

โครงการเลขที่ วศ.คพ. P008-2/2565

เรื่อง

แคลสแนป : ระบบจัดการการซื้อขายในร้านค้าปลีกอัตโนมัติด้วยตนเองโดยใช้ปัญญา
ประดิษฐ์

โดย

นายพงศกร รัตนพันธ์ รหัส 630610749
นางสาวศุภริภา ศิลปสิทธิ์ รหัส 690610969

โครงการนี้

เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเชียงใหม่
ปีการศึกษา 2565

PROJECT No. CPE P008-2/2565

**CapSnap : Retail self-checkout system using Computational Intelligence
Technique**

Pongsakorn Rattanapan 630610749

Suparida Silpasith 690610969

**A Project Submitted in Partial Fulfillment of Requirements
for the Degree of Bachelor of Engineering
Department of Computer Engineering
Faculty of Engineering
Chiang Mai University
2022**

หัวข้อโครงการ : แอปสแนป : ระบบจัดการการซื้อ-ขายในร้านค้าปลีกอัตโนมัติด้วยตนเองโดยใช้ปัญญาประดิษฐ์
: CapSnap : Retail self-checkout system using Computational Intelligence Technique
โดย : นายพงศกร รัตนพันธ์ รหัส 630610749
นางสาวศุภริภา ศิลปสิทธิ์ รหัส 690610969
ภาควิชา : วิศวกรรมคอมพิวเตอร์
อาจารย์ที่ปรึกษา : รศ.ดร. ศันสนีย์ เอื้อพันธ์วิริยะกุล
ปริญญา : วิศวกรรมศาสตรบัณฑิต
สาขา : วิศวกรรมคอมพิวเตอร์
ปีการศึกษา : 2565

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเชียงใหม่ ได้อนุมัติให้โครงการนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต (สาขาวิศวกรรมคอมพิวเตอร์)

..... หัวหน้าภาควิชาวิศวกรรมคอมพิวเตอร์
(รศ.ดร. สันติ พัทธกิจนุกูร)

คณะกรรมการสอบโครงการ

..... ประธานกรรมการ
(รศ.ดร. ศันสนีย์ เอื้อพันธ์วิริยะกุล)

..... กรรมการ
(ผศ.ดร. เกษมสิทธิ์ ตียพันธ์)

..... กรรมการ
(รศ.ดร. นิพนธ์ อีระอำพน)

หัวข้อโครงการ : แอปสแนป : ระบบจัดการการซื้อ-ขายในร้านค้าปลีกอัตโนมัติด้วยตนเองโดยใช้ปัญญาประดิษฐ์
: CapSnap : Retail self-checkout system using Computational Intelligence Technique
โดย : นายพงศกร รัตนพันธ์ รหัส 630610749
นางสาวศุภริญา ศิลปสิทธิ์ รหัส 690610969
ภาควิชา : วิศวกรรมคอมพิวเตอร์
อาจารย์ที่ปรึกษา : รศ.ดร. ศันสนีย์ เอื้อพันธ์วิริยะกุล
ปริญญา : วิศวกรรมศาสตรบัณฑิต
สาขา : วิศวกรรมคอมพิวเตอร์
ปีการศึกษา : 2565

บทคัดย่อ

CapSnap ระบบการซื้อของด้วยตนเองในร้านค้าปลีกโดยใช้ Computational Intelligence

เพื่อลดความยุ่งยากในการซื้อสินค้าในร้านค้าปลีกโดยใช้ application ที่ช่วยให้ลูกค้าสามารถสแกนและระบุรายการสินค้าด้วยตนเอง จากนั้นระบบจะใช้ computational intelligence เพื่อบอกรายละเอียดและราคาของสินค้าแต่ละชนิด โดยใช้ภาพจาก application ซึ่งจะเป็นประโยชน์ต่อคนที่มีปัญหาในการมองเห็นเพื่อช่วยในการเลือกซื้อสินค้า ภายในแอปพลิเคชันจะมีการเชื่อมต่อและดึงข้อมูลสินค้าของแต่ละร้านที่มีข้อมูลของสินค้าที่แตกต่างกัน เมื่อลูกค้าเข้าร้านค้าร้านไหนก็จะดึงข้อมูลของร้านค้านั้น และผู้ซื้อสามารถ check out รวมถึงชำระเงินอัตโนมัติเมื่อเดินจากออกร้าน นอกจากนี้ระบบยังมี web dashboard เพื่อช่วยจัดการสินค้าในคลังสำหรับร้านค้าเพื่อจัดการสินค้าและแสดงผลข้อมูลการชำระเงิน และสต็อกสินค้าของร้านค้า ช่วยให้ลูกค้าปลีกจัดการข้อมูลสินค้าในคลังและข้อมูลการขายได้อย่างมีประสิทธิภาพมากขึ้น

