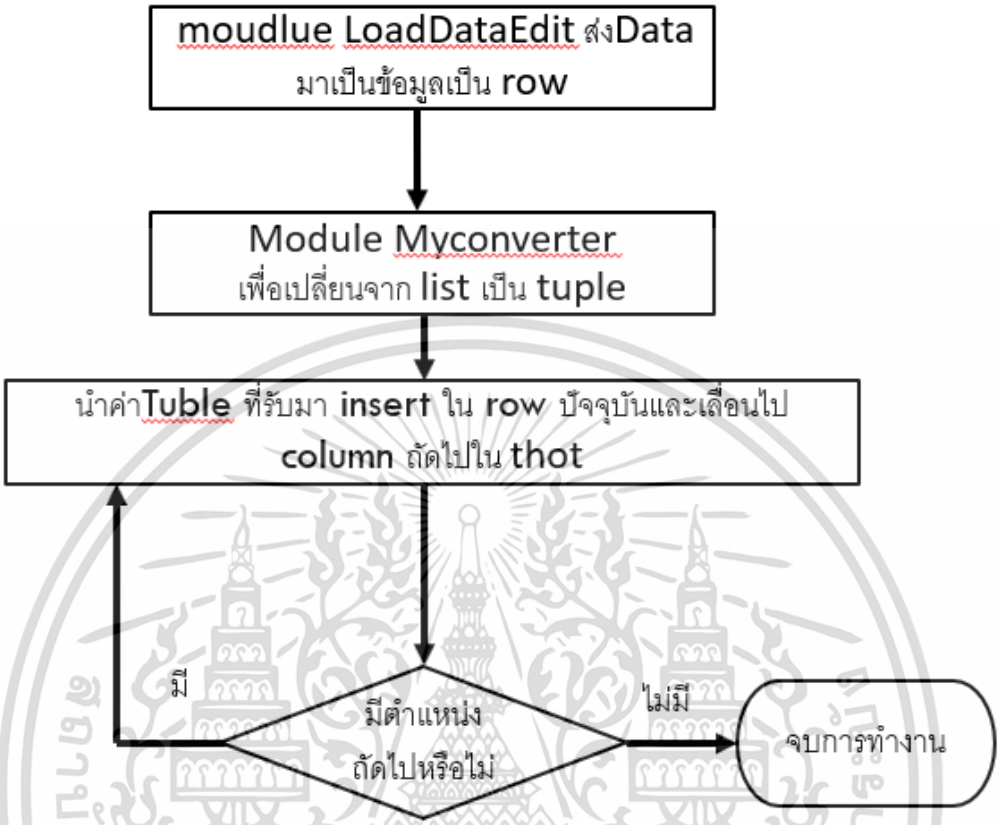
3.4.1.15 Module StatUi



รูป 3.22 Flowchart StatUi

Module StatUi ทำหน้าที่สร้างตาราง (Name,pices) เพื่อเรียกใช้งาน Module LoadDataStat เพื่อดึงข้อมูลแสดงลงในตาราง แสดงกราฟที่ polt มาจาก LoadDataStat แสดงค่า profit,total มุมขวาบน และมีหน้าที่คือเป็นเส้นทาง connect สู่หน้า HomeUi เมื่อคลิกที่ปุ่ม btnBack และเมื่อเปลี่ยนแปลงวันในปฏิทินจะทำการเรียก ModuleUpdatePlot เพื่อลบกราฟและทำการPlot กราฟขึ้นมาใหม่

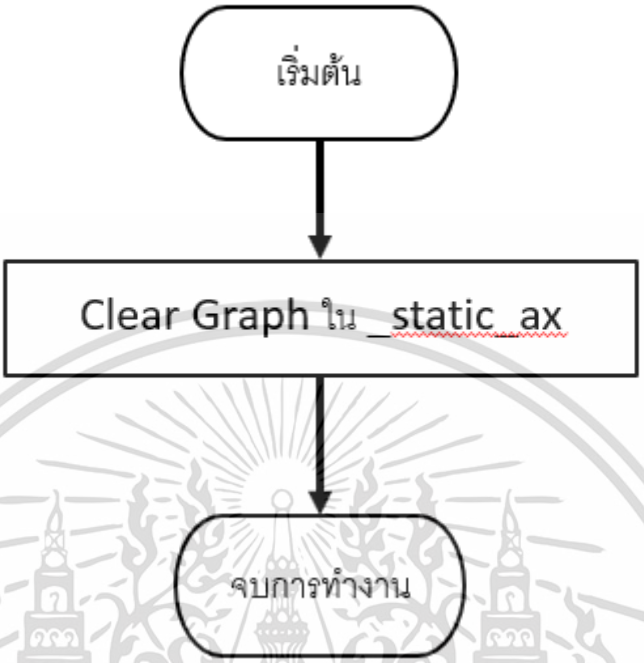
3.4.1.16 Module addTableStat



รูป 3.23 Flowchart addTableStat

Module addTableStat มีหน้าที่ในการรับค่าจาก Module LoadDataStat ที่ผ่านการเรียกใช้งาน Module Myconverter ส่ง tuple มา Insert ลงในตาราง thot โดยมี input เป็น Data

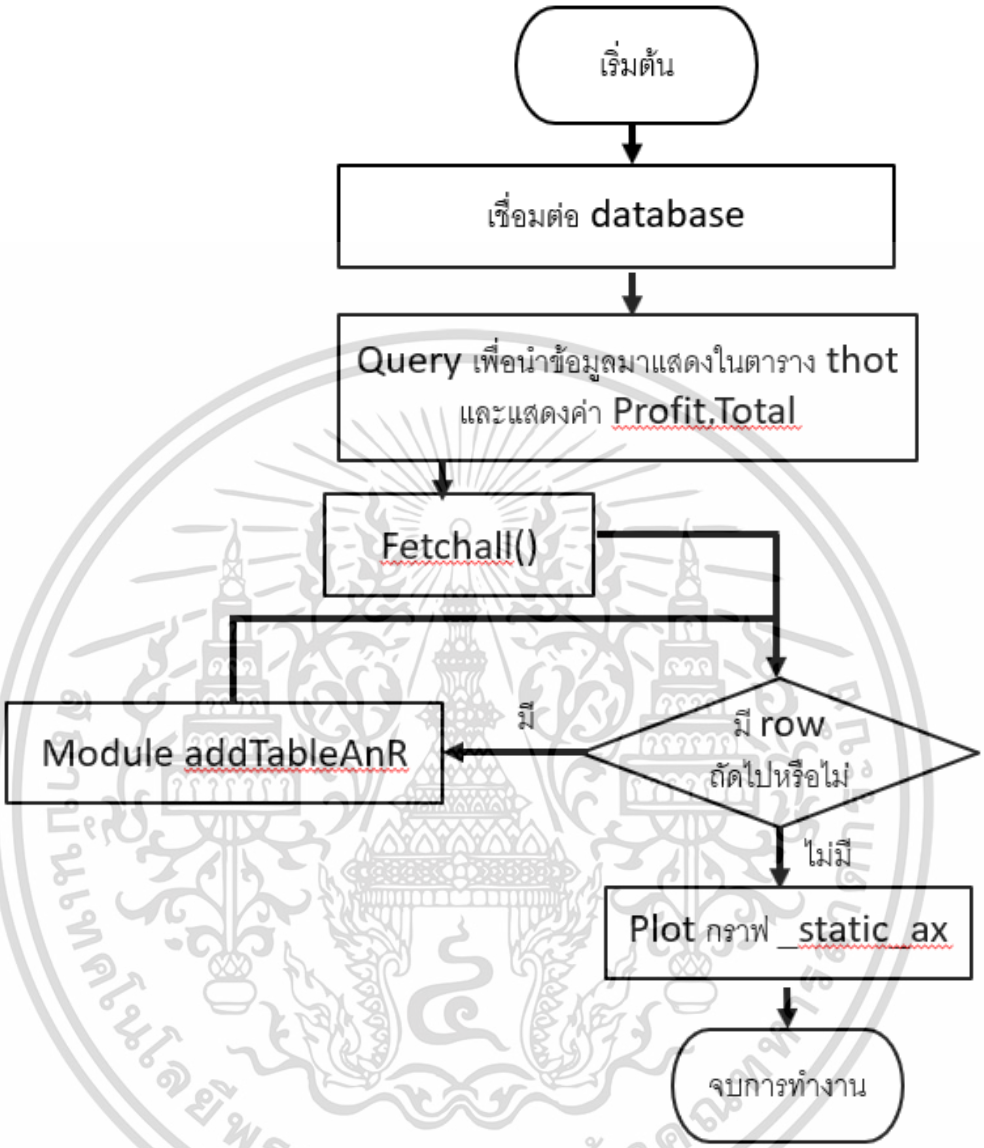
3.4.1.17 Module updateplot



รูป 3.24 Flowchart updateplot

Module updateplot มีหน้าที่ลบกราฟที่อยู่ใน _static_ax

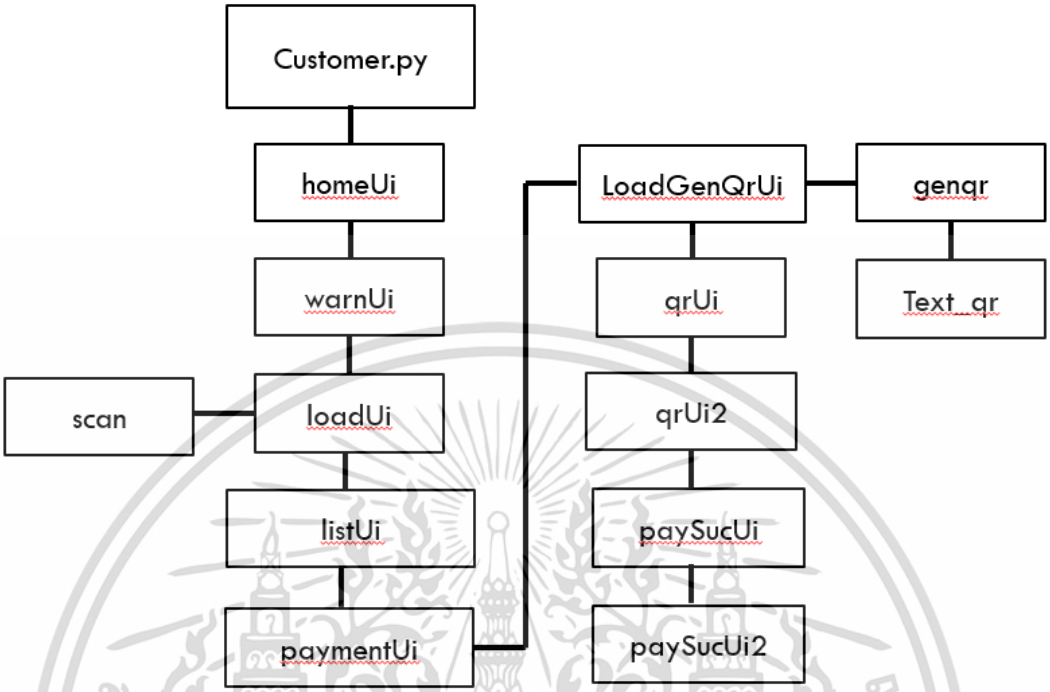
3.4.1.18 Module LoadDataStat



รูป 3.25 Flowchart LoadDataStat

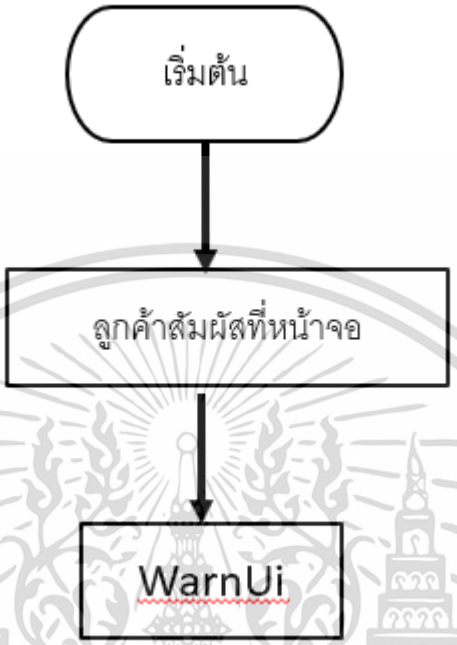
Module LoadDataStat เชื่อมต่อกับ Database ด้วย pymysql ต่อมาใช้คำสั่ง Fetchall ของ Query เพื่อนำข้อมูลของตาราง thot มาแสดง,Query เพื่อนำค่าของ profit ,total มาแสดง,Query เพื่อนำข้อมูลที่ได้มา plot กราฟ _static_ax เพื่อดูว่าแนวโน้มของกำไรไปในทิศทางไหน

