

รายงานความก้าวหน้าวิชา CE Project

ครั้งที่ 4

ระหว่างวันที่ 27 ก.พ. 66 ถึงวันที่ 24 มี.ค. 66

- ชื่อโครงการ (อังกฤษ) Marketplace for Image Processing Application using Task Management on Cluster Computing System

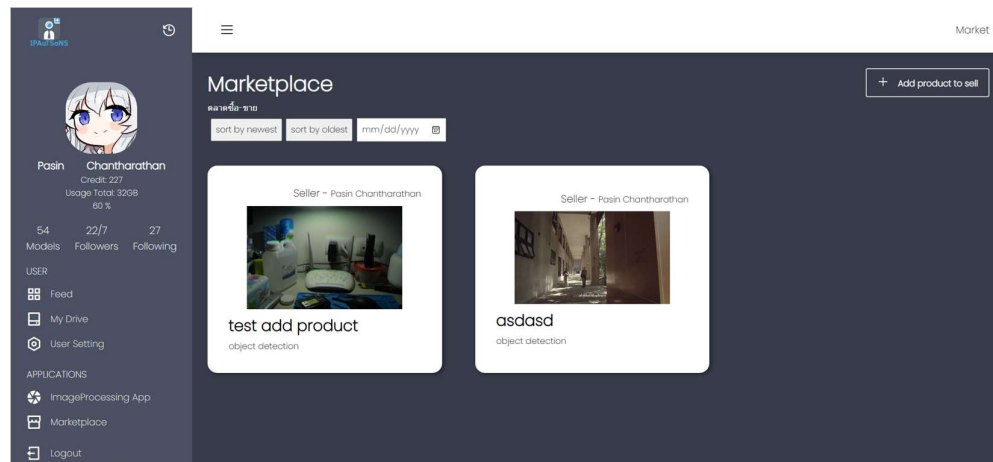
- การดำเนินงานมีความก้าวหน้า 84% (ใช้ค่า % **Complete** จาก MS Project)

มีความก้าวหน้าเพิ่มขึ้นจากรายงานความก้าวหน้า ครั้งก่อน 14 %

☐ เร็วกว่าแผน 0 วัน ☐ ช้ากว่าแผน 0 วัน

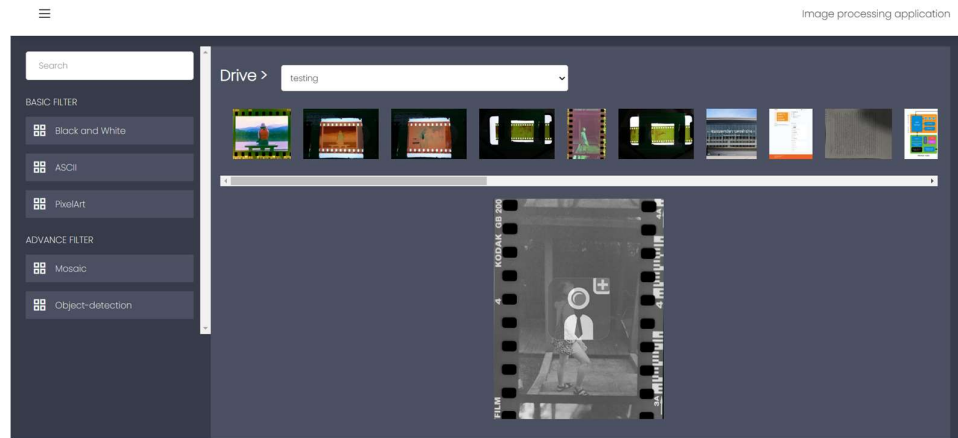
- รายละเอียดความก้าวหน้า

ในส่วนของ Web application นั้น หลังจากได้ทำการเพิ่มตัวในส่วนหน้าตลาดซื้อขาย (Marketplace) เข้าไปโดยจะมี 3 ส่วนคือ 1. ส่วนหน้ารวมของตลาด 2. ส่วนหน้าของสินค้าแต่ละชิ้น 3. ส่วนหน้าการเพิ่มสินค้า ซึ่งได้พัฒนาในส่วน Backend ของแต่ละหน้าเอาไว้เพื่อให้แสดงผลได้ตรงตามที่ต้องการเอาไว้

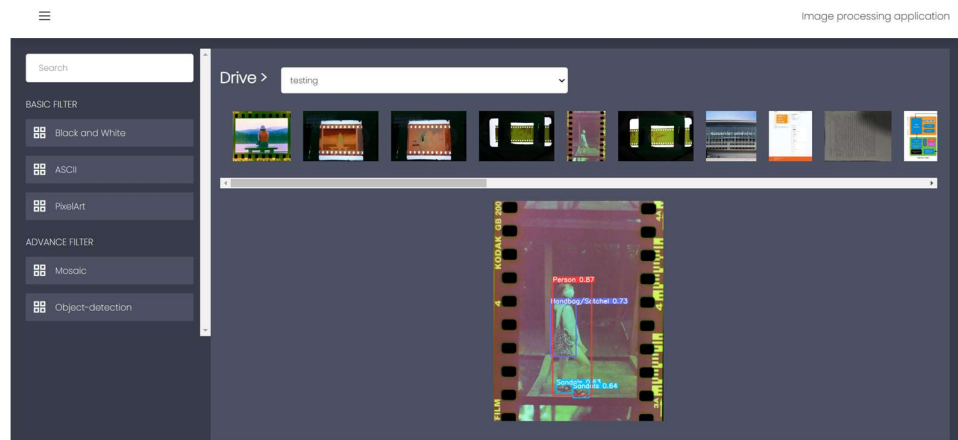


รูปของส่วนหน้ารวมแสดงสินค้าทั้งหมดที่จะมีการ sort และ filter หาสินค้า

และอีกส่วนที่มีการพัฒนาเพิ่มไปก็คือส่วนของการ Preview รูปภาพ ที่ได้ลองสร้างระบบการทำงานขึ้นมาโดยที่จะส่งรูปภาพไปประมวลผลผ่าน API สำหรับการประมวลผลรูปเดี่ยวโดยเฉพาะซึ่งทำให้มีความเร็วในการแสดงผลที่ดีกว่าการสร้าง job มาเพื่อประมวลผลรูปภาพรูปเดียว