กิตติกรรมประกาศ

Your acknowledgments go here. Make sure it sits inside the acknowledgment environment.

นายพงศกร รัตนพันธ์
นางสาวศุภริญา ศิลปสิทธิ์
25 พฤษภาคม 2563

สารบัญ

บทคัดย่อ	ข
กิตติกรรมประกาศ	ค
สารบัญ	ง
สารบัญรูป	จ
สารบัญตาราง	ฉ
1 บทนำ	1
1.1 ที่มาของโครงการ	1
1.2 วัตถุประสงค์ของโครงการ	1
1.3 ขอบเขตของโครงการ	1
1.4 ประโยชน์ที่ได้รับ	1
1.5 เทคโนโลยีและเครื่องมือที่ใช้	2
1.6 แผนการดำเนินงาน	2
1.7 บทบาทและความรับผิดชอบ	2
1.8 ผลกระทบด้านสังคม สุขภาพ ความปลอดภัย กฎหมาย และวัฒนธรรม	3
2 ทฤษฎีที่เกี่ยวข้อง	4
2.1 WebRTC for Streaming image	5
2.2 Transfer Learning	5
3 โครงสร้างและขั้นตอนการทำงาน	7
3.1 ชุดข้อมูลฝึกสอน Product database	7
3.2 Transfer Learning	7
3.2.1 Model Architecture	8
3.3 classification products	8
3.4 การพัฒนาweb dashboard และ application	9
3.5 แผนภาพกระแสข้อมูล (Data Flow Diagram)	9
4 การทดลองและผลลัพธ์	10
5 บทสรุปและข้อเสนอแนะ	11
5.1 สรุปผล	11
5.2 ปัญหาที่พบและแนวทางการแก้ไข	11
5.3 ข้อเสนอแนะและแนวทางการพัฒนาต่อ	11
ก The first appendix	13
ประวัติผู้เขียน	14

สารบัญรูป

2.1	Overall project structure	4
2.2	webrtc structure	5
2.3	The concept of transfer learning	6
3.1	Data Flow Diagram	9

สารบัญตาราง

1.1 Planning	2
------------------------	---

บทที่ 1

บทนำ

1.1 ที่มาของโครงการ

จากปัญหาที่พบเจอในการจัดการร้านค้าที่มีลูกค้าจำนวนมากและต้องการจ่ายเงินกับพนักงานเพื่อทำการเช็คเอาท์ การรอคิวที่นาน และ การใช้พื้นที่ในการรอคิวอาจจะทำให้ลูกค้าไม่พึงพอใจได้ และการที่ไม่อยากให้ลูกค้าต้องมาต่อแถวที่ยาว วิธีการหนึ่งคือทำให้ลูกค้าสามารถจ่ายเงินของสินค้าด้วยตัวเองได้ (self-service) เป็นการที่ลูกค้าบริการตนเองโดยการหยิบสินค้าและจ่ายเงินให้ทางร้านค้าด้วยตัวเอง เช่นผ่านทาง prompt-pay qr code ซึ่งในร้านค้าที่ลูกค้าบริการตัวเอง ก็จะมีปัญหาในการจัดการสินค้าในร้านค้า เนื่องจากในบางครั้งมีสินค้าที่หมด เนื่องจากไม่ทราบว่าขายสินค้าชิ้นใดไปแล้วเท่าไร เพราะไม่มีการจัดการสินค้าที่ดี โดยส่วนใหญ่ร้านค้าที่เป็น self-service พนักงานจะต้องคอยนับจำนวนสินค้าที่เหลืออยู่ภายในร้านเพื่อทราบว่าขายอะไรไปแล้ว หรือ รอให้สินค้าประเภทนั้นๆหมด แล้วจึงค่อยเติม stock ในทีเดียว

