# รายงานความก้าวหน้าวิชา CE Project

**ครั้งที่ 3**

**ระหว่างวันที่ 11 ก.ย. 65 ถึงวันที่ 18 ก.ย. 64**

1. ชื่อโครงงาน (อังกฤษ) Image Processing Application using Task Scheduling on Network System

## การดำเนินงานมีความก้าวหน้า 23% (ใช้ค่า **% Complete** จาก MS Project)

มีความก้าวหน้าเพิ่มขึ้นจากรายงานความก้าวหน้า ครั้งก่อน 21 %

□ เร็วกว่าแผน 0 วัน □ ช้ากว่าแผน 0 วัน

## รายละเอียดความก้าวหน้า

สำหรับการพัฒนาในส่วนของ Image Processing Application ผู้เรียนได้ออกแบบ และ จัดทำเขียนโปรแกรม Image Processing สำหรับใช้งานใน Application ออกมาเบี้องต้น 3 โปรแกรมดังนี้

* Black White Image
* ASCII Image
* Mosaic

ในส่วนของ Black White Image คือ การแปลงภาพสี RGB ที่มีอยู่ 3 Channel สีให้ออกมาเป็น 1 Channel สีจึงเกิดเป็นภาพสีขาว ดำ ขึ้นมาโดยใช้ตัวของ Python PIL ในการทำงาน มีตัวอย่างดังนี้

A picture containing text, person, indoor

Description automatically generated  
ภาพตัวอย่างของการทำ Black White Image

ในส่วนของ ASCII Image คือ การแปลงรูปภาพให้เป็น Pixel อยู่ในรูปของ Array และ นำตัวอักษร ASCII เช่น “ .,:irs?@9B&# ” มารวมแทนที่แต่ละ Pixel นั้นตามค่าตัวอักษรที่เราได้เรียงมาเป็นในรูปแบบของ Array List โดยสีของรูปภาพ ASCII นั้นจะขึ้นอยู่กับความเข้มของสีในภาพเดิมก่อนนำมาทำเป็น ASCII Image ยิ่งภาพมีสีเข้มตัวของ ASCII นั้นจะมีสีเข้มตามไล่ระดับจาก สีขาว สีเทา สีน้ำเงิน และ สีดำ ดังตัวอย่างนี้

Graphical user interface

Description automatically generated with medium confidence  
ภาพตัวอย่างของการทำ ASCII Image

ในส่วนของ Mosaic คือ การนำรูปภาพ 1 ภาพมาตัดแบ่งเป็น Tiles โดยกำหนดขนาดของ Tiles ออกมาได้ และ ยิ่งขนาดเล็กเท่าไหร่ความละเอียดของ Tiles ก็จะยิ่งสูงขึ้น โดยเมื่อแบ่ง Tiles เสร็จเราจึงนำรูปภาพอื่น ๆ จำนวนมากที่เราเตรียมไว้มาทำการ Feature Map กับตัว Tiles เพื่อทำให้ Tiles นั้น ๆ เกิดออกมาเป็นภาพอื่น ๆ ที่เราได้เตรียมไว้ และ ออกมาเหมือนกับภาพแบบ Mosaic ดังตัวอย่างนี้

Graphical user interface, website

Description automatically generated  
ภาพตัวอย่างของการทำ Mosaic

## ปัญหาที่เกิดขึ้นและแนวทางการแก้ไข

ปัญหาที่พบเนื่องจากตัวโปรแกรมทีได้เขียนมาต้องใช้เวลาในการจัดทำ Environment เพื่อนำไปทำงานใน Docker Image และ ใช้งานร่วมได้กับ Kubernetes จึงต้องใช้เวลาทดลอง และ ศึกษาเพิ่มเติมจากเดิมในการปรับแต่งตัวโปรแกรมให้ใช้งานได้อย่างมีประสิทธิภาพบนระบบ Containers

## สิ่งที่จะดำเนินการต่อไป

สิ่งที่จะดำเนินการต่อไปคือ

* เขียนโปรแกรม Docker Image สำหรับแอพพลิเคชั่น Image Processing
* ทดสอบแอพพลิเคชั่น Image Processing ที่ได้เขียนมาให้ทำงานอยู่ในรูปของ Container