

ระบบคาดการณ์ผลลัพธ์การผลิตบัณฑิตของหลักสูตร จากข้อมูลผลการเรียนของ  
นักศึกษา

Curriculum output prediction from student academic data

ณิษกานต์ สุขุมจิตพิทยัทย์

นรวิชญ์ อยู่บัว

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2565

ปริญญานิพนธ์ปี การศึกษา 2565

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบคาดการณ์ผลลัพธ์การผลิตบัณฑิตของหลักสูตร จากข้อมูลผลการเรียนของนักศึกษา

Curriculum output prediction from student academic data

ผู้จัดทำ

- |                   |                 |                       |
|-------------------|-----------------|-----------------------|
| 1. นางสาวณิชกานต์ | สุชุมจิตพิทยไธย | รหัสนักศึกษา 62010299 |
| 2. นายณรวิชญ์     | อยู่บัว         | รหัสนักศึกษา 62010465 |

อาจารย์ที่ปรึกษา

(ผศ. ดร. ธนัญชัย ตรีภาค)

# ระบบคาดการณ์ผลลัพธ์การผลิตบัณฑิตของหลักสูตร จากข้อมูลผล การเรียนของนักศึกษา

นางสาวณิกานต์	สุขุมจิตพิทยไทย์	62010299
นายณรวิชญ์	อยู่บัว	62010465
ผศ. ดร. ธนัญชัย	ตรีภาค	อาจารย์ที่ปรึกษา
ปีการศึกษา 2565		

## บทคัดย่อ

โครงการนี้จัดทำขึ้นเพื่อ พัฒนาระบบประมวลผลข้อมูลผลการเรียนของนักศึกษาในอดีต ข้อมูลของรายวิชาต่างๆ และข้อมูลจากแบบสำรวจการปฏิบัติงานของบัณฑิต เพื่อนำเสนอข้อมูลสถิติต่างๆ วิเคราะห์ข้อมูลผลการผลิตบัณฑิตเพื่อให้ได้ผลลัพธ์ว่าที่ผ่านมาหลักสูตรสามารถผลิตบัณฑิตกลุ่มใดได้บ้าง มีจำนวนมากน้อยเพียงใด สามารถพยากรณ์ว่าในอนาคตหลักสูตรสามารถผลิตบัณฑิตกลุ่มใดได้เป็นจำนวนเท่าใด เพื่อเป็นประโยชน์และอำนวยความสะดวกให้กรรมการหลักสูตรในการวางแผนการบริหารหลักสูตรในอนาคต และแสดงเป็นแผนภาพกราฟิกในการอำนวยความสะดวกให้หน่วยงานภายนอกได้รับทราบว่าหลักสูตรปัจจุบันของสถาบันสามารถผลิตบุคลากรที่มีความชำนาญด้านใดได้บ้าง

## **Curriculum output prediction from student academic data**

Ms. Nichakan Sukhumjitpitayotai 62010299

Mr. Narawich Youbua 62010465

Mr. Thanunchai Threepak Advisor

Academic Year 2022

### **Abstract**

The project was created to develop a data processing system that will measure the process of processing academic performance data from former students, data from various courses, and data from graduate employment surveys. We can present statistics and analyze graduate data to figure out which graduate groups have already been generated. It can forecast how many graduate groups that program will generate in the future based on historical data. This can help and facilitate course directors in future course administration planning, and it can be displayed as a graphical diagram in course management. The purpose of showing as a graphical representation is to let external agencies understand how the institution's existing curriculum can generate employees with expertise in any field.

## กิตติกรรมประกาศ

โครงการในภาคการศึกษานี้สำเร็จลุล่วงได้ด้วยดีจากความช่วยเหลือจากหลากหลายบุคคล โครงการในภาคการศึกษานี้จะผ่านไปไม่ได้หากปราศจากความช่วยเหลือจากบุคคลเหล่านี้ขอขอบคุณ อาจารย์ที่ปรึกษา ผศ. ดร. ธนัญชัย ตรีภาค ที่ให้ความช่วยเหลือในเรื่องต่าง ๆ ไม่ว่าจะเป็นการให้คำแนะนำถึงแนวทางการทำงานที่ดี การให้คำปรึกษาเพื่อหาทางออกเมื่อพบเจอกับปัญหา รวมถึงให้ความรู้เกี่ยวกับตัวงานทำให้งานต่าง ๆ เมื่อเจอปัญหาก็สามารถผ่านไปได้ด้วยดี

ขอขอบคุณคุณอาจารย์ในภาควิชาวิศวกรรมคอมพิวเตอร์ ที่ประสาทวิชาการความรู้มาตลอด 4 ปี ซึ่งความรู้หลาย ๆ แขนงก็ถูกใช้เป็นพื้นฐาน และเป็นส่วนหนึ่งของโครงการนี้

ขอขอบคุณเพื่อน ๆ วิศวกรรมคอมพิวเตอร์ที่ให้คำปรึกษา และแลกเปลี่ยนความรู้ซึ่งกันและกัน รวมถึงการรับฟังปัญหา

สุดท้ายนี้ขอขอบคุณบิดา มารดาและครอบครัว ที่เลี้ยงดูอบรมสั่งสอนและให้ความรู้คุณธรรม จริยธรรม และให้การสนับสนุนด้านการศึกษามาจนได้มีโอกาสมาทำโครงการนี้

ณิษกานต์ สุขุมจิตพิทยโยทัย

นรวิชญ์ อยู่บัว

# สารบัญ

## หน้า

บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VI
สารบัญรูป.....	VII

## บทที่ 1 บทนำ

1.1 ความเป็นมาของปัญหา.....	1
1.2 วัตถุประสงค์.....	1
1.3 ประโยชน์ของโครงการ.....	2
1.4 ข้อยกเว้นของโครงการ.....	2
1.5 แผนการดำเนินงาน.....	3

## บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 ทฤษฎีที่เกี่ยวข้อง.....	4
2.2 เครื่องมือที่เกี่ยวข้อง.....	12
2.3 งานวิจัยที่เกี่ยวข้อง.....	16

## บทที่ 3 การออกแบบ

3.1 โครงสร้างและการทำงานของระบบ.....	19
3.2 การทำงานของระบบ.....	19
3.3 Use Case Diagram.....	22
3.4 Sequence Diagram.....	30

## สารบัญ(ต่อ)

หน้า

3.5 การออกแบบฐานข้อมูล.....	36
-----------------------------	----

### บทที่ 4 ผลการดำเนินงาน

4.1 การสร้าง Django Framework เชื่อมต่อ MariaDB .....	39
4.2 การเตรียม Server ที่ใช้สำหรับ Deploy Application.....	42
4.3 การใช้งานฐานข้อมูลร่วมกับ Application.....	43
4.4 การพัฒนา Similarity Model.....	46
4.5 การนำชุดข้อมูลวิชาที่จัดกลุ่มไปใช้กับ Similarity Model.....	48
4.6 การพัฒนา Job Classification Model.....	52
4.7 การพัฒนาหน้าตาผู้ใช้งานของ Application.....	53
4.8 การนำ Model และการเรียกใช้ลงบน Django Application.....	57

### บทที่ 5 สรุป

5.1 บทสรุป.....	59
5.2 ปัญหาและอุปสรรคที่พบ.....	60
5.3 แนวทางในการพัฒนาต่อ.....	60
5.4 แนวทางในการพัฒนาต่อ.....	60

เอกสารอ้างอิง.....	61
--------------------	----

ภาคผนวก ก.....	63
----------------	----

## สารบัญตาราง

ตาราง	หน้า
ตาราง 3.1 Use Case คู่มือผลลัพธ์รายงานของบัณฑิตย้อนหลัง .....	23
ตาราง 3.2 Use Case คาดการณ์ผลลัพธ์รายงานที่ทำได้ของบัณฑิตในอนาคต.....	23
ตาราง 3.3 Use Case กรอกแบบฟอร์มเกรดสำหรับการวิเคราะห์ข้อมูล .....	24
ตาราง 3.4 Use Case แสดงค่าความถนัดของนักศึกษา .....	24
ตาราง 3.5 Use Case คาดการณ์เกรดแต่ละวิชาของนักศึกษา.....	25
ตาราง 3.6 Use Case แนะนำวิชาเลือกภาคให้แก่นักศึกษา.....	25
ตาราง 3.7 Use Case ให้ข้อมูลนักศึกษาโดยไม่ระบุตัวตนและข้อมูลหลักสูตร .....	26
ตาราง 3.8 Use Case อัปเดตข้อมูลนักศึกษาโดยไม่ระบุตัวตนและข้อมูลหลักสูตร .....	26
ตาราง 3.9 Use Case ลบข้อมูลนักศึกษาโดยไม่ระบุตัวตนและข้อมูลหลักสูตร .....	27
ตาราง 3.10 Use Case ล็อกอินเข้าใช้งาน .....	27
ตาราง 3.11 Use Case ทำนายพยากรณ์ผลลัพธ์และคาดการณ์ .....	28
ตาราง 3.12 Use Case จัดเตรียมข้อมูล .....	28
ตาราง 3.13 Use Case กรอกข้อมูลอาชีพ .....	29



# สารบัญรูป

รูป	หน้า
รูป 1.1 แผนการดำเนินงาน.....	3
รูป 2.1 ประเภทของ Machine Learning.....	6
รูป 2.2 รูปแบบของ Content-based Filtering .....	10
รูป 2.3 รูปแบบของ Memory-based.....	11
รูป 2.4 การทำงานของแอปพลิเคชันต่าง ๆ บน Docker Engine .....	13
รูป 2.5 ประเภทของ Machine Learning.....	17
รูป 3.1 โครงสร้างการทำงานของระบบ .....	19
รูป 3.2 Use Case Diagram .....	22
รูป 3.3 Sequence diagram สำหรับ UC-01 .....	30
รูป 3.4 Sequence diagram สำหรับ UC-02 .....	30
รูป 3.5 Sequence diagram สำหรับ UC-03 .....	31
รูป 3.6 Sequence diagram สำหรับ UC-04.....	31
รูป 3.7 Sequence diagram สำหรับ UC-05 .....	32
รูป 3.8 Sequence diagram สำหรับ UC-06 .....	32
รูป 3.9 Sequence diagram สำหรับ UC-07 .....	33
รูป 3.10 Sequence diagram สำหรับ UC-08.....	33
รูป 3.11 Sequence diagram สำหรับ UC-09 .....	34
รูป 3.12 Sequence diagram สำหรับ UC-10 .....	34
รูป 3.13 Sequence diagram สำหรับ UC-11 .....	35
รูป 3.14 Sequence diagram สำหรับ UC-12.....	35
รูป 3.15 Sequence diagram สำหรับ UC-13 .....	36
รูป 3.16 Data Schema .....	36
รูป 4.1 File Requirements.....	39
รูป 4.2 Docker file .....	40
รูป 4.3 File Docker - compose.yml.....	41

## สารบัญรูป(ต่อ)

รูป	หน้า
รูป 4.4 ภาพรวมของการตรวจสอบการทำงานของระบบ.....	41
รูป 4.5 Directory ที่เก็บ Django Application .....	42
รูป 4.6 ผลลัพธ์การสร้าง Docker Compose บน Server.....	43
รูป 4.7 ผลลัพธ์การค้นหาข้อมูลผ่าน API.....	44
รูป 4.8 ผลลัพธ์การแก้ไขข้อมูลผ่าน API.....	44
รูป 4.9 ผลลัพธ์การเพิ่มข้อมูลผ่าน API .....	45
รูป 4.10 ผลลัพธ์การลบข้อมูลผ่าน API.....	45
รูป 4.11 ชุดข้อมูลในการทดลอง .....	46
รูป 4.12 ชุดข้อมูลหลังจากผ่านการเตรียมข้อมูล .....	47
รูป 4.13 ตัวอย่างข้อมูล Data Subject .....	48
รูป 4.14 ข้อมูล Data Subject หลังจากคัดรหัสวิชาที่ซ้ำออก.....	49
รูป 4.15 ข้อมูล Data Subjectหลังจากจัดแต่งบทคัดย่อ.....	49
รูป 4.16 Coding การเทียบค่า Similarity.....	50
รูป 4.17 ชุดข้อมูลหลังจากผ่านการเตรียมข้อมูลผลลัพธ์จาก NLP .....	51
รูป 4.18 ผลลัพธ์การทดลองร่วมกับ NLP.....	51
รูป 4.19 ชุดข้อมูลแบบสอบถามการมีงานทำของบัณฑิต.....	52
รูป 4.20 ชุดข้อมูลที่ผ่านการ Transpose .....	53
รูป 4.21 Dockerfile React Project.....	54
รูป 4.22 docker-compose.yml React Project .....	54
รูป 4.23 ภาพรวมของการตรวจสอบการทำงานของ React Project.....	55
รูป 4.24 ภาพรวมของ React Project และ Django Application ที่เรียกดูผ่าน Web Browser ....	56
รูป 4.25 ทดสอบการติดต่อระหว่าง React Project และ Django Application.....	56
รูป 4.26 React Project ที่นำขึ้นไปยัง Server ที่เตรียมไว้ .....	57
รูป ก1 แผนภาพการทำงานโดยรวมของระบบ.....	64

# บทที่ 1

## บทนำ

### 1.1 ความเป็นมาของปัญหา

Data Analytics เป็นการวิเคราะห์ข้อมูลที่มีอยู่ตั้งแต่อดีตจนถึงปัจจุบัน ในกรณีที่ข้อมูลเพียงพอและเหมาะสมจะสามารถนำมาคาดการณ์แนวโน้ม ทำนายอนาคตที่เป็นประโยชน์ พยากรณ์สิ่งที่กำลังจะเกิดขึ้นหรือน่าจะเกิดขึ้นโดยใช้ข้อมูลในอดีตกับแบบจำลองทางสถิติ รวมถึงการให้คำแนะนำทางเลือกต่าง ๆ และผลของแต่ละทางเลือก

จากปัญหาที่ทางผู้จัดทำเล็งเห็นความสำคัญคือการนำข้อมูลผลการเรียนของนักศึกษาในอดีตมาใช้ประโยชน์ในการบริหารหลักสูตร และ นำมาวิเคราะห์ผลเพื่อช่วยในการวางแผนการเรียนของนักศึกษา ซึ่งการวางแผนในการเรียนของหลักสูตรจะสามารถช่วยอาจารย์และบุคลากรที่เกี่ยวข้องกับการศึกษาในด้านของการบริหารหลักสูตร เพื่อวางแผนการเพิ่มหรือลดจำนวนผู้เรียนในรายวิชาต่าง ๆ ซึ่งส่งผลต่อการผลิตบัณฑิตด้านต่าง ๆ ได้

ดังนั้นผู้จัดทำจึงได้เห็นถึงความสำคัญการประเมินสถานะขอหลักสูตร ของระบบแนะนำการวางแผนการคาดการณ์จากการใช้ความรู้ทางด้าน Data Analytics, Prediction และ Recommendation โดยใช้ข้อมูลผลการเรียนของนักศึกษาในอดีต เพื่อพัฒนาระบบช่วยเหลือและตอบโต้ภัยให้แก่นักศึกษาและบุคลากรทางการศึกษาหรือบุคคลที่เกี่ยวข้องได้

### 1.2 วัตถุประสงค์

- 1) เพื่อนำข้อมูลของผลการเรียนของนักศึกษาในอดีตและข้อมูลจากแบบสำรวจการมีงานทำของบัณฑิตมาใช้ ในการวางแผนการเรียนหรือประเมินอาชีพในอนาคตของนักศึกษาได้
- 2) ประมวลผลข้อมูลผลการเรียนของนักศึกษาในอดีต และข้อมูลจากแบบสำรวจการมีงานทำของบัณฑิต และทำแผนภาพกราฟิกเพื่อนำเสนอข้อมูล อำนวยความสะดวกให้กรรมการหลักสูตรในการวางแผนการ ทำงาน
- 3) เพื่อนำข้อมูลผลการเรียนของนักศึกษาในอดีต มาพัฒนาเป็นระบบแนะนำและวางแผนการเรียนตัวของ นักศึกษาได้

- 4) เพื่อนำข้อมูลการพยากรณ์อาชีพในอนาคตของนักศึกษาในสถาบันมาแสดงเป็นแผนภาพกราฟิกในการอำนวยความสะดวกให้หน่วยงานภายนอกได้รับทราบว่าหลักสูตรปัจจุบันของสถาบันสามารถผลิต บุคลากรที่มีความชำนาญด้านใดได้บ้าง

### 1.3 ประโยชน์ของโครงการ

- 1) ได้ระบบรวบรวมข้อมูลผลการเรียนของนักศึกษาและข้อมูลแบบสำรวจการทำงานของบัณฑิต แล้วนำมาวิเคราะห์และนำเสนอข้อมูลที่เป็นประโยชน์ในการบริหารหลักสูตรของกรรมการหลักสูตร
- 2) มีระบบที่สามารถแนะนำ วางแผน และประเมินอาชีพในอนาคตจากผลการเรียนของนักศึกษา

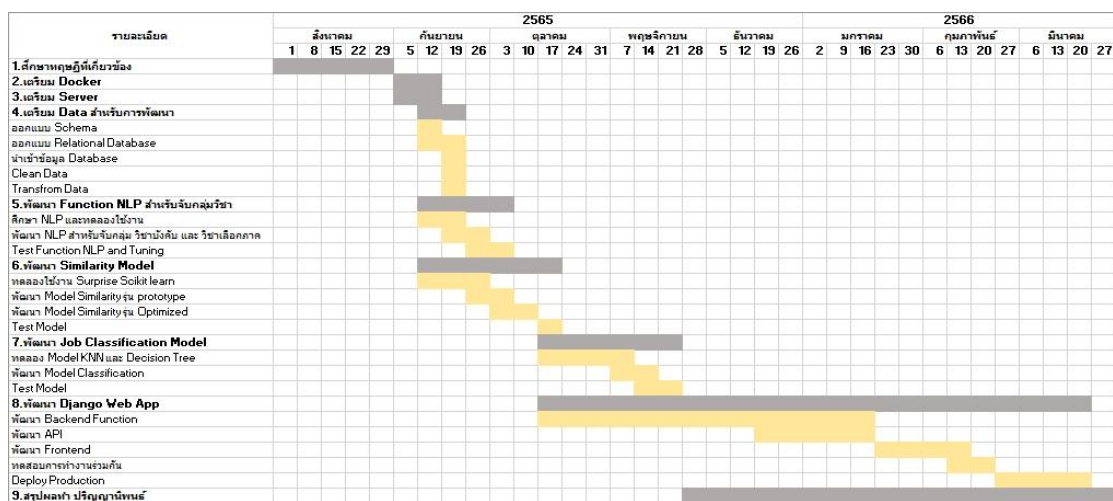
### 1.4 ข้อจำกัดของโครงการ

- 1) ข้อมูลผลการเรียนในอดีตย้อนหลังมีเพียง 2 ปี
- 2) ข้อมูลผลการเรียนในอดีตจะได้จากสำนักทะเบียนและประมวลผล โดยกรรมการหลักสูตรจะเป็นผู้ร้องขอข้อมูลดังกล่าวและนำเข้าระบบ
- 3) การทำนายต่าง ๆ จะใช้ข้อมูลเพียง 2 แหล่งคือข้อมูลผลการเรียนของนักศึกษาจากสำนักทะเบียนและประมวลผล และแบบสอบถามการมีงานทำของบัณฑิตเท่านั้น

## 1.5 แผนการดำเนินงาน

แผนการดำเนินงานในการพัฒนาโครงการตลอดระยะเวลา 2 ภาคการศึกษา ตั้งแต่เดือน

สิงหาคม พ.ศ. 2565 - มีนาคม พ.ศ. 2566 แสดงดังรูป 1.1



รูป 1.1 แผนการดำเนินงาน

## บทที่ 2

# ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

## 2.1 ทฤษฎีที่เกี่ยวข้อง

### 2.1.1 Classification and Predict

Classification and Prediction คือการจำแนกประเภทของข้อมูล โดยจะนำมาใช้ในการวิเคราะห์ข้อมูล ซึ่งกระบวนการดังกล่าวสามารถแบ่งได้เป็น 2 ส่วน

#### 2.1.1.1 Training Data

Training Data คือการนำข้อมูลที่ได้มาทำการเรียนรู้ให้กับคอมพิวเตอร์ เพื่อนำไปสร้างเป็นโมเดลแบบจำลองและวัดประสิทธิภาพของโมเดลแบบจำลองนั้น โดยจะทำการสร้างโมเดลซึ่งจะมีด้วยกันหลายวิธี เช่น Decision Tree, Naive Bayes, K Nearest Neighbors และ Neural Network เป็นต้น

#### 2.1.1.2 Predict

Predict คือการนำข้อมูลใหม่ที่รับมาเข้าโมเดลแบบจำลองที่เป็นผลลัพธ์จากการผ่านกระบวนการ Training Data ไปทำการคำนวณหรือพยากรณ์ประเภทของปัญหาในด้าน Classification

#### 1) Binary classification (การจำแนกแบบไบนารี)

เปรียบเทียบที่ดีที่สุดคือ ตัวแปรที่อยู่ในรูปแบบสองหมวดหมู่ เช่น ผลลัพธ์แบบ ใช่ หรือ ไม่ใช่ ตก หรือ ผ่าน หากเปรียบเทียบในรูปแบบของตัวเลขก็คือ 0 กับ 1 อัลกอริทึมที่ใช้คู่กับการจำแนกแบบไบนารี จะมีดังนี้ k-Nearest Neighbors Decision Trees หรือ Naive Bayes

#### 2) Multi-Class Classification (การจำแนกประเภทหลายคลาส)

ในการจำแนกรูปแบบนี้จะต่างกับการจำแนกแบบไบนารี โดยจะมีหมวดหมู่มากกว่าสอง ตัวอย่างของการจำแนกประเภทนี้ เช่น รูปภาพที่มีองค์ประกอบคล้ายคลึงกับรูปภาพที่อยู่ในฐานข้อมูลเพื่อค้นหาคำศัพท์ที่คาดว่าจะพิมพ์ใน predictive keyboard โดยผลลัพธ์ที่อาจเกิดขึ้นจะมีได้มากกว่า 2 หมวดหมู่ อัลกอริทึมที่ใช้คู่ไปกับการจำแนกประเภทนี้สามารถใช้อัลกอริทึมคล้ายกับการจำแนกแบบไบนารีได้

### 3) Multi-Label Classification (การจำแนกประเภทหลายเลเบล)

เปรียบให้เข้าใจง่ายโดยการยกตัวอย่างเช่น รูปภาพรูปหนึ่งสามารถมีรูปดอกไม้ ท้องฟ้าก้อนเมฆได้ แต่รูปภาพรูปนั้นจะจัดว่าเป็นหมวดหมู่รูปวาด รูปถ่าย หรือรูปเสีย Multi-Label Classification ก็คือการทำเลเบลให้กับชุดข้อมูล หรือการติดฉลากให้รูปนั้น ๆ ว่ามีดอกไม้หรือเปล่ามีก้อนเมฆหรือไม่ ส่วน Multi-Class Classification จะจำแนกว่ารูปนั้นเป็นรูปที่เกิดจากการวาดหรือรูปที่เกิดจากการถ่ายหรือรูปเสีย