ผู้จัดทำจึงพัฒนาระบบ self-service CapSnap ลดปัญหาการรอคิวของลูกค้าโดยการให้ลูกค้าสามารถบริการตนเองได้ โดยใช้ Computational Intelligence ในการแยกแยะสินค้า ซึ่งทำให้ลูกค้าไม่ต้องเลือกเพื่อบันทึกในระบบว่าซื้อสินค้าชนิดใด และสามารถช่วยให้ร้านค้าสามารถจัดการสินค้าได้อย่างมีประสิทธิภาพจากการเก็บบันทึกจำนวนการขายของสินค้าแต่ละชนิด และ เพิ่มประสิทธิภาพในการบริหารจัดการสินค้าในร้านค้า เพิ่มประสบการณ์ของลูกค้า และ ร้านค้าที่ต้องการเป็น self-service และ ลดการที่พนักงานต้องมาคอยจัดการนับสินค้าภายในร้าน

1.2 วัตถุประสงค์ของโครงการ

1. เพื่อช่วยให้ลูกค้าสามารถบริการตนเองในร้านค้าได้ และให้ร้านค้าสามารถตรวจสอบสินค้าที่ขายได้
2. เพื่อพัฒนาระบบ แยกแยะชนิดสินค้าโดยใช้ Computational Intelligence ได้
3. เพื่อสร้างเว็บไซต์และแอปโทรศัพท์เพื่อให้ผู้ใช้งานสามารถใช้งานระบบได้

1.3 ขอบเขตของโครงการ

ข้อมูลที่ใช้ในการ train ระบบ Computational Intelligence ในการแยกแยะชนิดสินค้า มาจาก ร้านค้า ห้อง 422 ในภาคคณะวิศวกรรมศาสตร์ สาขา มหาวิทยาลัยเชียงใหม่ โดยมีสินค้าประมาณ 100 ชนิด

1.4 ประโยชน์ที่ได้รับ

1. แอปโทรศัพท์ที่ลูกค้าสามารถแยกแยะชนิดสินค้าเพื่อให้ลูกค้าสามารถจ่ายเงินให้กับสินค้านั้นๆได้ด้วยตัวเอง โดยไม่ต้องคอยเลือกชนิดของสินค้านั้นๆ
2. เว็บไซต์ที่แสดงจำนวนการขายของสินค้าแต่ละชนิดเพื่อให้ร้านค้าสามารถจัดการสินค้าได้

1.5 เทคโนโลยีและเครื่องมือที่ใช้

1. Python + aiortc : สำหรับพัฒนาในส่วนของ Backend , การรับข้อมูล streaming จาก Application , การฝึกสอนโมเดล , และการ classify Product โดยไม่ต้องคอยเลือกชนิดของสินค้านั้นๆ
2. Laravel (Web Application Framework) : สำหรับพัฒนา Frontend ในส่วนหน้าเว็บไซต์ของร้านค้า web dashboard
3. Flutter + webrtc (Mobile Application Framework) : สำหรับพัฒนา Frontend ในส่วน Application ในโทรศัพท์ของลูกค้า และใช้ webrtc การ streaming ภาพจาก camera ไปยัง Backend
4. MongoDB : สำหรับเก็บข้อมูลชนิดของสินค้าที่เอาไปใช้ฝึกสอนโมเดล , ใช้ในการแสดงราคาบน Frontend และเก็บจำนวนของสินค้าที่ขายไปแล้ว

1.6 แผนการดำเนินงาน

ขั้นตอนการดำเนินงาน	ม.ค. 2566	ก.พ. 2566	มี.ค. 2566	เม.ย. 2566	พ.ค. 2566	มิ.ย. 2566	ก.ค. 2566	ส.ค. 2566	ก.ย. 2566	ต.ค. 2566	พ.ย. 2566	ธ.ค. 2566	ม.ค. 2567	ก.พ. 2567	มี.ค. 2567
Planning															
Document															
Back-end development															
App development															
Dashboard development															
Testing															

ตารางที่ 1.1: Planning

1.7 บทบาทและความรับผิดชอบ

นายพงศกร รัตนพันธ์ รหัส 630610749 ทำในส่วนของ Backend การรับข้อมูล streaming จาก Application , การฝึกสอนโมเดล , และการ classify Product , เก็บข้อมูลรูปภาพของสินค้าและราคาลง MongoDB และทำด้าน Application ซึ่งใช้ webrtc การ streaming ภาพจาก camera ไปยัง Backend