3.4.2 Module Customer



รูป 3.26 Modules ต่างๆภายใน Customer.py

3.4.2.1 Module homeUi

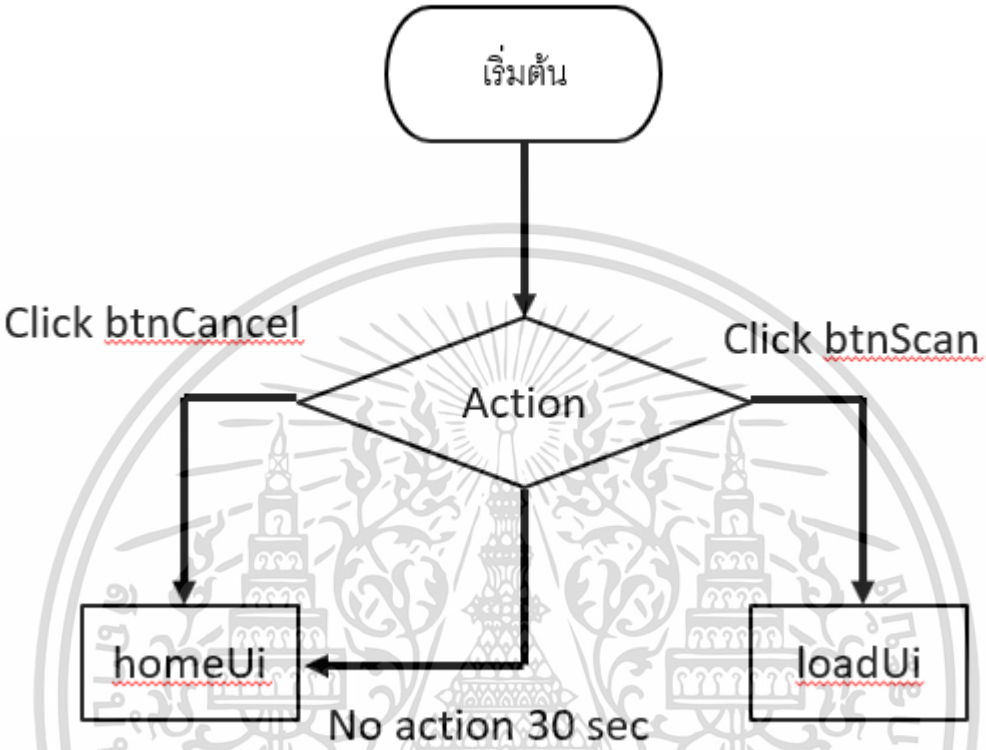


รูป 3.27 Flowchart homeUi

Module homeUi ทำหน้าที่เป็นหน้าแรกของโปรแกรมในส่วนของ Customer โดยจะทำการรอให้ลูกค้าที่ต้องการชำระเงินอัตโนมัติมาสัมผัสที่หน้าจอเพื่อเข้าสู่

Module warnUi

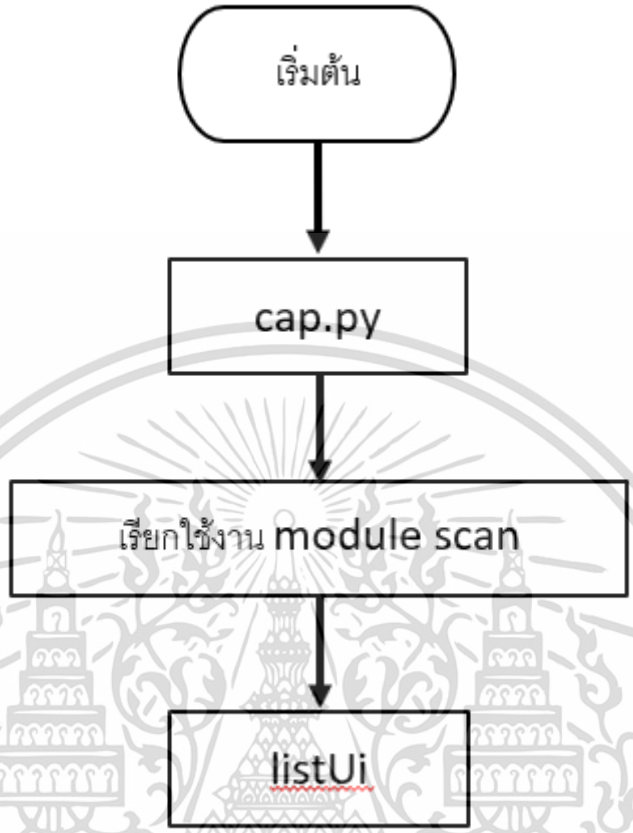
3.4.2.2 Module warnUi



รูป 3.28 Flowchart warnUi

Module warnUi ทำหน้าที่แสดง ข้อควรระวังให้แก่ผู้ใช้งานเพื่อลดความผิดพลาดของการทำงานโดยเมื่อคลิกที่ปุ่ม SCAN จะไปเรียก Module LoadUi เพื่อดำเนินรายการต่อไป หรือเมื่อกดที่ปุ่ม CANCEL จะทำการย้อนกลับไปหน้า homeUi หรือเมื่อไม่กระทำการใดๆนาน30วินาทีจะกลับสู่หน้า homeUi

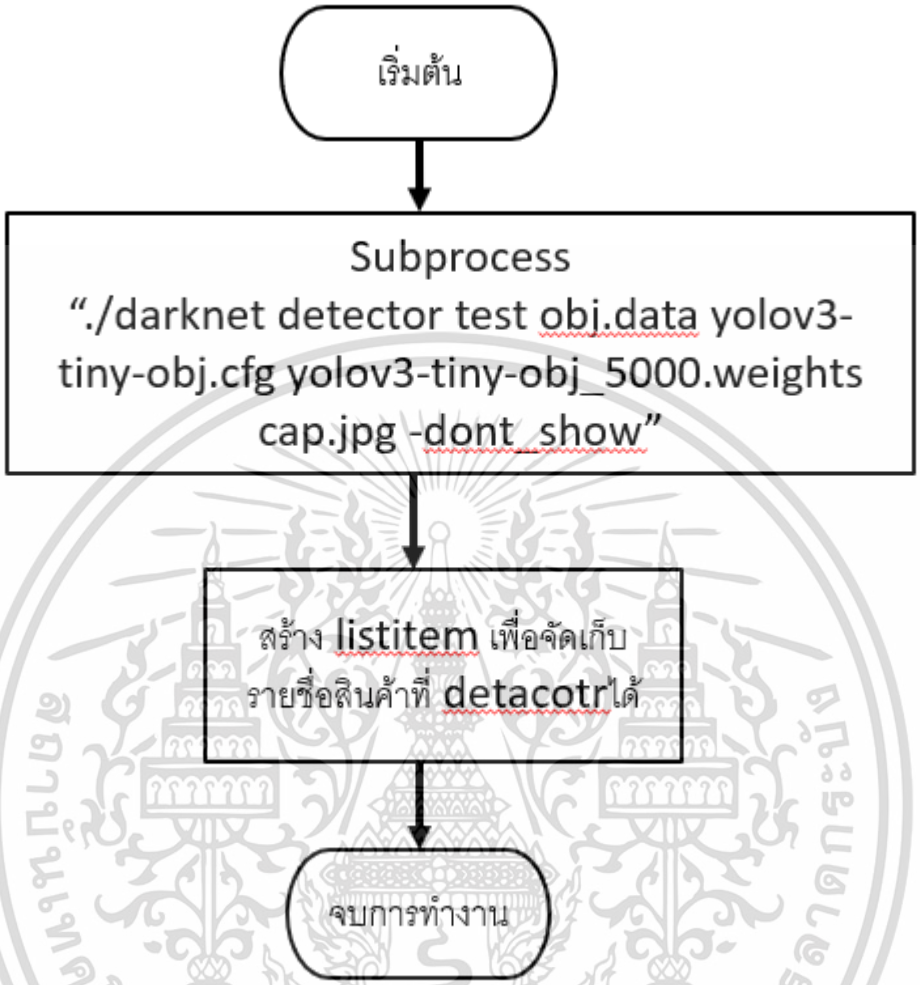
3.4.2.3 Module loadUi



รูป 3.29 Flowchart loadUi

Module loadUi มีหน้าที่แจ้งผู้ใช้งานว่ากำลัง loading ในเบื้องหน้าแต่เบื้องหลัง ทำการ subprocess ไปเรียก cap.py ให้ทำการบันทึกภาพสินค้าในช่องวางสินค้า ต่อมาก็ทำการ detector กับ model yolov3 ด้วย weight ที่เราเทรนมา และเข้าสู่ Module listUi โดยอัตโนมัติ

3.4.2.4 Module scan

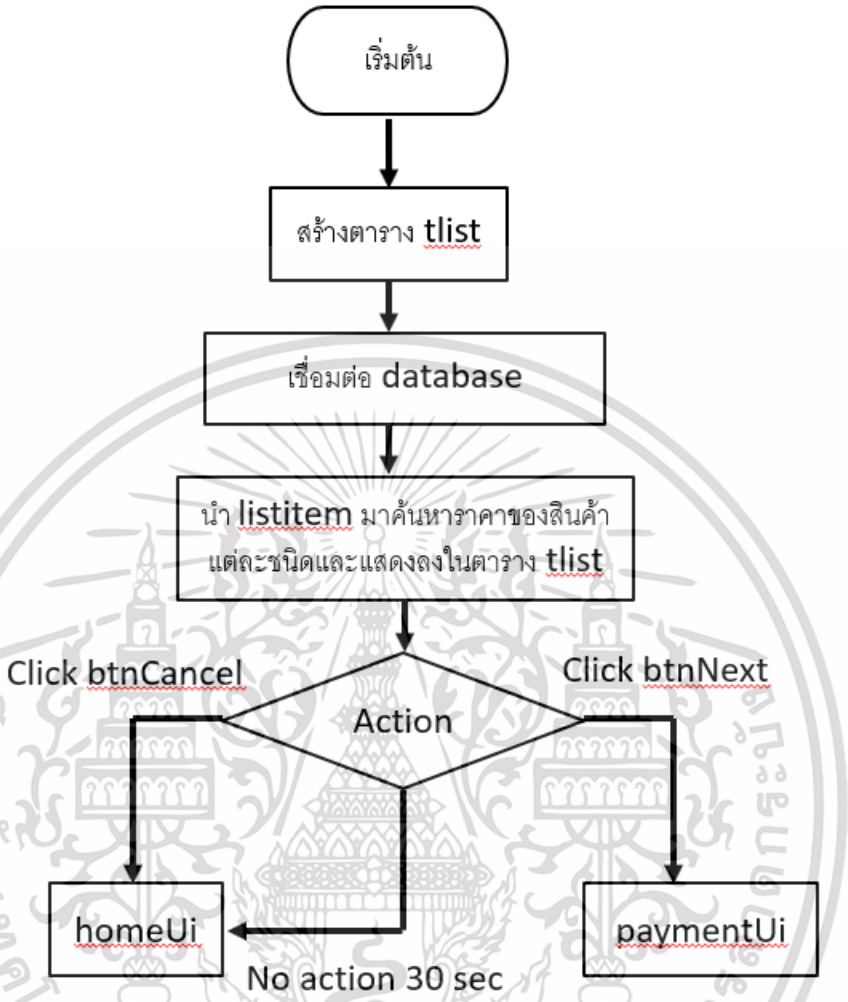


รูป 3.30 Flowchart scan

Module scan มีหน้าที่เพื่อเรียกใช้งาน darknet เพื่อ detector รูปภาพที่ถ่ายสินค้าที่ต้องการชำระเงิน โดยการใช้ subprocess “./darknet detector test obj.data yolov3-tiny-obj.cfg yolov3-tiny-obj_5000.weights cap.jpg -dont_show”