รูปของการ preview โดยใช้ Filter Black and white



รูปของการ preview โดยใช้ weight ที่มีความสามารถในการทำ Object detections

และได้ยังมีการปรับปรุงรูปแบบการทำงานของตัว web applications ใหม่โดยการใช้ local storage เข้ามาช่วยเสริมในการแสดงผลและเก็บข้อมูลเบื้องต้นของผู้ใช้ เพื่อลดการเรียกใช้ request จาก frontend ไปยังส่วน backend โดยจะเก็บในส่วน of JWT ที่ใช้ยืนยันตัวตน , ชื่อ-สกุล ของผู้ใช้ , ขนาดของ drive ที่ถูกใช้งานไป , Credit ที่มีอยู่ในระบบ , จำนวน product ที่เป็นเจ้าของ เป็นต้น ซึ่งถ้าหากมีส่วนไหนที่สามารถเก็บใน local storage เพิ่มสามารถลด Traffic ได้อีกก็จะนำมาใส่ไว้ในอนาคต

ในส่วนของ Server นั้น ได้ทำ Deployment ในส่วนของตัว Front-End และ Back-End ทั้ง 2 เครื่อง และได้ทำการสร้างตัว Service บน Kubernetes สำหรับการเข้าใช้งาน Web Application ทั้งพอร์ต 80 และ API Back-End ในพอร์ต 8000 และได้ทำการกำหนด Nginx คู่กับ Ingress ในการเข้ามาใช้งาน Service ของ Web Application ผ่าน 161.246.5.53 เพื่อเข้ามาใช้งาน Service ในส่วนของ Front-End ได้

ในส่วนของ API Process ได้ทำการสร้าง API สำหรับการทำงานด้าน Preview และ แก้ปัญหาในการใช้งาน Model ที่มีขนาดใหญ่ให้ทำงานอยู่ในรูปแบบ API เพื่อรองรับคำสั่งต่อไปโดยได้ใช้ FastAPI ในการสร้าง API ขึ้นมา โดยมี

1. BasicAppAPI สำหรับรองรับการทำงาน BasicApp ในการ Preview

2. AdvanceAppAPI สำหรับรองรับการทำงาน AdvanceApp ในการ Preview

3. YoloAPI สำหรับรองรับการทำงาน Model Detection ทั้ง Preview และ ในส่วนของการประมวลผล

โดยได้ทำการสร้าง Application สำหรับการใช้งานประมวลผลโมเดล YOLOv5 เพื่อการตรวจจับวัตถุโดยใช้ Weight ของผู้ใช้งานเรียบร้อยแล้ว และ ได้ทำการทดสอบการทำงานเรียบร้อยแล้ว

4. ปัญหาที่เกิดขึ้นและแนวทางการแก้ไข

ปัญหาที่พบในการพัฒนาของส่วนของ Web application คือในส่วนของการทำงาน preview ในช่วงแรกการเรียกใช้ API ในการประมวลผลรูปเพื่อ preview นั้นมีปัญหาจากการที่ส่งไฟล์รูปกลับมาในรูปแบบ Byte string ซึ่งนำไปประมวลผลด้านอื่นๆ ต่อได้ยาก เช่นการบีบอัด หรือ การแปลงขนาด เป็นต้น ซึ่งในปัญหานี้ได้มีการนำ Library เสริมเข้ามาใช้เพื่อช่วยในการแปลงค่าที่ได้มาในรูปแบบ Byte string ให้เป็น Pillow-Image เพื่อสามารถนำไปใช้งานต่อในส่วนอื่นๆ ได้รวมไปถึงการเข้ารหัสแบบ Base64 เพื่อให้สามารถส่งกลับมาแสดงผลที่ หน้า Frontend ได้ และปัญหาอีกส่วนคือในการ Upload file weight เพื่อจะลงขาย หน้า Product เจอปัญหาว่า ถ้าไฟล์ Weight มีขนาดใหญ่เกินไป มักจะมีปัญหากระหว่างการ Upload ซึ่งทำให้การ upload ทำไม่สำเร็จเลยมีแนวคิดที่จะแบ่ง file weight เป็นส่วนๆก่อนจะส่งแต่ละส่วนไปบันทึก

ปัญหาที่พบในการพัฒนาของส่วนของ Server คืออินเทอร์เน็ตของสถาบันมีการบล็อกการเข้าถึง และ การใช้งานบางส่วน จึงทำให้การดำเนินการล่าช้าลง และ ต้องปรับเปลี่ยนบางส่วนในงานทำเช่น พอร์ต หรือ การใช้งานบนเซิร์ฟเวอร์เท่านั้น

5. สิ่งที่จะดำเนินการต่อไป

สิ่งที่จะดำเนินการต่อไปคือ

- พัฒนาต่อในส่วนของการ Payment การหักค่าใช้จ่ายในการประมวลผล
- พัฒนาในส่วนของการ weight model checker เพื่อตรวจสอบ Weight ที่อัปโหลดขึ้นมาบน Web application
- ตัว API ของ GAN ในการทำ Preview
- Deployment ระบบทั้งหมด และ ทดสอบ เก็บสถิติ