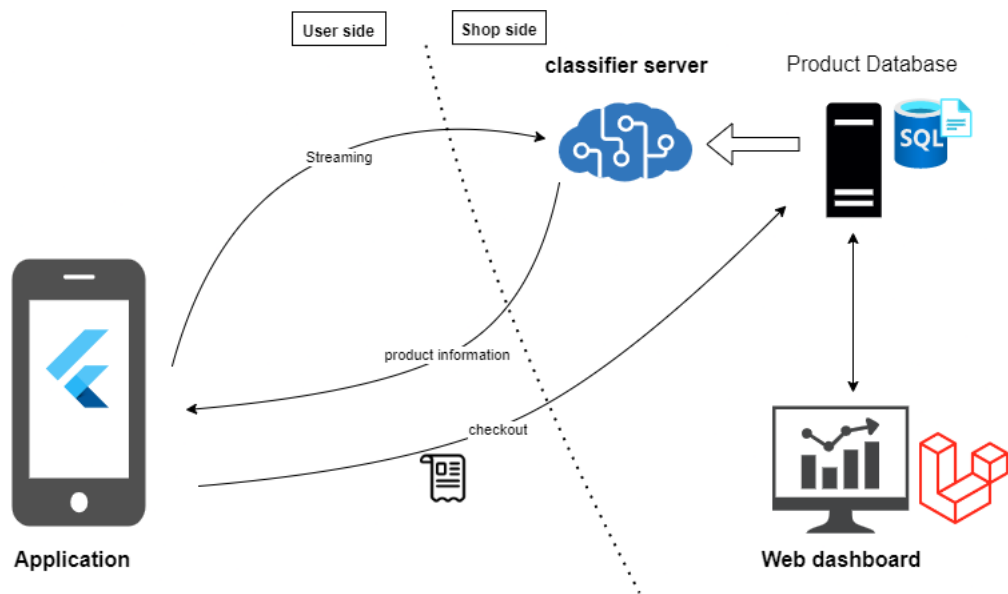
นางสาวศุภริญา ศิลปสิทธิ์ รหัส 690610969 ทำในส่วนของ Frontend ในส่วนหน้าเว็บไซต์ของร้านค้าที่แสดงข้อมูลจำนวนการขายสินค้าในแต่ละชนิดที่ขายไป web dashboard , เก็บข้อมูลรูปภาพของสินค้าและราคาลง MongoDB และ Frontend ในส่วน Application ในโทรศัพท์ของลูกค้า

1.8 ผลกระทบด้านสังคม สุขภาพ ความปลอดภัย กฎหมาย และวัฒนธรรม

โครงการนี้ลดความซับซ้อนและเวลาที่ลูกค้าจะต้องไปรอตอแถวเพื่อจ่ายเงินของสินค้า รวมถึงทำให้พนักงานของร้านค้า ไม่ต้องคอยนับจำนวนสินค้าในร้านค้า ในร้านค้าที่เป็นระบบ Self-Service อีกทั้งยังเป็นอีกทางเลือกหนึ่งในการที่ร้านค้าจะมาใช้ระบบ Self-Service ที่มีการจัดการที่ดี และ ส่งเสริมวัฒนธรรมในการบริการตนเองของลูกค้า