แลพจะทำการสร้าง listitem ขึ้นมาเพื่อจัดเก็บรายชื่อสินค้าที่ตรวจจับได้

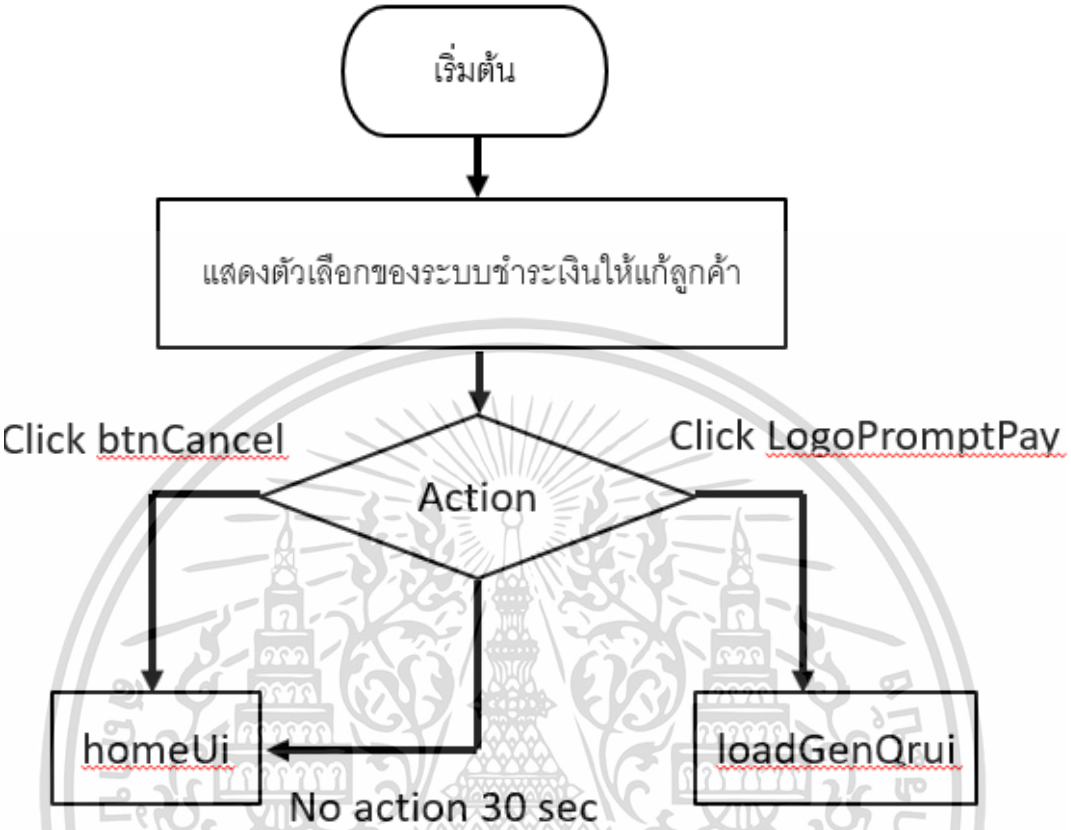
3.4.2.5 Module listUi



รูป 3.31 Flowchart listUi

Module listUi มีหน้าที่เพื่อสร้างตารางรายการสินค้าที่ต้องชำระจาก listitem ที่ได้มาจาก Module scan และเชื่อมต่อกับฐานข้อมูลเพื่อรับราคาสินค้ามาจากฐานข้อมูล เมื่อลูกค้าได้ตรวจสอบความถูกต้องและแน่ใจว่าจะต้องการจะชำระเงินก็ให้กดที่ปุ่ม NEXT เพื่อดำเนินการต่อ และเรียก Module payment หรือกด ที่ปุ่ม CANCEL เพื่อกลับสู่หน้า homeUi หรือเมื่อไม่กระทำการใดๆนาน 30วินาทีจะกลับสู่หน้า homeUi เช่นกัน

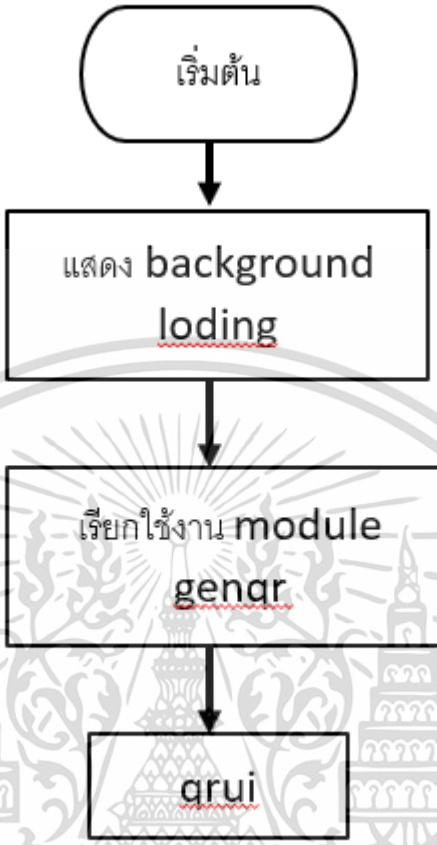
3.4.2.6 Module paymentUi



รูป 3.32 Flowchart paymentUi

Module paymentUi มีหน้าที่แสดงตัวเลือกให้ผู้ชำระเงินได้เลือกระบบการชำระเงินแต่เนื่องในเวอร์ชันนี้เราได้ทำให้สามารถจ่ายเงินได้แค่ระบบpromtpay โดยเมื่อคลิกที่ logo ของpromtpay จะทำการเรียก Module loadGenQrui หรือกด ที่ปุ่ม CANCEL เพื่อกลับสู่หน้า homeUi หรือเมื่อไม่กระทำการใดๆนาน30วินาทีจะกลับสู่หน้า homeUi เช่นกัน

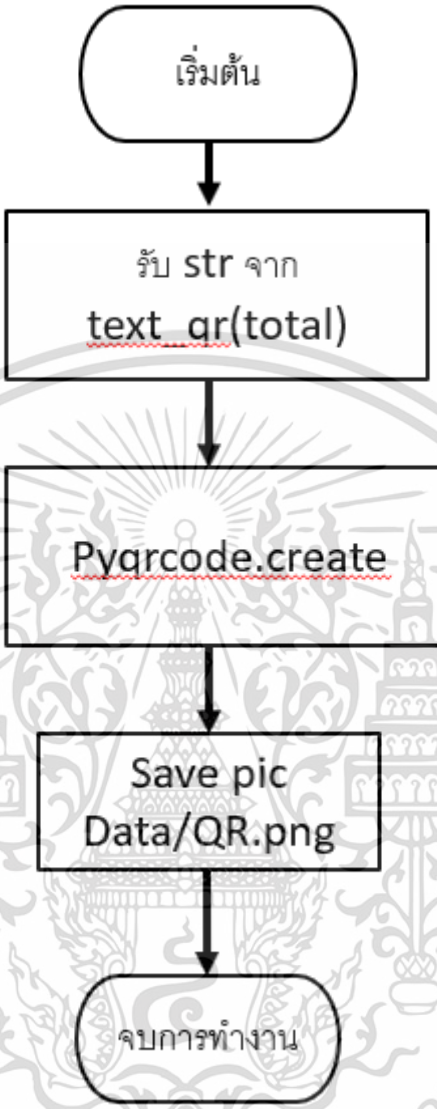
3.4.2.7 Module loadGenQRUi



รูป 3.33 Flowchart loadGenQRUi

Module loadGenQRUi ทำหน้าที่แสดงหน้า loading ให้แก่ผู้ใช้งาน, เรียก Module genqr เพื่อทำการสร้าง qrcode และเรียก Module qrUi เป็นลำดับต่อมาเพื่อไปหน้าที่ให้ลูกค้าscan qr code

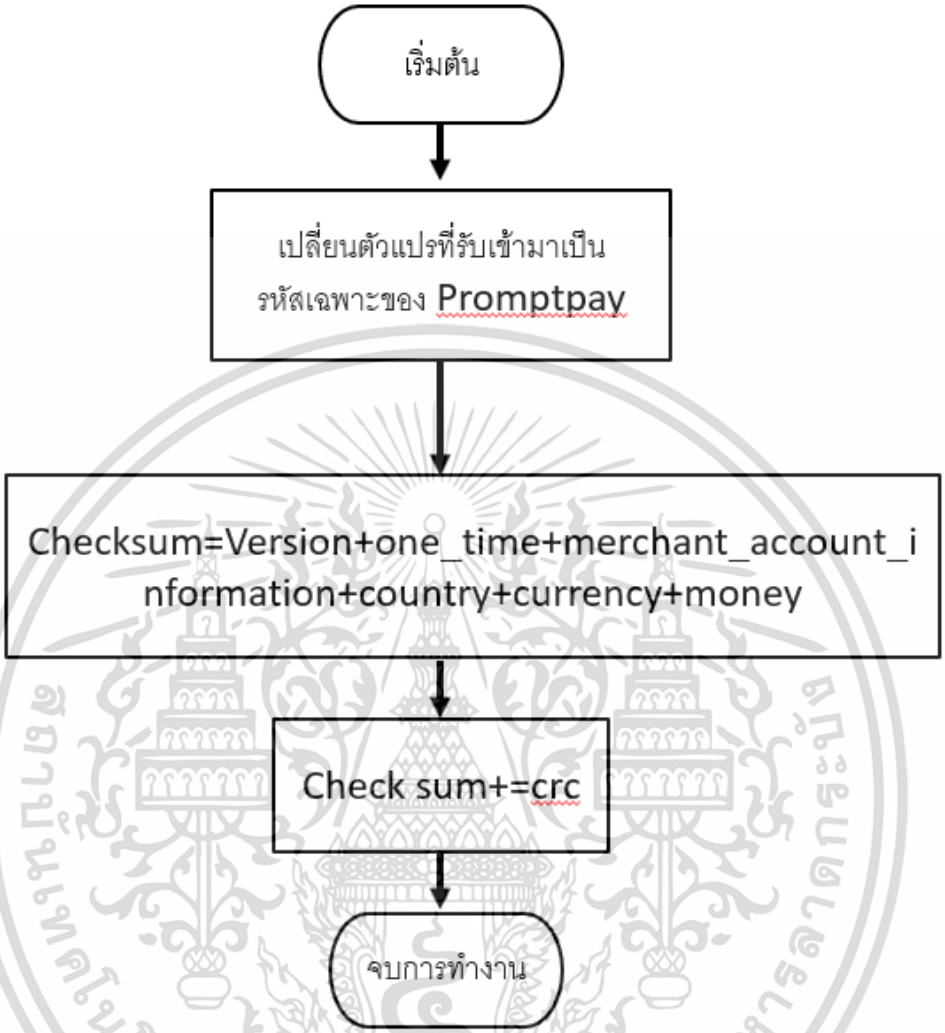
3.4.2.8 Module genqr



รูป 3.34 Flowchart genqr

Module genqr มีหน้าที่สร้างรูป qrcode จาก str ที่รับมาจาก text_qr และบันทึกลงใน Folder Data โดยที่ total เป็นยอดรวมของยอดจ่ายของสินค้าภายใน listitem

3.4.2.9 Module text_qr



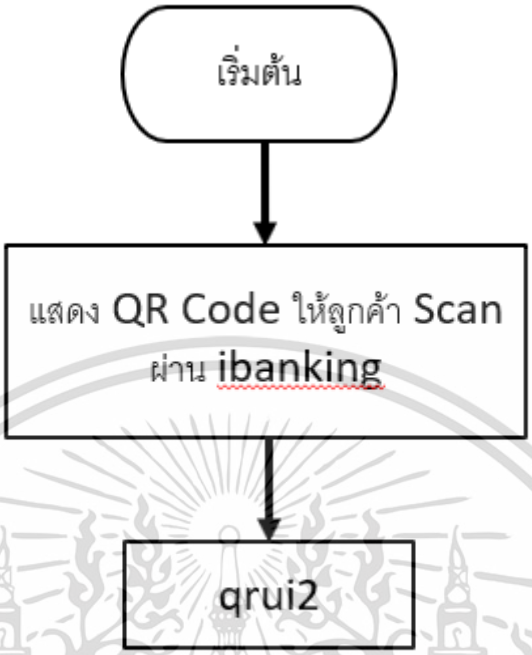
รูป 3.35 Flowchart text_qr

Module text_qr มีหน้าที่ในการ generate string เพื่อนำไปสร้าง qr code โดยรับตัวพารามิเตอร์คือ

account: เบอร์โทรของผู้ที่ต้องการรับเงินที่ชำระสินค้า one_time:คือต้องการใช้ code นี้ครั้งเดียวใช่หรือไม่