### 4) Imbalanced Classification (การจำแนกแบบข้อมูลไม่เท่าเทียม)

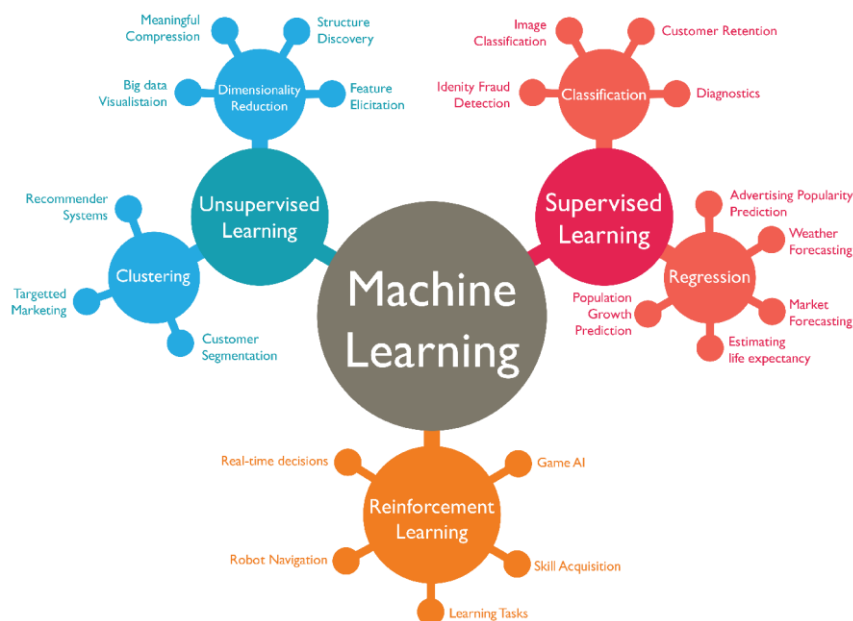
คือปัญหาที่เกิดจากข้อมูลที่มีไม่เท่าเทียมกัน (Imbalanced dataset) ตัวอย่างเช่นข้อมูลของการทุจริตโดยข้อมูลส่วนใหญ่ย่อมเป็นข้อมูลที่จัดว่า “ไม่ทุจริต” และจะมีเปอร์เซ็นต์น้อยที่จัดว่าเป็น “ทุจริต” เป็นต้น โดยจะเปรียบโดยง่ายคือกรณีที่ชุดข้อมูลมีการแยกประเภทกันแต่จำนวนของประเภทนั้นมีอัตราส่วนของข้อมูลที่ห่างกันค่อนข้างมาก

## 2.1.2 Machine Learning

Machine Learning คือ การทำให้ระบบของคอมพิวเตอร์นั้นสามารถเรียนรู้ได้ด้วยตนเอง โดยจะใช้ข้อมูล ด้วยวิธีการใส่ข้อมูลและผลลัพธ์เข้าไป เพื่อให้โปรแกรมนำผลลัพธ์นั้นไปประมวลผลและพยากรณ์ Output และ Input ของข้อมูลใหม่ โดยแบ่ง Machine Learning ออกได้เป็น 3 ประเภท คือ

- 1) Supervised Learning คือการเรียนรู้ที่เครื่องจักร หรือคอมพิวเตอร์นั้น จำเป็นต้องอาศัยข้อมูลใน การฝึกฝน เปรียบเสมือนกับการเรียนการสอนของเด็ก ซึ่งจำเป็นอย่างยิ่งที่จะต้องอาศัยชุดของข้อมูล ซึ่งประกอบไปด้วยชุดของข้อมูล และชุดของผลลัพธ์ของข้อมูลที่ต้องการจะนำมาให้ เครื่องจักรหรือคอมพิวเตอร์ในการเรียนรู้
- 2) Unsupervised Learning เป็นการเรียนรู้ที่ให้เครื่องจักรหรือคอมพิวเตอร์นั้น สามารถเรียนรู้ได้ด้วยตนเอง โดยไม่จำเป็นต้องมีค่าเป้าหมายของแต่ละชุดข้อมูล ซึ่งวิธีการนี้คือการที่มนุษย์นั้นจะเป็นผู้ใส่ชุดข้อมูล และกำหนดสิ่งที่ต้องการจากชุดข้อมูลเหล่านั้น โดยให้เครื่องจักรหรือคอมพิวเตอร์วิเคราะห์จากการจำแนกและทำการสร้างแบบแผนจากข้อมูลที่ได้รับมา

- 3) Reinforcement Learning เป็นการเรียนรู้สิ่งต่าง ๆ ผ่านจากการลองผิดลองถูก ภายใต้นโยบายที่จะเลือกกระทำสิ่งใดที่ทำให้ได้ผลลัพธ์มากที่สุด โดยจะทำการเรียนรู้จากการลองผิดลองถูกในสถานการณ์ในอดีตหรือระบบจำลอง และพยายามที่จะพัฒนาระบบการตัดสินใจของตัวเองให้ดีขึ้นอย่างต่อเนื่อง โดยที่อาจจะสามารถพัฒนาด้วยการพยายามสร้างแบบจำลองสถานการณ์ต่าง ๆ ขึ้นมา



รูป 2.1 ประเภทของ Machine Learning

(ที่มา : [medium.com](https://medium.com), 2018)

### 2.1.3 Extract-Transform-Load (ETL)

Extract-Transform-Load คือ กระบวนการ กระบวนการหนึ่งซึ่งอยู่ในระบบของ Data Warehouse ซึ่งเป็นระบบที่ออกแบบมาเพื่อที่จะสามารถดึงข้อมูลออกมาจากหลายแหล่ง โดยจะนำกระบวนการตรวจสอบคุณภาพของชุดข้อมูลมาประยุกต์ร่วมใช้ ซึ่งมีการเชื่อมโยงและปรับชุดของข้อมูลให้เป็นไปในรูปแบบเดียวกันทั้งหมดเพื่อให้ ชุดของข้อมูลจากหลากหลายแหล่งสามารถใช้งานร่วมกันได้ และทำการส่งมอบ

#### 2.1.3.1 Extract

เป็นกระบวนการเริ่มต้นของระบบที่ดึงข้อมูลจากแหล่งของข้อมูล จะประกอบด้วยข้อมูลจากหลากหลายแหล่งที่มา ข้อมูลที่อยู่ต่างที่กันนั้นอาจจะอยู่ในรูปแบบที่



แตกต่างกัน ยกตัวอย่างเช่น อาจอยู่ในรูปแบบของฐานข้อมูลคนละชนิด หรือ ไม่ใช่ฐานข้อมูล แท้จริงซึ่งอาจจะเป็นระบบไฟล์ข้อมูลธรรมดา

### 2.1.3.2 Transforming

Transforming ขั้นตอนการแปลงรูปแบบของข้อมูลนี้จะมีการใช้กฎหรือ ฟังก์ชัน (Function) มากมายเพื่อที่จะแปลงข้อมูลให้อยู่ในรูปแบบตามที่ต้องการก่อนที่จะนำ ข้อมูลเหล่านั้นเข้าไปยังปลายทาง ข้อมูลจากต้นทางนั้นบางแหล่งข้อมูลมีความจำเป็นน้อยมาก หรือแทบจะไม่ต้องการ การแปลงข้อมูลเลย แต่ในบางแหล่งอาจจะต้องการกระบวนการที่ ซับซ้อนในการแปลงข้อมูล ซึ่งจะกินทรัพยากรของระบบที่ใช้และเวลาในการประมวลผลของ ระบบ ซึ่งความซับซ้อนของข้อมูลจะขึ้นอยู่กับความต้องการของเชิงธุรกิจ หรือ เป้าหมายของ การนำข้อมูลไปใช้งาน โดยจะมีกระบวนการตัวอย่างต่อไปนี้

- 1) Selection คือ การเลือก Column ที่ต้องการที่จะนำไปใช้งานหรือ เก็บลงฐานข้อมูล ยกตัวอย่าง เช่น ถ้าต้นทางของข้อมูลมีอยู่ด้วยกัน 3 Column หรือ 3 attributes เช่น enroll\_num, age และ salary จะมีการแปลงข้อมูล เกิดขึ้นและ เลือกที่จะไม่มีการแปลงข้อมูลหาก พบว่า record นั้นมีค่าของข้อมูล column salary เป็นค่าว่าง
- 2) Translation คือ การแปลข้อมูล ตัวอย่างเช่น หากข้อมูลต้นทางนั้นมีการเก็บข้อมูลของเพศโดยให้ 1 เป็นเพศชาย และ 2 เป็นเพศหญิง จะต้องมีการแปลจากชุดตัวเลขที่กำหนดก่อนหน้านี้ให้ 1 = Male และ 2 = Female กระบวนการนี้เรียกว่า data cleaning หรือ กระบวนการทำความสะอาดข้อมูล
- 3) Encoding free form ยกตัวอย่างเช่นการ mapping จาก “Male” ไปเป็น “1” และ “Mr” ไปเป็น “M”
- 4) Filtering คือ กระบวนการกรองเฉพาะข้อมูลที่กำหนด
- 5) Sorting คือ กระบวนการเรียงข้อมูลที่ต้องการ
- 6) Joining คือ กระบวนการเชื่อมโยงข้อมูลระหว่างตารางข้อมูล
- 7) Aggregation คือ กระบวนการรวบรวม และ สรุปชุดข้อมูล ยกตัวอย่างเช่น การรวมยอด (summarize) ข้อมูลจากหลาย ๆ ระเบียบจนได้มาเป็นยอดขายรวม เป็นต้น

- 8) Transposing or pivoting คือการสลับทิศทางตำแหน่งของการแสดงข้อมูล เช่นการย้ายระเบียบไปเป็น Column หรือ ย้าย Column มาเป็นระเบียบ เพื่อให้ง่ายต่อการนำข้อมูลไปใช้

### 2.1.3.3 Loading

Loading กระบวนการโหลดข้อมูลเข้า โดยทั่วไปจะนำข้อมูลเข้าไปในระบบ Data Warehouse ทั้งนี้ขึ้นอยู่กับความต้องการขององค์กร หรือ ธุรกิจว่าจะให้ข้อมูลไหลไปในทิศทางใด บางองค์กร หรือ บางงานจะมีการสะสมของข้อมูล ความถี่ของการนำข้อมูลเข้าสู่ระบบ อาจจะมีการล้างข้อมูลแล้วทับข้อมูลใหม่ โดยทั่วไปแล้วข้อมูลของ Data Warehouse จะมีการใช้กันปีต่อปีเมื่อขึ้นปีใหม่แล้วจะมีการล้างข้อมูลของปีเก่า และ เก็บไว้ในระบบข้อมูลสำรอง เนื่องจากว่ากระบวนการนำข้อมูลเข้าจะต้องปฏิบัติตามพันธกิจกับฐานข้อมูล (Database) ดังนั้นจะต้องมีประเด็นเรื่องของ Database Constraints, Referential Integrity, Database Trigger เข้ามาเกี่ยวข้องด้วยในกระบวนการนำข้อมูลเข้า ซึ่งสิ่งเหล่านี้รวม ๆ แล้วเรียกว่า กระบวนการควบคุมคุณภาพของข้อมูล (Data Quality performance of E-T-L process)

### 2.1.4 Natural Language Processing (NLP)

Natural Language Processing (NLP) เป็นเครื่องมือที่ให้คอมพิวเตอร์เข้าใจภาษาของมนุษย์ที่มีความซับซ้อน เป็นศาสตร์หนึ่งที่สำคัญทางด้าน Machine Learning โดยเป็นสาขาวิชาหนึ่งที่ประกอบด้วยองค์ความรู้จากหลากหลายแขนง อาทิ ภาษาศาสตร์ (Linguistics) วิทยาการคอมพิวเตอร์ (Computer Science) ปัญญาประดิษฐ์ (Artificial Intelligence: AI) รวมไปถึงสถิติ (Statistics) โดย NLP มีมาตั้งแต่ช่วงกลางศตวรรษที่ 19 และมีการพัฒนามาเรื่อย ๆ จนถึงปัจจุบัน โดยแบ่งออกเป็น 3 ยุค ดังนี้

- 1) ยุค Rule-based Method (ช่วง ค.ศ.1950-1990)

ในยุคแรกของ NLP มีการใช้งานตามกฎ (Rule-based Method) โดยนักภาษาศาสตร์ผู้มีความเชี่ยวชาญโครงสร้างของภาษาที่สนใจ จะเป็นผู้เขียนกฎต่าง ๆ ขึ้นมาเพื่อให้คอมพิวเตอร์สามารถคำนวณข้อความของโจทย์ต่าง ๆ ได้

- 2) ยุค Machine Learning (ช่วง ค.ศ.1990-2010)

ในยุคนี้ พบว่ามีการเขียนกฎด้วยมือไม่สามารถตอบโจทย์ที่มีความซับซ้อนได้ จึงมีสิ่งที่ได้มาทดแทนในยุคนี้คือ ความสามารถของเครื่องคอมพิวเตอร์ รวมถึงความรู้ทางด้านสถิติ และ Machine Learning ซึ่งได้ถูกนำมาพัฒนาเพื่อ

ใช้ในการทำงานด้าน NLP โดยมีการนำเข้าข้อมูลเพื่อให้คอมพิวเตอร์สามารถเรียนรู้ด้วยตนเองแทนการใช้ผู้เชี่ยวชาญทางด้านภาษา

### 3) ยุค Deep Learning (ช่วง ค.ศ.2010-ปัจจุบัน)

ในยุคปัจจุบัน ด้วยพลังการคำนวณของคอมพิวเตอร์ที่มีการพัฒนาสูงขึ้นอย่างต่อเนื่อง ทำให้เทคโนโลยีที่มีความซับซ้อนสูงอย่าง การเรียนรู้เชิงลึก (Deep Learning) ถูกนำมาใช้งานแทนที่ Machine Learning ซึ่งใช้ความรู้ทางด้านสถิติแบบดั้งเดิมอย่างแพร่หลายมากขึ้น รวมถึงในงานด้าน NLP ด้วยเช่นกัน อาทิ การสร้างแบบจำลองทางภาษา (Language Model) และการวิเคราะห์โครงสร้างของข้อความ (Parsing)

ตัวอย่างการประยุกต์ใช้ NLP ในด้านต่าง ๆ

- 1) ด้านการทำงานวิจัย การวิจัยมีแหล่งของข้อมูลทางภาษานาใหญ่ ซึ่งทำให้ NLP สามารถเข้ามามีบทบาทได้อย่างหลากหลาย ตัวอย่างเช่น การใช้ Topic Model ในการจัดหมวดหมู่บทความ
- 2) ด้านพาณิชย์อิเล็กทรอนิกส์ การซื้อของผ่านช่องทางออนไลน์ เข้ามามีบทบาทสำคัญเป็นอย่างมากในระบบเศรษฐกิจ ซึ่งทำให้เกิดปริมาณธุรกรรมขนาดใหญ่ ไม่ว่าจะเป็น คำอธิบายสินค้าและบริการ การแสดงความคิดเห็นของผู้บริโภค รวมถึงการสนทนากันระหว่างผู้ซื้อและผู้ขายผ่านทางช่องแชต
- 3) ด้านการแพทย์ ข้อมูลทางการแพทย์มีการบันทึกข้อมูลด้วยข้อความ ตัวอย่างเช่น บทสนทนาระหว่างแพทย์และผู้ป่วย การวินิจฉัยโรคโดยแพทย์ และประวัติการรักษาของผู้ป่วย
- 4) ด้านกฎหมาย สำหรับงานด้าน มีข้อมูลทางด้านภาษาที่แตกต่างกันและหลากหลาย เช่นเดียวกัน เช่น ประมวลกฎหมายต่าง ๆ คำร้องต่อศาล คำให้การของกลุ่มความ และคำพิพากษาของศาล ซึ่งสามารถประยุกต์ใช้เครื่องมือ NLP ได้ในหลายมิติไม่ว่าจะเป็นการใช้ PoS Tagging และ NER เพื่อช่วยในการตีความประมวลกฎหมาย

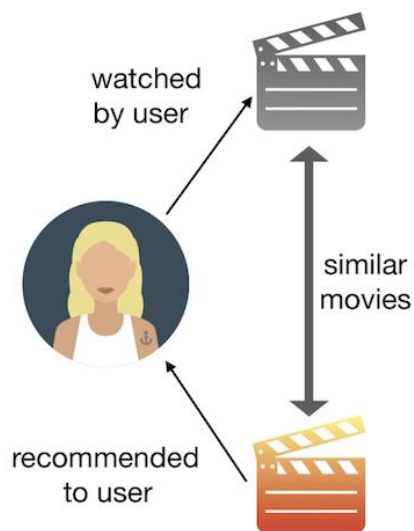
### 2.1.5 Recommendation System

Recommendation System เป็นระบบที่จะทำการแนะนำสิ่ง (item) ที่ “เหมาะสม” ให้แก่ผู้ใช้ โดย item เป็นได้ตั้งแต่ ข้าว เนื้อหา เพลง course เรียน ไปจนถึงสินค้าที่ขายในร้าน

online โดยสามารถแนะนำสิ่งที่ผู้ใช้สนใจได้ผ่านโมเดลที่ส่วนใหญ่มักจะถูกใช้กันมีอยู่ด้วยกันสามประเภท ได้แก่

#### 2.1.5.1 Content-based Filtering

เป็นรูปแบบของโมเดลที่จะแนะนำลักษณะของตัวบริการหรือสินค้าเป็นตัวตั้งและทำการแนะนำสิ่งที่มีลักษณะที่คล้ายกัน



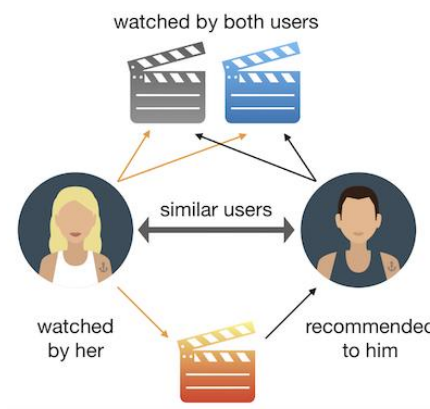
รูป 2.2 รูปแบบของ Content-based Filtering

(ที่มา : [towardsdatascience.com](https://towardsdatascience.com), 2018)

#### 2.1.5.2 Collaborative Filtering

เป็นรูปแบบโมเดลที่เรียนรู้จากพฤติกรรมของผู้ใช้กับผู้ใช้คนอื่น ๆ ที่คล้ายคลึงกัน

- 1) Memory-based เป็นการดูข้อมูลแล้วหาความสัมพันธ์ ระหว่างผู้ใช้ หรือ สินค้าจากข้อมูลโดยตรง



รูป 2.3 รูปแบบของ Memory-based

(ที่มา : [towardsdatascience.com](https://towardsdatascience.com), 2018)

- 2) Model-based ใช้เทคนิคของ machine learning เพื่อหา user embedding และ item embedding มาทำการทำนาย rating ที่ผู้ใช้จะให้กับสินค้า หรือ relevance score
- 3) Hybrid ใช้หลาย ๆ วิธีการมารวมกัน Hybrid system เป็นการนำรวมทั้งสองอัลกอริทึมของ Model-based และ Memory-based เอาไว้เพื่อทำให้ระบบการแนะนำสมบูรณ์ขึ้น ซึ่งระบบนี้ถูกนำไปใช้ในปัจจุบันมากที่สุดแทบจะทุกแพลตฟอร์มใหญ่ที่มีการแนะนำสินค้าและบริการ

### 2.1.5.3 Hybrid system

เป็นการนำมารวมทั้งสองระหว่าง Content-based Filtering และ Collaborative Filtering เพื่อทำให้ระบบการแนะนำสมบูรณ์ขึ้น

## 2.2 เครื่องมือที่เกี่ยวข้อง

### 2.2.1 Docker

Docker เป็นเครื่องมือแบบ open-source ที่ช่วยจำลองสภาพแวดล้อม ในการรัน service หรือ server โดยการสร้าง container เพื่อจัดการกับ library ต่างๆ และยังช่วยจัดการในเรื่องของ version control เพื่อให้ง่ายต่อการจัดการกับปัญหาต่างๆ ที่เกิดขึ้นองค์ประกอบต่างๆ ของ Docker

#### 1) Docker image

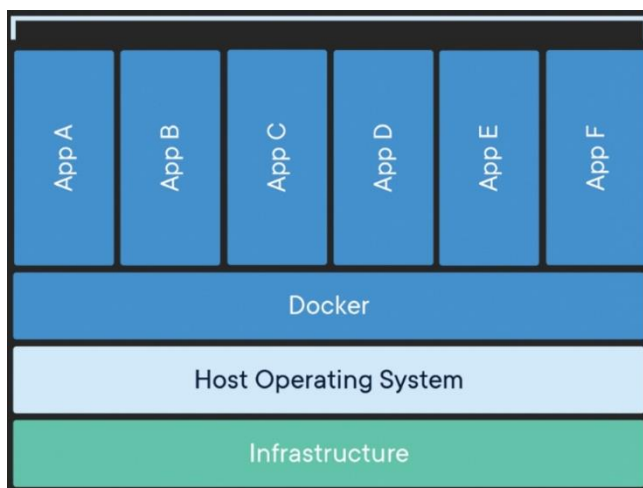
คือต้นแบบของ Container โดยข้างในจะเป็นระบบปฏิบัติการ Linux ที่มีการติดตั้ง Application และ มีการ Configuration เอาไว้ ซึ่งเกิดจากการ build ไฟล์ Docker file ขึ้นมาเป็น image

#### 2) Docker container

Docker container จะถูกสร้างมาจาก Docker Image ที่เป็นต้นแบบหรือแม่พิมพ์ เกิดเป็น container และจะได้ Service หรือ Application ที่สามารถเรียกใช้งานได้ทันที

#### 3) Docker registry

การสร้าง Docker Image แล้วนำไปเก็บรวบรวมไว้บน server ลักษณะเดียวกับการเก็บ Source Code ไว้บน (Github) โดย Docker registry ณ ปัจจุบันก็มีให้เลือกใช้งานได้อย่างหลากหลายโดยมี Docker Hub เป็น Docker registry หลักในการเรียกใช้(pull) Docker Image และนอกจากนี้ยังมีผู้ให้บริการ docker registry อื่นๆด้วย เช่น Gitlab, Quay.io, Google Cloud เป็นต้น



รูป 2.4 การทำงานของแอปพลิเคชันต่าง ๆ บน Docker Engine

(ที่มา : [docker.com](https://docker.com))

### 2.2.2 Django

Django Framework เป็นชุดของเครื่องมือ Framework สำหรับ การนำไปพัฒนา เว็บไซต์ด้วยภาษาของ Python โดยทุกวันนี้ Framework สำหรับการเขียนเว็บไซต์ด้วยภาษา Python มีค่อนข้างที่จะเยอะ ซึ่ง Django Framework ก็เป็นหนึ่งใน Framework สำหรับการพัฒนา เว็บไซต์ และทำเว็บไซต์ด้วยภาษา Python ด้วยเช่นกัน

คุณสมบัติของ Django Framework

- 1) Object-relational mapper คือ การกำหนด Data Model ในภาษา Python เพื่อใช้ในการทำงานด้านข้อมูล และช่วยสนับสนุน dynamic database-access API
- 2) Automatic admin interface คือ ส่วนในการสร้าง Interface อัตโนมัติสำหรับการ add, edit, delete และ search ด้วย Django Framework
- 3) Elegant URL design คือ การทำให้ URL มีความสั้น กระชับ สวยงาม และสื่อความหมายของหน้านั้น ๆ ได้อย่างชัดเจน
- 4) Template system คือ Django นั้นมีการออกแบบ Template Language เพื่อการเขียนแยกส่วนระหว่าง Design และ Business Logic
- 5) Cache system คือ ส่วนของการบันทึก หรือจัดการข้อมูลที่มีการดาวน์โหลดไปแล้ว เพื่อเพิ่มประสิทธิภาพการทำงานของเว็บไซต์ด้านความเร็ว และด้านอื่น ๆ
- 6) Internationalization คือ Django สนับสนุน Application ที่มีความหลากหลายด้านภาษาในการแสดงผล

### 2.2.3 Scikit-learn

Scikit-learn เป็นโมดูลหนึ่งของภาษา Python เป็นแพ็คเกจที่รวบรวม Library ด้านการเรียนรู้ของเครื่อง (Machine Learning) เอาไว้ และถูกออกแบบมาให้ทำงานร่วมกับ Library ของภาษา Python อย่าง NumPy และ SciPy ได้ดี

Scikit-learn ยังเป็น Open Source ที่เปิดให้สามารถเข้าไปพัฒนาต่อยอดได้และเป็นแหล่งรวม Library และอัลกอริทึมที่เน้นไปในด้านของ การเรียนรู้ของเครื่อง (Machine Learning) ซึ่งมีส่วนในการทำ แบบจำลองข้อมูล (Data Modeling) อีกหนึ่งปัจจัยที่ทำให้มีผู้ใช้เยอะ เพราะเป็น Interface ระดับสูง ทำให้มือใหม่สามารถเข้าใจภาพรวมและ ขั้นตอนการทำงานของ การเรียนรู้ของเครื่อง (Machine Learning) ได้เครื่องมือที่ผู้ใช้งานสามารถนำไปใช้ได้