บทที่ 2 ทฤษฎีที่เกี่ยวข้อง

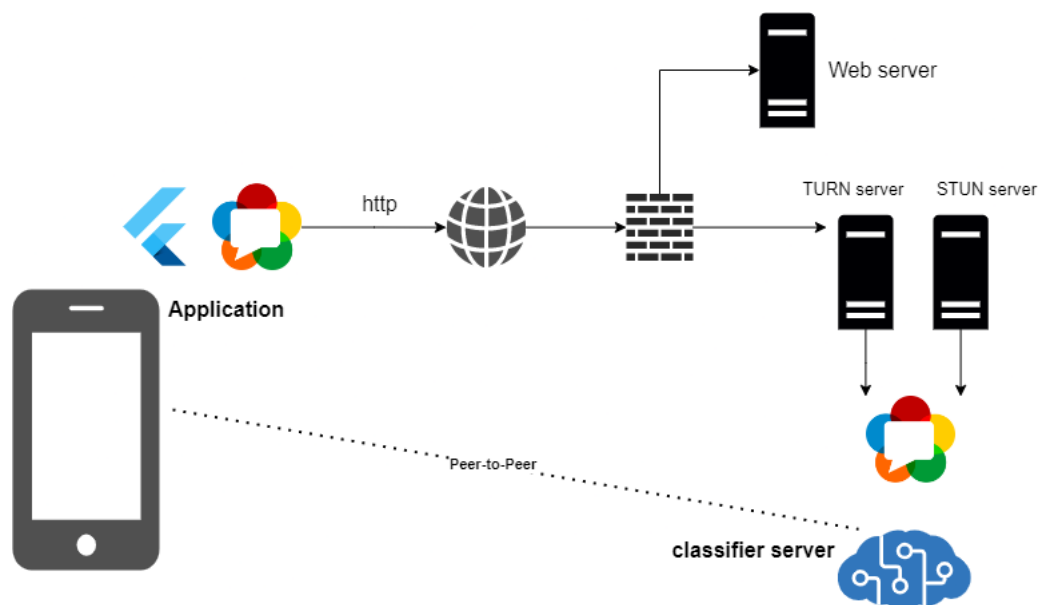
โครงงานนี้ได้นำองค์ความรู้ในด้านของ Computational Intelligence และ การ streaming แบบ Peer-to-peer ของรูปภาพจาก application ไปยัง backend ผ่าน WebRTC (Web Real Time Communications) เพื่อให้ backend ที่เป็น Computational Intelligence ทำการ classification products โดยโครงสร้างของระบบจะเป็นดังรูป 2.1



รูปที่ 2.1: Overall project structure

2.1 WebRTC for Streaming image

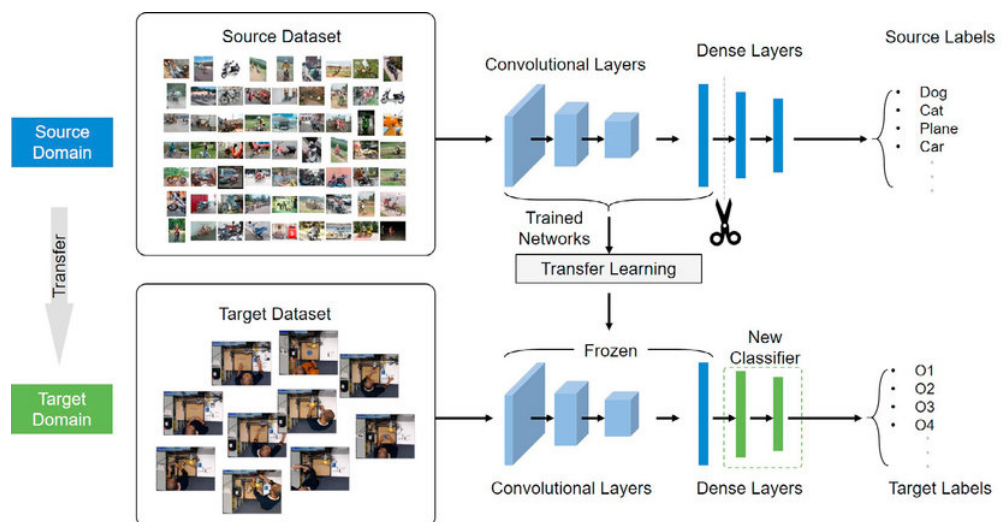
WebRTC (Web Real-Time Communication) เป็น open-source ที่ให้บริการ web browsers และ mobile applications ด้วยการสื่อสารแบบเรียลไทม์ (RTC) ผ่าน (API) ทำให้การ Communication ด้วยเสียงและวิดีโอได้ผ่าน Peer-to-peer โดยตรงตามรูป 2.2



