

## รายงานความก้าวหน้าวิชา CE Project

ครั้งที่ 6

ระหว่างวันที่ 07 พ.ย. 65 ถึงวันที่ 25 พ.ย. 65

1. ชื่อโครงการ (อังกฤษ) Image Processing Application using Task Scheduling on Network System
2. การดำเนินงานมีความก้าวหน้า 48% (ใช้ค่า % Complete จาก MS Project)

มีความก้าวหน้าเพิ่มขึ้นจากรายงานความก้าวหน้า ครั้งก่อน 4%

☐ เร็วกว่าแผน 0 วัน   ☐ ช้ากว่าแผน 0 วัน

3. รายละเอียดความก้าวหน้า

ในส่วนของ Web application นั้น ได้ทำการแก้ไขในส่วนของการสร้างไฟล์ YAML ในการสั่งให้ Kubernetes ทำงานประมวลผลภาพ เพื่อให้รองรับกับการสั่งงานที่ต้องการรายละเอียดมากขึ้นกว่าเดิม โดยสิ่งที่ส่งเดิมนั้นจะอยู่ในรูปต่อไปนี้

```

apiVersion: batch/v1
kind: Job
metadata:
  name: my-job-id
  namespace: jobdemonamespace
  labels:
    job_name: my-job-id
spec:
  template:
    metadata:
      labels:
        app: my-job-pod-id
        name: my-job-pod-id
    spec:
      containers:
        - image: "shuffler:latest"
          imagePullPolicy: Never
          name: "shuffler"
          command:
            - python3
            - -u
            - ./test.py "userID" "jobID" "appID" "Folder" "Fileseleted"
          args:
            - "Kubernetes"
          restartPolicy: Never

```

รูปของไฟล์ YAML ในการสั่งให้ Kubernetes ทำงานประมวลผลภาพ รูปแบบเก่า

โดยหลังจากที่ปรับเปลี่ยน ได้เพิ่มในส่วนวันเวลาที่ทำการสั่งสร้างไฟล์นี้ และ งานประมวลผล นอกนั้นก็ยังมีเรื่องของ Format ที่ต้องให้ตรงตามต้นแบบอีกด้วย โดยรูปภาพต่อไปนี้จะเป็นรูปภาพตัวอย่าง ไฟล์ YAML ในการสั่งให้ Kubernetes ทำงานประมวลผลภาพรูปแบบใหม่

```

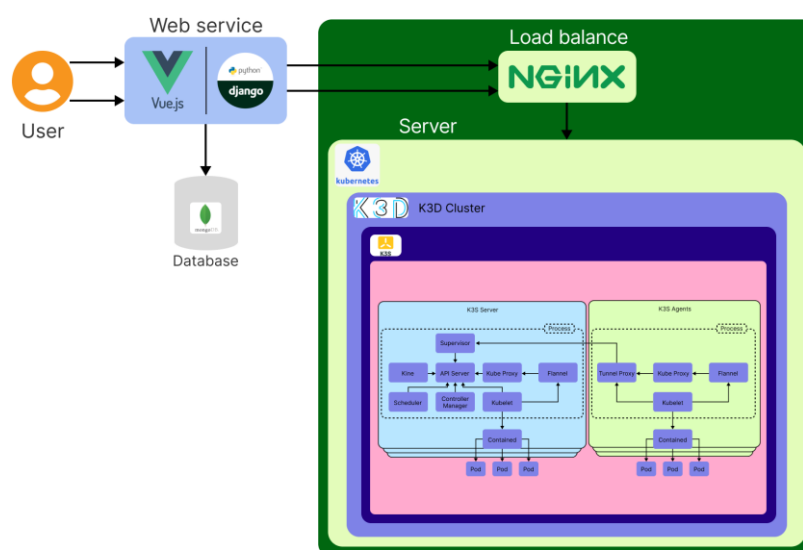
apiVersion: batch/v1
kind: Job
metadata:
  name: ascii+"iduser"+"dateที่สั่งงาน" // ห้ามตัวใหญ่ตัวเล็กเท่านั้น
spec:
  template:
    spec:
      containers:
        - name: ascii+"iduser"+"dateที่สั่งงาน" // ห้ามตัวใหญ่ตัวเล็กเท่านั้น
          image: suteesaraphan27/ascii
          volumeMounts:
            - name: myvolume
              mountPath: /www
          command: ["python","ASCII.py","1","/www/ea"] // ต้องเพิ่มเติมในส่วนของที่เก็บใน database
      restartPolicy: Never
      volumes:
        - name: myvolume
          persistentVolumeClaim:
            claimName: mypvc

```

รูปของไฟล์ YAML ในการสั่งให้ Kubernetes ทำงานประมวลผลภาพ รูปแบบใหม่

นอกจากนี้ในส่วนของ Web application เองก็ได้มีการปรับปรุง ไฟล์ CSS ใหม่เพื่อให้การแสดงผลต่างๆของ Web application นั้นอยู่ในรูปแบบของ Responsive และรองรับการแสดงผลได้ในหลากหลายอุปกรณ์