### 2.2.4 MariaDB

MariaDB คือ เป็น Open Source สำหรับจัดการกับฐานข้อมูล MariaDB เป็นหนึ่งในฐานข้อมูลที่ได้รับความนิยมมากที่สุดในโลก MariaDB ถูกพัฒนาขึ้นโดยนักพัฒนาเดิมของ MySQL เนื่องจากความกังวลที่เกิดขึ้นเมื่อ MySQL ถูกซื้อโดย Oracle Corporation ในปี 2009 ตอนนี้นักพัฒนาและผู้ดูแลของ MariaDB ได้รวมรายละเอียดกับฐานรหัส MySQL เพื่อให้แน่ใจว่า MariaDB มีการแก้ไขข้อบกพร่องที่เกี่ยวข้องเพิ่มลงใน MySQL

MariaDB ได้รับการพัฒนาเป็นซอฟต์แวร์โอเพนซอร์ส และเป็นฐานข้อมูลเชิงสัมพันธ์แบบ SQL สำหรับการเข้าถึงข้อมูล เวอร์ชันล่าสุดของ MariaDB มีคุณลักษณะ GIS และ JSON ด้วย

MariaDB เปลี่ยนข้อมูลเป็นฐานข้อมูลที่มีโครงสร้างในหลากหลายแอปพลิเคชัน ตั้งแต่ธนาคารไปจนถึงเว็บไซต์ต่างๆ เป็นการปรับปรุงและแทนที่ด้วยการแทนที่ของ MySQL เนื่องจากมีความรวดเร็วและสามารถปรับขนาดได้และมีระบบเวดล๊อคที่อุดมไปด้วยปลั๊กอินส์ เอนจินและเครื่องมืออื่น ๆ ทำให้สามารถใช้งานได้หลากหลาย

### 2.2.5 React

React เป็น JavaScript library ที่ใช้สำหรับสร้าง user interface ที่ให้เราสามารถเขียนโค้ดในการสร้าง UI ที่มีความซับซ้อนแบ่งเป็นส่วนเล็กๆออกจากกันได้ ซึ่งแต่ละส่วนสามารถแยกการทำงานออกจากกันได้อย่างอิสระ และทำให้สามารถนำชิ้นส่วน UI เหล่านั้นไปใช้ซ้ำได้

### 2.2.6 Node.JS

Node.js คือสภาพแวดล้อมการทำงานของภาษา JavaScript นอกจากเว็บเบราว์เซอร์ที่ทำงานด้วย V8 engine นั้นหมายความว่าเราสามารถนำ Node.js ในการพัฒนาแอปพลิเคชัน



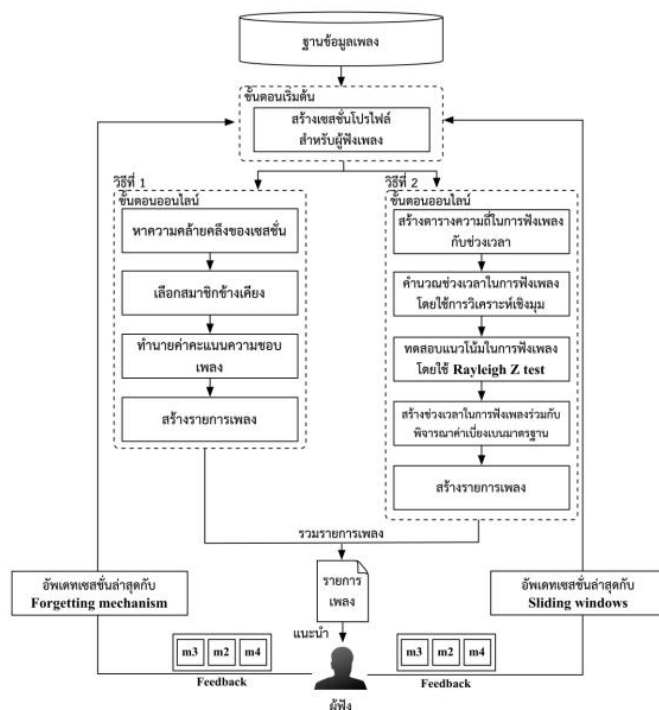
แบบ Command line แอปพลิเคชัน Desktop หรือแม้แต่เว็บเซิร์ฟเวอร์ได้ โดยที่ Node.js จะมี APIs ที่จะสามารถใช้สำหรับทำงานกับระบบปฏิบัติการ เช่น การรับค่าและการแสดงผล การอ่านเขียนไฟล์ และการทำงานกับเน็ตเวิร์ก และยังเป็นโปรแกรมที่สามารถใช้ได้ทั้งบน Windows, Linux และ Mac OS X โดยสามารถเขียนโปรแกรมในภาษา JavaScript และนำไปรันได้ทุกระบบปฏิบัติการที่สนับสนุนโดย Node.js

## 2.3 งานวิจัยที่เกี่ยวข้อง

### 2.3.1 การสร้างรายการเพลงโดยใช้การกรองร่วมแบบเซสชันที่เพิ่มขึ้นด้วยกลไกการลื้มและการวิเคราะห์สถิติเชิงมุม

สุเมธ ดาราพิสุท นำเสนองานวิจัยเรื่อง การสร้างรายการเพลงโดยใช้การกรองร่วมแบบเซสชันที่เพิ่มขึ้นด้วยกลไกการลื้มและการวิเคราะห์สถิติเชิงมุม โดยใช้ 2 วิธีร่วมกัน 1 การสร้างรายการเพลงจะพิจารณาการฟังเพลงในเซสชันปัจจุบันที่คล้ายกับเซสชันในอดีตของผู้ฟัง 2 สร้างรายการเพลงแนะนำโดยพิจารณาช่วงเวลา เฉพาะในการฟังเพลงซึ่งแตกต่างจากช่วงเวลาอื่นอย่างมีนัยสำคัญทางสถิติในรอบวันของผู้ฟังโดยใช้ การวิเคราะห์สถิติเชิงมุม และวัดประสิทธิภาพโดย ประสิทธิภาพ Hit Ratio และ Precision จากการทดลองพบว่าการใช้ 2 วิธีแยกกันนั้นได้ผลลัพธ์ที่น้อยก่นำมาใช้ร่วมกัน 0.18-0.22 % โดยวัตถุประสงค์ในการทำเพื่อวิเคราะห์ปัญหาที่เกิดขึ้นในการสร้างรายการเพลงแนะนำแบบออฟไลน์ พัฒนารูปแบบขั้นตอนวิธีการสร้างรายการเพลงแนะนำให้มีประสิทธิภาพเพิ่มขึ้นทั้งทางด้านความเร็วและความถูกต้องในการสร้างรายการเพลง

โดยในโครงงานของผู้จัดทำนั้นได้นำส่วนของไดอะแกรม Diagram ในงานวิจัยนี้มาใช้งาน โดยใช้วิธีที่ 1 ซึ่งของผู้จัดทำจะเป็น 1 หาความคล้ายคลึงของหมวดหมู่วิชา 2 เลือกสมาชิกข้างเคียง 3 ทำนายค่าผลลัพธ์การเรียนหรือเกรด 4 นำไปสร้างรายการสำหรับขั้นตอนต่อไป



รูป 2.5 ประเภทของ Machine Learning

(ที่มา : คาราพิสุทธิ์, 2016)

### 2.3.2 ระบบแนะนำสินค้าอาหารโดยใช้ระบบแนะนำแบบผสมผสาน

นิภาภรณ์ พันธุ์นาม นำเสนองานวิจัย ระบบแนะนำสินค้าอาหารโดยใช้ระบบแนะนำแบบผสมผสาน ใช้เทคนิค Content based filtering แบบหลักการ Cosine และสร้างแบบจำลองโดยใช้ lib Surprise ซึ่งมีอัลกอริทึม SVD, NMF, Baseline และ KNN และวัดประสิทธิภาพโดย RMSE, MAE จากการทดลองพบว่า 1 เทคนิคการกรองแบบอิงเนื้อหาหน้าวิธีการ TF-IDF เข้ามาช่วยในการทำ Vectorization ส่วนใหญ่ค่าความเหมือนออกมาค่อนข้างที่จะต่ำเนื่องจากข้อมูลที่น้อยเกินไป 2 เทคนิคการกรองข้อมูลแบบพึ่งพาผู้ใช้ร่วมผ่าน library Surprise ของ Scikitlearn ซึ่งโมเดลที่มีผลคะแนนโดยรวมดีที่สุดคืออัลกอริทึมของ SVD ซึ่งได้ค่า RMSE 1.2528 และ MAE 0.9376 และ 3 ระบบแนะนำแบบผสมผสาน โดยผลลัพธ์นั้นจะไม่ใช่ชัดเจนเนื่องจากวิธีนี้ได้มีการทำนายค่า Rating ซึ่งวิธีการของระบบแนะนำแบบผสมผสานนั้นได้มีนำเทคนิคการกรองแบบอิงเนื้อหา ที่ไม่ได้มีการทำนายค่าอะไรมารวมในการทำงานของแบบจำลองด้วย ซึ่งถ้าต้องการวัดผลลัพธ์สามารถอ้างอิงจากค่า RMSE, MAE ได้

โดยในงานโครงการของผู้จัดทำนั้นได้นำผลลัพธ์การทดลองของงานวิจัยนี้ที่สรุปได้ว่าผลคะแนนโดยรวมดีที่สุดคืออัลกอริทึม SVD เป็นตัวตัดสินใจในการเลือกใช้อัลกอริทึมนี้

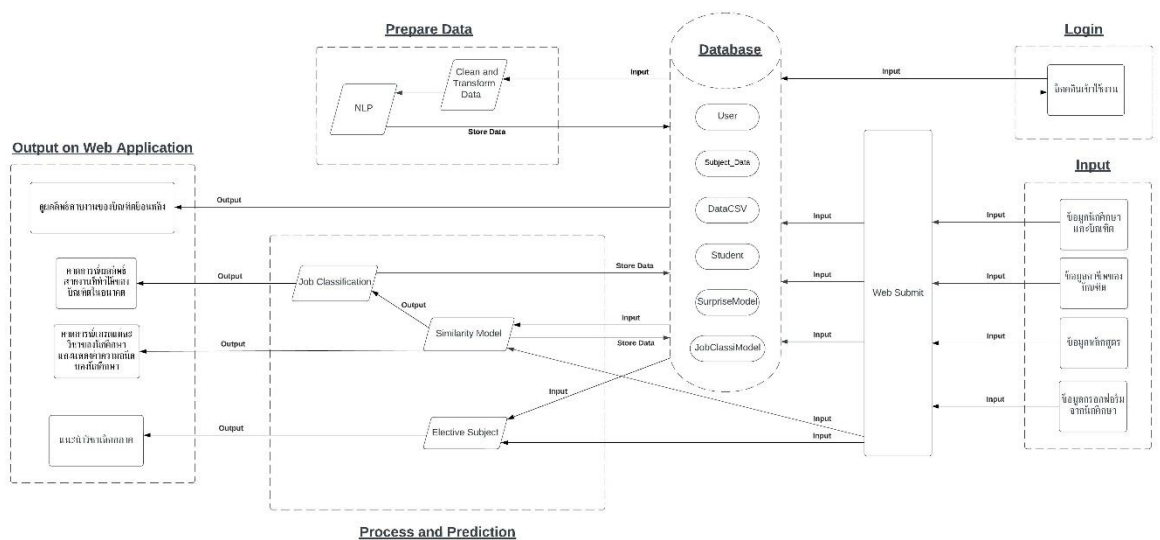
และได้นำวิธีการการวัดประสิทธิภาพของแบบจำลองนี้จากงานวิจัยมาปรับใช้ในรูปแบบเดียวกันกับตัวโครงการ

## บทที่ 3

### การออกแบบ

#### 3.1 โครงสร้างการทำงานของระบบ

โครงสร้างการทำงานของระบบได้อธิบายถึงการเชื่อมต่อระหว่างส่วนต่างๆของระบบ เริ่มตั้งแต่ส่วนของ Input ที่ทำหน้าที่รับข้อมูลเข้ามา แล้วเก็บไว้ในส่วนของ Database จากนั้นส่วนของ Prepare Data จะนำข้อมูลจากส่วนของ Database เมื่อทำเสร็จแล้ว จะทำการส่งกลับไปอัปเดตยัง Database ส่วนของ Process and Prediction จะนำข้อมูลที่ได้อัปโหลดเข้ามาเพื่อ Process ผลลัพธ์ออกมาแสดงผลบน Web Application



รูป 3.1 โครงสร้างการทำงานของระบบ

#### 3.2 การทำงานของระบบ

จากรูป 1 โครงสร้างการทำงานของระบบนั้นประกอบไปด้วยองค์ประกอบทั้งหมด 6 ส่วน ซึ่งมีรายละเอียด ดังนี้

### 3.2.1 Login

เป็นส่วนสำหรับไว้ให้กรรมการหลักสูตรได้ทำการล็อกอินเข้าสู่ระบบเพื่อให้กรรมการหลักสูตรป้อนข้อมูลเกรดของนักศึกษาปัจจุบัน เกรดและอาชีพของบัณฑิต และข้อมูลของหลักสูตร

### 3.2.2 Input

เป็นส่วนที่ทำการรับข้อมูลของนักศึกษาและบัณฑิต ข้อมูลของหลักสูตร แล้วเก็บเข้ายังส่วนของ Database และข้อมูลการกรอกฟอร์มของนักศึกษาจะส่งข้อมูลไปยังส่วนของ Process and Prediction โดยตรง

### 3.2.3 Database

ทำหน้าที่ในการจัดเก็บและบันทึกข้อมูล โดยจะประกอบไปด้วย Table User, Subject\_Data, DataCSV, Graduate, Student

### 3.2.4 Prepare Data

ทำหน้าที่เตรียมพร้อมข้อมูลเพื่อให้พร้อมต่อการนำไปใช้ในส่วนของการ Prediction and Prediction โดยแบ่งออกเป็น 2 ส่วนย่อย ดังนี้

#### 1) Clean and Transform Data

เป็นการเรียกข้อมูลจากใน Database มาทำให้ข้อมูลมีความถูกต้อง ครบถ้วน สมบูรณ์ ไม่มีค่าที่ผิดปกติ เพื่อเตรียมพร้อมสำหรับการประมวลผลข้อมูล โดยเลือกเฉพาะข้อมูลที่เกี่ยวข้องกับวัตถุประสงค์ของการวิเคราะห์ และ ประมวลผล พร้อมทั้งตัดข้อมูลส่วนที่ไม่ได้นำมาใช้

#### 2) NLP หรือ Natural Language Processing

นำข้อมูลรหัสวิชาเข้ากระบวนการ NLP เพื่อทำการหาค่า Similarity ของกลุ่มวิชาที่สามารถอยู่ในกลุ่มเดียวกันได้ โดยใช้บทคัดย่อของแต่ละวิชา เพื่อลดปัญหาการเปลี่ยนรหัสวิชาระหว่างหลักสูตร เมื่อทำเสร็จกระบวนการแล้วจะนำข้อมูลที่ได้กลับไปอัปเดตที่ Database

### 3.2.5 Process and Prediction

เป็นส่วนการประมวลผลหลักของระบบ ประกอบไปด้วย Process 3 ส่วนดังนี้

#### 1) Similarity Model

มีหน้าที่นำข้อมูลผลการเรียนของนักศึกษามาทำให้สมบูรณ์ ด้วยกรณีของนักศึกษาแต่ละคนมีเกรดแต่ละวิชาที่ไม่เหมือนกัน บางคนมีเกรดของวิชานี้ แต่อีกคนไม่มีเกรดของวิชานี้ จึงต้องทำให้ข้อมูลของนักศึกษามีเท่ากันเพื่อนำไปทำการ Prediction ในขั้นต่อไป โดยใช้กระบวนการหลักคือการทำ Recommender Systems และใช้ Library Surprise ของ Scikit ในภาษา Python เป็นตัวช่วย

#### 2) Job Classification

มีหน้าที่ทำนายและสถิติด้านความสามารถทางวิชาชีพของนักศึกษาในอนาคต

#### 3) Elective Subject

มีหน้าที่ประมวลผลจัดกลุ่มของวิชาเลือกภาค เพื่อนำไปแนะนำให้แก่นักศึกษาที่มีความสนใจเฉพาะจุดได้

### 3.2.6 Web Application

เป็นส่วนที่ทำหน้าที่ในการแสดงผลข้อมูลและเป็น interface สำหรับผู้ใช้งาน





ตาราง 3.1 Use Case คู่มือผลลัพธ์รายงานของบัณฑิตย้อนหลัง

<b>Use Case:</b> คู่มือผลลัพธ์รายงานของบัณฑิตย้อนหลัง
<b>Use Case ID:</b> UC-01
<b>Actor:</b> บุคคลทั่วไปหรือนักศึกษา
<b>Description:</b> สามารถดูผลลัพธ์รายงานของบัณฑิตย้อนหลังได้
<b>Precondition:</b> เข้า web application หน้าผลลัพธ์รายงานของบัณฑิตย้อนหลัง
<b>Flow of Events:</b> <ol style="list-style-type: none"> <li>1. เลือกหน้าคู่มือผลลัพธ์รายงานของบัณฑิตย้อนหลัง</li> <li>2. เลือกปีที่ต้องการดู</li> <li>3. เลือกหลักสูตรที่ต้องการดู</li> </ol>
<b>Postcondition:</b> หน้าเว็บแสดงผลผลลัพธ์รายงานของบัณฑิตย้อนหลัง

ตาราง 3.2 Use Case คาดการณ์ผลลัพธ์รายงานที่ทำได้ของบัณฑิตในอนาคต

<b>Use Case:</b> คาดการณ์ผลลัพธ์รายงานที่ทำได้ของบัณฑิตในอนาคต
<b>Use Case ID:</b> UC-02
<b>Actor:</b> บุคคลทั่วไปหรือนักศึกษา
<b>Description:</b> สามารถดูผลลัพธ์รายงานที่ทำได้ของบัณฑิตในอนาคตได้
<b>Precondition:</b> เข้า web application หน้าผลลัพธ์รายงานที่ทำได้ของบัณฑิตในอนาคต
<b>Flow of Events:</b> <ol style="list-style-type: none"> <li>1. เลือกหน้าคู่มือผลลัพธ์รายงานที่ทำได้ของบัณฑิตในอนาคต</li> <li>2. เลือกปีที่ต้องการดู</li> <li>3. เลือกหลักสูตรที่ต้องการดู</li> </ol>
<b>Postcondition:</b> หน้าเว็บแสดงผลผลลัพธ์รายงานที่ทำได้ของบัณฑิตในอนาคต

ตาราง 3.3 Use Case กรอกแบบฟอร์มเกรดสำหรับการวิเคราะห์ข้อมูล

<b>Use Case:</b> กรอกแบบฟอร์มเกรดสำหรับการวิเคราะห์ข้อมูล
<b>Use Case ID:</b> UC-03
<b>Actor:</b> บุคคลทั่วไปหรือนักศึกษา
<b>Description:</b> สามารถกรอกแบบฟอร์มการวิเคราะห์ข้อมูลได้
<b>Precondition:</b> เข้า web application หน้าสำหรับการวิเคราะห์ข้อมูลของนักศึกษา
<b>Flow of Events:</b> <ol style="list-style-type: none"> <li>1. เลือกหน้าสำหรับการวิเคราะห์ข้อมูลของนักศึกษา</li> <li>2. เลือกปีการศึกษา</li> <li>3. เลือกหลักสูตร</li> <li>4. โหลดแบบฟอร์มสำหรับการวิเคราะห์ข้อมูล</li> <li>5. อัปโหลดแบบฟอร์มสำหรับการวิเคราะห์ข้อมูล</li> </ol>
<b>Postcondition:</b> Redirect ไปหน้าแสดงผลวิเคราะห์ข้อมูลนักศึกษา

ตาราง 3.4 Use Case แสดงค่าความถนัดของนักศึกษา

<b>Use Case:</b> แสดงค่าความถนัดของนักศึกษา
<b>Use Case ID:</b> UC-04
<b>Actor:</b> บุคคลทั่วไปหรือนักศึกษา
<b>Description:</b> สามารถดูผลแสดงค่าความถนัดของนักศึกษาได้
<b>Precondition:</b> เข้า web application หน้าแสดงผลวิเคราะห์ข้อมูลนักศึกษา
<b>Flow of Events:</b> <ol style="list-style-type: none"> <li>1. เลือกหน้าแสดงผลสำหรับการวิเคราะห์ข้อมูลของนักศึกษา</li> </ol>
<b>Postcondition:</b> หน้าเว็บแสดงผลผลลัพธ์การวิเคราะห์ข้อมูลค่าความถนัดของนักศึกษา

ตาราง 3.5 Use Case คาดการณ์เกรดแต่ละวิชาของนักศึกษา

<b>Use Case:</b> คาดการณ์เกรดแต่ละวิชาของนักศึกษา
<b>Use Case ID:</b> UC-05
<b>Actor:</b> บุคคลทั่วไปหรือนักศึกษา
<b>Description:</b> สามารถดูผลแสดงค่าคาดการณ์เกรดแต่ละวิชาของนักศึกษาได้
<b>Precondition:</b> เข้า web application หน้าแสดงผลวิเคราะห์ข้อมูลนักศึกษา
<b>Flow of Events:</b> <ol style="list-style-type: none"> <li>1. เลือกหน้าแสดงผลสำหรับการวิเคราะห์ข้อมูลของนักศึกษา</li> </ol>
<b>Postcondition:</b> หน้าเว็บแสดงผลผลลัพธ์การวิเคราะห์ข้อมูลค่าคาดการณ์เกรดแต่ละวิชาของนักศึกษา

ตาราง 3.6 Use Case แนะนำวิชาเลือกภาคให้แก่นักศึกษา

<b>Use Case:</b> แนะนำวิชาเลือกภาคให้แก่นักศึกษา
<b>Use Case ID:</b> UC-06
<b>Actor:</b> บุคคลทั่วไปหรือนักศึกษา
<b>Description:</b> สามารถดูผลแนะนำวิชาเลือกให้ภาคให้แก่นักศึกษาได้
<b>Precondition:</b> เข้า web application หน้าแนะนำวิชาเลือกภาค
<b>Flow of Events:</b> <ol style="list-style-type: none"> <li>1. เลือกหน้าแนะนำวิชาเลือกภาค</li> <li>2. เลือกสิ่งที่น่าสนใจภายในตัวเลือกที่มีให้</li> </ol>
<b>Postcondition:</b> หน้าเว็บแนะนำวิชาเลือกภาค

ตาราง 3.7 Use Case ให้ข้อมูลนักศึกษาโดยไม่ระบุตัวตนและข้อมูลหลักสูตร

<b>Use Case:</b> ให้ข้อมูลนักศึกษาโดยไม่ระบุตัวตนและข้อมูลหลักสูตร
<b>Use Case ID:</b> UC-07
<b>Actor:</b> กรรมการหลักสูตร
<b>Description:</b> ให้ข้อมูลนักศึกษาโดยไม่ระบุตัวตนและข้อมูลหลักสูตรได้
<b>Precondition:</b> เข้า web application หน้าให้ข้อมูลนักศึกษาโดยไม่ระบุตัวตนและข้อมูลหลักสูตร
<b>Flow of Events:</b> <ol style="list-style-type: none"> <li>1. เลือกหน้าให้ข้อมูลนักศึกษาโดยไม่ระบุตัวตนและข้อมูลหลักสูตร</li> <li>2. อัปโหลดไฟล์ CSV</li> <li>3. กดปุ่มอัปโหลด</li> </ol>
<b>Postcondition:</b> อัปโหลดข้อมูลสำเร็จ

ตาราง 3.8 Use Case อัปเดตข้อมูลนักศึกษาโดยไม่ระบุตัวตนและข้อมูลหลักสูตร

<b>Use Case:</b> อัปเดตข้อมูลนักศึกษาโดยไม่ระบุตัวตนและข้อมูลหลักสูตร
<b>Use Case ID:</b> UC-08
<b>Actor:</b> กรรมการหลักสูตร
<b>Description:</b> อัปเดตนักศึกษาโดยไม่ระบุตัวตนและข้อมูลหลักสูตรได้
<b>Precondition:</b> เข้า web application หน้าให้ข้อมูลนักศึกษาโดยไม่ระบุตัวตนและข้อมูลหลักสูตร
<b>Flow of Events:</b> <ol style="list-style-type: none"> <li>1. เลือกหน้าให้ข้อมูลนักศึกษาโดยไม่ระบุตัวตนและข้อมูลหลักสูตร</li> <li>2. อัปเดตไฟล์ CSV</li> <li>3. กดปุ่มอัปโหลด</li> </ol>
<b>Postcondition:</b> อัปเดตข้อมูลสำเร็จ