Money: จำนวนยอดเงินที่ต้องการชำระ

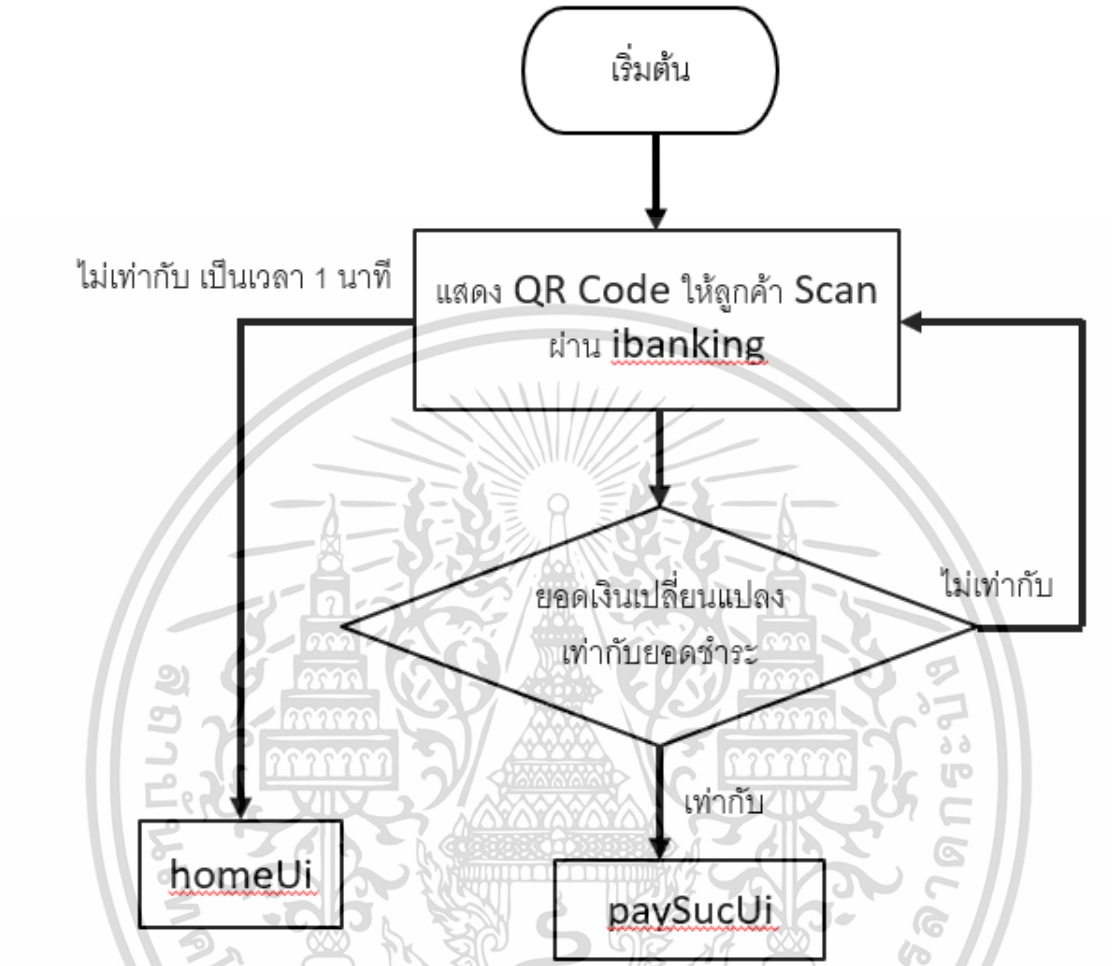
3.4.2.10 Module qrui



รูป 3.36 Flowchart qrui

Module qrui มีหน้าที่แสดง QR Code เพื่อให้ลูกค้าที่ต้องการชำระเงิน Scan QR Code ผ่าน iBanking เพื่อชำระเงิน และเรียกใช้งาน Module qrUi2

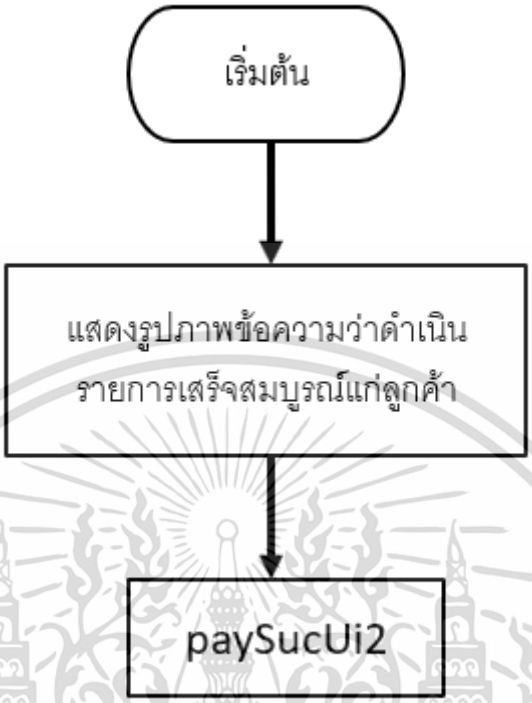
3.4.2.11 Module qrui2



รูป 3.37 Flowchart qrui2

Module qrui2 มีหน้าที่แสดง QR Code เพื่อให้ลูกค้าที่ต้องการชำระเงิน Scan QR Code ผ่าน iBanking เพื่อชำระเงิน และทำการตรวจสอบยอดเงินจากบัญชีธนาคารว่ามีการเปลี่ยนแปลงตามยอดเงินที่ต้องชำระหรือไม่ หรือถ้าหากยอดเงินไม่มีการเปลี่ยนแปลงเท่ากับยอดชำระเป็นเวลา 1 นาทีจะเรียก Module homeUi เพื่อกลับสู่หน้า homeUi

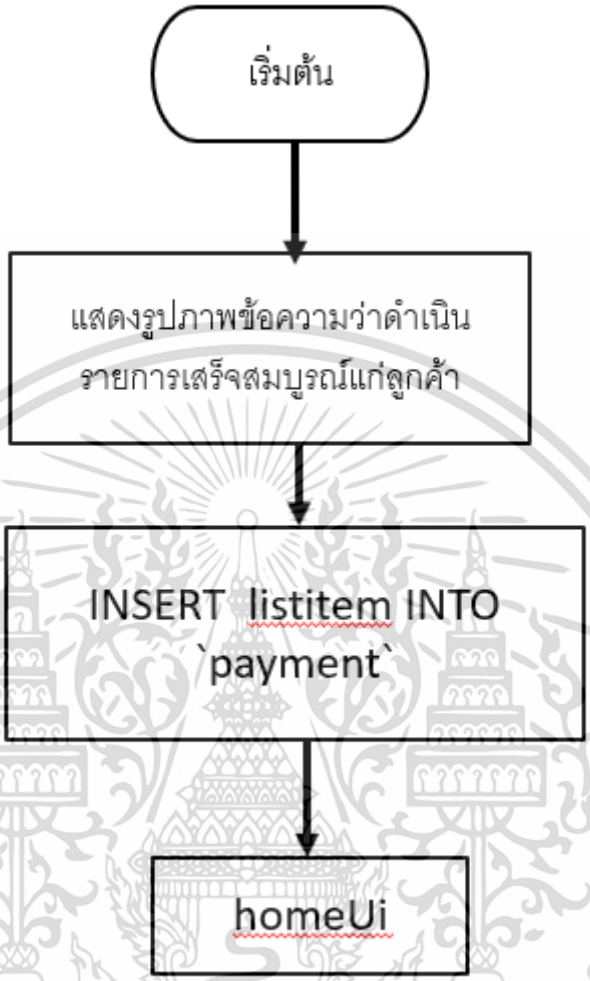
3.4.2.12 Module paySucUi



รูป 3.38 Flowchart paySucUi

Module paySucUi มีหน้าที่แสดงให้ผู้ใช้เห็นว่าได้ดำเนินรายการเสร็จสมบูรณ์ และเรียก Module paySucui2 เพื่อ update ข้อมูลลงในระบบฐานข้อมูล

3.4.2.13 Module paySucUi2



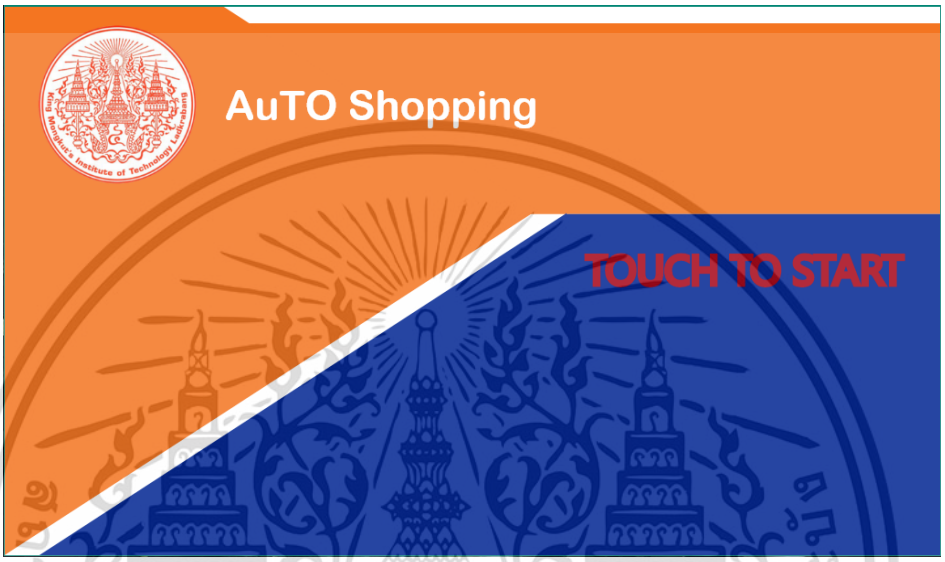
รูป 3.39 Flowchart paySucUi2

Module paySucUi2 มีหน้าที่แสดงให้ผู้ใช้เห็นว่าได้ดำเนินรายการเสร็จสมบูรณ์ และ update ข้อมูลลงในระบบฐานข้อมูลจาก listitem เพื่อบันทึกรายการขายและเรียก Module homeUi เพื่อพร้อมกลับสู่การเริ่มต้นทำงานอีกครั้ง

3.5 UI Design

3.5.1 Customer Application

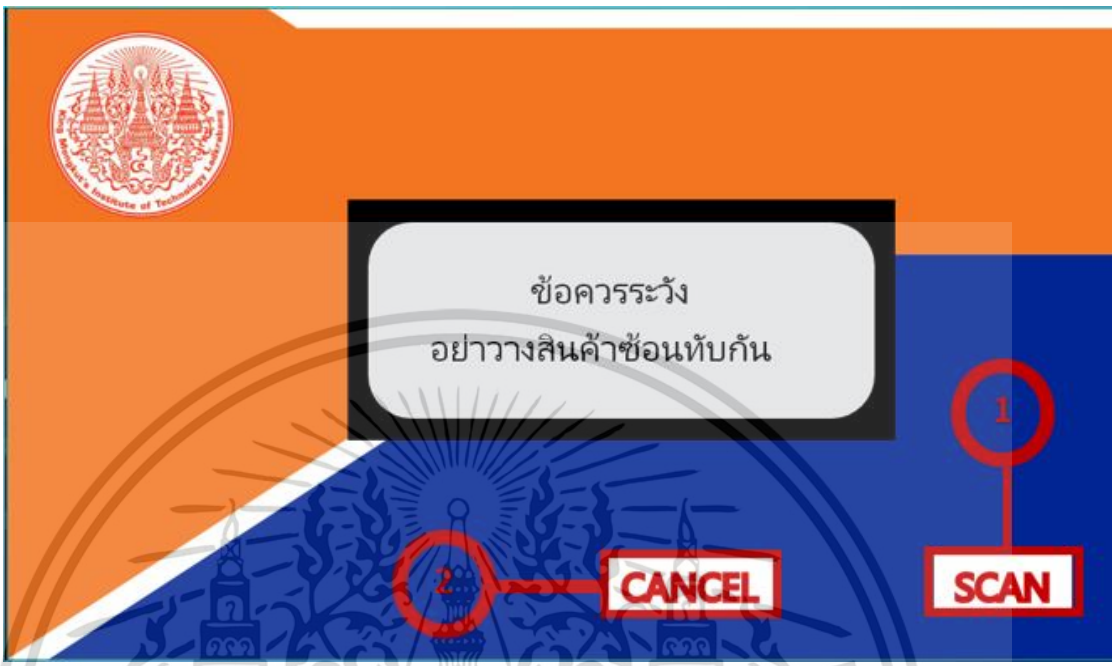
3.5.1.1 หน้า Home



รูปที่ 3.40 หน้า HOME

หน้า HOME เป็นหน้าเริ่มต้นเพื่อรอลูกค้ามาดำเนินการชำระเงินที่ตู้บริการ โดยลูกค้าต้องทำการสัมผัสที่หน้าจอ touch screen เพื่อดำเนินการต่อไป และจะกลับมาหน้าจอนี้เสมอเมื่อไม่มีการดำเนินการใดเป็นเวลา 60 วินาที

3.5.1.2 หน้า Warning



รูปที่ 3.41 หน้า Warning

หน้า Warning เป็นการแสดงคำเตือนเพื่อลดความผิดพลาดของระบบในการแส
กนสินค้าที่ลูกค้าต้องที่จะชำระเงินหรือกดยกเลิกรายการเพื่อกลับเข้าสู่หน้า HOME
และถ้าหากไม่ได้ทำรายการต่อภายใน 30 วินาที จะกลับเข้าสู่หน้า HOME เช่นกัน

