**ระบบวิเคราะห์ และ พยากรณ์ สำหรับการบริหารหลักสูตรวิศวกรรมคอมพิวเตอร์**

**Analytics and Prediction System for CE Curriculum administrators**

**ณิชกานต์ สุขุมจิตพิทโยทัย**

**นรวิชญ์ อยู่บัว**

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลกัสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2565

ปริญญานิพนธ์ปี การศึกษา 2564

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบวิเคราะห์ และ พยากรณ์ สำหรับการบริหารหลักสูตรวิศวกรรมคอมพิวเตอร์

Analytics and Prediction System for CE Curriculum administrators

ผู้จัดทำ

1. นางสาวณิชกานต์ สุขุมจิตพิทโยทัย รหัสนักศึกษา 62010299

2. นายนรวิชญ์ อยู่บัว รหัสนักศึกษา 62010465

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_อาจารย์ที่ปรึกษา

(ผศ. ดร. ธนัญชัย ตรีภาค)

**ระบบวิเคราะห์ และ พยากรณ์ สำหรับการบริหารหลักสูตรวิศวกรรมคอมพิวเตอร์**

นางสาวณิชกานต์ สุขุมจิตพิทโยทัย 62010299

นายนรวิชญ์ อยู่บัว 62010465

ผศ. ดร. ธนัญชัย ตรีภาค อาจารย์ที่ปรึกษา

ปีการศึกษา 2565

# บทคัดย่อ

โครงงานนี้จัดทำขึ้นเพื่อ พัฒนาระบบประมวลผลข้อมูลผลการเรียนของนักศึกษาในอดีต ข้อมูลของรายวิชาต่างๆ และข้อมูลจากแบบสำรวจการมีงานทำของบัณฑิต เพื่อนำเสนอข้อมูลสถิติต่างๆ วิเคราะห์ข้อมูลผลการผลิตบัณฑิตเพื่อให้ได้ผลลัพธ์ว่าที่ผ่านมาหลักสูตรสามารถผลิตบัณฑิตกลุ่มใดได้บ้าง มีจำนวนมากน้อยเพียงใด สามารถพยากรณ์ว่าในอนาคตหลักสูตรสามารถผลิตบัณฑิตกลุ่มใดได้เป็นจำนวนเท่าใด เพื่อเป็นประโยชน์และอำนวยความสะดวกให้กรรมการหลักสูตรในการวางแผนการบริหารหลักสูตรในอนาคต และแสดงเป็นแผนภาพกราฟิกในการอำนวยความสะดวกให้หน่วยงานภายนอกได้รับทราบว่าหลักสูตรปัจจุบันของสถาบันสามารถผลิตบุคลากรที่มีความชํานาญด้านใดได้บ้าง

**Analytics and Prediction System for CE Curriculum administrators**

Ms. Nichakan Sukhumjitpitayotai 62010299

Mr. Narawich Youbua 62010465

Mr. Thanunchai Threepak Advisor

Academic Year 2022

# Abstract

# กิตติกรรมประกาศ

โครงงานในภาคการศึกษานี้สำเร็จลุล่วงได้ด้วยดีจากความช่วยเหลือจากหลากหลายบุคคล โครงงานในภาคการศึกษานี้จะผ่านไปไม่ได้หากปราศจากความช่วยเหลือจากบุคคลเหล่านี้ขอขอบคุณอาจารย์ที่ปรึกษา ผศ. ดร. ธนัญชัย ตรีภาค ที่ให้ความช่วยเหลือในเรื่องต่าง ๆ ไม่ว่าจะเป็นการให้คำแนะนำถึงแนวทางการทำงานที่ดี การให้คำปรึกษาเพื่อหาทางออกเมื่อพบเจอกับปัญหา รวมถึงให้ความรู้เกี่ยวกับตัวงานทำให้งานต่าง ๆ เมื่อเจอปัญหาก็สามารถผ่านไปได้ด้วยดี

ขอขอบคุณคณาจารย์ในภาควิชาวิศวกรรมคอมพิวเตอร์ ที่ประสาทวิชาการความรู้มาตลอด 4 ปี ซึ่งความรู้หลาย ๆ แขนงก็ถูกใช้เป็นพื้นฐาน และเป็นส่วนหนึ่งของโครงงานนี้

ขอขอบคุณเพื่อน ๆ วิศวกรรมคอมพิวเตอร์ที่ให้คำปรึกษา และแลกเปลี่ยนความรู้ซึ่งกันและกันรวมถึงการรับฟังปัญหา

สุดท้ายนี้ขอขอบคุณบิดา มารดาและครอบครัว ที่เลี้ยงดูอบรมสั่งสอนและให้ความรู้คุณธรรมจริยธรรม และให้การสนับสนุนด้านการศึกษาจนได้มีโอกาสมาทำโครงงานนี้

ณิชกานต์ สุขุมจิตพิทโยทัย

นรวิชญ์ อยู่บัว

# สารบัญ

**หน้า**

**บทคัดย่อ**

**ABSTRACT**

**กิตติกรรมประกาศ**

**สารบัญ**

**สารบัญตาราง**

**สารบัญรูป**

**บทที่ 1 บทนำ**

**1.1 ความเป็นมาของปัญหา**

**1.2 วัตถุประสงค์**

**1.3 ประโยชน์ของโครงงาน**

**1.4 ข้อจำกัดของโครงงาน**

**1.5 แผนการดำเนินงาน**

**บทที่ 2 ทฤษฏีและงานวิจัยที่เกี่ยวข้อง**

**2.1 ทฤษฏีที่เกี่ยวข้อง**

**2.2 เครื่องมือที่เกี่ยวข้อง**

**2.3 งานวิจัยที่เกี่ยวข้อง**

**บทที่ 3 การออกแบบ**

**บทที่ 4 ผลการดำเนินงาน**

**บทที่ 5 สรุป**

**5.1 บทสรุป**

**5.2 ปัญหาและอุปสรรคที่พบ**

**5.3 แนวทางในการพัฒนาต่อ**

**เอกสารอ้างอิง**

**ภาคผนวก**

# สารบัญตาราง

ตาราง หน้า

# สารบัญรูป

รูป หน้า

**บทที่ 1**

**บทนำ**

* 1. **ความเป็นมาของปัญหา**

Data Analytics เป็นการวิเคราะห์ข้อมูลที่มีอยู่ตั้งแต่อดีตจนถึงปัจจุบัน ในกรณีที่ข้อมูลเพียงพอและเหมาะสมจะสามารถนำมาคาดการณ์แนวโน้ม ทำนายอนาคตที่เป็นประโยชน์ พยากรณ์สิ่งที่กำลังจะเกิดขึ้นหรือน่าจะเกิดขึ้นโดยใช้ข้อมูลในอดีตกับแบบจำลองทางสถิติรวมถึงการให้คำแนะนำทางเลือกต่าง ๆ และผลของแต่ละทางเลือก

จากปัญหาที่ทางผู้จัดทำเล็งเห็นความสำคัญคือการนำข้อมูลผลการเรียนของนักศึกษาในอดีตมาใช้ประโยชน์ในการบริหารหลักสูตร และ นำมาวิเคราะห์ผลเพื่อช่วยในการวางแผนการเรียนของนักศึกษา ซึ่งการวางแผนในการเรียนของหลักสูตรจะสามารถช่วยอาจารย์และบุคลากรที่เกี่ยวข้องกับการศึกษาในด้านของการบริหารหลักสูตร เพื่อวางแผนการเพิ่มหรือลดจำนวนผู้เรียนในรายวิชาต่าง ๆ ซึ่งส่งผลต่อการผลิตบัณฑิตด้านต่าง ๆ ได้