ตาราง 3.9 Use Case ลบข้อมูลนักศึกษาโดยไม่ระบุตัวตนและข้อมูลหลักสูตร

<b>Use Case:</b> อัปเดตข้อมูลนักศึกษาโดยไม่ระบุตัวตนและข้อมูลหลักสูตร
<b>Use Case ID:</b> UC-09
<b>Actor:</b> กรรมการหลักสูตร
<b>Description:</b> ลบนักศึกษาโดยไม่ระบุตัวตนและข้อมูลหลักสูตรได้
<b>Precondition:</b> เข้า web application หน้าให้ข้อมูลนักศึกษาโดยไม่ระบุตัวตนและข้อมูลหลักสูตร
<b>Flow of Events:</b> <ol style="list-style-type: none"> <li>1. เลือกหน้าให้ข้อมูลนักศึกษาโดยไม่ระบุตัวตนและข้อมูลหลักสูตร</li> <li>2. ลบไฟล์ CSV</li> <li>3. กดปุ่มยืนยัน</li> </ol>
<b>Postcondition:</b> ลบข้อมูลสำเร็จ

ตาราง 3.10 Use Case ล็อกอินใช้งาน

<b>Use Case:</b> ให้ข้อมูลนักศึกษาโดยไม่ระบุตัวตนและข้อมูลหลักสูตร
<b>Use Case ID:</b> UC-10
<b>Actor:</b> กรรมการหลักสูตร
<b>Description:</b> กรรมการหลักสูตรสามารถล็อกอินใช้งานได้
<b>Precondition:</b> เข้า web application หน้า login
<b>Flow of Events:</b> <ol style="list-style-type: none"> <li>1. กรอกข้อมูล username และ password</li> <li>2. กดปุ่ม Login</li> </ol>
<b>Postcondition:</b> Redirect ไปหน้าหลัก

ตาราง 3.11 Use Case ทำนายพยากรณ์ผลลัพธ์และคาดการณ์

<b>Use Case:</b> ทำนายพยากรณ์ผลลัพธ์และคาดการณ์
<b>Use Case ID:</b> UC-11
<b>Actor:</b> แอดมิน
<b>Description:</b> แอดมินสามารถทำนายพยากรณ์ผลลัพธ์และคาดการณ์ได้
<b>Precondition:</b> เป็นแอดมิน
<b>Flow of Events:</b> <ol style="list-style-type: none"> <li>1. ทำการทำนายพยากรณ์และคาดการณ์</li> <li>2. ส่งออกผลลัพธ์ไปยังหน้าเว็บ</li> </ol>
<b>Postcondition:</b> ทำนายพยากรณ์ผลลัพธ์และคาดการณ์ได้สำเร็จ

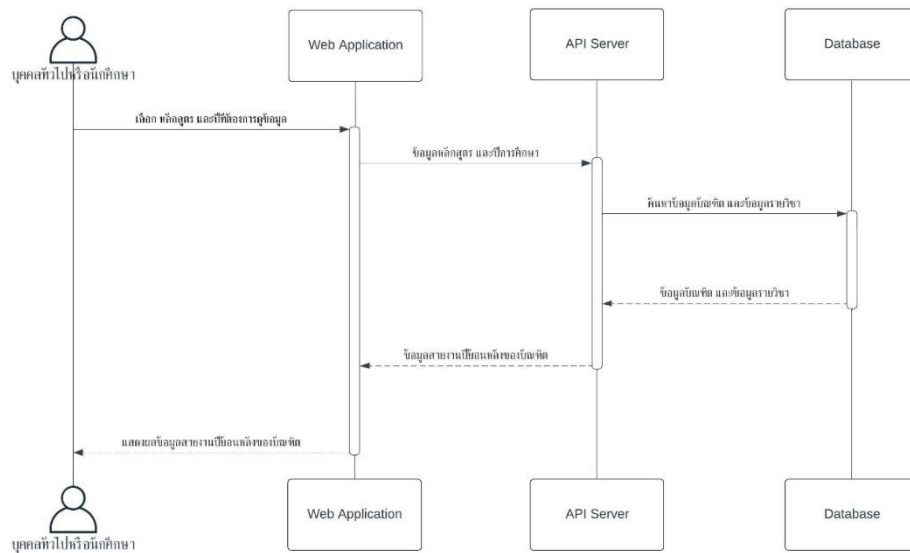
ตาราง 3.12 Use Case จัดเตรียมข้อมูล

<b>Use Case:</b> จัดเตรียมข้อมูล
<b>Use Case ID:</b> UC-12
<b>Actor:</b> แอดมิน
<b>Description:</b> จัดเตรียมข้อมูลสำหรับการทำนายและคาดการณ์ได้
<b>Precondition:</b> เป็นแอดมิน
<b>Flow of Events:</b> <ol style="list-style-type: none"> <li>1. นำเข้าข้อมูล</li> <li>2. จัดรูปแบบข้อมูล</li> <li>3. เก็บเข้าฐานข้อมูล</li> </ol>
<b>Postcondition:</b> จัดเตรียมข้อมูลสำหรับการทำนายและคาดการณ์ได้สำเร็จ

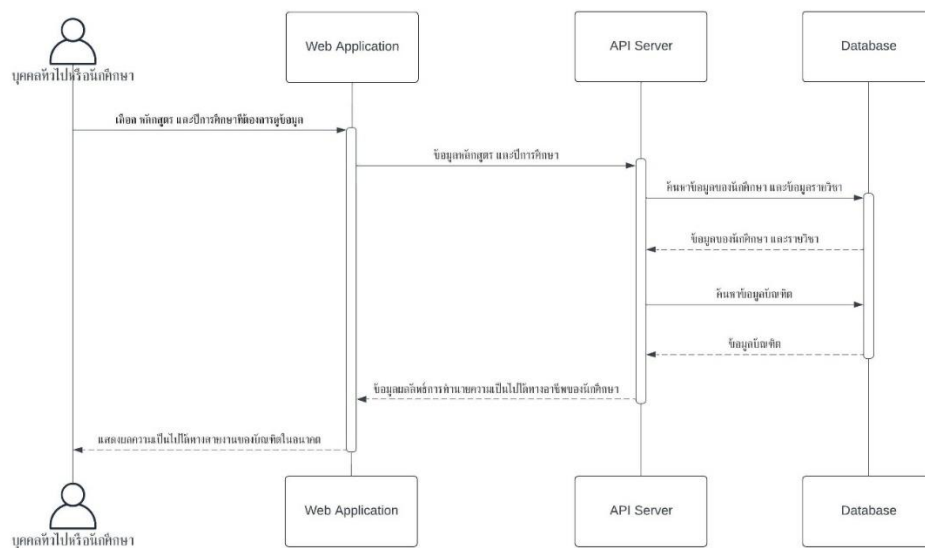
ตาราง 3.13 Use Case กรอกข้อมูลอาชีพ

<b>Use Case:</b> กรอกข้อมูลอาชีพ
<b>Use Case ID:</b> UC-13
<b>Actor:</b> นักศึกษา
<b>Description:</b> กรอกข้อมูลอาชีพแรกของการทำงานได้
<b>Precondition:</b> เข้า web application หน้ากรอกข้อมูลอาชีพ
<b>Flow of Events:</b> <ol style="list-style-type: none"> <li>1. เลือกหน้ากรอกข้อมูลอาชีพ</li> <li>2. ใส่รหัสนักศึกษา</li> <li>3. ใส่ตำแหน่งอาชีพ</li> <li>4. กดยืนยัน</li> </ol>
<b>Postcondition:</b> กรอกข้อมูลอาชีพสำเร็จ