รูปที่ 2.2: webrtc structure

2.2 Transfer Learning

เป็นเทคนิคที่นำ model ที่ผ่านการฝึกฝนจนแก้ ปัญหาในงานอื่นๆที่มีความคล้ายคลึงกัน นำมาเป็น model ตั้งต้น สำหรับ model ในการแก้ปัญหาใหม่ๆ ตัวอย่างเช่น โมเดลที่ได้รับการฝึกฝนให้จดจำวัตถุในภาพสามารถใช้ เพื่อระบุวัตถุที่คล้ายกันในภาพต่างๆ ได้ แม้ว่าภาพใหม่จะมีสภาพแสงหรือพื้นหลังต่างก็ตาม กฎเกณฑ์สำคัญคือการระบุคุณสมบัติทั่วไปหรือการเป็นตัวแทนที่ใช้ร่วมกันระหว่างโดเมนต้นทางและโดเมนเป้าหมาย วิธี Transfer Learning ที่ได้รับความนิยมมากที่สุดวิธีหนึ่งคือ fine-tuning คือการใช้โมเดลที่ผ่านการ pre-trained มาแล้ว นำมา train ต่อบน ชุดข้อมูลใหม่ และใช้ learning rate น้อยๆ เพื่อป้องกัน weight ที่เคยผ่านการ train จนมีความแม่นยำเปลี่ยนแปลงไปมาก จนไม่มีความแม่นยำ อีกวิธีหนึ่งคือ feature extraction ซึ่งเกี่ยวข้องกับการใช้ โมเดลที่ผ่านการ pre-trained เป็นตัวแยกคุณลักษณะของ ข้อมูล และ สร้าง model ใหม่เพื่อ train จากคุณลักษณะเหล่านี้ที่ model ตั้งต้นแบ่งแยกออกมาได้



รูปที่ 2.3: The concept of transfer learning

บทที่ 3

โครงสร้างและขั้นตอนการทำงาน

ในบทนี้จะกล่าวถึงหลักการ และการออกแบบระบบ

3.1 ชุดข้อมูลฝึกสอน Product database

ข้อมูลที่ใช้ในการ train mode โดยมีสินค้าประมาณ 100 ชนิด โดยจัดเก็บข้อมูลใช้กล้องมือถือ ในการถ่ายภาพในมุมต่างๆ ของสินค้าชนิดนั้นๆ ตามมุมต่างๆ จำนวนชนิดละ N รูป โดยจัดเก็บใน และทำการดึงข้อมูลมา train ผ่าน Google Colab โดยโครงสร้างการเก็บข้อมูลจะเป็นดังรูป

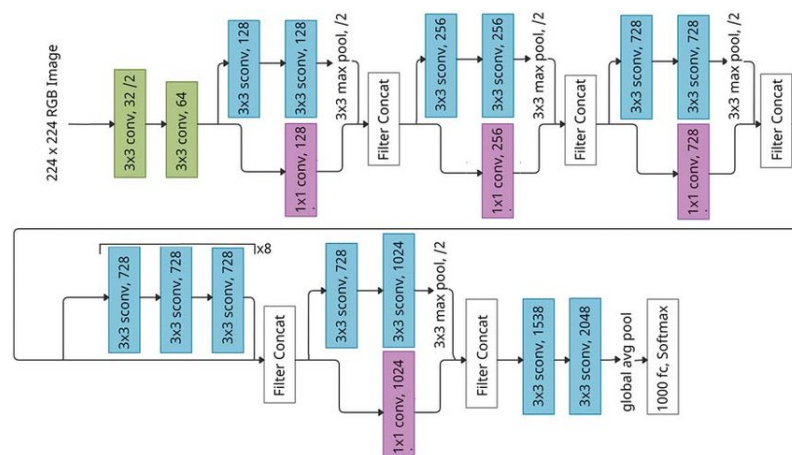
```
directory/  
  class_label_1/  
    image1  
    image2  
    .  
    .  
  class_label_2/  
    .  
    .
```

โดย

3.2 Transfer Learning

Xception is a convolutional neural network that is 71 layers deep. You can load a pre-trained version of the network trained on more than a million images from the ImageNet database [1]. The pretrained network can classify images into 1000 object categories, such as keyboard, mouse, pencil, and many animals.