ดังนั้นผู้จัดทำจึงได้เห็นถึงความสำคัญการประเมินสถานะขอหลักสูตร ของระบบแนะนำการวางแผนการคาดการณ์จากการใช้ความรู้ทางด้าน Data Analytics, Prediction และ Recommendation โดยใช้ข้อมูลผลการเรียนของนักศึกษาในอดีต เพื่อพัฒนาระบบช่วยเหลือ และตอบโจทย์ให้แก่นักศึกษาและบุคคลากรทางการศึกษาหรือบุคคลที่เกี่ยวข้องได้

* 1. **วัตถุประสงค์**

1. เพื่อนำข้อมูลของผลการเรียนของนักศึกษาในอดีตและข้อมูลจากแบบสำรวจการมีงานทำของบัณฑิตมาใช้ ในการวางแผนการเรียนหรือประเมินอาชีพในอนาคตของนักศึกษาได้
2. ประมวลผลข้อมูลผลการเรียนของนักศึกษาในอดีต และข้อมูลจากแบบสำรวจการมีงานทำของบัณฑิต และทำแผนภาพกราฟิกเพื่อนำเสนอข้อมูล อำนวยความสะดวกให้กรรมการหลักสูตรในการวางแผนการ ทำงาน
3. เพื่อนำข้อมูลผลการเรียนของนักศึกษาในอดีต มาพัฒนาเป็นระบบแนะนำและวางแผนการเรียนตัวของ นักศึกษาได้
4. เพื่อนำข้อมูลการพยากรณ์อาชีพในอนาคตของนักศึกษาในสถาบันมาแสดงเป็นแผนภาพกราฟิกในการ อำนวยความสะดวกให้หน่วยงานภายนอกได้รับทราบว่าหลักสูตรปัจจุบันของสถาบันสามารถผลิต บุคลากรที่มีความชํานาญด้านใดได้บ้าง
   1. **ประโยชน์ของโครงงาน**
5. ได้ระบบรวบรวมข้อมูลผลการเรียนของนักศึกษาและข้อมูลแบบสำรวจการทำงานของบัณฑิต แล้วนำมาวิเคราะห์และนำเสนอข้อมูลที่เป็นประโยชน์ในการบริหารหลักสูตรของกรรมการหลักสูตร
6. มีระบบที่สามารถแนะนำ วางแผน และประเมินอาชีพในอนาคตจากผลการเรียนของนักศึกษา
   1. **ข้อจำกัดของโครงงาน**
7. ข้อมูลผลการเรียนในอดีตย้อนหลังมีเพียง 2 ปี
8. ข้อมูลผลการเรียนในอดีตจะได้จากสำนักทะเบียนและประมวลผล โดยกรรมการหลักสูตรจะเป็นผู้ร้องขอข้อมูลดังกล่าวและนำเข้าระบบ
9. การทำนายต่าง ๆ จะใช้ข้อมูลเพียง 2 แหล่งคือข้อมูลผลการเรียนของนักศึกษาจากสำนักทะเบียนและประมวลผล และแบบสอบถามการมีงานทำของบัณฑิตเท่านั้น
   1. **แผนการดำเนินงาน**

แผนการดำเนินงานในการพัฒนาโครงงานตลอดระยะเวลา 2 ภาคการศึกษา ตั้งแต่เดือน สิงหาคม พ.ศ. 2565 - มีนาคม พ.ศ. 2566 แสดงดังรูป 1.1



รูป 1.1 แผนการดำเนินงาน

**บทที่ 2**

**ทฤษฎีและงานวิจัยที่เกี่ยวข้อง**

**2.1 ทฤษฏีที่เกี่ยวข้อง**

**2.1.1 Classification and Prediction**

Classification and Prediction คือการจำแนกประเภทของข้อมูล โดยนำมาใช้ในการวิเคราะห์ข้อมูล กระบวนการจำแนกของประเภทของข้อมูลแบ่งเป็น 2 ส่วน

1) Training Data คือการนำข้อมูลมาทำการเรียนรู้หรือสอนให้กับคอมพิวเตอร์เพื่อทำการสร้างโมเดลและวัดประสิทธิภาพของโมเดล โดยการสร้างโมเดลมีหลายวิธี เช่น Decision Tree, Naive Bayes, K Nearest Neighbors และ Neural Network เป็นต้น

2) Predict คือการนำข้อมูลใหม่ป้อนเข้าโมเดลที่ได้จากการ Training Data ไปทำการคำนวณหรือทำนาย

ประเภทของปัญหาด้าน Classification

1) Binary classification (การจำแนกแบบไบนารี)

เปรียบเทียบให้เข้าใจง่ายที่สุดก็คือ ตัวแปรที่แบ่งเป็นเพียงสองหมวดหมู่ เช่น ผลลัพธ์แบบ ใช่ หรือ ไม่ใช่ ซื้อ หรือ ไม่ซื้อ หากเปรียบเป็นตัวเลขก็คือ 0 กับ 1 อัลกอริทึมที่ใช้คู่กับการจำแนกแบบไบนารี ก็มี k-Nearest Neighbors Decision Trees หรือ Naive Bayes

2) Multi-Class Classification (การจำแนกประเภทหลายคลาส)

ต่างกับการจำแนกแบบไบนารีตรงที่มีหมวดหมู่มากกว่าสอง ยกตัวอย่างเช่น รูปภาพที่คล้ายคลึงกับรูปภาพที่อัปโหลดเพื่อค้นหาคำศัพท์ที่คาดว่าจะพิมพ์ใน predictive keyboard คือผลลัพธ์ที่มีได้มากกว่า 2 หมวดหมู่ อัลกอริทึมที่ใช้คู่กับการจำแนกประเภทหลายคลาสก็สามารถใช้อัลกอริทึมคล้ายกับที่ใช้กับการจำแนกแบบไบนารีได้

3) Multi-Label Classification (การจำแนกประเภทหลายเลเบล)

เพื่อการเปรียบเทียบให้เข้าใจง่ายขึ้น ตัวอย่างเช่น รูปภาพรูปหนึ่งสามารถมีรูปดอกไม้ ท้องฟ้าก้อนเมฆได้ แต่รูปภาพรูปนั้นจะจัดว่าเป็นหมวดหมู่รูปวาด รูปถ่าย หรือรูปเสีย Multi-Label Classification ก็คือการเลเบล หรือติดฉลากว่าในรูปนั้น ๆ มีดอกไม้หรือเปล่ามีก้อนเมฆหรือไม่ส่วน Multi-Class Classification จะจำแนกว่ารูปนั้นเป็นรูปวาดรูปถ่ายรูปเสีย

4) Imbalanced Classification (การจำแนกแบบข้อมูลไม่เท่าเทียม)

เป็นปัญหาที่เกิดจากข้อมูลที่ไม่เท่าเทียม (Imbalanced dataset) ตัวอย่างเช่นข้อมูลการทุจริตโดยข้อมูลส่วนใหญ่ย่อมเป็นข้อมูลที่จัดว่า “ไม่ทุจริต” และจะมีเปอร์เซ็นต์น้อยที่จัดว่าเป็น “ทุจริต” เป็นต้น

**2.1.2 Machine Learning**

Machine Learning คือ การทำให้ระบบของคอมพิวเตอร์เรียนรู้ได้ด้วยตนเอง โดยใช้ข้อมูล ด้วยวิธีการใส่ข้อมูลและผลลัพธ์เข้าไป เพื่อทำให้โปรแกรมนำไปประมวลผลและทำนาย Output และ Input ของข้อมูลใหม่ โดยแบ่ง Machine Learning ออกเป็น 3 ประเภท คือ

1) Supervised Learning คือการเรียนรู้ที่เครื่องจักรจะต้องอาศัยข้อมูลใน การฝึกฝน เปรียบเสมือนกับการเรียนการสอนของเด็กเล็ก โดยจำเป็นจะต้องอาศัยชุดข้อมูลต่าง ๆ ซึ่งประกอบไปด้วยชุดของข้อมูลและชุดผลลัพธ์ของข้อมูลที่ต้องการ