### 3.4 Sequence Diagram

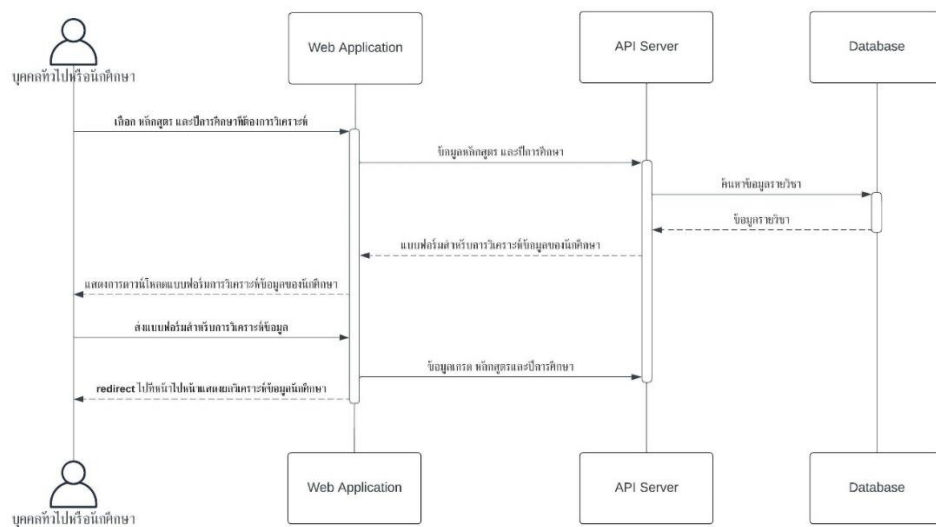


รูป 3.3 Sequence Diagram สำหรับ UC-01

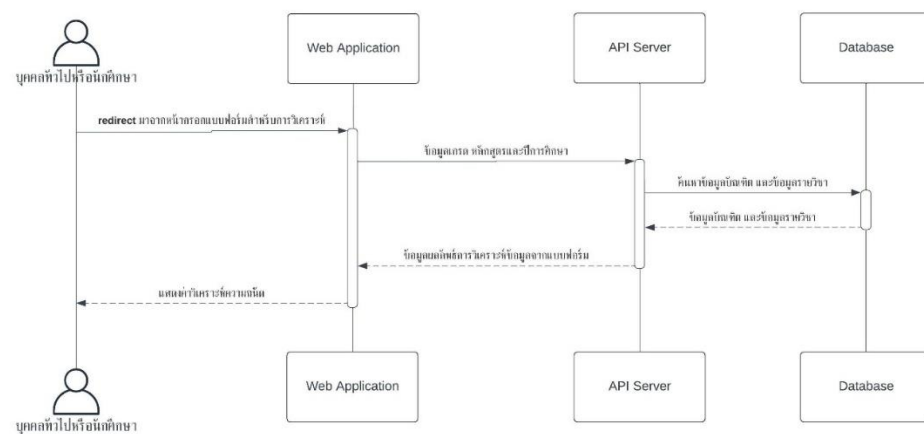


รูป 3.4 Sequence Diagram สำหรับ UC-02

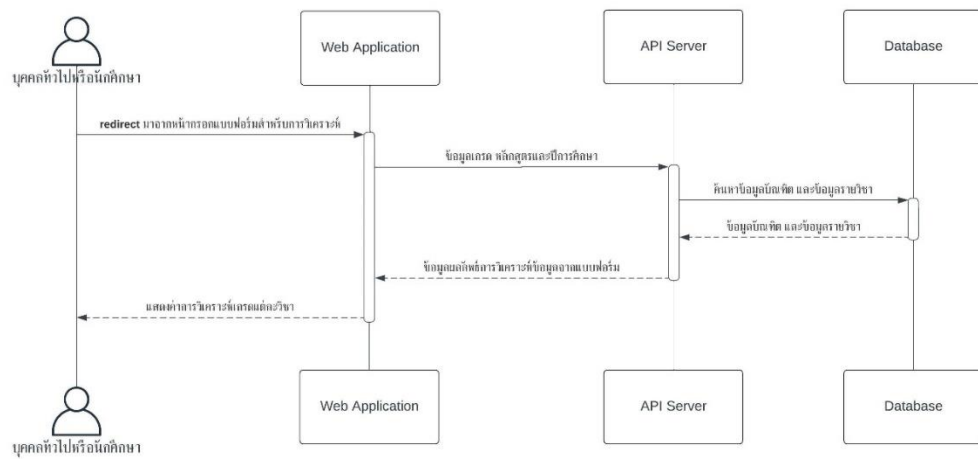




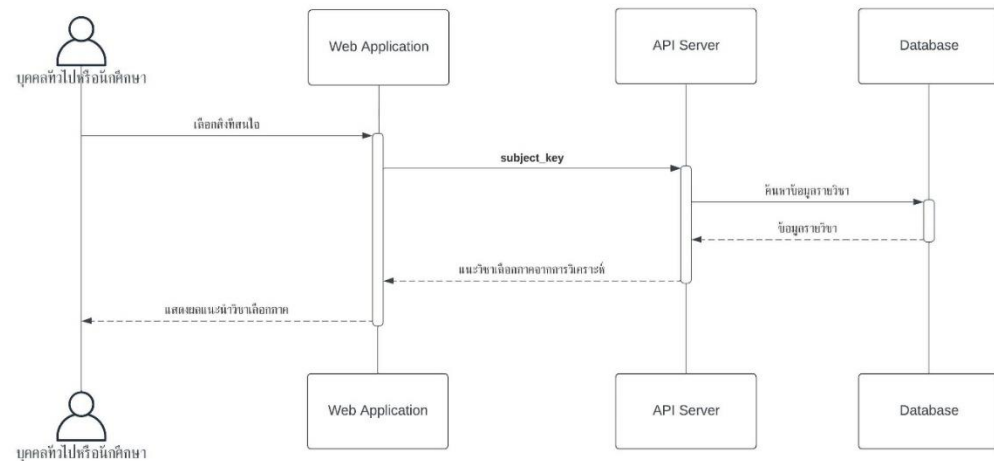
รูป 3.5 Sequence Diagram สำหรับ UC-03



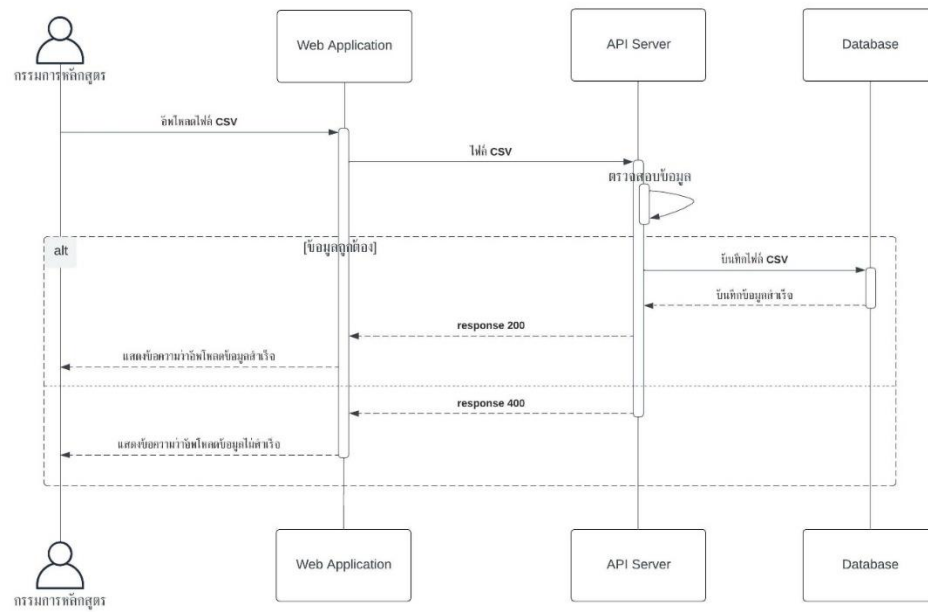
รูป 3.6 Sequence Diagram สำหรับ UC-04



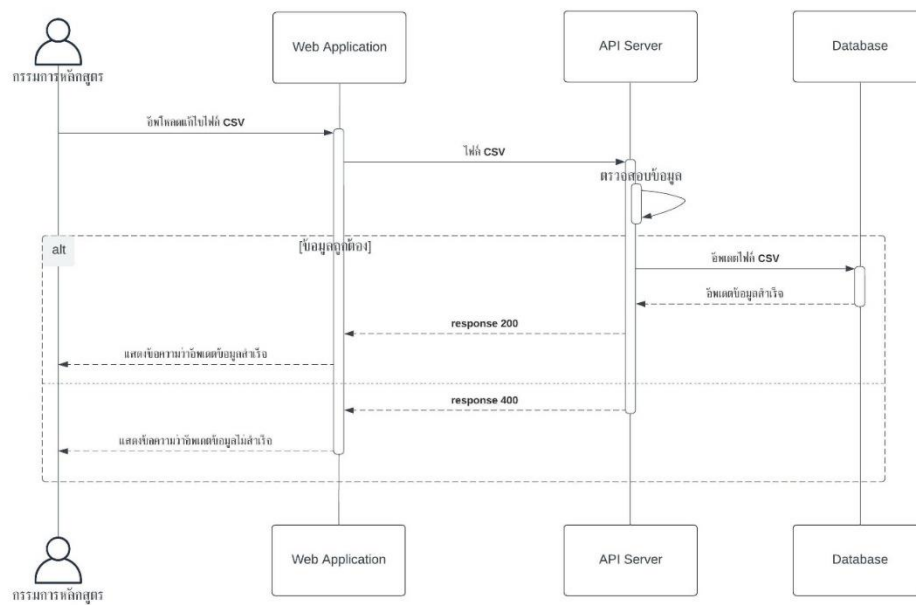
รูป 3.7 Sequence Diagram สำหรับ UC-05



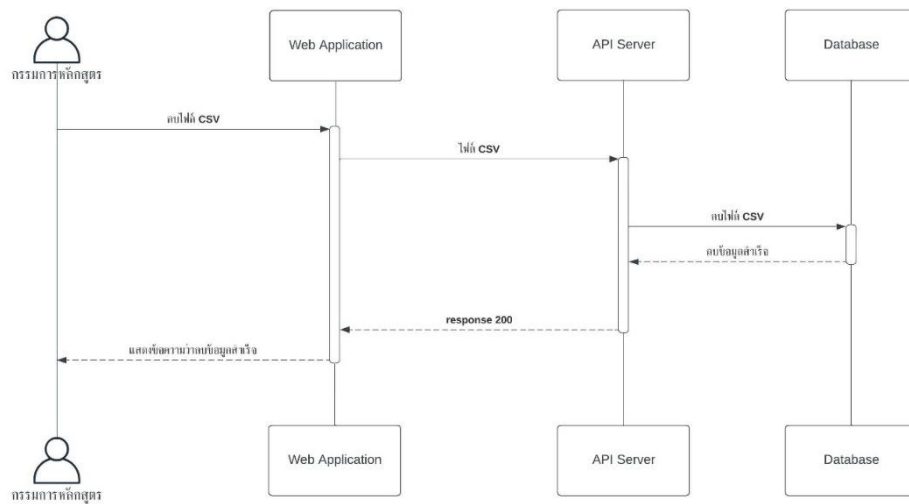
รูป 3.8 Sequence Diagram สำหรับ UC-06



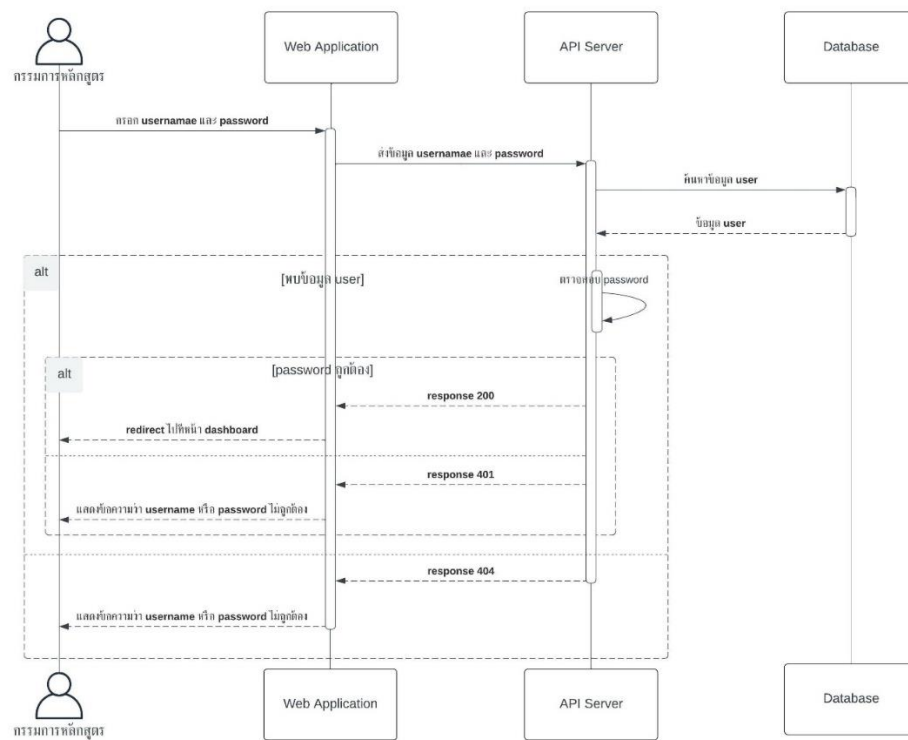
รูป 3.9 Sequence Diagram สำหรับ UC-07



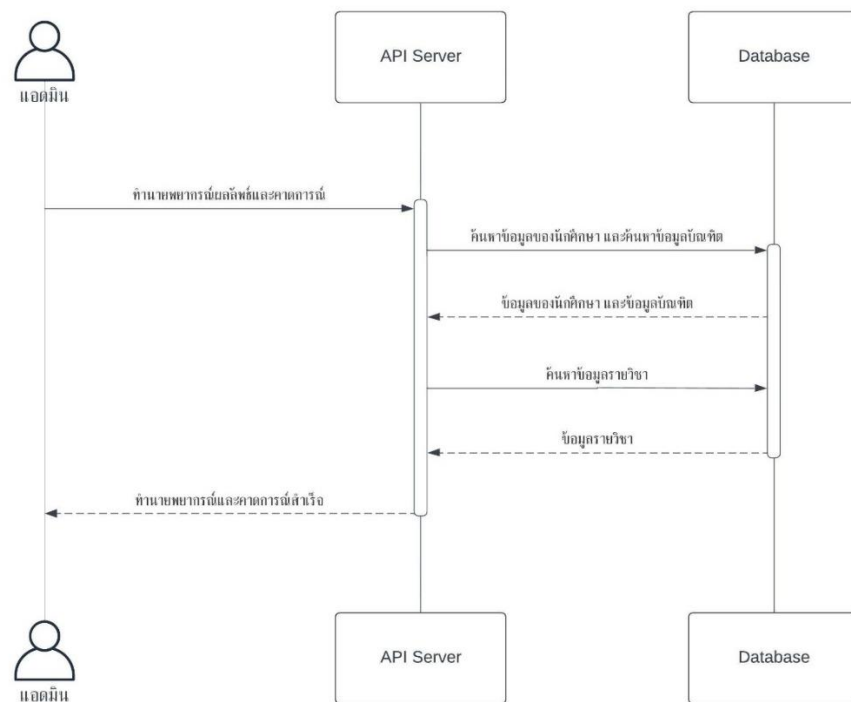
รูป 3.10 Sequence Diagram สำหรับ UC-08



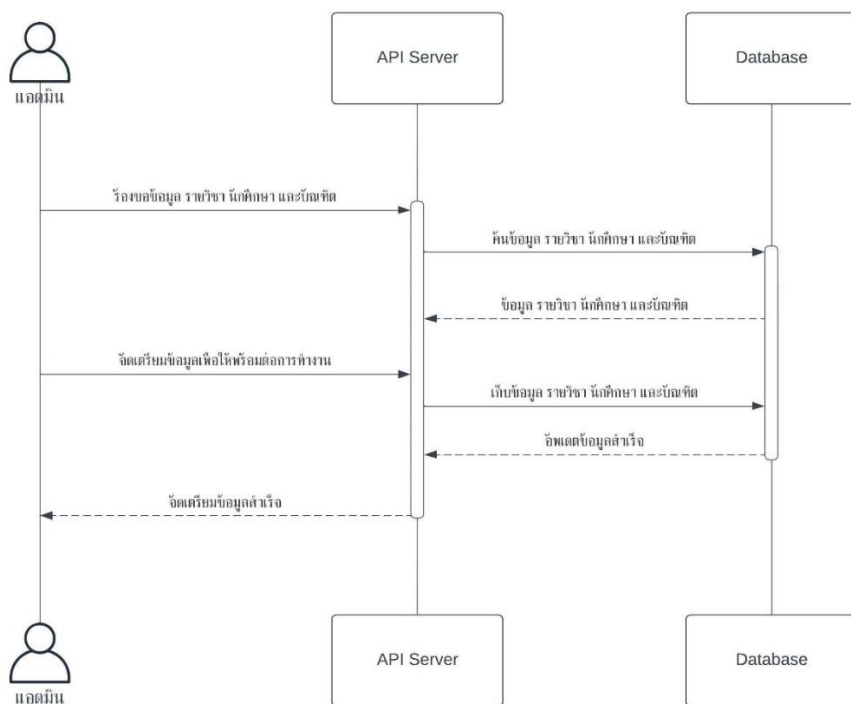
รูป 3.11 Sequence Diagram สำหรับ UC-09



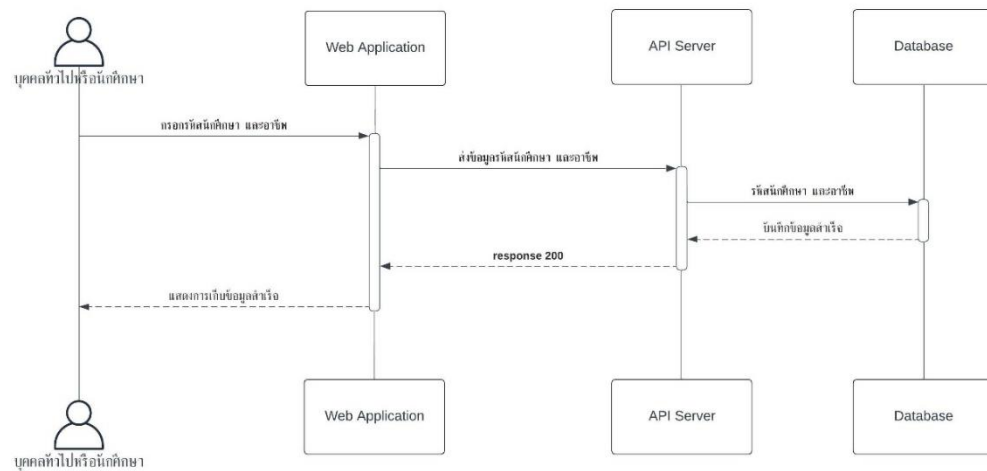
รูป 3.12 Sequence Diagram สำหรับ UC-10



รูป 3.13 Sequence Diagram สำหรับ UC-11



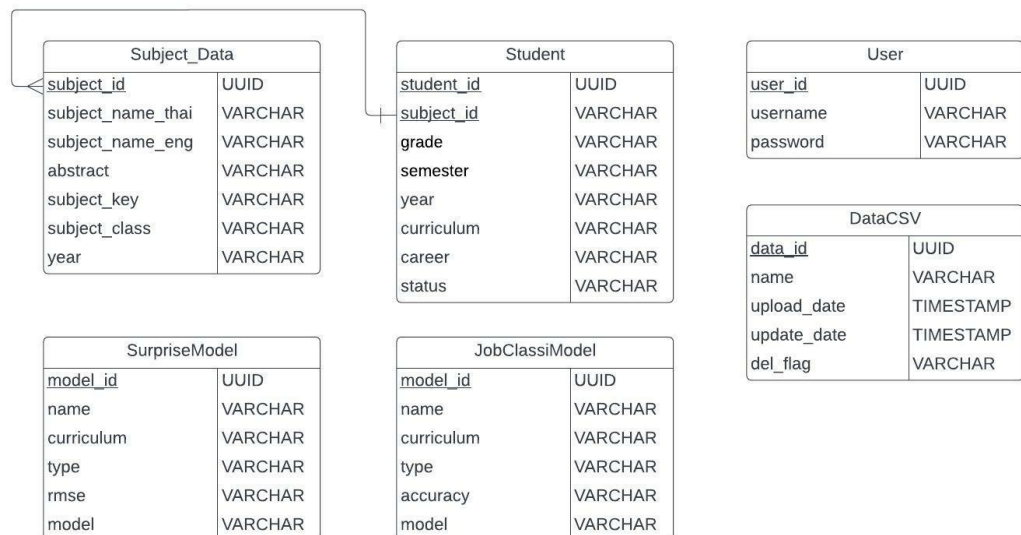
รูป 3.14 Sequence Diagram สำหรับ UC-12



รูป 3.15 Sequence Diagram สำหรับ UC-13

### 3.5 การออกแบบฐานข้อมูล

ระบบฐานข้อมูลของโครงการนี้ผู้จัดทำได้เลือกใช้ MariaDB ซึ่งเป็นฐานข้อมูลแบบ SQL ที่ถูกพัฒนามาเพื่อสำหรับเก็บข้อมูล โดยฐานข้อมูลของระบบประกอบไปด้วยทั้งหมด 5 ตาราง ดังรูป



รูป 3.16 แสดงความสัมพันธ์ระหว่างตารางของฐานข้อมูลในระบบ

### 3.5.1 ตาราง User

เป็นตารางที่ใช้เก็บข้อมูลผู้ใช้งานในระบบ โดยจะมี Attribute ดังนี้

1. user\_id : UUID สำหรับเก็บ id ของผู้ใช้
2. username : VARCHAR สำหรับเก็บ username ของผู้ใช้
3. password : VARCHAR สำหรับเก็บ password ของผู้ใช้

### 3.5.2 ตาราง Subject\_Data

เป็นตารางที่ใช้เก็บข้อมูลหลักสูตรวิชา โดยจะมี Attribute ดังนี้

1. subject\_id : UUID สำหรับเก็บ id ของรหัสวิชา
2. subject\_name\_thai : VARCHAR สำหรับเก็บชื่อรายวิชาภาษาไทย
3. subject\_name\_eng : VARCHAR สำหรับเก็บชื่อรายวิชาภาษาอังกฤษ
4. abstract : VARCHAR สำหรับเก็บบทคัดย่อรายวิชา
5. subject\_key : VARCHAR สำหรับเก็บ Keyword ของรายวิชาเลือกภาค
6. subject\_class : VARCHAR สำหรับเก็บหมวดหมู่ของแต่ละวิชา
7. year : VARCHAR สำหรับเก็บปีของเล่มหลักสูตรวิชา

### 3.5.3 ตาราง Student

เป็นตารางที่ใช้เก็บข้อมูลนักศึกษาโดยจะมี Attribute ดังนี้

1. student\_id : UUID สำหรับเก็บ id encrypt ของนักศึกษา
2. subject\_id : VARCHAR สำหรับเก็บ id ของรหัสวิชา
3. grade : VARCHAR สำหรับเก็บเกรดแต่ละรายวิชา
4. semester : VARCHAR สำหรับเก็บเทอมที่ลงเรียนรายวิชาของนักศึกษา
5. year : VARCHAR สำหรับเก็บปีที่ลงเรียนรายวิชาของนักศึกษา
6. curriculum : VARCHAR สำหรับเก็บรายชื่อหลักสูตรที่เรียนของนักศึกษา
7. career : VARCHAR สำหรับเก็บรายชื่อตำแหน่งอาชีพแรกของนักศึกษา
8. status : VARCHAR สำหรับเก็บสถานะการเป็นนักศึกษา

### 3.5.4 ตาราง DataCSV

เป็นตารางที่ใช้เก็บข้อมูลไฟล์ CSV โดยจะมี Attribute ดังนี้

1. data\_id : UUID สำหรับเก็บ id ของไฟล์ CSV
2. name : VARCHAR สำหรับเก็บชื่อของไฟล์ CSV
3. upload\_date : TIMESTAMP สำหรับเก็บเวลาและวันที่อัปโหลดไฟล์ CSV
4. update\_date : TIMESTAMP สำหรับเก็บเวลาและวันที่อัปเดตไฟล์ CSV

5. del\_flag : VARCHAR สำหรับเก็บสถานะไฟล์ CSV (0 = ไฟล์ยังมีอยู่, 1 = ไฟล์ถูกลบ)

### 3.5.5 ตาราง SurpriseModel

เป็นตารางที่ใช้เก็บ Surprise Model โดยจะมี Attribute ดังนี้

1. model\_id : UUID สำหรับเก็บ id ของไฟล์ Model
2. name : VARCHAR สำหรับเก็บชื่อของ Model
3. curriculum : VARCHAR สำหรับเก็บรายชื่อหลักสูตรที่เรียนของนักศึกษา
4. type : VARCHAR สำหรับเก็บประเภทของ Model
5. rmse : VARCHAR สำหรับเก็บค่าความแม่นยำของ Model
6. model : PICKLED OBJECT สำหรับเก็บตัว Model

### 3.5.6 ตาราง JobClassiModel

เป็นตารางที่ใช้เก็บ Job Classification Model โดยจะมี Attribute ดังนี้

1. model\_id : UUID สำหรับเก็บ id ของไฟล์ Model
2. name : VARCHAR สำหรับเก็บชื่อของ Model
3. curriculum : VARCHAR สำหรับเก็บรายชื่อหลักสูตรที่เรียนของนักศึกษา
4. type : VARCHAR สำหรับเก็บประเภทของ Model
5. rmse : VARCHAR สำหรับเก็บค่าความแม่นยำของ Model
6. model : PICKLED สำหรับเก็บตัว Model



## บทที่ 4

# ผลการดำเนินงาน

ในบทนี้จะอธิบายถึงการทดลองที่ได้ทำในภาคการศึกษานี้ เพื่อแสดงให้เห็นถึงความก้าวหน้าในโครงการ

### 4.1 การสร้าง Django Framework และเชื่อมต่อฐานข้อมูล MariaDB บน Docker Compose

โดยการทดลองการสร้างจะตัว Django Project ให้เชื่อมต่อกับ Maria DB ได้นั้น ทางผู้จัดทำได้ทำการเตรียม requirements.txt ซึ่งเป็น File เพื่อให้ Docker นั้นจัดเตรียม Environment ของตัวระบบดังรูปที่ 4.1

```
asgiref==3.5.2
blis==0.7.8
catalogue==2.0.8
certifi==2022.9.24
charset-normalizer==2.1.1
click==7.1.2
colorama==0.4.5
confection==0.0.3
cycler==0.11.0
cyrene==2.0.6
Cython==0.29.32
Django==3.2.16
django-cors-headers==3.13.0
django-rest-framework==3.14.0
en-core-web-sm @ https://github.com/explosion/spacy-models/releases/download/en_core_web_sm-3.4.0/en_core_web_sm-3.4.0-py3-none-any.whl
fonttools==4.37.1
gensim==3.6.0
greenlet==1.1.3
idna==2.10
Jinja2==2.11.3
joblib==1.2.0
kissolver==1.4.4
langcodes==3.3.0
MarkupSafe==2.0.1
matplotlib==3.2.2
murmurhash==1.0.8
mysql==0.0.3
mysqlclient==2.1.1
nltk==3.7
numpy==1.21.6
packaging==21.3
pandas==1.3.5
pandasql==0.7.3
pathy==0.6.2
Pillow==7.1.2
prezld==3.0.7
pydantic==1.9.2
PyMySQL==1.0.2
pyparsing==3.0.9
python-dateutil==2.8.2
pytz==2022.4
regex==2022.6.2
requests==2.28.0
scikit-learn==1.0.2
scikit-surprise==1.1.1
scipy==1.7.3
six==1.15.0
smart-open==5.2.1
spacy==1.4.1
spacy-legacy==3.0.10
spacy-loggers==1.0.3
SQLAlchemy==1.4.41
sqlparse==0.4.3
srsly==2.4.4
thinc==8.1.3
threadpoolctl==3.1.0
tqdm==4.64.1
types==4.2
typing_extensions==4.1.1
tzdata==2022.2
urllib3==1.24.3
wasabi==0.10.1
pickle5==0.0.13
pickleshare==0.7.5
django-picklefield==3.1
```

รูป 4.1 File Requirements

หลังจากได้มีการเตรียม File Requirements แล้วทางผู้จัดทำได้ทำการสร้าง Docker File เพื่อนำ Requirements ที่ได้มาจัดสรร Library ที่จำเป็นต่อการพัฒนาโครงการ โดยได้กำหนด Version ของ Python เป็น 3.7.14 ดังรูปที่ 4.2

```
You, last month | I author (You)
FROM python:3.7.14

ENV PYTHONUNBUFFERED 1
You, 3 months ago • Little set up d
RUN mkdir /requirements
WORKDIR /requirements
COPY requirements.txt /requirements/
RUN pip install -r requirements.txt

RUN mkdir /code
WORKDIR /code
```

รูป 4.2 Docker file

ต่อมาผู้จัดทำได้สร้าง File Docker Compose ขึ้นมาเพื่อนำ Docker Container ในส่วนของ Django Application และ MariaDB นำมารวมเป็น Docker Compose โดยกำหนด Internal Port ของ MariaDB เป็น 3306 และ External Port เป็น 3308 เพื่อจัดระเบียบในการใช้งานของฐานข้อมูล โดยทางผู้จัดทำออกแบบให้ Django Application ติดต่อกับฐานข้อมูลผ่าน Internal Port โดยกำหนด Port ของ Django Application เป็น 8000 ทั้ง Internal และ External Port ดังรูปที่ 4.3

```

version: '3'

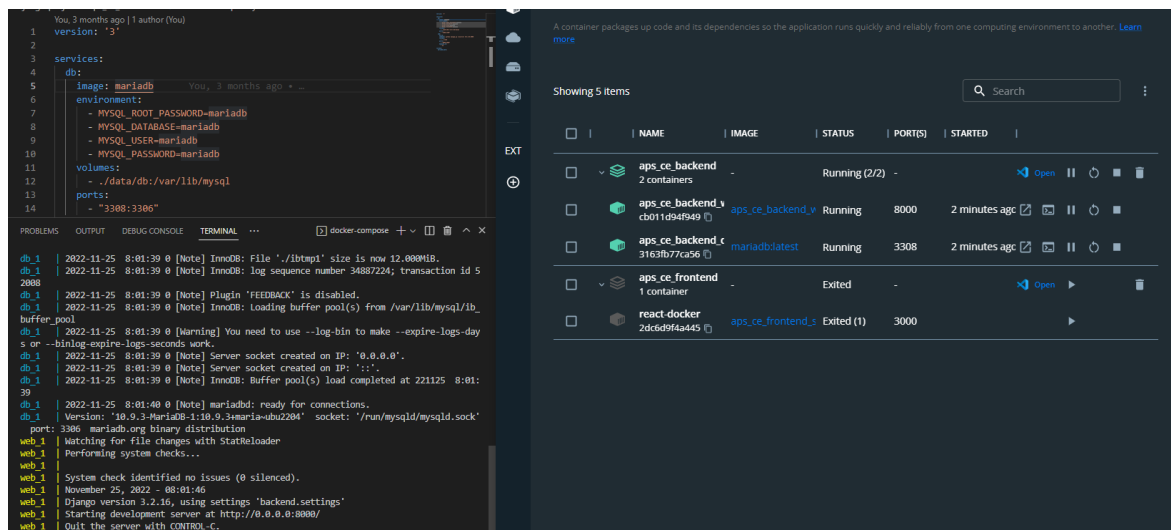
services:
  db:
    image: mariadb
    environment:
      - MYSQL_ROOT_PASSWORD=mariadb
      - MYSQL_DATABASE=mariadb
      - MYSQL_USER=mariadb
      - MYSQL_PASSWORD=mariadb
    volumes:
      - ./data/db:/var/lib/mysql
    ports:
      - "3308:3306"
  web:
    build: .
    command: python manage.py runserver 0.0.0.0:8000
    volumes:
      - ./code
    ports:
      - "8000:8000"
    depends_on:
      - db

volumes:
  mariadb_data:

```

รูป 4.3 File Docker - compose.yml

ซึ่งเมื่อหลังจากการสร้าง Docker Compose ขึ้นมาทางผู้จัดทำจะตรวจสอบการทำงานของระบบทั้งหมดผ่าน Application Docker Desktop ร่วมกับ Log Terminal ของ Visual Studio Code ดังรูปที่ 4.4



รูป 4.4 ภาพรวมของการตรวจสอบการทำงานของระบบ

จากผลการทดลองที่ 4.1 พบว่าการการสร้าง Django Framework และเชื่อมต่อฐานข้อมูล MariaDB บน Docker Compose พบว่าการทำงานของ Container ทั้ง Django Application และ

MariaDB Database นั้นทำงานร่วมกันเป็นไปอย่างราบรื่น และการตรวจสอบการทำงานของ Docker Compose ผ่าน Docker Desktop นั้นทำให้การทำงานและการตรวจสอบได้สะดวกยิ่งขึ้น

## 4.2 การเตรียม Server ที่ใช้สำหรับ Deploy Application

โดยทางผู้จัดทำได้ทำการทดลองการใช้งาน Server สำหรับ Deploy โดยได้ทำการสร้าง Directory ของผู้ทดลองขึ้นมาและได้ทำการทดลองติดตั้ง Program ที่จำเป็นในการรองรับการทำงานของ Application มี Python, MariaDB

ผลการทดลองพบว่าไม่สามารถติดตั้ง Python Version 3.10.2 ที่ผู้จัดทำต้องการได้ซึ่งปัญหาเกิดจากการแตก File Python ที่ติดตั้งไม่สำเร็จ

โดยทางผู้จัดทำจึงแก้ปัญหาโดยการนำ Docker Container ไป Deploy บน Server แทน โดยทางผู้จัดทำได้ทำการนำ Directory ที่เก็บ Django Application รวมทั้ง Docker File, requirements.txt และ docker-compose.yml โค้ดผ่าน Github ลงไปยัง server ที่เตรียมไว้ ดังรูปที่ 4.5

```
[analytic@itanalytic1 ~]$ ls
62010465  mariadb_repo_setup
[analytic@itanalytic1 ~]$ cd 62010465/
[analytic@itanalytic1 62010465]$ ls
project  test.py
[analytic@itanalytic1 62010465]$ cd project
[analytic@itanalytic1 project]$ ls
APS_CE  tempFol
[analytic@itanalytic1 project]$ cd APS_CE
[analytic@itanalytic1 APS_CE]$ ls
django project
```

รูป 4.5 Directory ที่เก็บ Django Application

โดยต่อมาทางผู้จัดทำได้ทำการทดลองใช้คำสั่งในการสร้าง Docker Compose ขึ้นมาใน Server โดยจะสร้าง Docker Compose ผ่านการอ่าน File Docker File, requirements.txt และ docker-compose.yml ที่มากับ Directory ของ Django Application ดังรูปที่ 4.6

```

Successfully built fef4f47f41a1
Successfully tagged apsebackend_web:latest
WARNING: Image for service web was built because it did not already exist. To rebuild this image you must use 'docker-compose build' or 'docker-compose up --Creating apsebackend_db_1 ...
Creating apsebackend_db_1 ... done
Attaching to apsebackend_db_1, apsebackend_web_1
db_1 | 2022-11-27 16:45:40+00:00 [Note] [Entrypoint]: Entrypoint script for MariaDB Server 1:10.9.2+maria-ubu2204 started.
db_1 | 2022-11-27 16:45:40+00:00 [Note] [Entrypoint]: Switching to dedicated user 'mysql'
db_1 | 2022-11-27 16:45:40+00:00 [Note] [Entrypoint]: Entrypoint script for MariaDB Server 1:10.9.2+maria-ubu2204 started.
db_1 | 2022-11-27 16:45:40+00:00 [Note] [Entrypoint]: MariaDB upgrade not required
db_1 | 2022-11-27 16:45:40 0 [Note] mariadbd (server 10.9.2+MariaDB-1:10.9.2+maria-ubu2204) starting as process 1 ...
db_1 | 2022-11-27 16:45:40 0 [Note] InnoDB: Compressed tables use zlib 1.2.11
db_1 | 2022-11-27 16:45:40 0 [Note] InnoDB: Using transactional memory
db_1 | 2022-11-27 16:45:40 0 [Note] InnoDB: Number of transaction pools: 1
db_1 | 2022-11-27 16:45:40 0 [Note] InnoDB: Using crc32 + pclmulqdq instructions
db_1 | 2022-11-27 16:45:40 0 [Note] mariadbd: O_TMPFILE is not supported on /tmp (disabling future attempts)
db_1 | 2022-11-27 16:45:40 0 [Warning] mariadbd: io_uring queue init() failed with ENOSYS: check seccomp filters, and the kernel version (newer than 5.1 required)
db_1 | 2022-11-27 16:45:40 0 [Warning] InnoDB: liburing disabled: falling back to innodb_use_native_aio=OFF
db_1 | 2022-11-27 16:45:40 0 [Note] InnoDB: Initializing buffer pool, total size = 128.000MiB, chunk size = 2.000MiB
db_1 | 2022-11-27 16:45:40 0 [Note] InnoDB: Completed initialization of buffer pool
db_1 | 2022-11-27 16:45:40 0 [Note] InnoDB: File system buffers for log disabled (block size=512 bytes)
db_1 | 2022-11-27 16:45:40 0 [Note] InnoDB: 128 rollback segments are active.
db_1 | 2022-11-27 16:45:40 0 [Note] InnoDB: Setting file './ibtmp1' size to 12.000MiB. Physically writing the file full; Please wait ...
db_1 | 2022-11-27 16:45:40 0 [Note] InnoDB: File './ibtmp1' size is now 12.000MiB.
db_1 | 2022-11-27 16:45:40 0 [Note] InnoDB: log sequence number 46667; transaction id 14
db_1 | 2022-11-27 16:45:40 0 [Note] InnoDB: Loading buffer pool(s) from /var/lib/mysql/ib_buffer_pool
db_1 | 2022-11-27 16:45:40 0 [Note] Plugin 'FEEDBACK' is disabled.
db_1 | 2022-11-27 16:45:40 0 [Warning] You need to use --log-bin to make --expire-logs-days or --binlog-expire-logs-seconds work.
db_1 | 2022-11-27 16:45:40 0 [Note] Server socket created on IP: '0.0.0.0'
db_1 | 2022-11-27 16:45:40 0 [Note] Server socket created on IP: '::'
db_1 | 2022-11-27 16:45:40 0 [Note] InnoDB: Buffer pool(s) load completed at 221127 16:45:40
db_1 | 2022-11-27 16:45:40 0 [Note] mariadbd: ready for connections.
Version: '10.9.2+MariaDB-1:10.9.2+maria-ubu2204' socket: '/run/mysqld/mysqld.sock' port: 3306 mariadb.org binary distribution
web_1 | Matching for file changes with StatReloader
web_1 | Performing system checks...
web_1 |
web_1 | System check identified no issues (0 silenced).
web_1 |
web_1 | You have 18 unapplied migration(s). Your project may not work properly until you apply the migrations for app(s): admin, auth, contenttypes, sessions.
web_1 | Run 'python manage.py migrate' to apply them.
web_1 | November 27, 2022 - 16:45:41
web_1 | Django version 4.1, using settings 'backend.settings'
web_1 | Starting development server at http://0.0.0.0:8000/
web_1 | Quit the server with CONTROL-C.

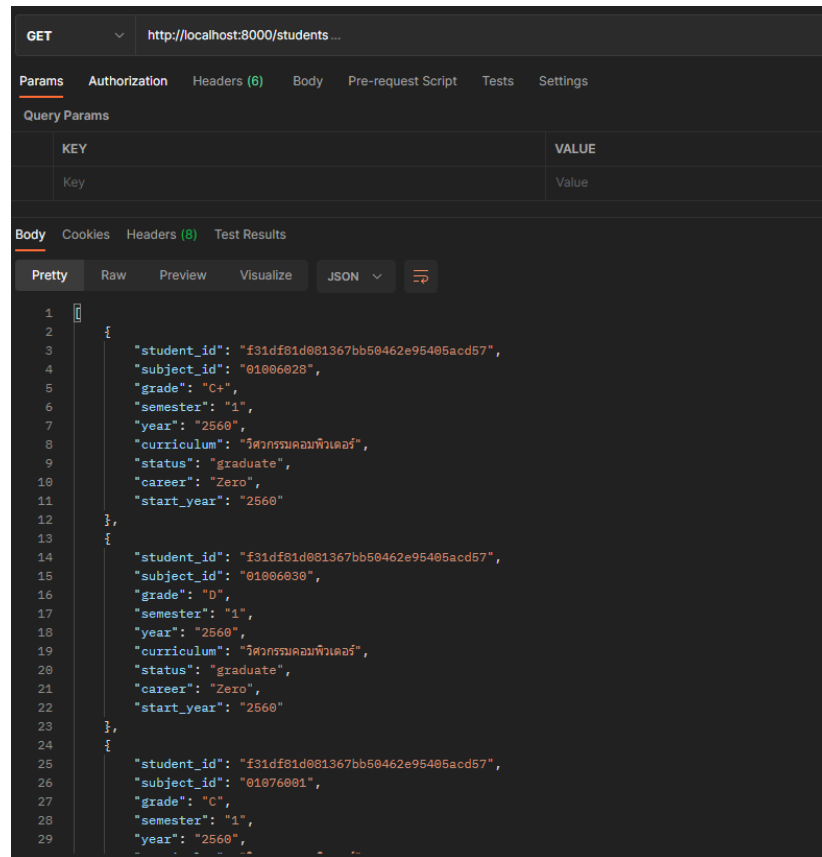
```

