คู่มือการติดตั้ง

1. **ภาพรวมระบบ**

ร้านค้าสำหรับแอพพลิเคชั่นการประมวลผลภาพซึ่งจัดการงานบนระบบ ประมวลผลแบบกลุ่ม พัฒนาด้วย Django, VueJS ในส่วนของ Web Application

ในส่วนของ API สำหรับการประมวลผลแสดงผลภาพ ใช้ Python Script โดยมีการเรียกใช้งานผ่าน FastAPI

1. **ความต้องการของระบบ**

1) เครื่อง Server ต้องติดตั้ง Docker

2) เครื่อง Server สำหรับ Deploy ต้องมีการติดตั้ง Kubernetes

3) เครื่อง Server ต้องมีการติดตั้ง Storage แยกในระบบ Network

4) clone repo github จาก https://github.com/SuteeSaraphan/IPAuTSoNS

1. **ขั้นตอนการติดตั้ง**

**3.1 การติดตั้งส่วนของเว็บแอพพลิเคชั่นบน Kubernetes**

1) ในส่วนของ Front-End ให้ใช้ไฟล์ frontend.yaml ที่อยู่ในโฟลเดอร์ IPAuTSoNS/webb/frontend.yaml ในการสั่งงานด้วยคำสั่ง kubectl apply -f frontend.yaml

2) ในส่วนของ Back-End ให้ใช้ไฟล์ backend.yaml ที่อยู่ในโฟลเดอร์ IPAuTSoNS/webb/backend.yaml ในการสั่งงานด้วยคำสั่ง kubectl apply -f backend.yaml

3) เมื่อสั่งงานด้วยคำสั่งดังข้อที่ 1 และ 2 แล้ว สามารถทำการตรวจเช็คจาก Kubernetes ได้ว่าได้ทำการติดตั้งเสร็จสิ้นแล้ว หรือ ไม่ โดยจะแสดงผลดังภาพ

ภาพ

4) ทำการสั่งสร้าง Service เพื่อเข้าใช้งานในส่วนของ Front-End และ Back-End ด้วยคำสั่งในไฟล์ frontend-serive.yaml และ backend-service.yaml ในโฟลเดอร์ IPAuTSoNS/webb/ โดยมีคำสั่งดังนี้ kubectl apply -f frontend-service.yaml backend-service.yaml

5) เมื่อทำการสร้าง Service เสร็จสิ้นสามารถเข้าทดสอบ Front-End โดยการเข้าไปที่เว็บไซต์ IP ของเครื่อง Server และ ด้วย Port 80 ดั่งภาพตัวอย่างนี้

ภาพ

6) เมื่อทำการสร้าง Service เสร็จสิ้นสามารถเข้าทดสอบ Back-End โดยการเข้าไปที่เว็บไซต์ IP ของเครื่อง Server และ ด้วย Port 8000 ดั่งภาพตัวอย่างนี้

ภาพ

**3.2 การติดตั้งส่วนของ API แอพพลิเคชั่นบน Kubernetes**

1) ในส่วนของ API การแสดงผลตัวอย่างแอพพลิเคชั่นประมวลผลภาพเบื้องต้น ให้ใช้ไฟล์ basicapi.yaml ที่อยู่ในโฟลเดอร์ IPAuTSoNS/webb/basicapi.yaml ในการสั่งงานด้วยคำสั่ง kubectl apply -f basicapi.yaml

2) ในส่วนของ API การแสดงผลตัวอย่างแอพพลิเคชั่นประมวลผลภาพด้วย Model YoloV5 ให้ใช้ไฟล์ yoloapi.yaml ที่อยู่ในโฟลเดอร์ IPAuTSoNS/webb/yoloapi.yaml ในการสั่งงานด้วยคำสั่ง kubectl apply -f yoloapi.yaml

3) ในส่วนของ API การแสดงผลตัวอย่างแอพพลิเคชั่นประมวลผลภาพด้วย Model GAN ให้ใช้ไฟล์ ganapi.yaml ที่อยู่ในโฟลเดอร์ IPAuTSoNS/webb/ganapi.yaml ในการสั่งงานด้วยคำสั่ง kubectl apply -f ganapi.yaml

ภาพ

4) ในส่วนของ API ในการบีบอัดไฟล์ในรูปแบบ zip ไฟล์ ให้ใช้ไฟล์ zipapi.yaml ที่อยู่ในโฟลเดอร์ IPAuTSoNS/webb/zipapi.yaml ในการสั่งงานด้วยคำสั่ง kubectl apply -f zipapi.yaml

5) ทำการสั่งสร้าง Service เพื่อเข้าใช้งานในส่วนของ BasicAPI, YoloAPI, GANAPI และ ZipAPI ด้วยคำสั่งในไฟล์ basicapi-serive.yaml, yoloapi-service.yaml, ganapi-service.yaml และ zipapi-service.yaml ในโฟลเดอร์ IPAuTSoNS/webb/ โดยมีคำสั่งดังนี้ kubectl apply -f basicapi-serive.yaml, yoloapi-service.yaml, ganapi-service.yaml zipapi-service.yaml

6) เมื่อทำการสร้าง Service เสร็จสิ้นสามารถเข้าทดสอบ BasicAPIโดยการเข้าไปที่เว็บไซต์ IP ของเครื่อง Server และ ด้วย Port 4020 ดั่งภาพตัวอย่างนี้

ภาพ

7) เมื่อทำการสร้าง Service เสร็จสิ้นสามารถเข้าทดสอบ YoloAPIโดยการเข้าไปที่เว็บไซต์ IP ของเครื่อง Server และ ด้วย Port 4050 ดั่งภาพตัวอย่างนี้

ภาพ

8) เมื่อทำการสร้าง Service เสร็จสิ้นสามารถเข้าทดสอบ GANAPIโดยการเข้าไปที่เว็บไซต์ IP ของเครื่อง Server และ ด้วย Port 4070 ดั่งภาพตัวอย่างนี้

ภาพ

9) เมื่อทำการสร้าง Service เสร็จสิ้นสามารถเข้าทดสอบ ZipAPIโดยการเข้าไปที่เว็บไซต์ IP ของเครื่อง Server และ ด้วย Port 4090 ดั่งภาพตัวอย่างนี้

ภาพ