จะมีปุ่มที่สามารถคลิกได้ 2 ปุ่ม

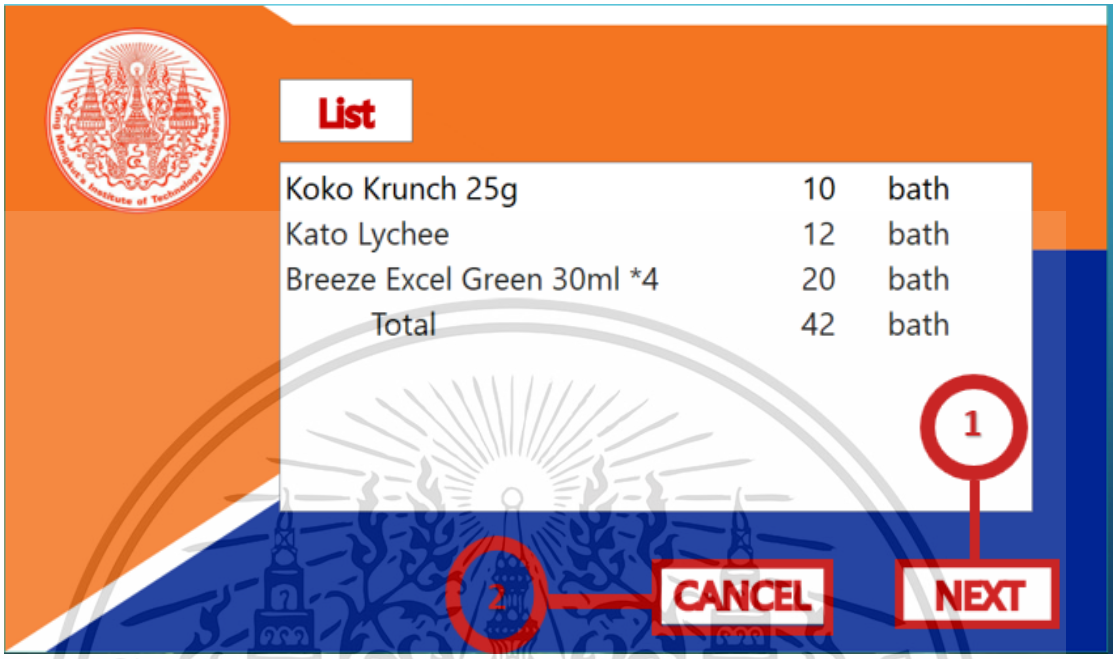
1. SCAN - คลิกเพื่อสั่งให้กล้องตรวจจับสินค้าในช่องวางสินค้า
2. CANCEL - คลิกพื่อย้อนกลับสู่หน้า HOME

3.5.1.3 หน้า Loading 1

รูปที่ 3.42 หน้า Loading 1

หน้า Loading หลังจากกดปุ่ม SCAN เครื่องจะทำการถ่ายภาพวัตถุเพื่อนำไปตรวจจับสิ่งของโดย Model ที่ทำการอบรมมาแล้ว

3.5.1.4 หน้า List



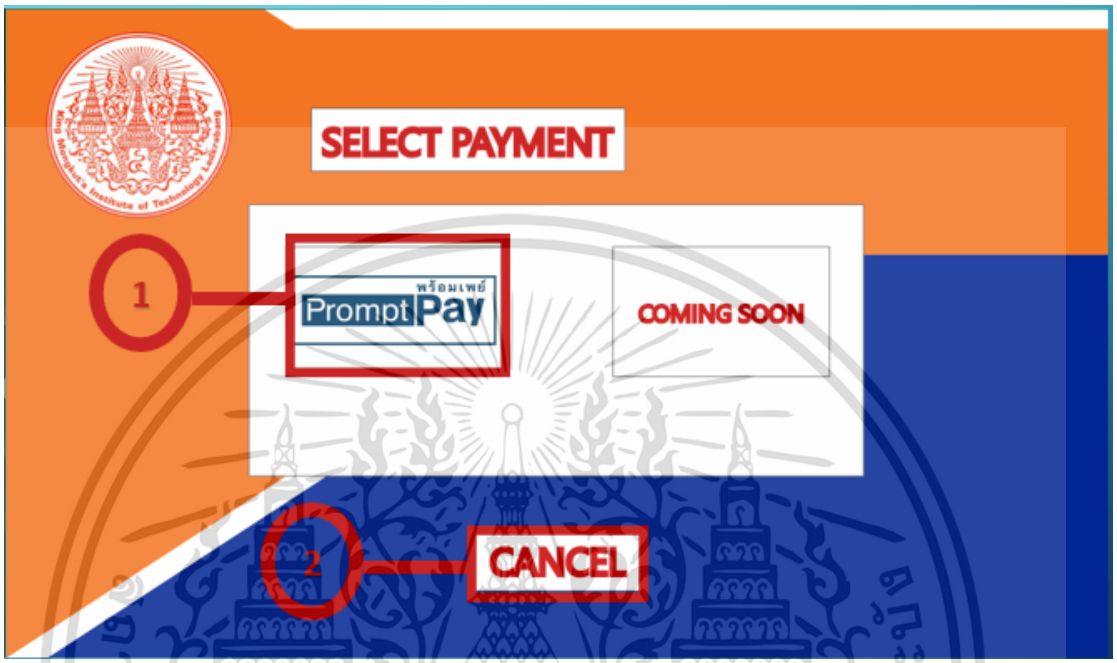
รูปที่ 3.43 หน้า List

หน้า List เป็นการแสดงรายการสินค้าที่ต้องชำระเพื่อให้ลูกค้าตรวจสอบว่าถูกต้องหรือไม่ ถ้าถูกต้องก็สามารถกดที่ปุ่มชำระเงินเพื่อดำเนินการต่อ หรือไม่ก็ควรที่จะกดยกเลิกรายการเพื่อกลับเข้าสู่หน้า HOME และถ้าหากไม่ได้ทำรายการต่อภายใน 30 วินาที จะกลับเข้าสู่หน้า HOME เช่นกัน

จะมีปุ่มที่สามารถคลิกได้ 2 ปุ่ม

1. NEXT - คลิกเพื่อสั่งให้กล้องตรวจจับสินค้าในช่องวางสินค้า
2. CANCEL - คลิกพื่อย้อนกลับสู่หน้า HOME

3.5.1.5 หน้า payment



รูปที่ 3.44 หน้า payment

ลูกค้าสามารถเลือกช่องทางการชำระเงินโดยการสัมผัสที่โลโก้ ของบริการแต่ในแอพพลิเคชั่นนี้รองรับเพียงช่องทอง PromptPay เท่านั้น

จะมีปุ่มที่สามารถคลิกได้ 2 ปุ่ม

1. Icon PromptPay - คลิกเพื่อดำเนินการจ่ายเงินด้วยระบบ PromptPay และเข้าสู่หน้า QRUI เพื่อทำการจ่ายเงินด้วย QRCODE
2. CANCEL - คลิกพื่อย้อนกลับสู่หน้า HOME

3.5.1.6 หน้า Loading 2

รูปที่ 3.45 หน้า Loading 2

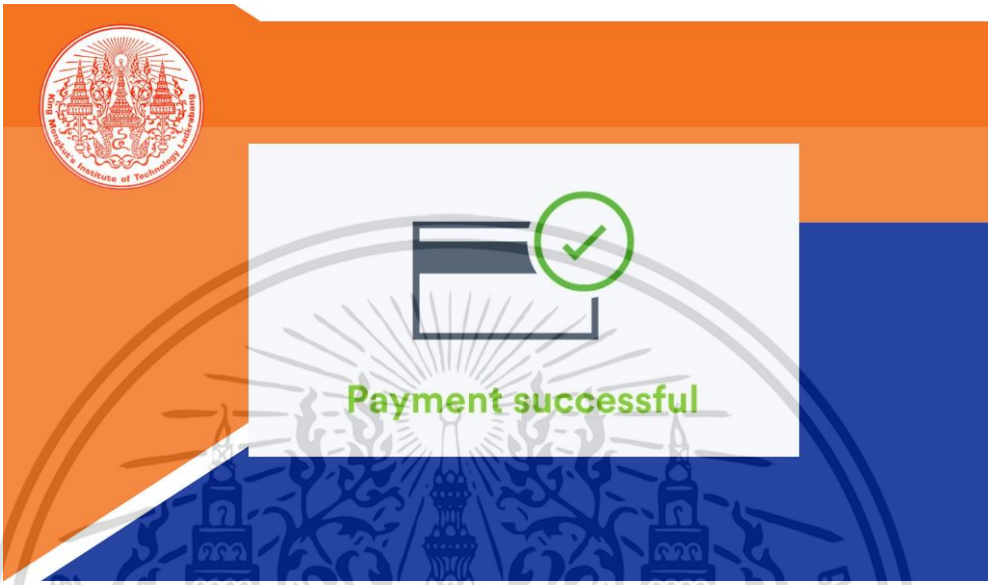
เมื่อลูกค้าเลือกการชำระเงินผ่านระบบ PromptPay จะเข้าสู่หน้า Loading เพื่อทำการสร้างQR Code เพื่อให้ลูกค้าสแกนเพื่อชำระเงิน

3.5.1.7 หน้า QRUI

รูปที่ 3.46 หน้า QRUI

หน้าQRUI จะทำการแสดง QR Code เพื่อให้ลูกค้าสแกนผ่าน QR SCANNER ของ iBanking ต่าง ๆ เพื่อทำงานโอนเงินตามยอดเงินที่ต้องชำระ โดยลูกค้าต้องทำการแสกน QRCODE เพื่อเข้าสู่หน้า Pay Successful หรือเมื่อเวลาผ่านไปและไม่ได้ทำการ scan QRCODE จะกลับสู่หน้า HOME

3.5.1.8 หน้า pay_Successful

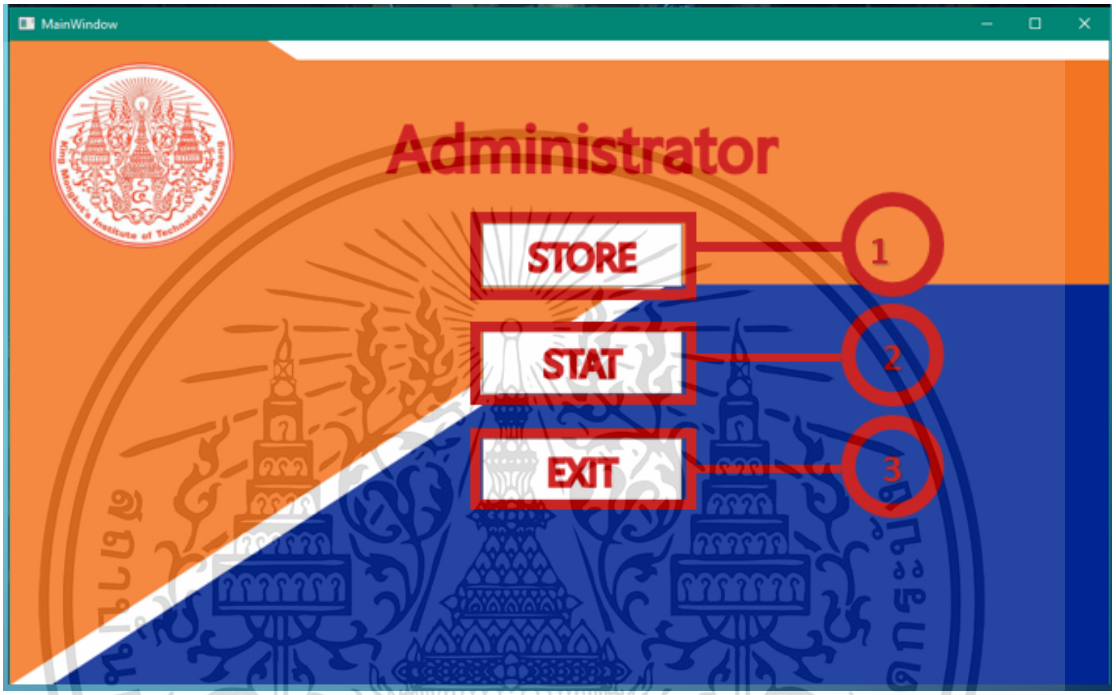


รูปที่ 3.47 หน้า pay_Successful

หน้า pay_Successful จะแสดงเมื่อลูกค้าทำการโอนเงินตรงกับยอดกับที่ต้องชำระ โดยตัวโปรแกรมจะเข้าไปดึงยอดเงินของธนาคารมาเพื่อตรวจสอบและจะกลับเข้าสู่หน้า Home โดยอัตโนมัติในเวลาต่อมา