## รูป 4.6 ผลลัพธ์การสร้าง Docker Compose บน Server

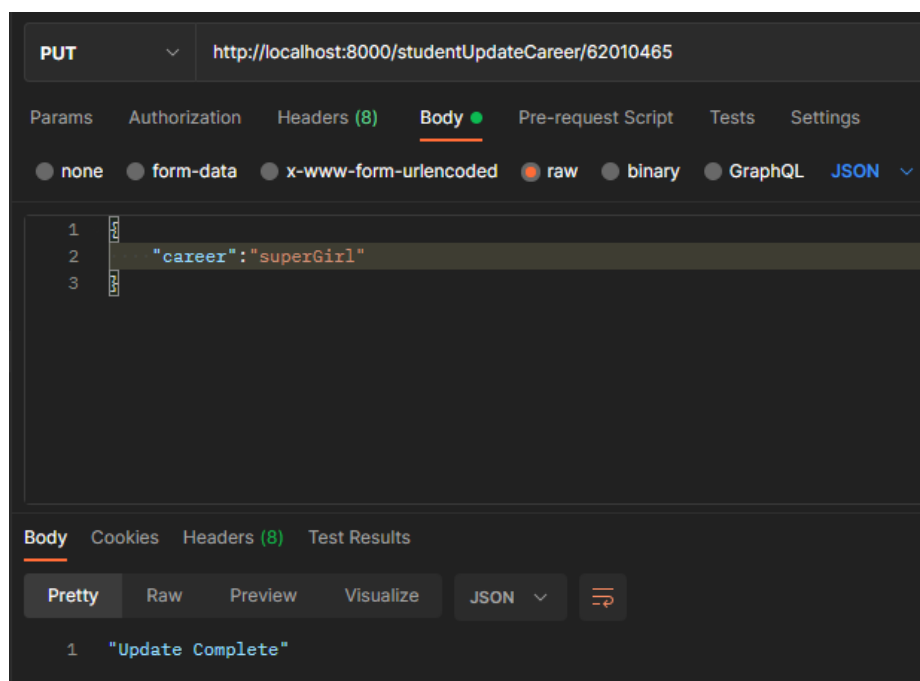
จากผลการทดลองที่ 4.2 พบว่าสามารถนำ Django Application มาบน Server ที่เตรียมไว้ได้ และสามารถสร้าง Docker Compose ได้ โดยปัญหาที่พบคือไม่สามารถทดสอบการ Response ของ API จาก Application ได้ โดยทางผู้จัดทำได้สันนิษฐานสาเหตุอาจเกิดจาก Server ที่ทางผู้จัดทำได้จัดเตรียมอาจไม่ได้เปิด Port การติดต่อที่ทางผู้จัดทำได้จัดสรรไว้ หรือ อาจเกิดจากที่ผู้จัดทำไม่ได้ดำเนินการ Forwarding Port เพื่อให้ Application ติดต่อกับภายนอกได้

## 4.3 การใช้งานฐานข้อมูลร่วมกับ Application

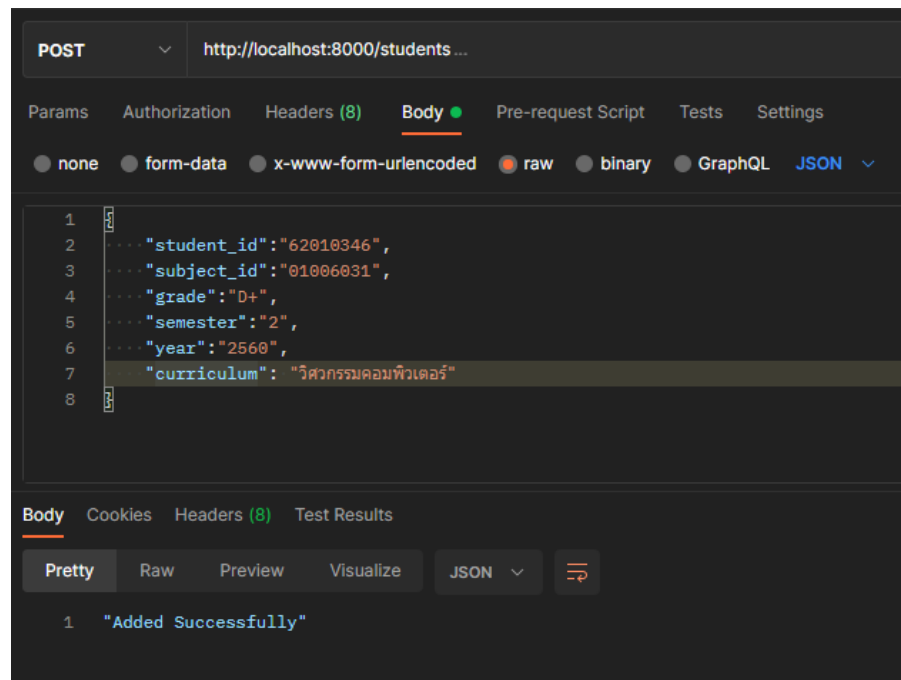
โดยการทดลองการใช้งานร่วมกันของฐานข้อมูลกับ Application นั้นทางผู้จัดทำได้ทำการทดลองการใช้งานผ่านทางเครื่องคอมพิวเตอร์ส่วนบุคคลของผู้จัดทำเองในรูปแบบ Local Host ซึ่งการทำงานร่วมกันของฐานข้อมูลกับ Application นั้นทางผู้จัดทำได้ทดลองเป็นการทำงานในรูปแบบการใช้ API ในการติดต่อระหว่างฐานข้อมูลกับ Application ทั้งกระบวนการค้นหาข้อมูล ดังรูปที่ 4.7, เพิ่มข้อมูลดังรูปที่ 4.8, แก้ไขข้อมูลดังรูปที่ 4.9 และ การลบข้อมูลดังรูปที่ 4.10



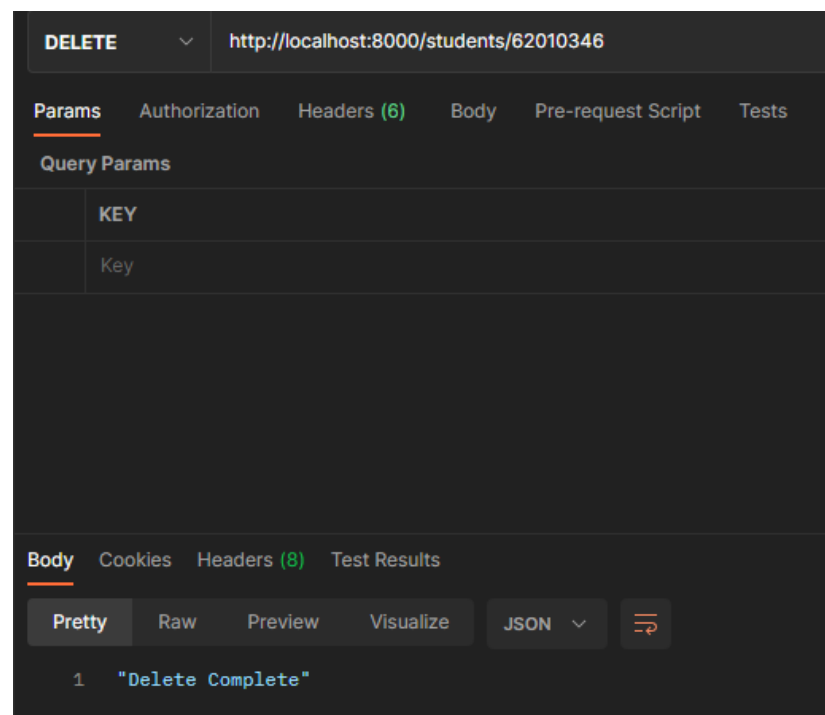
รูป 4.7 ผลลัพธ์การค้นหาข้อมูลผ่าน API



รูป 4.8 ผลลัพธ์การแก้ไขข้อมูลผ่าน API



รูป 4.9 ผลลัพธ์การเพิ่มข้อมูลผ่าน API



รูป 4.10 ผลลัพธ์การลบข้อมูลผ่าน API

จากการทดลองที่ 4.3 พบว่าการทำงานร่วมกันของ Application ในส่วนของ Backend และฐานข้อมูลเป็นไปได้ด้วยดี

#### 4.4 การพัฒนา Similarity Model สำหรับการเติมเกรดที่หายไปของข้อมูลบัณฑิต

โดยการทดลองการพัฒนา Similarity Model นี้ทางผู้จัดทำได้ทำการทดลองใน Google Colab โดยทางผู้จัดทำได้เลือกใช้อัลกอริทึม SVD ของ Surprise จาก Library ของ Scikit-Learn โดยการทดลองนี้ผู้จัดทำได้นำชุดข้อมูลผลลัพธ์การเรียนของบัณฑิตปีการศึกษา 2560 และ 2561 มาทดลองโดยผลลัพธ์ที่พึงประสงค์ของ Similarity Model ที่ผู้จัดทำคาดหวังคือ สามารถเติมชุดข้อมูลผลลัพธ์การเรียนของบัณฑิตในวิชาที่บัณฑิตผู้นั้นไม่ได้ลงทะเบียนไว้จากหลักการของ อัลกอริทึม SVD โดยผู้จัดทำได้กำหนด Parameter ว่า Ratings คือ Grade ของบัณฑิต UserID คือ รหัสนักศึกษาของบัณฑิต และ ProductID คือรหัสวิชา โดยชุดข้อมูลที่ได้นั้นประกอบไปด้วย ข้อมูลของบัณฑิตหลายหลักสูตรด้วยกันโดยทางผู้จัดทำได้เลือกใช้แค่เพียงหลักสูตร วิศวกรรม คอมพิวเตอร์ และ วิศวกรรมคอมพิวเตอร์(ต่อเนื่อง) ดังรูปที่ 4.11 โดยจะแบ่งส่วนของการ ทดลองออกเป็น 2 ส่วนคือ การเตรียมข้อมูลการทดลอง และ การ Train Model

	A	B	C	D	E	F	G
1	student_id	subject_id	grade	semester	year	curriculum	
2	f31df81d08	1006028	C+	1	2560	วิศวกรรมคอมพิวเตอร์	
3	f31df81d08	1006030	D	1	2560	วิศวกรรมคอมพิวเตอร์	
4	f31df81d08	1076001	C	1	2560	วิศวกรรมคอมพิวเตอร์	
5	f31df81d08	1076002	B+	1	2560	วิศวกรรมคอมพิวเตอร์	
6	f31df81d08	90201001	C+	1	2560	วิศวกรรมคอมพิวเตอร์	
7	f31df81d08	90306003	B+	1	2560	วิศวกรรมคอมพิวเตอร์	
8	f31df81d08	1006031	F	2	2560	วิศวกรรมคอมพิวเตอร์	
9	f31df81d08	1076003	D	2	2560	วิศวกรรมคอมพิวเตอร์	
10	f31df81d08	1076004	D+	2	2560	วิศวกรรมคอมพิวเตอร์	
11	f31df81d08	1076012	D	2	2560	วิศวกรรมคอมพิวเตอร์	
12	f31df81d08	90201002	C	2	2560	วิศวกรรมคอมพิวเตอร์	
13	f31df81d08	90402008	C+	2	2560	วิศวกรรมคอมพิวเตอร์	
14	f31df81d08	1006031	D+	3	2560	วิศวกรรมคอมพิวเตอร์	
15	f31df81d08	3456227	A	3	2560	วิศวกรรมคอมพิวเตอร์	
16	f31df81d08	1076006	D+	1	2561	วิศวกรรมคอมพิวเตอร์	
17	f31df81d08	90108007	B	1	2561	วิศวกรรมคอมพิวเตอร์	
18	f31df81d08	90201026	B	1	2561	วิศวกรรมคอมพิวเตอร์	
19	f31df81d08	1076005	D+	2	2561	วิศวกรรมคอมพิวเตอร์	
20	f31df81d08	90106002	B	2	2561	วิศวกรรมคอมพิวเตอร์	
21	f31df81d08	90306008	F	2	2561	วิศวกรรมคอมพิวเตอร์	
22	f31df81d08	90401013	B+	2	2561	วิศวกรรมคอมพิวเตอร์	
23	f31df81d08	3456216	A	3	2561	วิศวกรรมคอมพิวเตอร์	
24	f31df81d08	90201016	B	3	2561	วิศวกรรมคอมพิวเตอร์	
25	f31df81d08	1076008	C+	1	2562	วิศวกรรมคอมพิวเตอร์	
26	f31df81d08	1076032	F	1	2562	วิศวกรรมคอมพิวเตอร์	
27	f31df81d08	1076263	C+	1	2562	วิศวกรรมคอมพิวเตอร์	
28	f31df81d08	1076013	C	2	2562	วิศวกรรมคอมพิวเตอร์	
29	f31df81d08	1076014	B+	2	2562	วิศวกรรมคอมพิวเตอร์	
30	f31df81d08	1076023	A	2	2562	วิศวกรรมคอมพิวเตอร์	
31	f31df81d08	1076423	B+	2	2562	วิศวกรรมคอมพิวเตอร์	

รูป 4.11 ชุดข้อมูลในการทดลอง



#### 4.4.1 การเตรียมข้อมูลการทดลอง

โดยในทุกรูปแบบการทดลองทางผู้จัดทำจะ ทำการจัดการเตรียมการชุดข้อมูลในรูปแบบเดียวกัน โดยทางผู้จัดทำจะ ทำการนำข้อมูลชุดที่ไม่ใช้งานออกในที่นี้คือชุดข้อมูลของบัณฑิตที่ไม่ได้อยู่ในหลักสูตรวิศวกรรมคอมพิวเตอร์ และ วิศวกรรมคอมพิวเตอร์(ต่อเนื่อง) และได้ทำการเปลี่ยนรูปแบบของข้อมูล Column Grade จากรูปแบบของตัวอักษรให้อยู่ในรูปแบบของตัวเลข และแยกส่วนของชุดข้อมูลออกตามหลักสูตร วิศวกรรมคอมพิวเตอร์ และ วิศวกรรมคอมพิวเตอร์(ต่อเนื่อง) เพื่อทำการสร้าง Model ให้ทั้งสองหลักสูตรโดยจะทำการจัดรูปแบบของ Column ของข้อมูลตามนี้ student\_id, subject\_id, grade, semesters, year, curriculum และ job ดังรูปที่ 4.12

```
[ "วิศวกรรมคอมพิวเตอร์": [
  student_id subject_id grade semester year \
0 f31df81d081367bb50462e95405acd57 1006028 2.5 1 2560
1 f31df81d081367bb50462e95405acd57 1006030 1 1 2560
2 f31df81d081367bb50462e95405acd57 1076001 2 1 2560
3 f31df81d081367bb50462e95405acd57 1076002 3.5 1 2560
4 f31df81d081367bb50462e95405acd57 1006031 0 2 2560
...
6850 9591ba44152cb89516c3c373d3fc066f 1076140 None 1 2564
6851 60ca1ecc1b410f648f31c4c8c0c1326d 1076101 None 1 2564
6852 60ca1ecc1b410f648f31c4c8c0c1326d 1076103 None 1 2564
6853 60ca1ecc1b410f648f31c4c8c0c1326d 1076104 None 1 2564
6854 60ca1ecc1b410f648f31c4c8c0c1326d 1076140 None 1 2564

curriculum job
0 วิศวกรรมคอมพิวเตอร์ None
1 วิศวกรรมคอมพิวเตอร์ None
2 วิศวกรรมคอมพิวเตอร์ None
3 วิศวกรรมคอมพิวเตอร์ None
4 วิศวกรรมคอมพิวเตอร์ None
...
6850 วิศวกรรมคอมพิวเตอร์ None
6851 วิศวกรรมคอมพิวเตอร์ None
6852 วิศวกรรมคอมพิวเตอร์ None
6853 วิศวกรรมคอมพิวเตอร์ None
6854 วิศวกรรมคอมพิวเตอร์ None

[6855 rows x 7 columns]],
"วิศวกรรมคอมพิวเตอร์ (ต่อเนื่อง)": [
  student_id subject_id grade semester year \
0 10eb70b54b130d82f2c70add2d31091b 1006031 1 1 2560
1 10eb70b54b130d82f2c70add2d31091b 1076004 2 1 2560
2 10eb70b54b130d82f2c70add2d31091b 1076006 1 1 2560
3 10eb70b54b130d82f2c70add2d31091b 1076012 1.5 1 2560
4 10eb70b54b130d82f2c70add2d31091b 1076009 1.5 2 2560
...
1320 8b6c9ad43a4fd390abdebfb034b4b330 1076031 None 1 2564
1321 8b6c9ad43a4fd390abdebfb034b4b330 1076103 None 1 2564
1322 8b6c9ad43a4fd390abdebfb034b4b330 1076104 None 1 2564
1323 8b6c9ad43a4fd390abdebfb034b4b330 1076112 None 1 2564
1324 8b6c9ad43a4fd390abdebfb034b4b330 1076118 None 1 2564

curriculum job
0 วิศวกรรมคอมพิวเตอร์ (ต่อเนื่อง) None
1 วิศวกรรมคอมพิวเตอร์ (ต่อเนื่อง) None
2 วิศวกรรมคอมพิวเตอร์ (ต่อเนื่อง) None
3 วิศวกรรมคอมพิวเตอร์ (ต่อเนื่อง) None
4 วิศวกรรมคอมพิวเตอร์ (ต่อเนื่อง) None
...
1320 วิศวกรรมคอมพิวเตอร์ (ต่อเนื่อง) None
```

รูป 4.12 ชุดข้อมูลหลังจากผ่านการเตรียมข้อมูล

#### 4.4.2 การ Train Model

หลังจากที่ผู้จัดทำได้ทำการเตรียมชุดข้อมูลเรียบร้อยแล้วจะนำชุดข้อมูลที่ได้มาทำการ Tuning Hyperparameter ด้วย Grid Search โดยช่วงของ Parameter ที่ได้กำหนดสำหรับ อัลกอริทึม SVD มีดังนี้ n\_factors = [20, 50, 100] และ n\_epochs = [5, 10, 20] และได้ทำการแบ่ง

ชุดข้อมูลในการทดสอบออกเป็น 80 (สำหรับ Train) ต่อ 20 (สำหรับทดสอบ) ส่วน และทำการวัดประสิทธิภาพโดยใช้ค่า RMSE

จากผลลัพธ์การทดลองที่ 4.4 พบว่าชุดข้อมูลในส่วนของหลักสูตร วิศวกรรม คอมพิวเตอร์ มีค่า RMSE อยู่ที่ 0.6319 และ ใน ส่วนของ หลักสูตร วิศวกรรม คอมพิวเตอร์(ต่อเนื่อง) คอมพิวเตอร์ มีค่า RMSE อยู่ที่ 0.7871 ซึ่งเมื่อวัดจากช่วงของ Rating หรือ Grade ของชุดข้อมูลที่อยู่ตั้งแต่ 0 ถึง 4

ซึ่งยังไม่เป็นที่น่าพอใจทางผู้จัดทำเลยทำการสันนิษฐานว่าการที่จำนวนของวิชาที่มี จำนวนที่มากนั้นอาจจะส่งผลต่อการคำนวณของอัลกอริทึม จึงแก้ปัญหาโดยการจัดกลุ่มวิชา

#### 4.5 การพัฒนา NLP สำหรับการจับกลุ่มวิชา

เนื่องจากการทำนายผลเกรดของนักศึกษานั้น ต้องประกอบไปด้วยข้อมูล subject\_id หรือ ข้อมูลรหัสของรายวิชานั้นๆ ซึ่งหลักสูตรรายวิชามีการถูกปรับทุกๆ 4 ปี จึงทำให้เกิดปัญหาในการ Train Model เพราะมีข้อมูลรหัสรายวิชาที่ไม่เหมือนกัน แต่ว่าตัววิชานั้นมีความคล้ายคลึงกับตัววิชาเดิม ทางผู้จัดทำจึงนำบทคัดย่อของรายวิชามาจัดกลุ่มวิชา โดยการหาค่า Similarity ผ่านกระบวนการ NLP ความคล้ายคลึงของบทคัดย่อแต่ละวิชา โดยมีตัวอย่างของข้อมูลดังรูปที่ 4.13

	A	B	C	D	E	F
1	subject_id	subject_name_thai	subject_name_eng	abstract	subject_key	year
2	1006030	แคลคูลัส 1	CALCULUS 1	Function, Limit, Continuity and their applications, Mathematical induction, Introduction to derivative, Differ	2560	
3	1006031	แคลคูลัส 2	CALCULUS 2	Functions of several variables and theirs applications, Vector algebra in three dimensions, Polar coordinate	2560	
4	1006032	สมการเชิงอนุพันธ์และพีชคณิตเชิงเส้นขั้นพื้นฐาน	ELEMENTARY DIFFERENTIAL EQUATIONS AND LINEAR ALGEBRA	Systems of linear equations and solutions. vector and space; Matrices, Solution of linear equations by ma	2560	
5	1006004	การฝึกงานอุตสาหกรรม	INDUSTRIAL TRAINING	During their four-year selected studies, students are required to complete a short-term industrial placeme	2560	
6	1006028	เตรียมความพร้อมสำหรับวิศวกร	PRE-ENGINEER ACTIVITIES	Participates in activities organized by the Faculty of Engineering of advising and preparing students for suc	2560	
7	1076003	วงจรไฟฟ้าและอิเล็กทรอนิกส์	CIRCUITS AND ELECTRONICS	Fundamentals electric circuit. Ohm's law, Kirchhoff's law, Thevenin's and Norton's theorems, superpositor	2560	
8	1076253	ความน่าจะเป็นและสถิติ	PROBABILITY AND STATISTICS	Introduction to probability and statistics; probability; random variables; discrete probability distributions; coi	2560	