2) Unsupervised Learning เป็นการเรียนรู้ที่ให้เครื่องจักรนั้นสามารถ เรียนรู้ได้ด้วยตนเอง โดยไม่ต้องมีค่าเป้าหมายของแต่ละข้อมูล ซึ่งวิธีการคือมนุษย์จะเป็นผู้ใส่ข้อมูลต่าง ๆ และกำหนดสิ่งที่ต้องการจากข้อมูลเหล่านั้น โดยให้เครื่องจักรวิเคราะห์จากการจำแนกและสร้างแบบแผนจากข้อมูลที่ได้รับมา

 3) Reinforcement Learning เป็นการเรียนรู้สิ่งต่าง ๆ จากการลองผิดลองถูก ภายใต้แนวคิดที่ว่าจะเลือกกระทำสิ่งที่ทำให้ได้ผลลัพธ์มากที่สุด โดยทำการเรียนรู้จากการลองผิดลองถูกในสถานการณ์ในอดีตหรือระบบจำลองและพยายามที่จะพัฒนาระบบการตัดสินใจของตัวเองให้ดีขึ้นเรื่อย ๆ โดยที่อาจจะพัฒนาด้วยการพยายามสร้างแบบจำลองสถานการณ์ต่าง ๆ

* + 1. **Extract-Transform-Load (ETL)**

Extract-Transform-Load คือ กระบวนการหนึ่งในระบบ Data Warehouse โดยระบบที่ออกแบบเอาไว้จะดึงข้อมูลออกมาจากหลายๆ ที่นำกระบวนการตรวจสอบคุณภาพของข้อมูลมาประยุกต์ใช้ มีการเชื่อมโยงและปรับข้อมูลให้เป็นไปในรูปแบบเดียวกันเพื่อให้ ข้อมูลจากหลายๆ แหล่งสามารถใช้งานร่วมกันได้ และทำการส่งมอบ

1) Extract เป็นกระบวนการเริ่มต้นของระบบดึงข้อมูลจากแหล่งของข้อมูล จะประกอบด้วยข้อมูลจากหลายๆ ที่ข้อมูลที่อยู่ต่างที่กันนั้นอาจจะอยู่ในรูปแบบที่แตกต่างกัน ยกตัวอย่างเช่น อาจจะอยู่ในรูปแบบของฐานข้อมูลคนละชนิด หรือ ไม่ใช่ฐานข้อมูลแท้จริงซึ่งอาจจะเป็นระบบไฟล์ข้อมูลธรรมดา (Flat Files)

2) Transforming ขั้นตอนการแปลงข้อมูลนี้จะมีการใช้กฏหรือฟังชั่น (Function) มากมายเพื่อที่จะแปลงข้อมูลให้ได้ตามที่ต้องการก่อนที่จะนำข้อมูลเข้าไปยังปลายทาง ข้อมูลจากต้นทางบางแหล่งข้อมูลมีความจำเป็นน้อยมากหรือแทบจะไม่ต้องการ การแปลงข้อมูลเลย แต่ในบางแหล่งอาจจะต้องการกระบวนการที่ซับซ้อน ซึ่งจะกินทรัพยากรระบบและเวลาในการประมวลผล ทั้งนี้ ความซับซ้อนของข้อมูลนั้นขึ้นอยู่กับความต้องการของธุรกิจ หรือ เป้าหมายของการนำข้อมูลไปใช้งาน โดยมีกระบวนการตัวอย่างต่อไปนี้

1) Selection คือ เลือกเฉพา Column ที่ต้องการที่จะ Load ยกตัวอย่าง เช่น ถ้าต้นทางข้อมูลมี 3 Column หรือจะเรียกว่ามี 3 attributes เช่น roll\_no, age และ salary จะมีการแปลงข้อมูล และ เลือกที่จะไม่มีการแปลงข้อมูลหากว่า column salary มีค่าเป็น Null

2) Translation คือ การแปลข้อมูล ยกตัวอย่างเช่น หากข้อมูลต้นทางมีการเก็บข้อมูลเพศว่า 1 เป็นเพศชาย และ 2 เป็นเพศหญิง จะต้องมีการแปลจากโค้ดที่กำหนดก่อนหน้านี้ให้ 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

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

* + 1. **Natural Language Processing (NLP)**

Natural Language Processing (NLP) คือเครื่องมือที่ช่วยให้คอมพิวเตอร์เข้าใจภาษาที่ซับซ้อนของมนุษย์ เป็นศาสตร์ที่สำคัญทางด้าน Machine Learning โดยมันเป็นสาขาวิชาหนึ่งที่ประกอบด้วยองค์ความรู้จากหลากหลายแขนง อาทิ ภาษาศาสตร์ (Linguistics) วิทยาการคอมพิวเตอร์ (Computer Science) ปัญญาประดิษฐ์ (Artificial Intelligence: AI) รวมถึงสถิติ (Statistics) โดย NLP มีจุดกำเนิดมาตั้งแต่ช่วงกลางศตวรรษที่ 19 และได้มีการพัฒนาต่อยอดมาเรื่อย ๆ จนถึงปัจจุบัน แบ่งออกเป็น 3 ยุค ดังนี้

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

ในยุคแรก NLP ถูกใช้งานด้วยวิธีการตามกฎ (Rule-based Method) โดยนักภาษาศาสตร์ผู้มีความเชี่ยวชาญโครงสร้างของภาษาที่สนใจ จะเป็นผู้เขียนกฎต่าง ๆ ขึ้นมา เพื่อให้คอมพิวเตอร์สามารถนำไปคำนวณเพื่อหาคำตอบของโจทย์ที่ต้องการได้

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

ในยุคต่อมา พบว่าการเขียนกฎด้วยมือไม่สามารถตอบสนองต่อโจทย์ที่มีความซับซ้อนมาก ๆ ได้ อย่างไรก็ตาม สิ่งที่ได้มาทดแทนในยุคนี้คือ ประสิทธิภาพของเครื่องคอมพิวเตอร์ รวมถึงความรู้ทางด้านสถิติ และ Machine Learning ซึ่งได้ถูกนำมาพัฒนาเพื่อใช้ในการทำงานด้าน NLP โดยมีการนำเข้าข้อมูลเพื่อให้คอมพิวเตอร์สามารถเรียนรู้ด้วยตนเองแทนการใช้ผู้เชี่ยวชาญทางด้านภาษา

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

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

ตัวอย่างการประยุกต์ใช้ NLP ในด้านต่าง ๆ

1) ด้านการทำงานวิจัย วงการวิจัยเป็นหนึ่งในแหล่งของข้อมูลทางภาษาขนาดใหญ่ ซึ่งเปิดโอกาสให้ NLP สามารถเข้ามามีบทบาทได้อย่างหลากหลาย ตัวอย่างเช่น การประยุกต์ใช้ Topic Model ในการจัดหมวดหมู่เอกสารเพื่อวิเคราะห์หาหัวข้อของบทความงานวิจัย

2) ด้านพาณิชย์อิเล็กทรอนิกส์ การใช้จ่ายเพื่อสั่งซื้อสินค้าผ่านช่องทางออนไลน์อย่างแพลตฟอร์ม E-commerce เข้ามามีบทบาทสำคัญเป็นอย่างมากในระบบเศรษฐกิจ ซึ่งนอกจากจะก่อให้เกิดปริมาณการทำธุรกรรมที่มหาศาลแล้ว ยังมีการนำเข้าข้อมูลประเภทข้อความขนาดใหญ่เช่นเดียวกัน ไม่ว่าจะเป็น คำอธิบายสินค้าและบริการ การแสดงความคิดเห็นของผู้บริโภค รวมถึงการสนทนากันระหว่างผู้ซื้อและผู้ขายผ่านทางช่องแชท

3) ด้านการแพทย์ ข้อมูลทางการแพทย์มีจำนวนไม่น้อยที่มีลักษณะเป็นข้อความ ซึ่งสามารถนำไปใช้วิเคราะห์ต่อได้ ตัวอย่างเช่น บทสนทนาระหว่างแพทย์และผู้ป่วย การวินิจฉัยโรคโดยแพทย์ และประวัติการรักษาของผู้ป่วย