3.5.2 Admin Application

3.3.2.1 หน้า Home



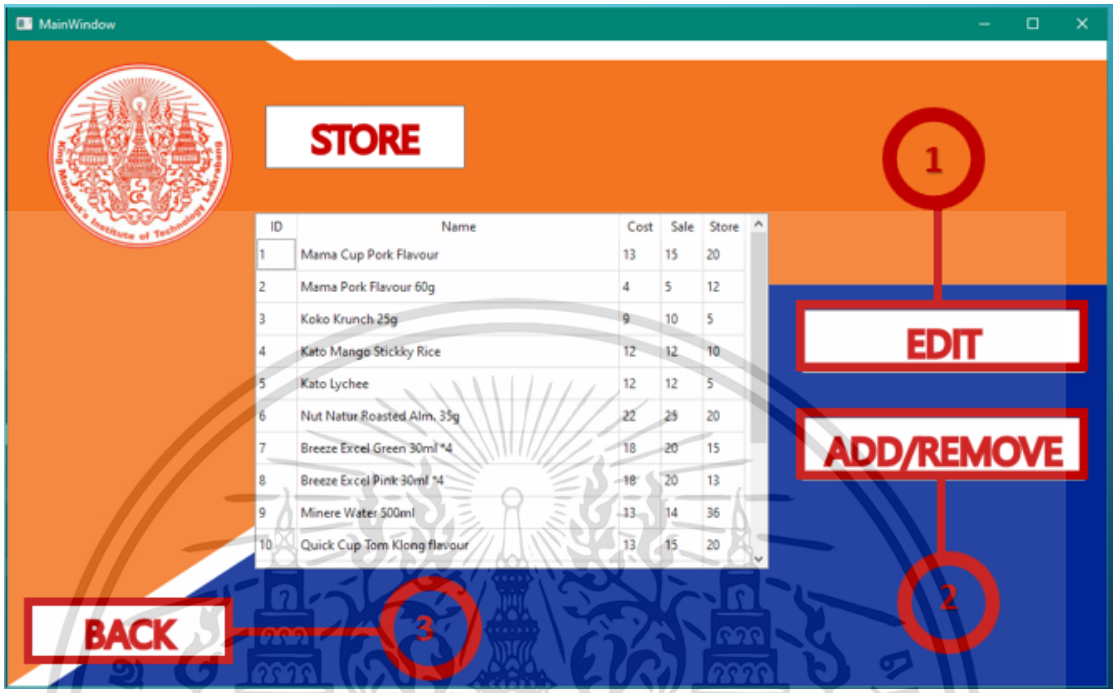
รูปที่ 3.48 หน้า home

จะมีเมนูให้เลือกเพื่อเข้าสู่หน้าถัดไป Store เพื่อตรวจสอบคลังสินค้าหรือแก้ไข STAT เพื่อดูสถิติของรายการขาย

มีเมนูให้เลือก 3เมนู

1. STORE-คลิกเพื่อเข้าสู่หน้า STORE สามารถดูลายละเอียดและแก้ไขคลังสินค้าได้
2. STAT-คลิกเพื่อเข้าสู่หน้า STAT สามารถดูค่าสถิติเกี่ยวกับยอดขายได้
3. EXIT- คลิกเพื่อออกจากโปรแกรม

3.5.2.2 หน้า Store



รูปที่ 3.49 หน้า Store

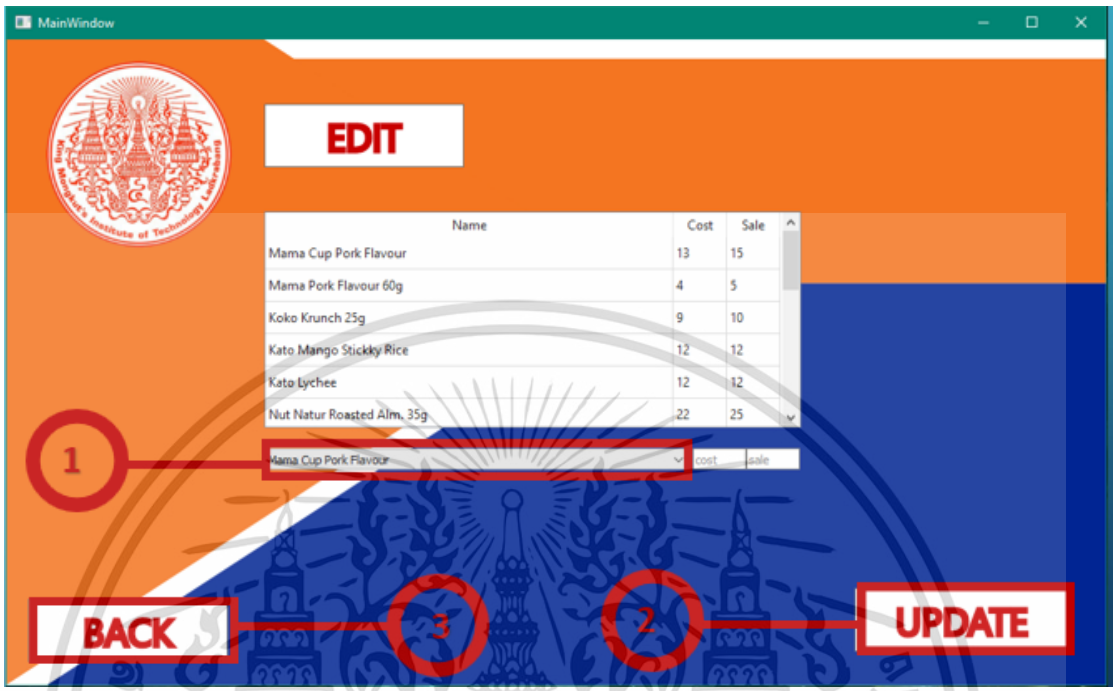
จะแสดงข้อมูลสินค้าต่าง ๆในฐานข้อมูลมีเมนูให้เลือกเพื่อแอดมินสามารถเลือกเมนู EDIT เพื่อแก้ไขราคาต้นทุน(Cost) และราคาขาย(Sale) หรือเข้าสู่เมนู ADD/REMOVE เพื่อเพิ่มหรือลดจำนวนสินค้าจากฐานข้อมูลได้

จะมีปุ่มที่สามารถคลิกได้ 3 ปุ่ม

1. EDIT - คลิกเพื่อเข้าสู่หน้า EDIT เพื่อสามารถแก้ไขราคาสินค้าได้
2. ADD/REMOVE - คลิกที่ เพื่อเข้าสู่หน้า EDIT เพื่อสามารถแก้ไขราคาสินค้าได้

BACK - คลิกพื่อย้อนกลับสู่หน้า HOME

3.5.2.3 หน้า EDIT



รูปที่ 3.50 หน้า EDIT

หน้า EDIT จะแสดงตารางชื่อสินค้า ราคาต้นทุนและราคาขาย สามารถเลือกชื่อสินค้าและใส่ราคาต้นทุนหรือราคาขายเพื่ออัพเดทราคาสินค้าได้

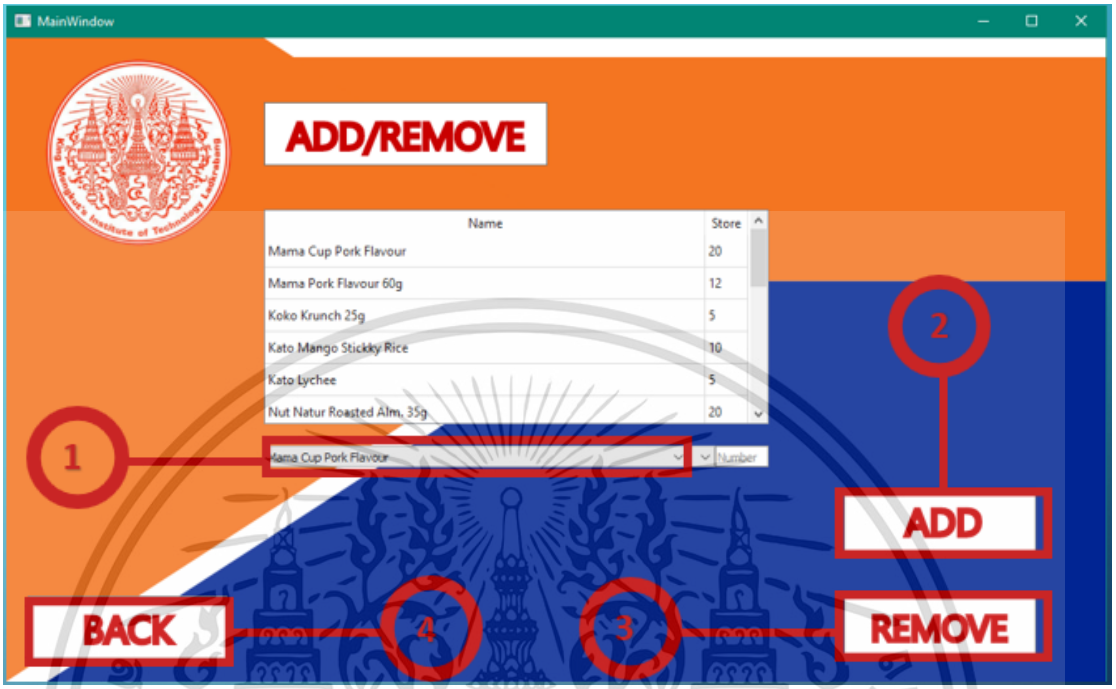
จะมีปุ่มที่สามารถคลิกได้ 3 ปุ่ม

1.Combobox ใต้ตาราง – คลิกเพื่อเลือกชื่อสินค้าที่ต้องการแก้ไข และใส่ตัวเลขที่ต้องการแก้ไขในช่องว่างหลัง combobox

2.Update - คลิกเมื่อหลังจากใส่ค่าที่ต้องการเปลี่ยนแปลงแล้วเพื่อupdate ลงในฐานข้อมูล

3.BACK - คลิกพื่อย้อนกลับสู่หน้า STORE

3.5.2.4 หน้า AddAndRemove



รูปที่ 3.51 หน้า AddAndRemove

หน้า AddAndRemove จะแสดง ชื่อสินค้า และจำนวนสินค้าใน Store สามารถเลือกชื่อสินค้าและใส่ตัวเลขเพื่อเพิ่มหรือลดจำนวนของสินค้าได้

จะมีปุ่มที่สามารถคลิกได้ 4 ปุ่ม

1. Combobox ใต้ตาราง – คลิกเพื่อเลือกชื่อสินค้าที่ต้องการแก้ไข และใส่ตัวเลขที่ต้องการแก้ไขในช่องว่างหลัง
2. ADD - คลิกเมื่อหลังจากใส่ค่าที่ต้องการเปลี่ยนแปลงแล้วเพื่อเพิ่มจำนวนสินค้าตามจำนวนที่กรอกลงในฐานข้อมูล
3. REMOVE - คลิกเมื่อหลังจากใส่ค่าที่ต้องการเปลี่ยนแปลงแล้วเพื่อลดจำนวนสินค้าตามจำนวนที่กรอกลงในฐานข้อมูล
4. BACK - คลิกพื่อย้อนกลับสู่หน้า STORE

3.5.2.5 หน้า STAT

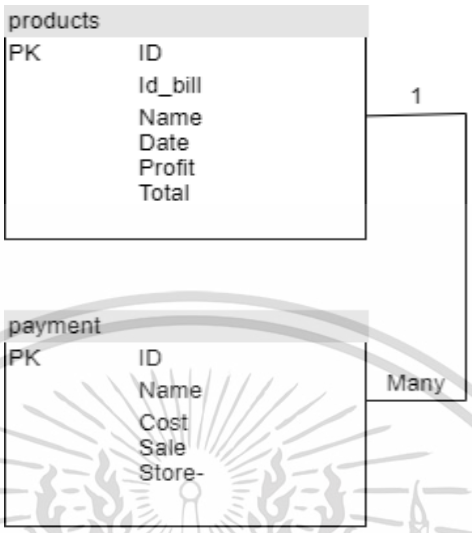
รูปที่ 3.52 หน้า STAT

ตัวเลขทางด้านบนจะแสดงกำไรและยอดขาย กราฟทางด้านซ้ายจะแสดงกำไรระหว่างวันที่กำหนด ส่วนตารางทางด้านขวาจะแสดงจำนวนสินค้าที่ขายได้มากที่สุดในช่วงเวลาที่ทำการเลือก โดยเมื่อเข้าครั้งแรกจะย้อนหลังไป 7 วัน

จะมีปุ่มที่สามารถคลิกได้ 3 ปุ่ม

1. Start Date - คลิกวันเพื่อเลือกวันที่ต้องการเริ่มต้นที่จะต้องการแสดงผลของข้อมูล
2. END Date - คลิกวันเพื่อเลือกวันที่ต้องการสิ้นสุดที่จะต้องการแสดงผลของข้อมูล
3. BACK - คลิกพื่อย้อนกลับสู่หน้า HOME