รูป 4.13 ตัวอย่างข้อมูล Data Subject

ขั้นตอนแรกทำการนำเข้าสู่ข้อมูลจาก Table subject\_data สำหรับการหาค่า Similarity ใน Table subject\_data เป็นการรวมข้อมูลรายวิชาของแต่ละหลักสูตร จึงทำให้มีรหัสวิชาบาง รายวิชาซ้ำ ใช้วิธีการแก้ไขโดยการ Drop Record ที่ซ้ำออก แล้วเก็บ Record แรกเอาไว้ ผลลัพธ์ ดังรูปที่ 4.14

index	subject_id	subject_name_thai	subject_name_eng	abstract	subject_key	year
0	0	แคลคูลัส 1	CALCULUS 1	Function, Limit, Continuity and their applicat...	NaN	2560
1	1	แคลคูลัส 2	CALCULUS 2	Functions of several variables and theirs appl...	NaN	2560
2	2	สมการอนุพันธ์และพีชคณิตเชิงเส้นพื้นฐาน	ELEMENTARY DIFFERENTIAL EQUATIONS AND LINEAR A...	Systems of linear equations and solutions. vec...	NaN	2560
3	3	การฝึกงานอุตสาหกรรม	INDUSTRIAL TRAINING	During their four-year selected studies, stude...	NaN	2560
4	4	เตรียมความพร้อมสำหรับวิศวกร	PRE-ENGINEER ACTIVITIES	Participates in activities organized by the Fa...	NaN	2560
...	...	...	...	...	...	...
110	145	เครือข่ายสวิตซ์ไอพี	IP SWITCHED NETWORKS	The main objective of this course is to provid...	โครงสร้างพื้นฐานของระบบเน็ตเวิร์ค	2564
111	147	ผู้ประกอบการกับวิศวกร	ENTREPRENEURSHIP AND THE ENGINEER	Discusses basic concepts of marketing, busines...	วิชาบูรณาการ	2564
112	148	การประกอบการและการจัดการด้านเทคโนโลยีสารสนเทศ	IT ENTREPRENEURSHIP AND MANAGEMENT	This course covers the basic of IT entrepreneu...	วิชาบูรณาการ	2564
113	149	การวางแผนเชิงกลยุทธ์โดยใช้บอร์ดและการ์ดเกม	STRATEGIC PLANNING USING BOARD AND CARD GAME	This course covers the strategic planning, stu...	วิชาบูรณาการ	2564
114	150	การจัดการโครงการเทคโนโลยีสารสนเทศ	IT PROJECT MANAGEMENT	Meaning of Project; Importance of Information ...	วิชาบูรณาการ	2564

115 rows x 7 columns

รูป 4.14 ข้อมูล Data Subject หลังจากคัดรหัสวิชาที่ซ้ำออก

หลังจากนั้นตัวบทคัดย่อที่จะนำไปหาค่า Similarity นั้น มีค่าเชื่อมต่างๆ ไว้ ซึ่งส่งผลต่อการนำไปหาค่า NLP จึงนำค่าเชื่อมเหล่านั้นออกจากตัวบทคัดย่อของแต่ละหลายวิชา เพื่อให้เหลือแต่คำที่เฉพาะเจาะจงมากขึ้น ผลลัพธ์ดังรูปที่ 4.15

index	subject_id	subject_name_thai	subject_name_eng	abstract	subject_key	year
0	0	แคลคูลัส 1	CALCULUS 1	Mathematical Limit derivative functions series...	NaN	2560
1	1	แคลคูลัส 2	CALCULUS 2	line - valued functions theory etc. planes coo...	NaN	2560
2	2	สมการอนุพันธ์และพีชคณิตเชิงเส้นพื้นฐาน	ELEMENTARY DIFFERENTIAL EQUATIONS AND LINEAR A...	linear z-transformation initial Solution serie...	NaN	2560
3	3	การฝึกงานอุตสาหกรรม	INDUSTRIAL TRAINING	takes short-term environments. put This reflec...	NaN	2560
4	4	เตรียมความพร้อมสำหรับวิศวกร	PRE-ENGINEER ACTIVITIES	technology preparing Laboratory project educat...	NaN	2560
...	...	...	...	...	...	...
110	145	เครือข่ายสวิตซ์ไอพี	IP SWITCHED NETWORKS	Spanning 3 teaches This security building also...	โครงสร้างพื้นฐานของระบบเน็ตเวิร์ค	2564
111	147	ผู้ประกอบการกับวิศวกร	ENTREPRENEURSHIP AND THE ENGINEER	products implementation appreciate entrepreneu...	วิชาบูรณาการ	2564
112	148	การประกอบการและการจัดการด้านเทคโนโลยีสารสนเทศ	IT ENTREPRENEURSHIP AND MANAGEMENT	including This investment business Self-learn...	วิชาบูรณาการ	2564
113	149	การวางแผนเชิงกลยุทธ์โดยใช้บอร์ดและการ์ดเกม	STRATEGIC PLANNING USING BOARD AND CARD GAME	application strategic. board game card compute...	วิชาบูรณาการ	2564
114	150	การจัดการโครงการเทคโนโลยีสารสนเทศ	IT PROJECT MANAGEMENT	resource technology integration information pr...	วิชาบูรณาการ	2564

115 rows x 7 columns

รูป 4.15 ข้อมูล Data Subject หลังจากจัดแต่งบทคัดย่อ

เมื่อบทคัดย่อมีความพร้อมต่อการนำเข้าหาค่า NLP แล้ว ต่อมาทำการสร้าง Dict เอาไว้ โดยให้ค่า Key เป็นเลขกลุ่ม โดยเริ่มจาก 0 ค่า Value ทำการเก็บเป็น List ของรหัสวิชาต่างๆที่อยู่ในกลุ่มเดียวกัน ต่อมาได้ทำการวนลูปตามจำนวนของวิชาทั้งหมด และทำการเทียบค่า Similarity ของแต่ละวิชา โดยจะจัดกลุ่มจากค่า Similarity ที่มีค่ามากกว่าเท่ากับ 0.90 ให้อยู่ในกลุ่มเดียวกัน และสร้าง Column เพิ่ม เพื่อเก็บค่าเลขกลุ่มเอาไว้ ตัวอย่างของ code ดังรูปที่ 4.16

```

▶ thisdict = {}
all_docs = [nlp(row) for row in df['abstract']]
num = 0
for i in range(len(all_docs)):
    check = 0
    sims = []
    if list(thisdict.keys()) == []:
        thisdict[num] = [df.loc[df.index == i, 'subject_id'].iloc[0]]
        num += 1
    else:
        for x in thisdict:
            temp = df.loc[df['subject_id'] == thisdict[x][0], 'abstract'].iloc[0]
            sim = all_docs[i].similarity(nlp(temp))
            sims.append(sim)
        maxsim = max(sims)
        for x in thisdict:
            temp = df.loc[df['subject_id'] == thisdict[x][0], 'abstract'].iloc[0]
            sim = all_docs[i].similarity(nlp(temp))
            if sim >= 0.90 and sim == maxsim:
                thisdict[x].append(df.loc[df.index == i, 'subject_id'].iloc[0])
                check = 1
                break
        if check != 1:
            thisdict[num] = [df.loc[df.index == i, 'subject_id'].iloc[0]]
            num += 1

```

รูป 4.16 Coding การเทียบค่า Similarity

การนำชุดข้อมูลวิชาที่จัดกลุ่มไปใช้กับ Similarity Model โดยการเตรียมข้อมูลในการทดลองนั้นจำเป็นต้องนำชุดข้อมูลกลุ่มของวิชามารวมกับชุดข้อมูลของผลลัพธ์การเรียนของบัณฑิต แล้วรวมกลุ่มข้อมูลของผลลัพธ์การเรียนของบัณฑิต

ในวิชาที่อยู่ในกลุ่มเดียวกันโดยรวมข้อมูลผ่านการทำค่าเฉลี่ย แล้วจึงนำข้อมูลไปทดลองกับการ Train แบบจำลอง Similarity Model โดยจะทำการจัดรูปแบบของ Column ของข้อมูลตามนี้ student\_id, grade, semesters, year, curriculum, subjectTypes และ job ผลลัพธ์ดังรูปที่ 4.17

	student_id	grade	semester	year	curriculum	subjectTypes	job
0	0197dc3d32f1d32bfff2a3bff89e69f9	4.0	1	2562	วิศวกรรมคอมพิวเตอร์	4	None
1	0197dc3d32f1d32bfff2a3bff89e69f9	3.0	1	2562	วิศวกรรมคอมพิวเตอร์	อื่นๆ	None
2	025f749d7a5d9b5c3f3d57b68e1de9e9	2.0	2	2562	วิศวกรรมคอมพิวเตอร์	0	None
3	025f749d7a5d9b5c3f3d57b68e1de9e9	1.0	3	2560	วิศวกรรมคอมพิวเตอร์	1	None
4	025f749d7a5d9b5c3f3d57b68e1de9e9	0.0	1	2562	วิศวกรรมคอมพิวเตอร์	10	None
...	...	...	...	...	...	...	...
4026	fe38534f82b88c58b9acc94cd2280246	3.5	2	2561	วิศวกรรมคอมพิวเตอร์	5	None
4027	fe38534f82b88c58b9acc94cd2280246	3.0	2	2562	วิศวกรรมคอมพิวเตอร์	6	None
4028	fe38534f82b88c58b9acc94cd2280246	3.0	1	2561	วิศวกรรมคอมพิวเตอร์	7	None
4029	fe38534f82b88c58b9acc94cd2280246	4.0	1	2561	วิศวกรรมคอมพิวเตอร์	8	None
4030	fe38534f82b88c58b9acc94cd2280246	3.5	1	2563	วิศวกรรมคอมพิวเตอร์	9	None

รูป 4.17 ชุดข้อมูลหลังจากผ่านการเตรียมข้อมูลผลลัพธ์จาก NLP

จากผลลัพธ์การทดลองที่ 4.5 พบว่าชุดข้อมูลในส่วนของหลักสูตร วิศวกรรมคอมพิวเตอร์ มีค่า RMSE อยู่ที่ 0.5751 และ ในส่วนของหลักสูตร วิศวกรรมคอมพิวเตอร์(ต่อเนื่อง) คอมพิวเตอร์ มีค่า RMSE อยู่ที่ 0.7351 ซึ่งเมื่อวัดจากช่วงของ Rating หรือ Grade ของชุดข้อมูลที่อยู่ตั้งแต่ 0 ถึง 4 ดังรูปที่ 4.18

Evaluating RMSE, MAE of algorithm SVD on 10 split(s).												
	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	Mean	Std
RMSE (testset)	0.5513	0.6143	0.6097	0.5788	0.5607	0.6176	0.5590	0.5890	0.5866	0.5837	0.5851	0.0224
MAE (testset)	0.4304	0.4778	0.4643	0.4506	0.4521	0.4797	0.4408	0.4513	0.4684	0.4575	0.4573	0.0148
Fit time	0.29	0.19	0.16	0.10	0.11	0.10	0.11	0.10	0.10	0.13	0.14	0.06
Test time	0.00	0.01	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00
RMSE: 0.5751												
0.5751494925828684												
Evaluating RMSE, MAE of algorithm SVD on 10 split(s).												
	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	Mean	Std
RMSE (testset)	0.6924	0.8274	0.7174	0.8376	0.8569	0.8976	0.8220	0.8543	0.7384	0.6775	0.7922	0.0741
MAE (testset)	0.5158	0.6480	0.5949	0.6754	0.6342	0.7137	0.6487	0.6766	0.5898	0.5484	0.6246	0.0584
Fit time	0.02	0.02	0.02	0.02	0.02	0.02	0.02	0.02	0.02	0.02	0.02	0.00
Test time	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
RMSE: 0.7352												
0.7351639239250899												

รูป 4.18 ผลลัพธ์การทดลองร่วมกับ NLP

จากผลลัพธ์ที่ได้เลยสามารถสรุปได้ว่าการสันนิษฐานของผู้จัดทำในเรื่องของจำนวนวิชาที่มีมากเกินไปทำให้ค่า RMSE เพิ่มขึ้นนั้นถูกต้อง และผลลัพธ์ที่ได้ออกมาเป็นที่น่าสนใจเพียงพอแล้ว

#### 4.6 การพัฒนา Job Classification Model

โดยการทดลองนี้เนื่องจากข้อมูลที่มีมานั้นมีไม่มากพอสำหรับการสร้าง Model ทำให้ทางผู้จัดทำไม่สามารถตัดสินใจเลือกอัลกอริทึมในการนำไปใช้งานจริงได้ โดยมีข้อมูลที่ใช้ดังรูปที่ 4.19

โดยการทดลองนี้ได้อัลกอริทึมในการสร้าง Model ด้วยกันสองอัลกอริทึมคือ KNN Classifier และ Decision Tree Classifier จาก Library Scikit-learn โดยชุดข้อมูลที่ใช้จะมี ชุดข้อมูลที่เป็นผลลัพธ์ที่ได้จาก Similarity Model และชุดข้อมูลแบบสอบถามการมีงานทำของบัณฑิต โดยการวัดประสิทธิภาพของ Model จะใช้ค่า Accuracy ที่ได้จากการคำนวณของ Function Metrics Accuracy จาก Library Scikit-Learn c

	A	B
1	student_id	job
2	6c61b48a3e68d7810fe6c47aa69bf123	Software Tester
3	c7605751af1a39f5d7e35159038b5211	Software Engineer
4	0a079386e7934625831ece1087ae915f	Mobile Developer
5	55222ca81ca73662048c05ba69243685	Mobile Developer
6	a1145995e321897b60617525829e507c	Software Engineer
7	24d411247cb41b2ecea0fe3d3dbc94ab	Software Engineer
8	46a1e8e78f7340b7454d0041238fdf30	Software Engineer
9	a85bfa671ba7196bc95c9aecce7ffe86	Software Engineer
10	2b25ba620f6313a62a0727fdae38d13e	Software Engineer
11	0faeb665c628f8c3a41f327a57a8575c	Software Tester
12	84244e4340b039daa5fa474edfebb110	UX UI Designer
13	90b6b34e9a6d0b846c0b176115e7d5b6	business intelligence
14	44199bcce9cc2e3d6f46c8489d2689f1	Software Engineer
15	bb4e051171993c63c9e0717260052615	Software Engineer
16	e72ad5d4d7441b947de1872ef908826c	Personal bussiness
17	04b011dab88861100de77e5bc03f8dcf	Software Engineer
18	5c0a54e49d6c2a5732f34ed0f319f439	Software Engineer
19	a0e1e9b5b3446e27eec066af0cd36cb0	Software Engineer
20	f2e5835eb5336ebf0a83469f7822928f	Data Analyst
21	cb1d53d24af443c5a7cad5236a45849f	Software Engineer
22	a480f70b9f13a85a152e3308d9b227dd	Software Engineer
23	223b34b45d624c6e2015f08036b92a94	Software Engineer
24	65365996c5164fe1983a09e6fef3687e	UX UI Designer

รูป 4.19 ชุดข้อมูลแบบสอบถามการมีงานทำของบัณฑิต

โดยขั้นตอนต่อมาคือการเตรียมข้อมูล ทางผู้จัดทำได้ทำการนำผลลัพธ์ที่ได้จาก Similarity Model มาเติมเต็มข้อมูลผลลัพธ์การเรียนรู้ที่ขาดหายไปของบัณฑิต แล้วทางผู้จัดทำได้ทำการรวมข้อมูลของสองชุดข้อมูลเข้าด้วยกันโดยใช้ student\_id ในการรวมกันของสองชุดข้อมูล โดยถ้า

ในชุดข้อมูลของผลลัพธ์ที่ได้จาก Similarity Model มีข้อมูลรหัสนักศึกษาที่ชุดข้อมูลแบบสอบถามการมีงานทำของบัณฑิตจะไม่นำมาเป็นข้อมูลในการสร้าง Model หลังจากนั้นจะทำการ Transpose ข้อมูลให้อยู่ในรูปแบบดังรูปที่ 4.20 เพื่อนำไป Train Model

	student_id	0	1	2	3	4	5	6	7	8	...	23	25	31	33	35	42	50	51	อื่นๆ	job
12	0a079386e7934625831ece1087ae915f	2.50	2.00	4.0	4.0	2.5	3.0	3.50	3.5	3.0	...	3.20	3.40	3.06	3.12	3.00	3.37	3.03	3.48	3.50	Mobile Developer
19	0f6119a28800d31fb88ceddb554573	2.50	4.00	4.0	4.0	2.0	3.0	2.00	2.5	1.0	...	3.18	3.50	3.18	3.12	3.50	3.40	3.13	3.51	3.73	Data Engineer
20	0faeb665c628f8c3a41f327a57a8575c	2.50	3.00	4.0	4.0	3.0	2.0	3.25	3.5	3.5	...	3.31	3.51	3.21	3.28	3.36	3.60	3.22	3.65	4.00	Software Tester
24	15ee5c71ce09039ae9c104849bd3a47	2.00	3.00	4.0	4.0	2.5	3.5	2.50	3.5	3.0	...	3.19	3.52	3.18	3.12	3.50	3.40	3.22	3.60	4.00	Software Engineer
38	1e288fce0b9e3c2028f79cb33365db1	1.50	2.00	4.0	4.0	2.5	3.0	2.50	3.5	2.5	...	3.25	3.42	3.07	3.15	3.50	3.39	3.08	3.43	3.75	IT Infrastructure
55	2b25ba620f6313a62a0727fdae38d13e	2.50	1.50	4.0	4.0	2.5	2.5	2.75	4.0	3.0	...	3.22	3.40	3.09	3.07	3.00	3.40	3.15	3.57	3.70	Software Engineer
72	3ac00e4c4710ac58f180405f45da8205	2.50	3.00	4.0	4.0	2.5	3.5	3.25	3.5	3.0	...	3.40	3.57	3.30	3.26	4.00	3.54	3.27	3.75	3.84	Software QA
75	3b387161d63eca3a4f643b339c2072a0	3.00	3.50	4.0	4.0	2.5	3.5	4.00	4.0	3.5	...	3.55	3.64	3.43	3.42	3.62	3.60	3.43	3.75	4.00	Software Engineer
83	44ae02b1e659304d4c84bbbe9a0d2ed5	3.00	3.00	4.0	4.0	2.5	2.5	3.50	3.5	3.0	...	3.33	3.46	3.19	3.24	3.50	3.43	3.24	3.54	3.00	Software Engineer
95	52a5ed08a47d50ed4a4cad3b7ac6d312	3.00	3.50	4.0	4.0	4.0	4.0	3.75	4.0	3.0	...	3.62	3.86	3.63	3.55	3.73	3.92	3.00	4.00	4.00	Mobile Developer
117	63ff5b2e627f0162eb706e16098a6001	3.25	3.25	4.0	4.0	3.0	2.5	2.00	3.5	2.0	...	3.50	3.13	2.93	2.94	2.96	3.09	2.97	3.50	3.51	Software Tester
121	65365996c5164fe1983a09e6f3687e	2.00	2.00	4.0	4.0	2.0	3.5	1.50	3.0	2.5	...	3.11	3.41	3.17	2.96	3.50	3.23	3.08	3.49	3.75	UX UI Designer
136	732d85012d4d210e0b85b353c494326d	2.00	1.50	4.0	4.0	2.5	3.5	2.75	3.5	2.5	...	3.30	3.49	3.21	3.11	3.36	3.48	3.50	3.59	4.00	UX UI Designer
140	78fee460c034f47f85e9e1a827e4fca	3.50	4.00	4.0	4.0	3.5	2.5	4.00	4.0	4.0	...	3.76	4.00	3.65	3.72	3.50	4.00	3.82	4.00	4.00	Software Engineer
143	7d52033d2d37502c500d662a47a1644	2.00	3.50	4.0	4.0	3.5	3.0	3.75	4.0	3.5	...	3.61	3.83	3.53	3.50	3.00	3.83	3.64	3.98	4.00	Software Engineer
154	839f9b47419d9b8f442514d1da429e34	1.50	3.00	4.0	4.0	1.5	4.0	3.00	3.5	2.5	...	3.35	3.52	3.22	3.16	3.32	3.41	3.23	3.58	3.78	Software Engineer
190	a00b4a2d58030dede1e020f47521ae6	2.00	3.50	4.0	4.0	2.5	2.5	2.50	3.5	2.5	...	3.09	3.26	3.10	2.92	3.11	3.14	3.02	3.42	4.00	UX UI Designer
192	a208cc29835c203e85bd2e42897b654f	3.50	3.00	4.0	4.0	2.5	3.0	3.25	3.0	3.0	...	3.26	3.50	3.20	3.31	3.00	3.52	3.35	3.78	3.50	Data Engineer
227	bb4e051171993c63c9e0717260052615	3.00	2.50	4.0	4.0	2.5	2.5	3.00	3.5	1.5	...	3.19	3.29	3.06	2.96	3.11	3.23	2.95	3.37	3.58	Software Engineer
236	c7605751af1a39f5d7e35159038b5211	2.00	2.00	4.0	4.0	4.0	2.5	3.25	4.0	4.0	...	3.67	3.96	3.56	3.43	3.74	4.00	3.53	3.91	4.00	Software Engineer
238	c8a67d514a8e641f8abceba7c0db1cb5	2.50	2.00	4.0	4.0	2.5	3.5	3.50	3.5	2.0	...	3.36	3.62	3.32	3.22	3.37	3.56	3.00	3.77	4.00	Study on Master's degree
260	d71edbe586f9b607c1a18b04e60310f	3.00	2.50	4.0	4.0	2.0	2.5	2.00	3.5	2.0	...	3.38	3.49	3.15	3.15	3.50	3.46	3.24	3.57	3.50	Software Engineer
288	f1897be137c1342811f9007fbf100416	2.00	2.00	4.0	4.0	1.5	3.5	3.00	3.5	3.5	...	3.30	3.57	3.25	3.33	3.50	3.47	3.21	3.65	3.75	Mobile Developer

รูป 4.20 ชุดข้อมูลผ่านการ Transpose

จากผลลัพธ์การทดลองที่ 4.6 พบว่าค่า Accuracy ของอัลกอริทึม KNN Classifier อยู่ที่ 0.4 และ Accuracy ของอัลกอริทึม Decision Tree Classifier อยู่ที่ 0.2 จากคะแนนเต็ม 1 ซึ่งการทางผู้จัดทำได้สันนิษฐานว่าข้อผิดพลาดที่เกิดขึ้นมาจากจำของชุดข้อมูลแบบสอบถามการมีงานทำของบัณฑิตที่มีน้อยเกินไป

## 4.7 การพัฒนาหน้าตาผู้ใช้งานของ Application

โดยจะแบ่งส่วนของการทดลองออกเป็น 3 ส่วนคือ การทดลองสร้าง React Project โดย Docker Compose, การทดลองการทำงานร่วมกันของ React Project และ Django Application และ การนำ Docker Compose ของ React Project ขึ้นไปอยู่บน Server

### 4.7.1 การทดลองสร้าง React Project โดย Docker Compose

โดยการทดลองนี้ทางผู้จัดทำได้ทำการเตรียม File สำหรับการสร้างโดยมี Dockerfile ที่ระบุการสร้าง Directory ของ React Project และ Version ของ NodeJS ดังรูปที่ 4.21 ต่อมาคือ File docker-compose.yml ดังรูปที่ 4.22



```

You, 3 months ago | 1 author (You)
1 # base image
2 FROM node
3
4 # set working directory
5 RUN mkdir /usr/src/app
6 WORKDIR /usr/src/app
7
8 # add `/usr/src/app/node_modules/.bin` to $PATH
9 ENV PATH /usr/src/app/node_modules/.bin:$PATH
10
11 # install and cache app dependencies
12 COPY package.json /usr/src/app/package.json
13 RUN npm install
14 RUN npm install react-scripts -g
15 EXPOSE 3000
16 # start app
17 CMD ["npm", "start"]