4) ด้านกฎหมาย สำหรับงานด้านกฎหมาย ก็มีปริมาณข้อมูลทางภาษาจำนวนมากและหลากหลายเช่นเดียวกัน อาทิ ประมวลกฎหมายต่าง ๆ คำร้องต่อศาล คำให้การของคู่ความ และคำพิพากษาของศาล ซึ่งสามารถประยุกต์ใช้เครื่องมือ NLP ได้ในหลายมิติไม่ว่าจะเป็นการใช้ PoS Tagging และ NER เพื่อช่วยในการตีความประมวลกฎหมาย

**2.1.5 Recommendation System**

Recommendation System เป็นระบบที่จะแนะนำสิ่ง (item) ที่ “เหมาะสม” ให้กับผู้ใช้ โดย item เป็นได้ตั้งแต่ ข่าว เนื้อหา เพลง course เรียน ไปจนถึงสินค้าที่ขายในร้าน online โดยระบบสามารถให้คำแนะนำที่ต่างกันตามความชอบของแต่ละผู้ใช้ได้ โมเดลโดยส่วนใหญ่ที่มักจะถูกใช้กันมีอยู่ด้วยกันสามประเภท ได้แก่

1) Content-based Filtering เป็นรูปแบบโมเดลที่แนะนำลักษณะของตัวบริการหรือสินค้าเป็นตัวตั้งแล้วแนะนำสินค้าและบริการอื่นที่มีคุณลักษณะคล้าย ๆ กัน

2) **Collaborative** Filtering เป็นรูปแบบโมเดลที่เรียนรู้จากพฤติกรรมของผู้ใช้กับผู้ใช้คนอื่น ๆ ที่คล้ายคลึงกัน

1) Memory-based ซึ่งจะเน้นไปในทางที่ดูข้อมูลแล้วหาความสัมพันธ์ ระหว่างผู้ใช้หรือ สินค้าจากข้อมูลตรง ๆ

2) Model-based ใช้เทคนิคของ machine learning เพื่อหา user embedding และ item embedding มาทำนาย rating ที่ผู้ใช้จะให้กับสินค้า หรือ relevance score

3) Hybrid ใช้หลาย ๆ วิธีการมารวมกัน Hybrid system เป็นการมัดรวมทั้งสองอัลกอริทึมด้านบนเอาไว้เพื่อทำให้ระบบการแนะนำสมบูรณ์ขึ้น ซึ่งระบบนี้ถูกนำไปใช้ในปัจจุบันมากที่สุดแทบจะทุกแพลทฟอร์มใหญ่ที่มีการแนะนำสินค้าและบริการ

3) Hybrid system เป็นการมัดรวมทั้งสองระหว่าง Content-based Filtering และ Collaborative Filtering เพื่อทำให้ระบบการแนะนำสมบูรณ์ขึ้น

## **2.2** เครื่องมือที่เกี่ยวข้อง

**2.2.1 Docker**

Docker คือแพลตฟอร์มซอฟต์แวร์ที่ช่วยให้คุณสร้าง ทดสอบ และติดตั้งแอปพลิเคชันใช้จริงได้อย่างรวดเร็ว Docker จะบรรจุซอฟต์แวร์ลงไปในหน่วยที่เป็นมาตรฐานเรียกว่า คอนเทนเนอร์ ซึ่งจะมีทุกสิ่งที่ซอฟต์แวร์ต้องใช้ในการเรียกใช้งาน รวมทั้งไลบรารี เครื่องมือสำหรับระบบ โค้ด และรันไทม์ เมื่อใช้ Docker คุณจะสามารถติดตั้งใช้จริงและปรับขนาดแอปพลิเคชันให้เหมาะกับทุกสภาพแวดล้อมและทราบว่าโค้ดของคุณจะเรียกใช้ได้อย่างอย่างรวดเร็ว

A screenshot of a computer

Description automatically generated with medium confidence การทำงานของ Docker ทำงานโดยการช่วยสร้างวิธีมาตรฐานในการเรียกใช้โค้ด Docker เป็นระบบปฏิบัติการสำหรับคอนเทนเนอร์ คอนเทนเนอร์จะจำลองระบบปฏิบัติการของเซิร์ฟเวอร์ ซึ่งคล้ายคลึงกับวิธีการที่เครื่องเสมือนจำลอง (ลดความจำเป็นในการจัดการโดยตรง) ฮาร์ดแวร์ของเซิร์ฟเวอร์ Docker ได้รับการติดตั้งลงบนแต่ละเซิร์ฟเวอร์และสร้างคำสั่งง่ายๆ ที่คุณสามารถใช้ในการสร้าง เริ่มต้น หรือหยุดคอนเทนเนอร์

องค์ประกอบต่างๆของ Docker

Docker image

คือต้นแบบของ Container ข้างในจะเป็น Linux ที่มีการติดตั้ง Application และ มีการ Configuration เอาไว้แล้ว ซึ่งเกิดมาจากการ build ไฟล์ Docker file ขึ้นมาเป็น image

Docker container

Docker container จะถูกสร้างมาจาก Docker Image ที่เป็นต้นแบบ เกิดเป็น container จะได้ Service หรือ Application ที่สามารถเรียกใช้งานได้ทันที

Docker registry

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

**2.2.2 Django**

Django Framework เป็นชุดเครื่องมือ Framework สำหรับ การพัฒนาเว็บไซต์ด้วยภาษา Python ซึ่งความเป็นจริงแล้วทุกวันนี้มี Framework สำหรับการเขียนเว็บไซต์ด้วยภาษา Python ค่อนข้างเยอะ ซึ่ง Django Framework ก็เป็นหนึ่งใน Framework สำหรับการพัฒนาเว็บไซต์ และทำเว็บไซต์ด้วยภาษา Python ด้วยเช่นกัน

คุณสมบัติของ Django Framework

1) Object-relational mapper คือ การกำหนด Data Model ในภาษา Python เพื่อการทำงานด้านข้อมูล และสนับสนุน dynamic database-access API

2) Automatic admin interface คือ ส่วนของการสร้าง Interface อัตโนมัติสำหรับการ add, edit , delete และ search ด้วย Django Framework

3) Elegant URL design คือ การทำให้ URL มีความสวยงาม สั้น กระชับ และสื่อความหมายของหน้านั้น ๆ ได้อย่างชัดเจน เหมาะสมกับการทำ SEO ในปัจจุบัน

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 เป็นแพ็กเกจที่รวบรวม Library ด้าน การเรียนรู้ของเครื่อง (Machine Learning) เอาไว้ และถูกออกแบบมาให้ทำงานร่วมกับ Library ของภาษา Python อย่าง NumPy และ SciPy ได้ดี

นอกจากนี้ Scikit-learn ยังเป็น Open Source ที่เปิดให้ผู้ที่สนใจสามารถเข้าไปพัฒนาต่อยอดได้และที่ทำให้ทุกคนต่างยอมรับคือเป็นแหล่งรวม Library และอัลกอริทึมที่เน้นไปในด้านของ การเรียนรู้ของเครื่อง (Machine Learning) ซึ่งมีส่วนในการทำ แบบจำลองข้อมูล (Data Modeling) ทั้งนั้น อีกหนึ่งปัจจัยที่ทำให้มีผู้ใช้เยอะ และเป็นเครื่องมือที่แนะนำสำหรับมือใหม่ เพราะเป็น Interface ระดับสูง ให้มือใหม่สามารถเข้าใจภาพรวมและ ขั้นตอนการทำงาน ของการเรียนรู้ของเครื่อง (Machine Learning) ได้เครื่องมือที่ผู้ใช้งานสามารถนำไปใช้ในโปรเจกต์ได้