ซึ่งสามารถ



3.2.1 Model Architecture

โดยโมเดลในโครงการนี้จะใช้ Xception pre-trained มาใช้ในการแยกคุณลักษณะของข้อมูล และสร้างโมเดลมาต่อท้ายเพื่อ เรียนรู้จากข้อมูลที่ Xception แยกออกมาได้

Model: "model_1"		
Layer (type)	Output Shape	Param #
=====		
input_4 (InputLayer)	[(None, 224, 224, 3)]	0
xception (Functional)	(None, 7, 7, 2048)	20861488
flatten_1 (Flatten)	(None, 108352)	0
dense_3 (Dense)	(None, 32)	3211296
dense_4 (Dense)	(None, 16)	528
dense_5 (Dense)	(None, 48)	816
=====		
Total params: 24,074,120		
Trainable params: 3,212,640		
Non-trainable params: 20,861,480		

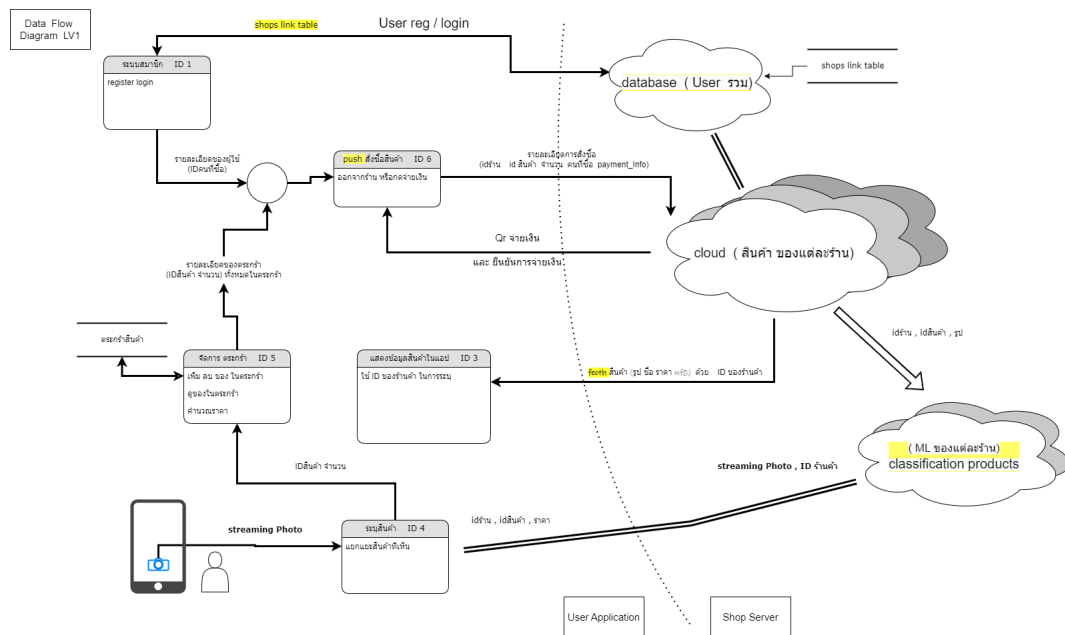
นำมาต่อด้วย classifier ซึ่งเป็น Dense Layers สำหรับการ classify ชนิดของ products จากชุดข้อมูล Subsection 1 text

3.3 classification products

3.4 การพัฒนา web dashboard และ application

ออกแบบ UI/UX ของเว็บไซต์และ application ด้วย Figma ในส่วน application จะใช้งาน Flutter และ ภาษา Dart และใช้ module WebRTC ในการติดต่อกับ classifier ของ Backend

3.5 แผนภาพกระแสข้อมูล (Data Flow Diagram)



รูปที่ 3.1: Data Flow Diagram

บทที่ 4
การทดลองและผลลัพธ์

บทที่ 5

บทสรุปและข้อเสนอแนะ

- 5.1 สรุปผล
- 5.2 ปัญหาที่พบและแนวทางการแก้ไข
- 5.3 ข้อเสนอแนะและแนวทางการพัฒนาต่อ

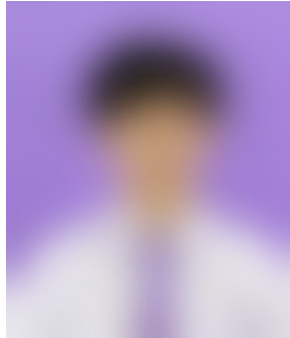
ภาคผนวก

ภาคผนวก ก

The first appendix

Text for the first appendix goes here.

ประวัติผู้เขียน



Your biosketch goes here. Make sure it sits inside the biosketch environment.