```

รูป 4.21 Dockerfile React Project

```

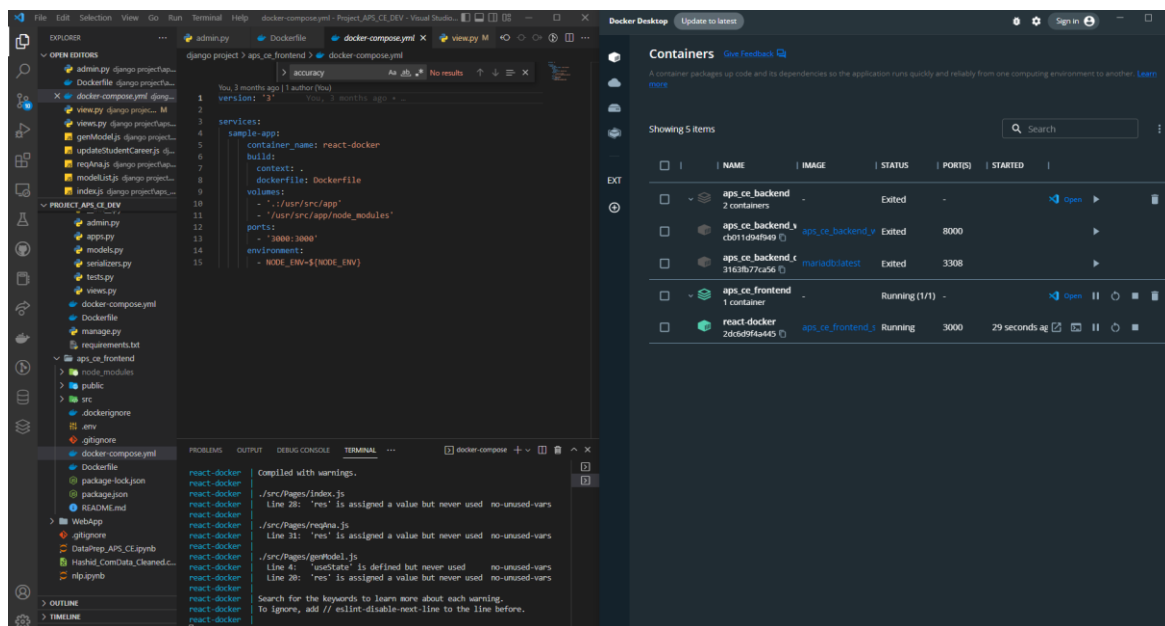
You, 3 months ago | 1 author (You)
version: '3'
services:
  sample-app:
    container_name: react-docker
    build:
      context: .
      dockerfile: Dockerfile
    volumes:
      - './usr/src/app'
      - '/usr/src/app/node_modules'
    ports:
      - '3000:3000'
    environment:
      - NODE_ENV=${NODE_ENV}

```

รูป 4.22 docker-compose.yml React Project

ซึ่งเมื่อหลังจากการสร้าง docker compose ขึ้นมาทางผู้จัดทำจะตรวจสอบการทำงานของระบบทั้งหมดผ่าน Application Docker Desktop ร่วมกับ log terminal ของ Visual Studio Code ดังรูปที่ 4.23 โดยได้กำหนด Port ในการเข้าถึงทั้งภายนอกและภายในเป็น 3000



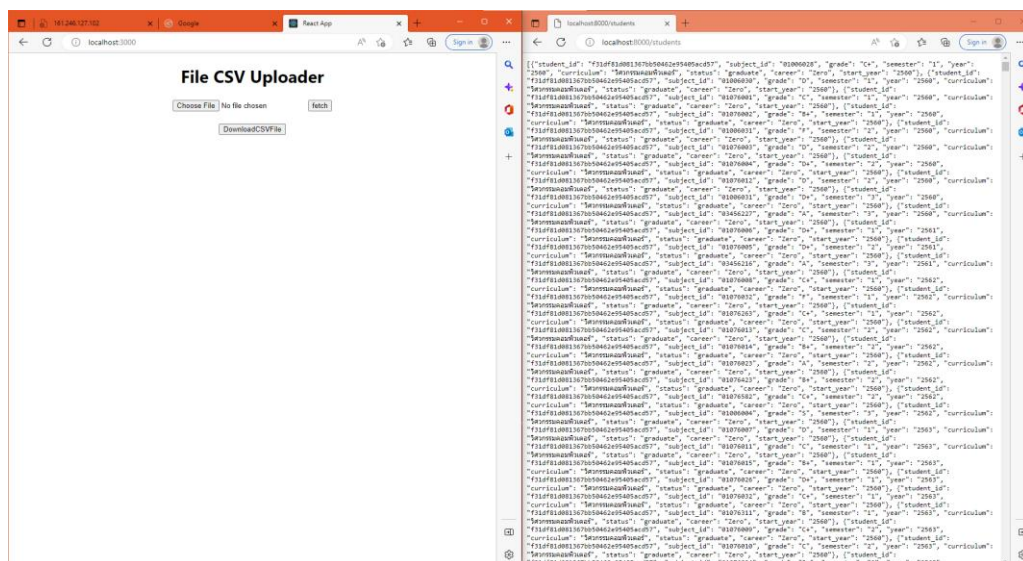


รูป 4.23 ภาพรวมของการตรวจสอบการทำงานของ React Project

จากผลการทดลองที่ 4.7.1 พบว่าสามารถสร้าง Docker Compose ของ React Project ได้ และสามารถตรวจสอบการทำงานต่างๆ ของ React Project ผ่าน Visual Studio Code Terminal และ Application Docker Desktop ได้

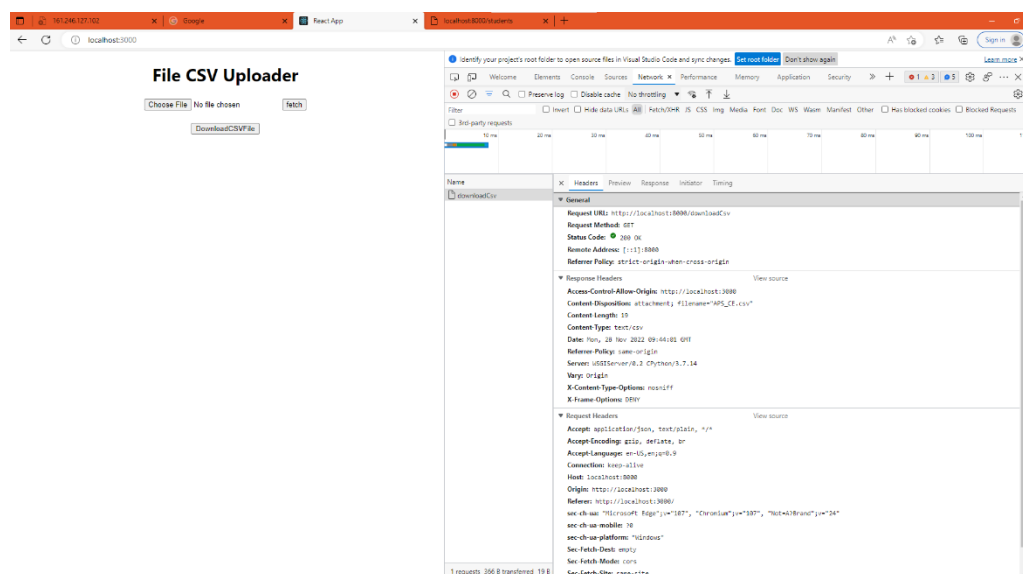
#### 4.7.2 การทดลองการทำงานร่วมกันของ React Project และ Django Application

โดยในการทดลองนี้ทางผู้จัดทำได้กำหนดสภาพแวดล้อมของการทดลองโดยให้ทั้งสองส่วน (React Project และ Django Application) ทำงานอยู่บนเครื่องคอมพิวเตอร์ส่วนบุคคลในรูปแบบ Local Host ดังรูปที่ 4.24 โดยด้านซ้ายคือหน้าหลักของ React Project และด้านขวาคือหน้าของ Django Application



รูป 4.24 ภาพรวมของ React Project และ Django Application ที่เรียกดูผ่าน Web Browser

จากผลลัพธ์การทดลองที่ 4.7.2 พบว่าทั้งสองสามารถติดต่อส่งข้อมูลให้กันและกัน  
ได้ราบรื่น โดย React Project จะติดต่อกับ Django Application ผ่าน Port 8000 และ Django  
Application จะติดต่อกับ React Project ผ่าน Port 3000 ผลลัพธ์ดังรูปที่ 4.25



รูป 4.25 ทดสอบการติดต่อระหว่าง React Project และ Django Application

#### 4.7.3 การนำ Docker Compose ของ React Project ขึ้นไปอยู่บน Server

โดยในขั้นตอนนี้นักผู้จัดทำได้นำ Directory ของ React Project ไปยัง Server ที่ได้จัดเตรียมไว้ โดยจะเป็น Server เดียวกันกับในหัวข้อการทดลอง 4.2 ผลลัพธ์ดังรูปที่ 4.26

```
[analytic@itanalytic1 ~]$ cd django/
[analytic@itanalytic1 django]$ cd aps_ce_frontend/
[analytic@itanalytic1 aps_ce_frontend]$ ls
Dockerfile package.json package-lock.json public README.md src
[analytic@itanalytic1 aps_ce_frontend]$ nano docker-compose.yml
[analytic@itanalytic1 aps_ce_frontend]$
[analytic@itanalytic1 aps_ce_frontend]$ sudo nano docker-compose.yml
[analytic@itanalytic1 aps_ce_frontend]$ [analytic@itanalytic1 aps_ce_frontend]$
[analytic@itanalytic1 aps_ce_frontend]$ ls
docker-compose.yml Dockerfile package.json package-lock.json public README.md src
```

รูป 4.26 React Project ที่นำขึ้นไปยัง Server ที่เตรียมไว้

โดยต่อมาได้ทำการทดลองโดยการสร้าง Docker Compose ของ React Project ขึ้นมา โดยจากผลลัพธ์การทดลองที่ 4.7.3 พบว่าสามารถสร้าง Docker Compose ของ React Project ได้สำเร็จและตัว React Project สามารถดำเนินการได้อย่างราบรื่น

จากผลลัพธ์การทดลองที่ 4.7 พบว่าสามารถสร้าง React Project จาก Docker Compose ได้และติดต่อกับ Django Application ได้อย่างราบรื่น และสามารถนำ React Project Directory ขึ้นไปสร้าง Docker Compose บน Server ได้และตัว React Project เองสามารถทำงานได้เป็นปกติ โดยปัญหาที่พบคือ ไม่สามารถติดต่อกับ React Project ที่ทำงานอยู่บน Server ณ Port 3000 ได้ซึ่งผู้จัดทำได้สันนิษฐานว่าปัญหานี้จะเป็นปัญหาแบบเดียวกันกับการทดลองที่ 4.2

#### 4.8 การนำกระบวนการสร้าง Model และการเรียกใช้ลงบน Django Application

โดยในการทดลองนี้นักผู้จัดทำจะนำกระบวนการสร้างและเรียกใช้ที่ได้ทำการทดลองไปในการทดลองที่ 4.4 และ 4.5 โดยจะไม่นำการทดลองที่ 4.6 มาเนื่องจากผลลัพธ์การทดลองยังไม่เป็นที่น่าพอใจเพียงพอ ซึ่งจะนำกระบวนการสร้างและเรียกใช้ Similarity Model และ NLP ให้อยู่ในรูปแบบของ API และ Call Function โดยกระบวนการทำงาน และการเรียกใช้งานของ Similarity Model และ NLP มีดังนี้

##### 4.8.1 กระบวนการ การทำงานของ NLP

- 1) ผู้ใช้งานนำ File CSV ของรายวิชาที่ต้องการเพิ่มในระบบ Upload เข้าไป
- 2) ได้ผลลัพธ์ออกมาเป็น List ของ Dictionary ที่เก็บ Class และวิชาใน Class
- 3) นำ List ที่ได้มารวมกับตารางจาก File CSV ที่ผู้ใช้งาน Upload
- 4) รูปให้ข้อมูลอยู่ในรูปแบบที่สามารถเพิ่มลงในฐานข้อมูลได้

- 5) ข้อมูลเพิ่มเข้าฐานข้อมูล
- 6) แสดงผลลัพธ์การเรียกใช้งาน API

#### 4.8.2 กระบวนการ การสร้าง Similarity Model

- 1) ผู้ใช้งานเลือกรูปแบบของ Model ที่จะสร้าง (หลักสูตร, รูปแบบการพยากรณ์ รายวิชา หรือ กลุ่มของวิชา)
- 2) ค้นหาข้อมูลของบันทึกในฐานข้อมูล
- 3) นำข้อมูลมาสร้าง Model
- 4) แปลงรูปแบบข้อมูลของ Model ให้อยู่ในรูปแบบ Pickled File
- 5) คำนวณหา RMSE ของ Model นั้น
- 6) จัดรูปให้ข้อมูลอยู่ในรูปแบบที่สามารถเพิ่มลงในฐานข้อมูลได้
- 7) นำข้อมูลเพิ่มเข้าฐานข้อมูล
- 8) แสดงผลลัพธ์การเรียกใช้งาน API

#### 4.8.3 กระบวนการ การเรียกใช้งาน Similarity Model

- 1) ผู้ใช้งาน Upload File CSV ผลลัพธ์การเรียนของตน
- 2) ผู้ใช้งานเลือกใช้ Model ที่อยู่ในฐานข้อมูล
- 3) ข้อมูลรายวิชาจากฐานข้อมูล
- 4) ค้นหาข้อมูลบันทึกจากฐานข้อมูล
- 5) นำข้อมูลรายวิชามาสร้าง List ของ Dictionary ที่เก็บ Class และวิชาใน Class
- 6) นำข้อมูลบันทึกมาหารายวิชาที่ผู้ใช้งานยังไม่ได้ลงทะเบียน
- 7) ข้อมูลของผู้ใช้งานไปพยากรณ์หาผลลัพธ์
- 8) แสดงผลลัพธ์การเรียกใช้งาน API

จากผลลัพธ์การทดลองที่ 4.8 พบว่าทั้งสามกระบวนการการทำงานนั้นสามารถทำงานได้บน Django Application ได้อย่างราบรื่น

## บทที่ 5

### สรุป

#### 5.1 บทสรุป

โครงการนี้มีวัตถุประสงค์เพื่อพัฒนาระบบประมวลผลข้อมูลผลการเรียนของนักศึกษาในอดีต ข้อมูลของรายวิชาต่างๆ และข้อมูลจากแบบสำรวจการมีงานทำของบัณฑิต เพื่อนำเสนอข้อมูลสถิติต่างๆ วิเคราะห์ข้อมูลผลการผลิตบัณฑิตเพื่อให้ได้ผลลัพธ์ว่าที่ผ่านมาหลักสูตรสามารถผลิตบัณฑิตกลุ่มใดได้บ้าง มีจำนวนมากน้อยเพียงใด สามารถพยากรณ์ว่าในอนาคตหลักสูตรสามารถผลิตบัณฑิตกลุ่มใดได้เป็นจำนวนเท่าใด เพื่อเป็นประโยชน์และอำนวยความสะดวกให้กรรมการหลักสูตรในการวางแผนการบริหารหลักสูตรในอนาคต และแสดงเป็นแผนภาพกราฟิกในการอำนวยความสะดวกให้หน่วยงานภายนอกได้รับทราบว่าหลักสูตรปัจจุบันของสถาบันสามารถผลิตบุคลากรที่มีความชำนาญด้านใดได้บ้าง โดยการทำงานของ Application นี้มี Web Application, Server, Database

##### 5.1.1 การทำงานในส่วนของ Web Application

- 1) Web Application มีระบบสร้าง File CSV Template สำหรับให้ผู้ใช้งานได้ผลลัพธ์การเรียนของตน
- 2) Web Application สามารถสร้าง Similarity Model แล้วนำลงฐานข้อมูลได้
- 3) Web Application สามารถนำเข้ารายวิชาแล้วเรียกใช้การจับกลุ่มรายวิชาด้วย NPL ได้
- 4) Web Application สามารถนำเข้าแบบสำรวจการมีงานทำของบัณฑิตแล้วแก้ไขข้อมูลของบัณฑิตคนนั้นในฐานข้อมูลได้
- 5) Web Application สามารถนำเข้า File CSV ผลลัพธ์การเรียนของผู้ใช้งานได้
- 6) Web Application มีระบบรองรับการร้องขอผลลัพธ์การพยากรณ์ได้
- 7) Web Application สามารถติดต่อกับ Database ได้

##### 5.1.2 การทำงานในส่วนของ Server

- 1) Server สามารถรัน Web Application ผ่าน Docker ได้
- 2) Server สามารถสร้าง Database ผ่าน Docker ได้

### 5.1.3 การทำงานในส่วนของ Database

- 1) Database สามารถเก็บข้อมูลผลลัพธ์การเรียนรู้ของบัณฑิตได้
- 2) Database สามารถเก็บข้อมูลรายวิชาได้
- 3) Database สามารถเก็บข้อมูลของ Model ได้

## 5.2 ปัญหาและอุปสรรคที่พบ

- 1) การติดต่อกับ Web Application ที่ทำงานอยู่ใน Server ที่ได้เตรียมไว้ โดยแนวทางในการแก้ปัญหาคือการศึกษาร่วมกันเกี่ยวกับการ Deployment และการ Forwarding Port
- 2) ข้อมูลจากแบบสอบถามการปฏิบัติงานของบัณฑิตที่มีน้อยเกินไปที่จะนำไป Train Model Job Classification โดยแนวทางในการแก้ไขปัญหาคือการที่จะต้องทำแบบสอบถามใหม่อีกครั้ง

## 5.3 แนวทางในการพัฒนาต่อ

- 1) นำข้อมูลจากแบบสอบถามการปฏิบัติงานของบัณฑิตที่ทำการเก็บรวบรวมใหม่มาทดลองแล้วพัฒนา Model Job Classification ต่อไป
- 2) พัฒนาส่วนของหน้าตาการใช้งานของผู้ใช้งาน
- 3) พัฒนาส่วนของการแสดงผลข้อมูลแบบภาพ
- 4) หาแนวทางในการ Deployment บน Sever ที่ได้เตรียมไว้

## บรรณานุกรม

สุเมธ ดาราพิสุทธิ. 2559. “การสร้างรายการเพลงโดยใช้การกรองร่วมแบบเซตชั้นที่เพิ่มขึ้นด้วยกลไกการลิ้มและการวิเคราะห์สถิติเชิงมุม.” วิทยานิพนธ์วิทยาศาสตรมหาบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์, มหาวิทยาลัยบูรพา.

นิภาภรณ์ พันธุ์นาม. 2563. “ระบบแนะนำสินค้าอาหารโดยใช้ระบบแนะนำแบบผสมผสาน.” สารนิพนธ์วิทยาศาสตรมหาบัณฑิต สาขาวิชาวิทยาการข้อมูล, มหาวิทยาลัยศรีนครินทรวิโรฒ.

Punna Rirkvaleekul. 2020. Machine Learning – คำศัพท์ที่ควรรู้. [Online].  
Available : <https://twinsynergy.co.th/machine-learning>.

พิพัฒน์ สมโลก. 2020. Machine Learning สิ่งใกล้ตัวแห่งโลกยุคใหม่. [Online].  
Available : <https://www.depa.or.th/th/article-view/article11-2563>.

อาทิตย์ สกุลเมือง. 2022. Natural Language Processing (NLP): เครื่องมือที่ช่วยให้คอมพิวเตอร์เข้าใจภาษามนุษย์. [Online]. Available : <https://bigdata.go.th/big-data-101/what-is-natural-language-processing>.

DUSADEEVIROJ. 2020. REVIEW ETL PROCESS. [Online].  
Available : <https://www.fusionsol.com/blog/review-etl-process>.

LDA. 2019. Recommendation System ระบบผู้ช่วยแนะนำที่รู้ใจเรามากกว่าตัวเราเอง. [Online].  
Available : <https://www.ldaworld.com/recommendation-system-lda>.

สรรพโชค สิงหสุวรรณ. 2022. การใช้งาน Docker เบื้องต้น. [Online].  
Available : [https://race.nSTRU.ac.th/home\\_ex/blog/topic/show/4200](https://race.nSTRU.ac.th/home_ex/blog/topic/show/4200)

Pattanapong Cherthong. 2016. ทำความรู้จัก Docker และ Software Container. [Online].

Available : <https://medium.com/@teamteam>

2013. เริ่มต้นการเขียนเว็บไซต์ ทำเว็บไซต์ด้วยภาษา Python กับ Django Framework. [Online].

Available : <https://www.amplysoft.com/knowledge/what-is-django-framework-python.html>

Pavarudh. 2022. ระบบจัดการฐานข้อมูล MariaDB คืออะไร. [Online].

Available : <https://km.cc.swu.ac.th/archives/4177>.

Pavarudh. 2021. ทำความรู้จักกับ Node.js. [Online].

Available : <http://marcuscode.com/tutorials/nodejs/introducing-nodejs>.



## ภาคผนวก ก

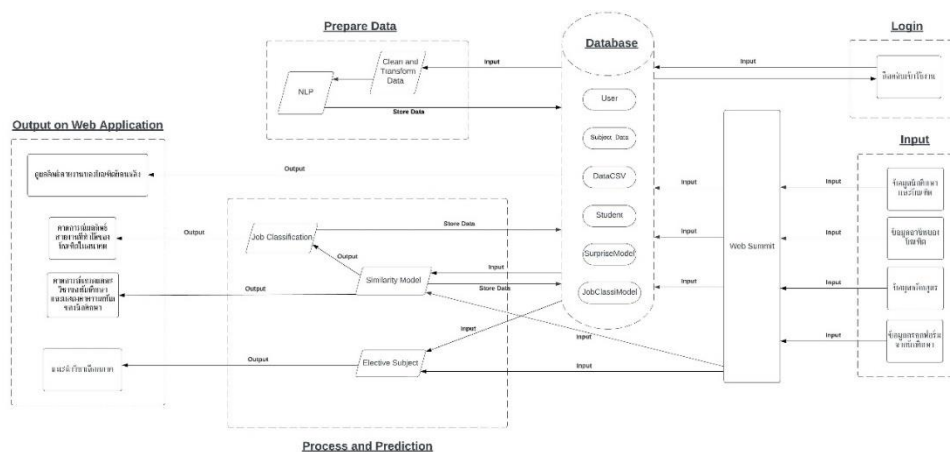
### การพิจารณาปริมาณงานขั้นต่ำสำหรับโครงการที่ 1

การพิจารณาปริมาณงานขั้นต่ำสำหรับโครงการที่ 1 นั้น เป็นการพิจารณาปริมาณความคืบหน้าของโครงการ 1 ของนักศึกษาชั้นปีการศึกษาที่ 4 คณะวิศวกรรมศาสตร์ สาขาภาควิชาวิศวกรรมคอมพิวเตอร์ ณ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง เพื่อให้ทราบว่าการพัฒนาโครงการ 1 นั้นมีความคืบหน้าหรือปริมาณงานขั้นต่ำที่มากเพียงสำหรับการพิจารณาการให้ผ่านหรือไม่ ซึ่งนักศึกษาชั้นปีที่ 4 คณะวิศวกรรมศาสตร์ สาขาภาควิชาวิศวกรรมคอมพิวเตอร์ จำเป็นต้องอภิปรายปริมาณงานของตนให้คณะกรรมการทราบโดยมีรายละเอียดดังนี้

1. ชื่อโครงการ (ภาษาอังกฤษ) Curriculum output prediction from student academic data
2. ประเภทของโครงการ

ระบุ	ประเภทโครงการ	ปริมาณงานขั้นต่ำ
	1. HW + SW	ระบบที่เชื่อมโยงทุกส่วนเข้าด้วยกัน
	2. SW Dev	Complete Design Front End and Back End และ Implement 30% ของฟังก์ชันทั้งหมด
	3. Research	มีผลการทดลองจาก algorithm หลักด้วยข้อมูลที่ใช้งานจริง

3. แจกแจงปริมาณงานที่ได้ทำในวิชาโครงการ 1 อิงตามประเภทโครงการ
  - โครงสร้างของระบบ การออกแบบดังรูปที่ ก1 นั้นเป็นการออกภาพรวมของระบบแล้วได้นำไปเป็นรากฐานในการพัฒนาของโครงการ



รูป ก1 แผนภาพการทำงานโดยรวมของระบบ

- ส่วนการ Implement ฟังก์ชัน
  - 1) ฟังก์ชัน Map Subject Group สำเร็จ
  - 2) ฟังก์ชัน similarity หลักสูตรปกติ สำเร็จ
  - 3) ฟังก์ชัน similarity หลักสูตรต่อเนื่อง สำเร็จ
  - 4) ฟังก์ชัน Analytic Data ไม่สำเร็จ
  - 5) ฟังก์ชัน Sum Occupations ไม่สำเร็จ
  - 6) ฟังก์ชัน Recommend Elective Subject ไม่สำเร็จ
  - 7) API Simple CSV Generator สำเร็จ
  - 8) API Request All Prediction Handler ไม่สำเร็จ
  - 9) API Request All Outcome Predict of All Student ไม่สำเร็จ
  - 10) API Calculate overall (old) Outcome ไม่สำเร็จ
  - 11) API Request Elective Subject ไม่สำเร็จ
  - 12) API Add Update Edit สำเร็จ
  - 13) API Login Handler ไม่สำเร็จ

Implementation สำเร็จทั้งหมด 5 ฟังก์ชันจากทั้งหมด 13 ฟังก์ชันคิดเป็น 38.46%