1) Classification

หนึ่งในตัวอย่างของ Classification คือการแยกอีเมลว่าเป็นสแปมหรือไม่ Classification ถือว่าเป็นหนึ่งในแขนงของ Supervised Learning (การเรียนรู้แบบมีผู้สอนนั้น คือการทำให้คอมพิวเตอร์สามารถหาคำตอบของปัญหาได้ด้วยตัวเอง หลังจากเรียนรู้จากชุดข้อมูลตัวอย่างไปแล้วระยะหนึ่ง) การเรียนรู้ของอัลกอริทึมจากชุดข้อมูล (Datasets) ที่มีคำตอบที่ถูกต้อง

2) Regression

เป็นเครื่องมือเพื่อเข้าใจความสัมพันธ์ระหว่างข้อมูล Input กับ Output ซึ่งก็ถือว่าเป็นอีกหนึ่งแขนงของ Supervised Learning เช่นกัน ตัวอย่างของอัลกอริทึมนี้คือการเข้าใจพฤติกรรมของราคาหุ้น

3) Clustering

คือการจำแนกกลุ่มข้อมูลที่มีคุณสมบัติคล้ายกัน เครื่องมือนี้ถือว่าเป็นแขนงของ Unsupervised Learning (เป็นการเรียนรู้ที่ไม่มีผู้สอน โดยที่เครื่องจักรนั้น ๆ จะทำการเรียนรู้สิ่งต่าง ๆ ด้วยตัวเอง โดยไม่ต้องตั้งค่าเป้าหมายของแต่ละข้อมูล) และตัวอย่างของการใช้อัลกอริทึมนี้คือการจำแนกกลุ่มลูกค้าตามพื้นที่

4) Model selection

อัลกอริทึมเพื่อใช้เปรียบเทียบ ตรวจสอบ และเลือกโมเดล พารามิเตอร์ที่เหมาะสมกับชุดข้อมูลที่สุดในโปรเจ็ค จะช่วยเพิ่มความแม่นยำของอัลกอริทึม การเรียนรู้ของเครื่อง (Machine Learning) ได้

5) Pre-processing

ในขั้นตอนของ Data Analysis หรือการเข้าใจและวิเคราะห์ข้อมูล อาจต้องมีการแก้ไขให้ข้อมูลอยู่ในรูปแบบที่เรานำไปใช้งานได้ เครื่องมือตัวนี้ของ Scikit-learn จะสามารถช่วยจัดการกับข้อมูลได้

**2.2.3 MariaDB**

MariaDB คือ ฟรีซอฟต์แวร์แบบโอเพนซอสต์สำหรับจัดการกับฐานข้อมูล MariaDB เป็นหนึ่งในฐานข้อมูลที่ได้รับความนิยมมากที่สุดในโลก MariaDB ถูกพัฒนาขึ้นโดยนักพัฒนาเดิมของ MySQL เนื่องจากความกังวลที่เกิดขึ้นเมื่อ MySQL ถูกซื้อโดย Oracle Corporation ในปี 2009 ตอนนี้นักพัฒนาและผู้ดูแลของ MariaDB ได้รวมรายเดือนกับฐานรหัส MySQL เพื่อให้แน่ใจว่า MariaDB มีการแก้ไขข้อบกพร่องที่เกี่ยวข้องเพิ่มลงใน MySQL

MariaDB ได้รับการพัฒนาเป็นซอฟต์แวร์โอเพ่นซอร์ส และเป็นฐานข้อมูลเชิงสัมพันธ์ซึ่งมีอินเทอร์เฟส SQL สำหรับการเข้าถึงข้อมูล เวอร์ชันล่าสุดของ MariaDB มีคุณลักษณะ GIS และ JSON ด้วย

MariaDB เปลี่ยนข้อมูลเป็นฐานข้อมูลที่มีโครงสร้างในหลากหลายแอพพลิเคชั่น ตั้งแต่ธนาคารไปจนถึงเว็บไซต์ต่างๆ เป็นการปรับปรุงและแทนที่ด้วยการแทนที่ของ MySQL เนื่องจากมีความรวดเร็วและสามารถปรับขนาดได้และมีระบบแวดล้อมที่อุดมไปด้วยปลั๊กอินเอนจินและเครื่องมืออื่น ๆ ทำให้สามารถใช้งานได้หลากหลาย

MariaDB เป็นระบบจัดการฐานข้อมูลที่เป็น Open Source (ฟรี) มักใช้เป็นอีกทางเลือกทดแทนในการจัดการ MySQL ของแพลตฟอร์มยอดนิยมอย่าง LAMP stack หรือ Linux, Apache, MySQL, PHP/Python/Perl นอกจากนี้ MariaDB ถูกพัฒนาออกมาเพื่อต้องการมาแทนที่ MySQL ในระบบเหล่านี้

นอกจากนี้ยังทำงานบน Windows และ macOS และระบบปฏิบัติการอื่น ๆ อีกมากมาย ใช้แทนระบบฐานข้อมูล MySQL ใน LAMP (Linux + Apache + MariaDB + PHP), LEMP (Linux + Engine-X + MariaDB + PHP) รวมทั้ง XAMPP (Apache + MariaDB + PHP + PHPMyAdmin) ในทุกระบบปฏิบัติการ

**2.2.4 React**

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

**2.2.5 Node.JS**

Node.js คือสภาพแวดล้อมการทำงานของภาษา JavaScript นอกเว็บเบราว์เซอร์ที่ทำงานด้วย V8 engine นั่นหมายความว่าเราสามารถใช้ Node.js ในการพัตนาแอพพลิเคชันแบบ Command line แอพพลิเคชัน Desktop หรือแม้แต่เว็บเซิร์ฟเวอร์ได้ โดยที่ Node.js จะมี APIs ที่เราสามารถใช้สำหรับทำงานกับระบบปฏิบัติการ เช่น การรับค่าและการแสดงผล การอ่านเขียนไฟล์ และการทำงานกับเน็ตเวิร์ก เป็นต้น

Node.js ถูกพัฒนาและทำงานด้วยใช้ Chrome V8 engine สำหรับคอมไพล์ภาษา JavaScript ให้เป็นภาษาเครื่องด้วยการคอมไพล์แบบ Just-in-time (JIT) เพื่อเพิ่มประสิทธิภาพการทำงานของภาษา JavaScript จากที่แต่เดิมมันเป็นภาษาที่มีการทำงานแบบ Interpreted

Node.js เป็นโปรแกรมที่สามารถใช้ได้ทั้งบน Windows, Linux และ Mac OS X นั่นหมายความว่าคุณสามารถเขียนโปรแกรมในภาษา JavaScript และนำไปรันได้ทุกระบบปฏิบัติการที่สนับสนุนโดย Node.js นี่เป็นแนวคิดของการเขียนครั้งเดียวแต่ทำงานได้ทุกที่ (Write once, run anywhere)