3.6 การออกแบบ Database



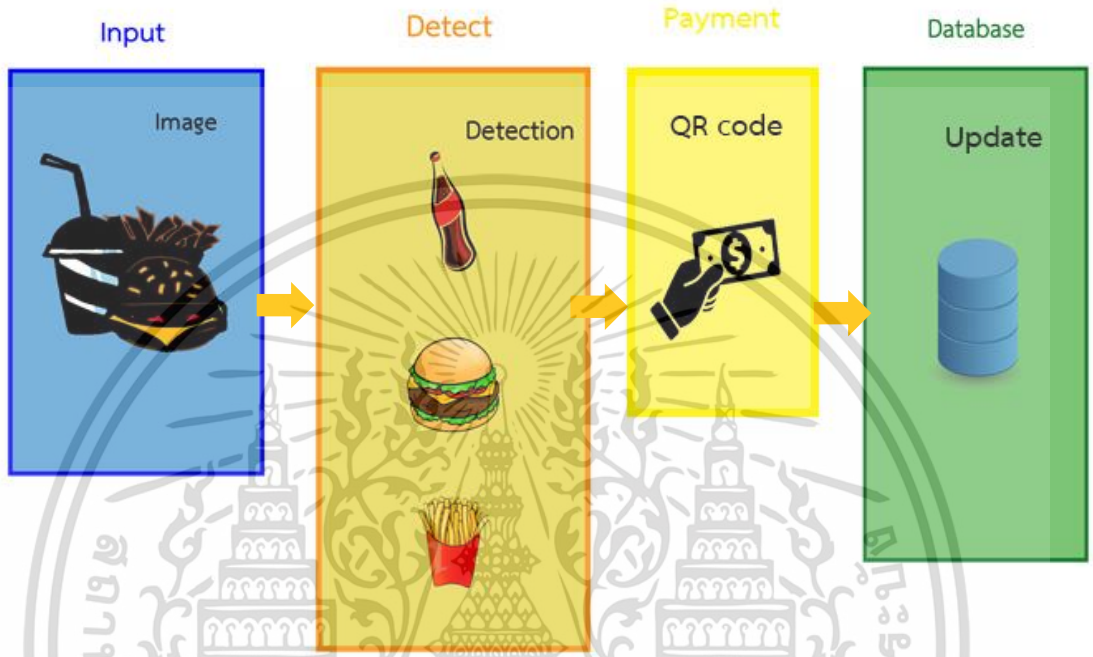
รูป 3.53 Entity Relationship Diagram

ตาราง 3.10 Data Dictionary

Table	Column	Data type	PK or FK	Description	FK.Ref. Table
product	ID	INT	PK	ไอดีของสินค้า	
	Name	TEXT		ชื่อสินค้า	
	Cost	INT		ต้นทุนของสินค้า	
	Sale	INT		ราคาขายของสินค้า	
	Store	INT		จำนวนสินค้าในคลัง	
payment	ID	INT	PK	ไอดีของรายการ	
	Id_bill	INT		ไอดีของบิล	
	Name	TEXT		ชื่อสินค้า	
	Date	DATE		วันที่ทำรายการ	
	Profit	INT		กำไรจากการขาย	
	Total	INT		ยอดเงินจากการขาย	

3.7 ภาพรวมระบบ

3.7.1 ภาพรวมการทำงานภายในระบบ



รูป 3.54 System Diagram

จาก Diagram จะสามารถแบ่งการทำงาน เป็น 4 ส่วนหลัก ได้แก่

1) Input: เป็นไฟล์ภาพที่ถ่ายได้จากกล่องของอุปกรณ์

2) Detect: เป็นการนำ Input ที่ถ่ายได้นั้นไปตรวจจับวัตถุและระบุสินค้า

3) Payment: เป็นการนำสินค้าที่ตรวจพบไปขอข้อมูลจากฐานข้อมูลและคิดคำนวณเป็นรายการค่าใช้จ่าย

4) Database: เป็นอัพเดทการทำรายการไปยังฐานข้อมูลเพื่อเป็นประวัติการซื้อขาย

3.8 การดำเนินการวิเคราะห์และระบุตัวตนสินค้า

การตรวจจับสินค้าสำหรับโครงงานนี้ได้เทรน Model ผ่าน Framework yolov3 โดยจะรับภาพจากกล้องในตอนที่ลูกค้าสัมผัสที่ปุ่ม SCAN เข้ามาตรวจจับ โดยรัน Model ที่ผ่านการอบรมมาผ่าน Darknet โดยจะมีขนาดภาพเป็น 608*608 pixel เนื่องจากขนาดภาพมีผลต่อความแม่นยำและระยะเวลาในการอบรม และ Darknet จะทำการแสดงผลการตรวจจับในหน้า LIST

3.9 ขั้นตอนการฝึกอบรม

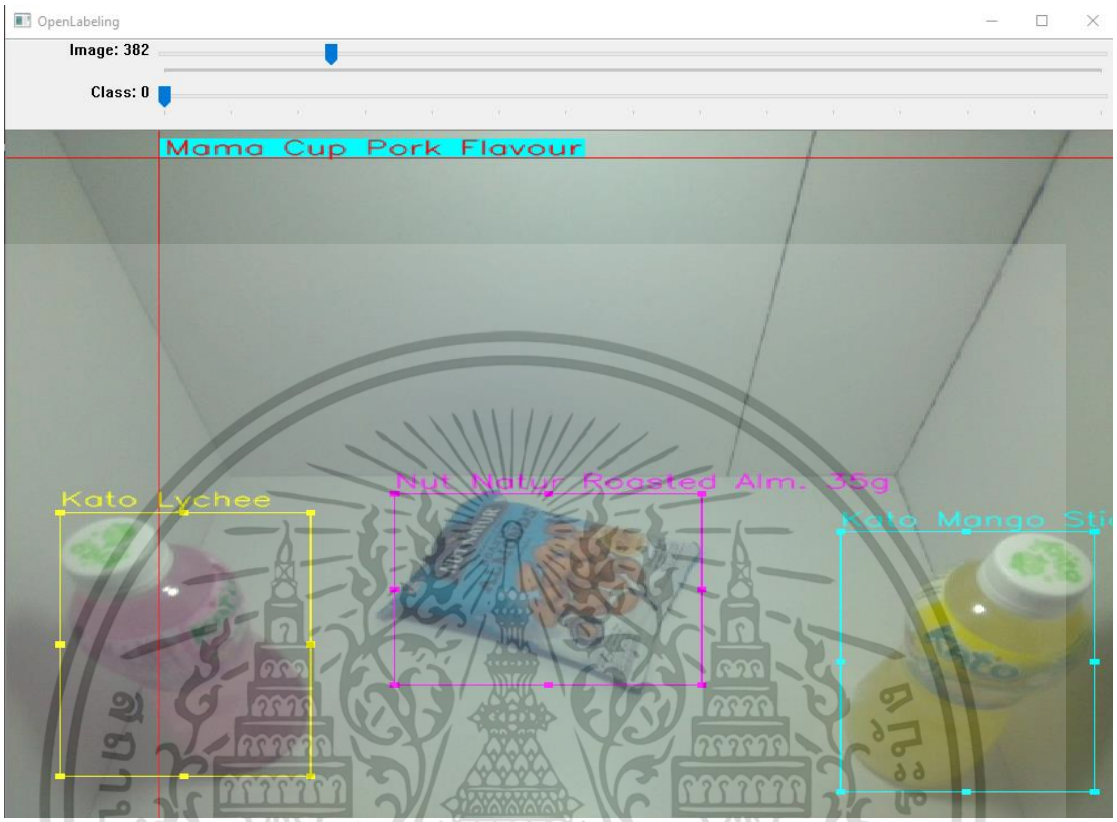
1. เลือก model เพื่อใช้ในการตรวจจับวัตถุในโครงงานนี้เราเลือกใช้ Tiny Yolov3 เนื่องจากใช้ทรัพยากรด้าน Hardware และเวลาในการฝึกอบรมที่น้อยกว่า Yolov3

Performance on the COCO Dataset

Model	Train	Test	mAP	FLOPS	FPS	Cfg	Weights
SSD300	COCO trainval	test-dev	41.2	-	46		link
SSD500	COCO trainval	test-dev	46.5	-	19		link
YOLOv2 608x608	COCO trainval	test-dev	48.1	62.94 Bn	40	cfg	weights
Tiny YOLO	COCO trainval	test-dev	23.7	5.41 Bn	244	cfg	weights
SSD321	COCO trainval	test-dev	45.4	-	16		link
DSSD321	COCO trainval	test-dev	46.1	-	12		link
R-FCN	COCO trainval	test-dev	51.9	-	12		link
SSD513	COCO trainval	test-dev	50.4	-	8		link
DSSD513	COCO trainval	test-dev	53.3	-	6		link
FPN FRCN	COCO trainval	test-dev	59.1	-	6		link
Retinanet-50-500	COCO trainval	test-dev	50.9	-	14		link
Retinanet-101-500	COCO trainval	test-dev	53.1	-	11		link
Retinanet-101-800	COCO trainval	test-dev	57.5	-	5		link
YOLOv3-320	COCO trainval	test-dev	51.5	38.97 Bn	45	cfg	weights
YOLOv3-416	COCO trainval	test-dev	55.3	65.86 Bn	35	cfg	weights
YOLOv3-608	COCO trainval	test-dev	57.9	140.69 Bn	20	cfg	weights
YOLOv3-tiny	COCO trainval	test-dev	33.1	5.56 Bn	220	cfg	weights
YOLOv3-spp	COCO trainval	test-dev	60.6	141.45 Bn	20	cfg	weights

รูปที่ 3.55 Performance on the COCO Dataset

2. ทำการ label วัตถุที่ต้องการตรวจจับทำการตรวจจับ เพื่อทำให้โมเดลได้เรียนรู้รูปลักษณ์และสีของวัตถุชนิดนี้



รูปที่ 3.56 ตัวอย่างการ Label image

3.จัดทำ File Text ใน Format ดังนี้

Row format: image_file_path box1 box2 ... boxN;

Box format: x_min,y_min,x_max,y_max,class_id (no space).

```
path/to/img1.jpg 50,100,150,200,0 30,50,200,120,3
path/to/img2.jpg 120,300,250,600,2
...
```

รูปที่ 3.57 ตัวอย่าง format ในfile train.text

4. ทำการฝึกอบรมด้วย darknet โดยประมวลผลผ่าน Google Colab

```
[ ] !./darknet detector train data_for_colab/obj.data data_for_colab/yolov3-tiny-obj.cfg data_for_colab/yolov3-tiny.conv.15 -dont_show

 CUDA-version: 8000 (10010), GPU count: 1
 OpenCV version: 3.2.0
yolov3-tiny-obj
 compute_capability = 610, cudnn_half = 0
net.optimized_memory = 0
mini_batch = 2, batch = 8, time_steps = 1, train = 1
   layer   filters  size/strd(dil)      input                output
   0 conv     16       3 x 3/ 1    608 x 608 x   3 ->  608 x 608 x  16 0.319 BF
   1 max                2x 2/ 2    608 x 608 x  16 ->  304 x 304 x  16 0.006 BF
   2 conv     32       3 x 3/ 1    304 x 304 x  16 ->  304 x 304 x  32 0.852 BF
   3 max                2x 2/ 2    304 x 304 x  32 ->  152 x 152 x  32 0.003 BF
   4 conv     64       3 x 3/ 1    152 x 152 x  32 ->  152 x 152 x  64 0.852 BF
   5 max                2x 2/ 2    152 x 152 x  64 ->   76 x  76 x  64 0.001 BF
   6 conv    128       3 x 3/ 1     76 x  76 x  64 ->   76 x  76 x 128 0.852 BF
```

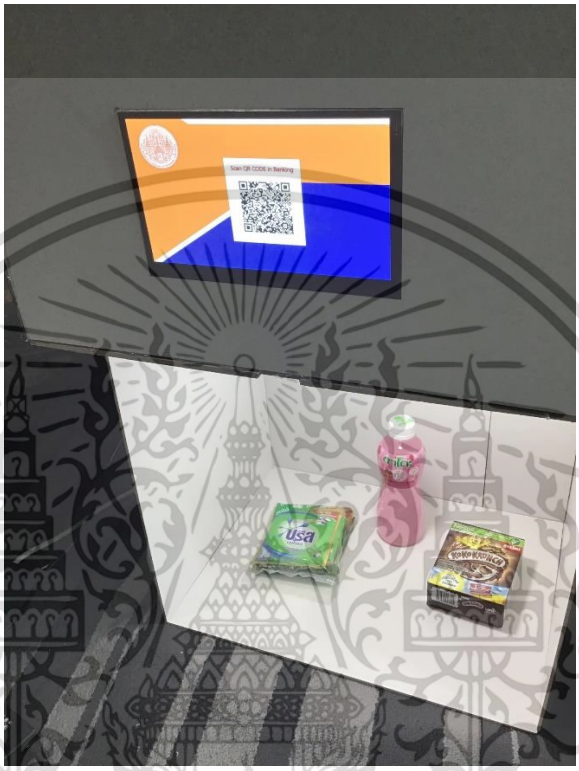
รูปที่ 3.58 ตัวอย่างการฝึกอบรมด้วย Darknet บน COLAB

5. จะได้ file นามสกุล .weights มาเพื่อสามารถ นำไปประมวลผลกับภาพเพื่อตรวจจับวัตถุได้ จะได้ผลการตรวจจับดังรูป

รูปที่ 3.59 ตัวอย่างการตรวจจับวัตถุจากภาพ

3.10 การจัดทำเครื่องชำระสินค้าอัตโนมัติ

การออกแบบโมเดลเครื่องชำระสินค้าอัตโนมัติเลือกขนาดที่ไม่ใหญ่มากและถ่ายภาพได้มุมกว้างเนื่องจากกล้องจับภาพระยะไกลทำให้มีความแม่นยำลดลง และเลือกจอ Touch Screen ขนาด 7 นิ้ว เพื่อแสดงผล