ในส่วนของการรายงานการดำเนินงานนั้น ได้ดำเนินการเขียนในบทที่ 3 เรื่องการออกแบบชิ้นงานลงไป โดยมีส่วนที่ต้องทำเพิ่มก็คือ Flow chart ของชิ้นงานและการปรับแก้ส่วนการออกแบบบางอย่างเพื่อให้สามารถรองรับการพัฒนาต่อไปในอนาคต เช่นการออกแบบ ER diagram ที่ต้องเพิ่มในส่วนของการเก็บสถิติของการใช้งานเพื่อมาใช้งานในส่วนของการทำประมวลสถิติและนำมาประกอบกับผลการ Monitoring Hardware ต่างๆ ให้สอดคล้องกัน รวมไปถึงหน้า User interface ต่างๆ ที่หลังจากที่ผู้เรียนได้เรียนรู้ในวิชา USER EXPERIENCE AND USER INTERFACE DESIGN ก็นำมาปรับแก้ไข Design เพื่อให้สวยงามมากขึ้น และยังมีการวาด Diagram Overview หรือ ภาพรวมของระบบออกมาเพื่อที่สามารถนำไปประกอบรายงานและ ใช้เพื่อเสริมในการนำเสนออีกด้วย โดย diagram ที่วาดออกมาจะเป็นรูปต่อไปนี้



รูปของ Diagram Overview หรือ ภาพรวมของระบบ

ในส่วนของ Server นั้น ได้ทำการทดสอบการสั่งงานของ Image Processing App ที่ได้ทำการอัปโหลดขึ้นไปบน Docker Hub โดยทดสอบการเตรียมการ OverHead Time ของระบบการสั่งงาน และการเสร็จสิ้นงาน โดยเปรียบเทียบกับการทำงานแบบปกติพบว่าการทำงานแบบปกติมีเวลาในการทำงานที่เร็วกว่าสำหรับงาน 1 งานเพราะเนื่องจาก OverHead Time ในการสั่งงานบน Kubernetes ใช้เวลานานกว่า แต่ในกรณีที่มีการสั่งงาน 5 งานในเวลาใกล้เคียงกันพบว่าใช้เวลาเร็วกว่าเนื่องจากใช้หน่วยประมวลผลแยกไปยังเครื่องอื่น ๆ ถึงแม้จะนับเวลา OverHead Time แล้วก็ตาม

#### 4. ปัญหาที่เกิดขึ้นและแนวทางการแก้ไข

ปัญหาที่พบในการพัฒนาของส่วนของ Web application คือการสร้างไฟล์ YAML ในการสั่งให้ Kubernetes ทำงานประมวลผลรูปนั้น ตัวไฟล์ YAML ค่อนข้าง sensitive เป็นอย่างมากกับ Format ของตัวไฟล์ ที่ต้องการให้การสร้างไฟล์นั้นทำได้ค่อนข้างยากและติดปัญหาผิด Format บ่อยครั้ง ซึ่งต้องปรับแก้ไขเพื่อให้สร้างไฟล์ YAML ออกมาได้ตรงตาม Format เดิม และทำงานได้ไม่ผิดพลาด

ปัญหาที่พบในการพัฒนาของส่วนของ Server คือการสั่งงานครั้งแรกจะใช้เวลาในการเตรียมการ Image ของ Docker ค่อนข้างนาน แต่หลังจากนั้นจะเรียกใช้จาก Cache จึงใช้เวลาเร็วกว่าการเรียกใช้งานครั้งแรกจึงได้มองหาแนวทางในการใช้ Kubernetes Private registries เพื่อใช้งาน Image ของ Docker บน Local เพื่อให้ไม่มีเวลาในการโหลด และ เก็บ Cache ไว้บนเครื่องตั้งแต่ต้น

ในส่วนของการเขียนรายงาน ด้วยการที่บทที่ 3 นั้นเกี่ยวข้องกับด้านการออกแบบ ทำให้เต็มไปด้วย Diagram หรือแผนภาพจำนวนมาก และ บางชนิดก็มีวิธีการหรือข้อกำหนดในการวาดที่ไม่เหมือนกัน จึงทำให้เกิดการเข้าใจผิดกันภายในสมาชิกกลุ่มเรื่องรูปแบบของ Diagram ที่ถูกต้อง จึงได้มีการตกลงเกี่ยวกับมาตรฐานการวาด Diagram ขึ้นมาเพื่อให้สมาชิกทำงานร่วมกันได้อย่างราบรื่น

#### 5. สิ่งที่จะดำเนินการต่อไป

สิ่งที่จะดำเนินการต่อไปคือ

- ศึกษาทดลอง Private Registries
- เก็บสถิติการใช้เวลาในการสั่งงาน Kubernetes และ ปรับปรุงแก้ไขให้ใช้เวลาได้เร็วขึ้น
- ปรับปรุงส่วนของหน้า image processing application ที่จะไว้สำหรับติดต่อกับ Server ในการประมวลผล image processing