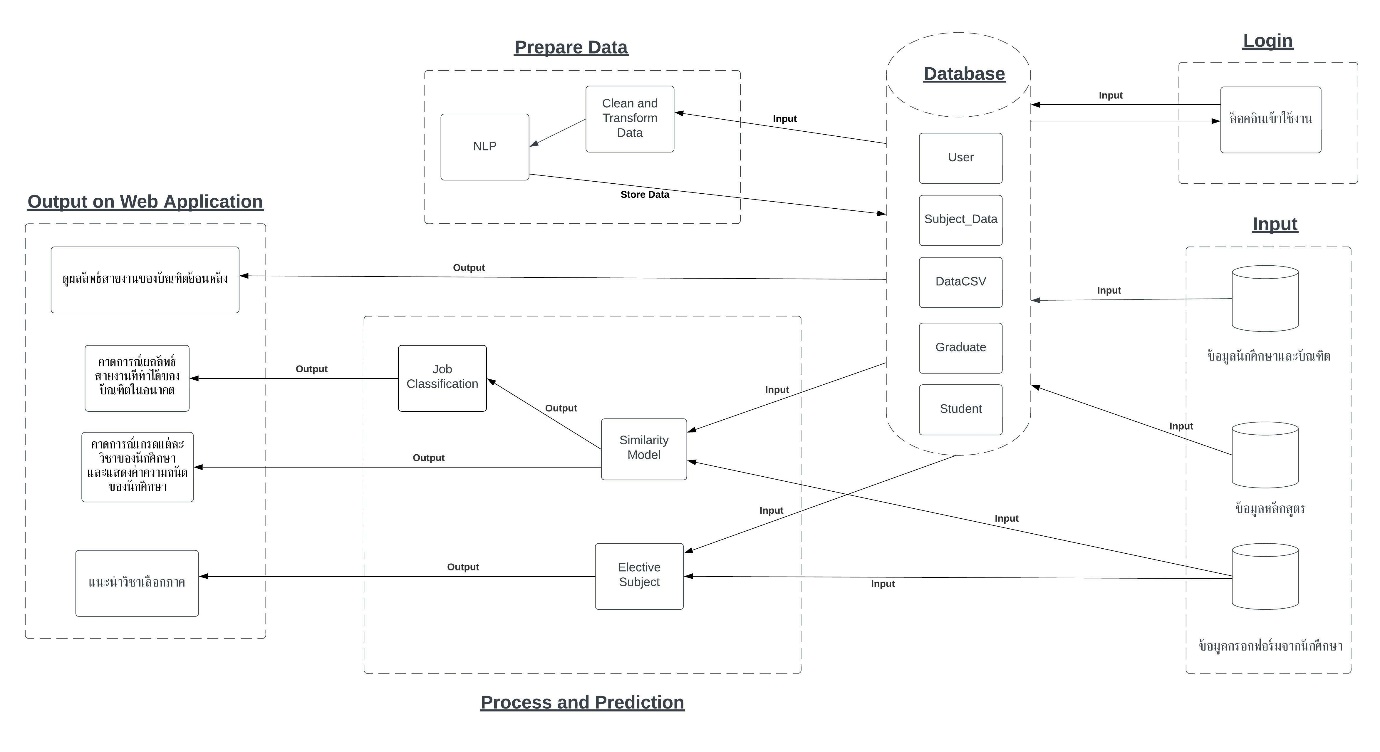
**2.3** งานวิจัยที่เกี่ยวข้อง

**บทที่ 3**

**การออกแบบ**

**3.1 โครงสร้างการทำงานของระบบ**

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

****

**รูป 3.1 โครงสร้างการทำงานของระบบ**

**3.2 การทำงานของระบบ**

จากรูป 1 โครงสร้างการทำงานของระบบนั้นประกอบไปด้วยองค์ประกอบทั้งหมด 6 ส่วน ซึ่งมีรายละเอียด ดังนี้

1. Login

เป็นส่วนสำหรับไว้ให้กรรมการหลักสูตรได้ทำการล๊อคอินเข้าสู่ระบบเพื่อให้กรรมการหลักสูตรป้อนข้อมูลเกรดของนักศึกษาปัจจุบัน เกรดและอาชีพของบัณฑิต และข้อมูลของหลักสูตร

2. Input

เป็นส่วนที่ทำการรับข้อมูลของนักศึกษาและบัณฑิต ข้อมูลของหลักสูตร แล้วเก็บเข้ายังส่วนของ Database และข้อมูลการกรอกฟอร์มของนักศึกษาจะส่งข้อมูลไปยังส่วนของ Process and Prediction โดยตรง

3. Database

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

4. Prepare Data

ทำหน้าที่เตรียมพร้อมข้อมูลเพื่อให้พร้อมต่อการนำไปใช้ในส่วนของ Prediction and Prediction โดยแบ่งออกเป็น 2 ส่วนย่อย ดังนี้

1. Clean and Transform Data

เป็นการเรียกข้อมูลจากใน Database มาทำให้ข้อมูลมีความถูกต้อง ครบถ้วน สมบูรณ์ ไม่มีค่าที่ผิดปกติ เพื่อเตรียมพร้อมสำหรับการประมวลผลข้อมูล โดย เลือกเฉพาะข้อมูลที่เกี่ยวข้องกับวัตถุประสงค์ของการวิเคราะห์ และประมวลผล พร้อมทั้งตัดข้อมูลส่วนที่ไม่ได้นำมาใช้ออก

2. NLP หรือ Natural Language Processing

นำข้อมูลรหัสวิชามาเข้ากระบวนการ NLP เพื่อทำการหาค่า Similarity ของกลุ่ม วิชาที่สามารถอยู่ในกลุ่มเดียวกันได้ โดยใช้บทคัดย่อของแต่ละวิชา เพื่อลดปัญหา การเปลี่ยนรหัสวิชาระหว่างหลักสูตร เมื่อทำเสร็จกระบวนการแล้ว จะนำข้อมูลที่ ได้ กลับไปอัพเดตที่ Database

5. Process and Prediction

เป็นส่วนการประมวลผลหลักของระบบ ประกอบไปด้วย Process 3 ส่วนดังนี้

1. Similarity Model

มีหน้าที่นำข้อมูลผลการเรียนของนักศึกษามาทำให้สมบูรณ์ ด้วยกรณีของนักศึกษา แต่ละคนมีเกรดแต่ละวิชาที่ไม่เหมือนกัน บางคนมีเกรดของวิชานี้ แต่อีกคนไม่มี เกรดของวิชานี้ จึงต้องทำให้ข้อมูลของนักศึกษามีเท่ากันเพื่อนำไปทำการ Prediction ในขั้นต่อไป โดยใช้กระบวนการหลักคือการทำ Recommender Systems และใช้ Library Surprise ของ Scikit ในภาษา Python เป็นตัวช่วย

2. Job Classification

มีหน้าที่ทำนายและสถิติด้านความสามารถทางวิชาชีพของนักศึกษาในอนาคต

3. Elective Subject

มีหน้าที่ประมวลผลจัดกลุ่มของวิชาเลือกภาค เพื่อนำไปแนะนำให้แก่นักศึกษาที่มี ความสนใจเฉพาะจุดได้

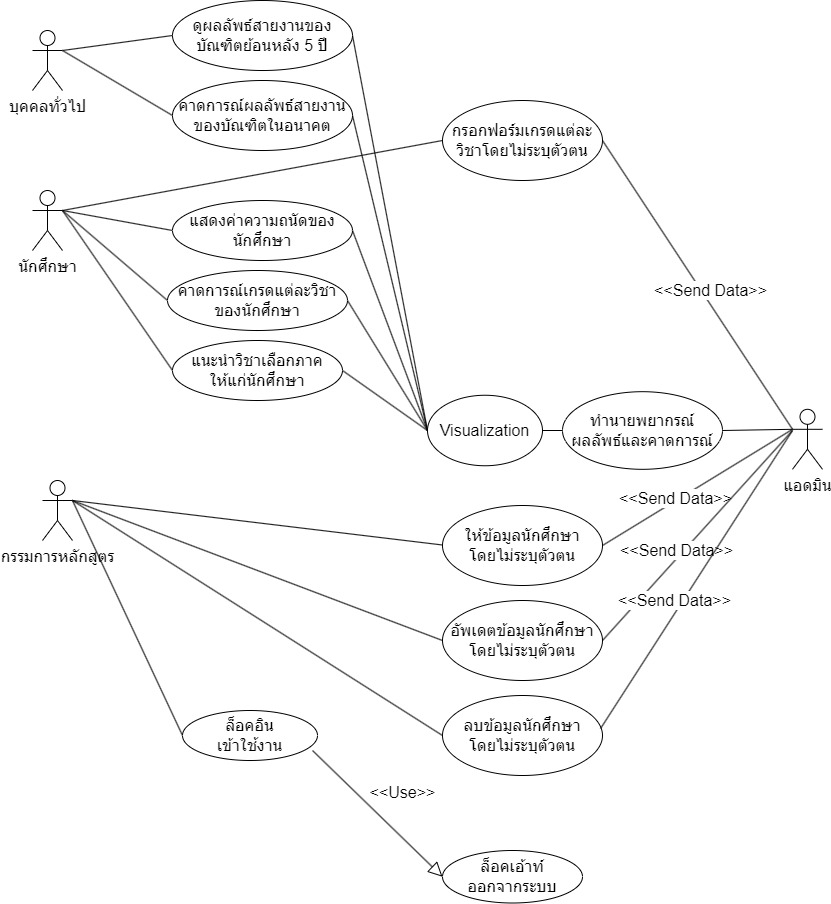
6. Web Application

เป็นส่วนที่ทำหน้าที่ในการแสดงผลข้อมูลและเป็น interface สำหรับผู้ใช้งาน

**3.****3 Use Case Diagram**

การใช้งานระบบจะแบ่งผู้ใช้ออกเป็น 4 ประเภท ได้แก่ บุคคลทั่วไปหรือนักศึกษา กรรมการหลักสูตร และ แอดมิน