รูปที่ 3.60 โมเดลอุปกรณ์

ตาราง 3.11 รายการวัสดุ อุปกรณ์ที่ใช้ในการจัดทำเครื่องตรวจจับสินค้า

ลำดับ	รายการวัสดุ อุปกรณ์
1	Nvidia Jetson Nano
2	Raspberry Pi Camera V2
3	พัดลมรับายความร้อน 1.5 นิ้ว
4	จอ Touch Screen 7 นิ้ว
5	หลอดไฟ 5 W

บทที่ 4

การทดลอง

4.1 การทดลองที่ 1

วัตถุประสงค์ :

ศึกษาการอบรบโมเดล และตรวจจับสินค้า เพื่อการเรียนรู้เชิงลึกของโมเดล Yolov3 และปรับค่า Config TRAINING Input size=416*416 Batch=64 Subdivisions=2

รายละเอียดการทดลอง :

ใช้รูปภาพที่ถ่ายจากเครื่องตรวจจับสินค้าในการทดลองอบรมโมเดล ซึ่งโดยรูปภาพทั้งหมด 100 รูป ต่อ Class

ผลลัพธ์การทดลอง :

ได้ความแม่นยำที่ต่ำมาก เนื่องจากข้อมูลที่นำไปฝึกอบรม ในบางรูปวัตถุกลับด้านอาจทำให้ความแม่นยำลดลงเนื่องจาก วัตถุในคลาสเดียวกันไม่เหมือนกัน

ตาราง 4.1การทดลองที่ 1

Input size=416*416 Batch=64 Subdivisions=2	average IoU(%)	mAP (%)
1000	1.77%	2.36%
2000	2.47%	2.75%
3000	2.61%	1.41%
4000	2.38%	1.16%
5000	1.62%	1.49%

4.2 การทดลองที่ 2

วัตถุประสงค์ :

ศึกษาการอบรบโมเดล และตรวจจับสินค้า เพื่อการเรียนรู้เชิงลึกของโมเดล Yolov3 และปรับค่า Config TRAINING Input size=416*416 Batch=64 Subdivisions=2

รายละเอียดการทดลอง :

เพิ่มรูปภาพที่ใช้ในการฝึกอบรมโมเดล ซึ่งโดยรูปภาพทั้งหมด 150 รูป ต่อ Class จากเดิม 100 รูป ต่อ Class และลดรูปที่กลับด้านให้น้อยลง

ผลลัพธ์การทดลอง :

ได้ผลลัพธ์การทดลองแม่นยำขึ้นกว่าเดิมอย่างมาก และเริ่มOverfitting ที่ประมาณ 4000 epoch

ตาราง 4.2 การทดลองที่ 2

Input size=416*416 Batch=64 Subdivisions=2	average IoU(%)	mAP (%)
1000	44.27%	58.93%
2000	74.69%	98.73%
3000	74.87%	98.41%
4000	77.65%	98.79%
5000	77.71%	96.21%

4.3 การทดลองที่ 3

วัตถุประสงค์ :

ศึกษาการอบรบโมเดล และตรวจจับสินค้า เพื่อการเรียนรู้เชิงลึกของโมเดล Yolov3 และปรับค่า Config TRAINING Input size=608*608 Batch=1 Subdivisions=2

รายละเอียดการทดลอง :

เพิ่มขนาดความละเอียดของรูปภาพที่ใช้ในการฝึกอบรม เป็น 608*608 pixel และทดลองปรับ batch size เป็น 1 เนื่องจาก GPU memory error เพราะมี memory ไมเพียงพอ ซึ่งโดยเพิ่มรูปภาพทั้งหมด 300 รูป ต่อ Class จากเดิม 150 รูปต่อ Class

ผลลัพธ์การทดลอง :

ความแม่นยำลดลง โดยเกี่ยวเนื่องจากการปรับ config ให้มีประสิทธิภาพลดลงเนื่องจากทรัพยากรด้าน Hardware ไม่สามารถทำงานที่ input size 608*608 Batch=64 Subdivisions=2 ได้ และได้ลดลงมายัง 32 ก็ยังไม่สามารถทำการฝึกอบรมได้ ดังนั้นจึงตั้งค่าไว้เท่ากับ 1 และเพิ่มรูปภาพที่ใช้ในการฝึกอบรมเป็น 2 เท่าจากการทดลองครั้งที่ 2 เพื่อไม่ให้ได้ความแม่นยำที่ต่ำจนเกินไป และเริ่มOverfitting ที่ประมาณ 5000 epoch

ตาราง 4.3 การทดลองที่ 3

Input size=608 Batch=1 Subdivisions=2	average IoU(%)	mAP (%)
1000	10.29%	10.7%
2000	44.82%	66.17%
3000	39.63%	40.66%
4000	34.53%	32.37%
5000	65.99%	88.82%
6000	44.65%	51.31%

4.4 การทดลองที่ 4

วัตถุประสงค์ :

ศึกษาการอบรบโมเดล และตรวจจับสินค้า เพื่อการเรียนรู้เชิงลึกของโมเดล Yolov3 และปรับค่า Config TRAINING Input size=608*608 Batch=8 Subdivisions=2

รายละเอียดการทดลอง :

เนื่องจากการทดลองครั้งที่ 3 ทำให้คาดเดาจะได้ความแม่นยำเพิ่มมากขึ้นเมื่อปรับค่าconfig ให้มากขึ้น ในครั้งนี้เราจึงเพิ่ม Batch size ให้สูงที่สุดที่ทรัพยากรด้าน Hardware สามารถทำงานได้

ผลลัพธ์การทดลอง :

ได้ความแม่นยำสูงและเสถียรกว่าเดิมอย่างมาก และเริ่มOverfitting ที่ประมาณ 5000 epoch

ตาราง 4.4 การทดลองที่ 4

Input size=608*608 Batch=8 Subdivisions=4	average IoU(%)	mAP (%)
1000	24.25%	35.24%
2000	55.87%	75.71%
3000	67.78%	83.31%
4000	73.09%	91.65%
5000	76.42%	94.43%
6000	70.33%	87.71%

4.5 สรุปผลที่ได้จากการทดลอง

สามารถสรุปได้ว่าในตัวแปรที่ควบคุมได้ง่ายที่เพิ่มความแม่นยำของโมเดลให้มากขึ้นนั้นก็คือ INPUT SIZE , จำนวนรูปภาพที่นำมาเข้าการฝึกอบรม,Batch ,Subdivisions โดยที่ถ้าต้องการความแม่นยำที่สูงควรตั้งค่า Batch size 64 และ Subdivisions 16 ตามที่ Darknet แนะนำไว้สำหรับ Yolov3 แต่ก็แลกมาด้วยการใช้ทรัพยากรด้าน Hardware ที่ค่อนข้างสูง

บทที่ 5

บทสรุปและข้อเสนอแนะ

5.1 บทสรุป

สามารถสรุปได้ว่าในตัวแปรที่ควบคุมได้ง่ายที่เพิ่มความแม่นยำของโมเดลให้มากขึ้นนั้นก็คือ INPUT SIZE , จำนวนรูปภาพที่นำมาเข้าการฝึกอบรม,Batch ,Subdivisions โดยที่ถ้าต้องการความแม่นยำที่สูงควรใช้ Yolov3.cfg ในการฝึกอบรมจะได้ความแม่นยำที่สูงมากแต่ก็แลกมาด้วยซึ่งทรัพยากรและเวลาที่ใช้ในการฝึกอบรมที่เพิ่มขึ้นอย่างมาก ควรใช้ภาพ 2000 เป็นอย่างน้อยต่อ 1 Class ใน Class ที่มีหลายรูปแบบหน้าตา แต่เนื่องด้วยการที่คลาสของโครงงานนี้มีอยู่รูปลักษณ์เดียวจึงใช้รูปภาพในการฝึกอบรมที่ไม่มากแต่ได้ความแม่นยำที่ไม่แย่จนเกินไป ส่วนตัวแปรที่ควบคุมได้ยาก อย่างเช่นความชัด แสง และความละเอียดของภาพที่นำมาตรวจจับวัตถุ รูปแบบของวัตถุในรูปภาพที่ทำการนำมาเข้าการฝึกอบรมนั้นสามารถทำให้ความแม่นยำลดลงได้

5.2 ปัญหาและอุปสรรค

1) การที่ได้ความแม่นยำที่ต่ำนั้นเนื่องจากมีข้อมูลที่นำไปฝึกอบรมน้อยเกินไป และ การจะได้มาซึ่งความแม่นยำในการตรวจจับวัตถุอาจต้องใช้ Dataset จำนวนมากและGPU ที่มีRAM สูงเนื่องจาก GPU Memory ไม่เพียงพออบรม

2) Software ที่ใช้ในการทดลองมีระบบปฏิบัติการที่แตกต่างกันโดยการเขียนโปรแกรมในตอนเริ่มต้นนั้นทำบน Windows 10 แต่ตัวเครื่องที่ใช้ในการประมวลผลแอพพลิเคชั่นนั้นเป็น Linux ทำให้บางเครื่องมือนั้น ไม่สามารถทำงานได้เหมือนกัน

3) Hardware ที่ใช้ในการทดลองมีสถาปัตยกรรมที่แตกต่างกันโดยการเขียนโปรแกรมในตอนเริ่มต้นนั้นทำบน PC ซึ่งเป็น X86 แต่ตัวเครื่องที่ใช้ในการประมวลผลแอปพลิเคชั่นนั้นเป็น aarch64 ทำให้บางเครื่องมือนั้น ไม่สามารถทำงานได้เหมือนกันและความเร็วในการประมวลผลต่างกันมาก

4) ตัวแปรในการอบรม Model การเรียนรู้เชิงลึกนั้นมีมากมายจนเกินไป ทำให้ยากในการหา model ที่ดีที่สุดในระยะเวลาอันสั้น อย่างเช่น ขนาดวัตถุ สีของวัตถุ ความชัดของภาพ ขนาดของภาพ มุมของกล้อง จำนวนข้อมูลที่นำไปอบรม ความเสมือนจริงของข้อมูลทดสอบและ Config ต่าง ๆ ในโมเดลที่อบรม

5.3 แนวทางแก้ปัญหา

1. ศึกษาก Tool ที่ใช้ในระบบปฏิบัติการและสถาปัตยกรรมให้เหมาะสมแก่ตัว อุปกรณ์ที่ประมวลผลแอพพลิเคชั่น
2. ฝึกอบรมโมเดลในเครื่องที่มี GPU RAM สูงๆ เพื่อความแม่นยำมากขึ้นและ นำโมเดลที่ได้ออกมาประมวลผลในเครื่องทั่วไป
3. เมื่อมีข้อมูลที่นำไปฝึกอบรมน้อย ควรปรับสิ่งแวดล้อมในระบบในภาพที่นำไปอบรมเนื่องใกล้เคียงกับภาพที่ต้องนำมาตรวจจับวัตถุ จะมีความแม่นยำเพิ่มมากขึ้น
4. ลดการทำงานด้านกราฟิกลงเพื่อเพิ่มประสิทธิภาพในการทำงานของเครื่อง เนื่องจากเมมโมรี่ของอุปกร์ไม่เพียงพอต่อการรันหลายแอพพลิเคชั่นพร้อม ๆ กัน

5.4 ข้อเสนอแนะ

1) หากต้องการโมเดลที่มีความแม่นยำสูงต้องใช้ ข้อมูลในการอบรมอย่างมากและปรับ Config ใน file นามสกุล .cfg ของ Yolov3 ให้มีประสิทธิภาพยิ่งขึ้น แต่ก็ใช้ทรัพยากรด้าน Hardware สูงมากขึ้นเช่นกัน

2.) เพิ่มระบบสมาชิกเพื่อที่จะทำให้สามารถจัดโปรโมชั่นพิเศษและวิเคราะห์ทางด้านสถิติได้ละเอียดยิ่งขึ้น โดยทำการจัดแบ่ง เพศ อายุ อาชีพ ก็สามารถนำไปทำนายอนาคตเพื่อเพิ่มกำไรในอนาคตได้

3) เพิ่มช่องทางการจ่ายเงินเพื่อความสะดวกแก่ลูกค้า

บรรณานุกรม

Joseph Chet Redmon. 2018.YOLO:Real-Time Object Detection.

[Online]Available: https://pjreddie.com/darknet/yolo/

Matterport, Inc. 2017. Mask R-CNN for object detection and instance segmentation on Keras and TensorFlow.

[Online]Available : https://github.com/matterport/Mask_RCNN

Kaiming He, Georgia Gkioxari, Piotr Dollár and Ross Girshick. 2017. Mask R-CNN Computer Vision and Pattern Recognition.

[Online]Available : https://arxiv.org/pdf/1703.06870

Heng2j. 2018. Mask R-CNN for Object Detection and Segmentation.

[Online]Available : https://github.com/heng2j/Mask_RCNN

Arthur Ouaknine. 2018 Deep Learning Algorithm for Object Detection.

[Online]Available: https://medium.com/zylapp/review-of-deep-learning-algorithms-for-object-detection-c1f3d437b852