1. บุคคลทั่วไปหรือนักศึกษา มีสิทธิ์เข้าถึงการดูข้อมูลการทำนายผล
2. กรรมการหลักสูตร มีสิทธ์เข้าถึงในการส่งไฟล์ข้อมูลของนักศึกษา
3. แอดมิน มีสิทธ์เข้าถึงการทำงานทั้งหมดของระบบวิเคราะห์ และ พยากรณ์

****

**รูป 3.3 Use Case Diagram**

**ตาราง 3.1 Use Case ดูผลลัพธ์สายงานของบัณฑิตย้อนหลัง**

|  |
| --- |
| **Use Case:** ดูผลลัพธ์สายงานของบัณฑิตย้อนหลัง |
| **Use Case ID:** UC-01 |
| **Actor:** บุคคลทั่วไปหรือนักศึกษา |
| **Description:** สามารถดูผลลัพธ์สายงานของบัณฑิตย้อนหลังได้ |
| **Precondition:** เข้า web application หน้าผลลัพธ์สายงานของบัณฑิตย้อนหลัง |
| **Flow of Events:**   1. เลือกหน้าดูผลลัพธ์สายงานของบัณฑิตย้อนหลัง 2. เลือกปีที่ต้องการดู 3. เลือกหลักสูตรที่ต้องการดู |
| **Postcondition:** หน้าเว็บแสดงผลผลลัพธ์สายงานของบัณฑิตย้อนหลัง |

**ตาราง 3.2 Use Case คาดการณ์ผลลัพธ์สายงานที่ทำได้ของบัณฑิตในอนาคต**

|  |
| --- |
| **Use Case:** คาดการณ์ผลลัพธ์สายงานที่ทำได้ของบัณฑิตในอนาคต |
| **Use Case ID:** UC-02 |
| **Actor:** บุคคลทั่วไปหรือนักศึกษา |
| **Description:** สามารถดูผลลัพธ์สายงานที่ทำได้ของบัณฑิตในอนาคตได้ |
| **Precondition:** เข้า web application หน้าผลลัพธ์สายงานที่ทำได้ของบัณฑิตในอนาคต |
| **Flow of Events:**   1. เลือกหน้าดูผลลัพธ์สายงานที่ทำได้ของบัณฑิตในอนาคต 2. เลือกปีที่ต้องการดู 3. เลือกหลักสูตรที่ต้องการดู |
| **Postcondition:** หน้าเว็บแสดงผลผลลัพธ์สายงานที่ทำได้ของบัณฑิตในอนาคต |

**ตาราง 3.3 Use Case กรอกแบบฟอร์มเกรดสำหรับการวิเคราะห์ข้อมูล**

|  |
| --- |
| **Use Case:** กรอกแบบฟอร์มเกรดสำหรับการวิเคราะห์ข้อมูล |
| **Use Case ID:** UC-03 |
| **Actor:** บุคคลทั่วไปหรือนักศึกษา |
| **Description:** สามารถกรอกแบบฟอร์มการวิเคราะห์ข้อมูลได้ |
| **Precondition:** เข้า web application หน้าสำหรับการวิเคราะห์ข้อมูลของนักศึกษา |
| **Flow of Events:**   1. เลือกหน้าสำหรับการวิเคราะห์ข้อมูลของนักศึกษา 2. เลือกปีการศึกษา 3. เลือกหลักสูตร 4. โหลดแบบฟอร์มสำหรับการวิเคราะห์ข้อมูล 5. อัพโหลดแบบฟอร์มสำหรับการวิเคราะห์ข้อมูล |
| **Postcondition:** Redirect ไปหน้าแสดงผลวิเคราะห์ข้อมูลนักศึกษา |

**ตาราง 3.4 Use Case แสดงค่าความถนัดของนักศึกษา**

|  |
| --- |
| **Use Case:** แสดงค่าความถนัดของนักศึกษา |
| **Use Case ID:** UC-04 |
| **Actor:** บุคคลทั่วไปหรือนักศึกษา |
| **Description:** สามารถดูผลแสดงค่าความถนัดของนักศึกษาได้ |
| **Precondition:** เข้า web application หน้าแสดงผลวิเคราะห์ข้อมูลนักศึกษา |
| **Flow of Events:**   1. เลือกหน้าแสดงผลสำหรับการวิเคราะห์ข้อมูลของนักศึกษา |
| **Postcondition:** หน้าเว็บแสดงผลผลลัพธ์การวิเคราะห์ข้อมูลค่าความถนัดของนักศึกษา |

**ตาราง 3.5 Use Case คาดการณ์เกรดแต่ละวิชาของนักศึกษา**

|  |
| --- |
| **Use Case:** คาดการณ์เกรดแต่ละวิชาของนักศึกษา |
| **Use Case ID:** UC-05 |
| **Actor:** บุคคลทั่วไปหรือนักศึกษา |
| **Description:** สามารถดูผลแสดงค่าคาดการณ์เกรดแต่ละวิชาของนักศึกษาได้ |
| **Precondition:** เข้า web application หน้าแสดงผลวิเคราะห์ข้อมูลนักศึกษา |
| **Flow of Events:**   1. เลือกหน้าแสดงผลสำหรับการวิเคราะห์ข้อมูลของนักศึกษา |
| **Postcondition:** หน้าเว็บแสดงผลผลลัพธ์การวิเคราะห์ข้อมูลค่าคาดการณ์เกรดแต่ละวิชาของนักศึกษา |

**ตาราง 3.6 Use Case แนะนำวิชาเลือกภาคให้แก่นักศึกษา**

|  |
| --- |
| **Use Case:** แนะนำวิชาเลือกภาคให้แก่นักศึกษา |
| **Use Case ID:** UC-06 |
| **Actor:** บุคคลทั่วไปหรือนักศึกษา |
| **Description:** สามารถดูผลแนะนำวิชาเลือกให้ภาคให้แก่นักศึกษาได้ |
| **Precondition:** เข้า web application หน้าแนะนำวิชาเลือกภาค |
| **Flow of Events:**   1. เลือกหน้าแนะนำวิชาเลือกภาค 2. เลือกสิ่งที่สนใจภายในตัวเลือกที่มีให้ |
| **Postcondition:** หน้าเว็บแนะนำวิชาเลือกภาค |

**ตาราง 3.7 Use Case ให้ข้อมูลนักศึกษาโดยไม่ระบุตัวตนและข้อมูลหลักสูตร**

|  |
| --- |
| **Use Case:** ให้ข้อมูลนักศึกษาโดยไม่ระบุตัวตนและข้อมูลหลักสูตร |
| **Use Case ID:** UC-07 |
| **Actor:** กรรมการหลักสูตร |
| **Description:** ให้ข้อมูลนักศึกษาโดยไม่ระบุตัวตนและข้อมูลหลักสูตรได้ |
| **Precondition:** เข้า web application หน้าให้ข้อมูลนักศึกษาโดยไม่ระบุตัวตนและข้อมูลหลักสูตร |
| **Flow of Events:**   1. เลือกหน้าให้ข้อมูลนักศึกษาโดยไม่ระบุตัวตนและข้อมูลหลักสูตร 2. อัพโหลดไฟล์ CSV 3. กดปุ่มอัพโหลด |
| **Postcondition:** อัพโหลดข้อมูลสำเร็จ |

**ตาราง 3.8 Use Case อัพเดตข้อมูลนักศึกษาโดยไม่ระบุตัวตนและข้อมูลหลักสูตร**

|  |
| --- |
| **Use Case:** อัพเดตข้อมูลนักศึกษาโดยไม่ระบุตัวตนและข้อมูลหลักสูตร |
| **Use Case ID:** UC-08 |
| **Actor:** กรรมการหลักสูตร |
| **Description:** อัพเดตนักศึกษาโดยไม่ระบุตัวตนและข้อมูลหลักสูตรได้ |
| **Precondition:** เข้า web application หน้าให้ข้อมูลนักศึกษาโดยไม่ระบุตัวตนและข้อมูลหลักสูตร |
| **Flow of Events:**   1. เลือกหน้าให้ข้อมูลนักศึกษาโดยไม่ระบุตัวตนและข้อมูลหลักสูตร 2. อัพเดตไฟล์ CSV 3. กดปุ่มอัพโหลด |
| **Postcondition:** อัพเดตข้อมูลสำเร็จ |

**ตาราง 3.9 Use Case ลบข้อมูลนักศึกษาโดยไม่ระบุตัวตนและข้อมูลหลักสูตร**

|  |
| --- |
| **Use Case:** อัพเดตข้อมูลนักศึกษาโดยไม่ระบุตัวตนและข้อมูลหลักสูตร |
| **Use Case ID:** UC-09 |
| **Actor:** กรรมการหลักสูตร |
| **Description:** ลบนักศึกษาโดยไม่ระบุตัวตนและข้อมูลหลักสูตรได้ |
| **Precondition:** เข้า web application หน้าให้ข้อมูลนักศึกษาโดยไม่ระบุตัวตนและข้อมูลหลักสูตร |
| **Flow of Events:**   1. เลือกหน้าให้ข้อมูลนักศึกษาโดยไม่ระบุตัวตนและข้อมูลหลักสูตร 2. ลบไฟล์ CSV 3. กดปุ่มยืนยัน |
| **Postcondition:** ลบข้อมูลสำเร็จ |

**ตาราง 3.10 Use Case ล็อคอินเข้าใช้งาน**

|  |
| --- |
| **Use Case:** ให้ข้อมูลนักศึกษาโดยไม่ระบุตัวตนและข้อมูลหลักสูตร |
| **Use Case ID:** UC-10 |
| **Actor:** กรรมการหลักสูตร |
| **Description:** กรรมการหลักสูตรสามารถล็อคอินเข้าใช้งานได้ |
| **Precondition:** เข้า web application หน้า login |
| **Flow of Events:**   1. กรอกข้อมูล username และ password 2. กดปุ่ม Login |
| **Postcondition:** Redirect ไปหน้าหลัก |

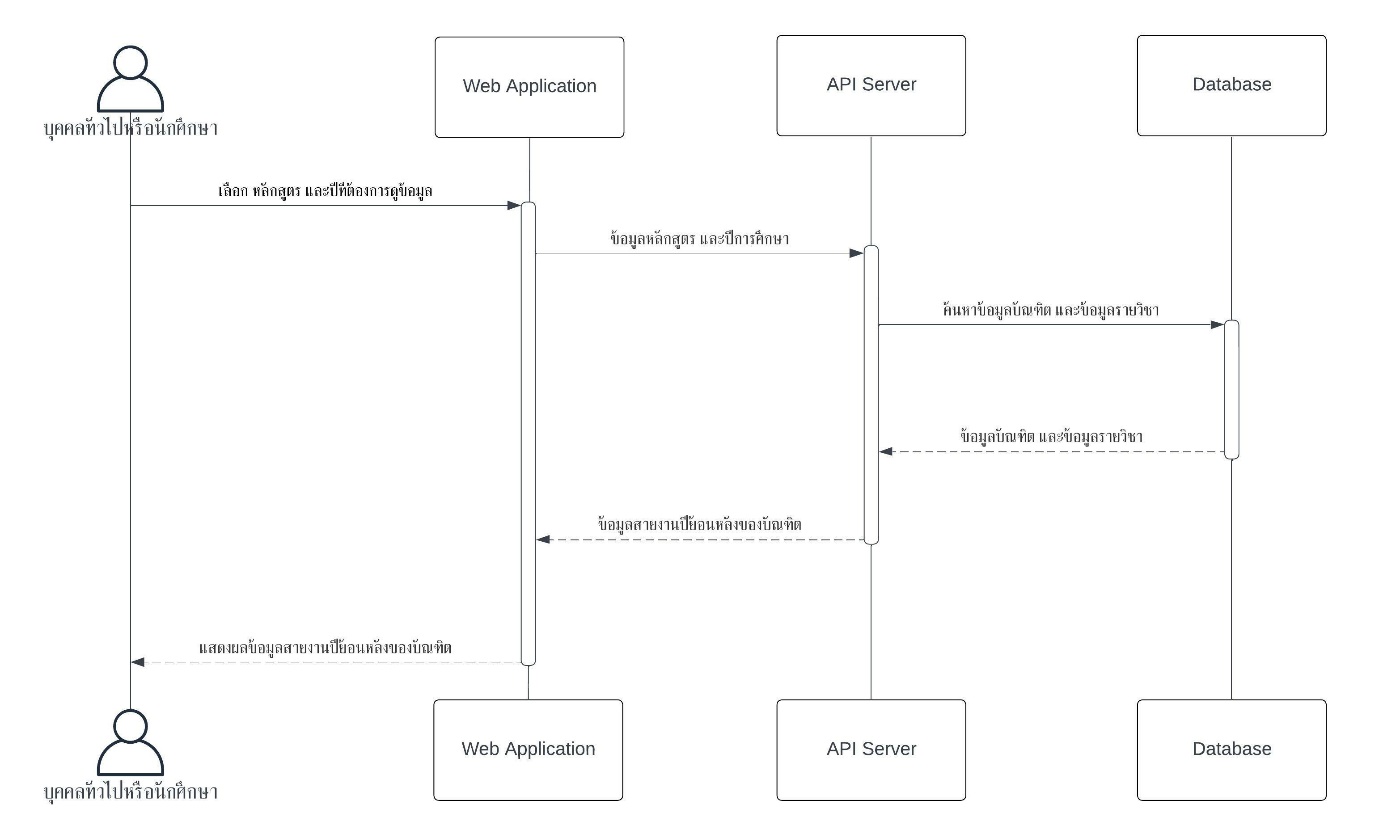
**ตาราง 3.11 Use Case ทำนายพยากรณ์ผลลัพธ์และคาดการณ์**

|  |
| --- |
| **Use Case:** ทำนายพยากรณ์ผลลัพธ์และคาดการณ์ |
| **Use Case ID:** UC-11 |
| **Actor:** แอดมิน |
| **Description:** แอดมินสามารถทำนายพยากรณ์ผลลัพธ์และคาดการณ์ได้ |
| **Precondition:** เป็นแอดมิน |
| **Flow of Events:**   1. ทำการทำนายพยากรณ์และคาดการณ์ 2. ส่งออกผลลัพธ์ไปยังหน้าเว็บ |
| **Postcondition:** ทำนายพยากรณ์ผลลัพธ์และคาดการณ์ได้สำเร็จ |

**ตาราง 3.12 Use Case จัดเตรียมข้อมูล**

|  |
| --- |
| **Use Case:** จัดเตรียมข้อมูล |
| **Use Case ID:** UC-12 |
| **Actor:** แอดมิน |
| **Description:** จัดเตรียมข้อมูลสำหรับการทำนายและคาดการณ์ได้ |
| **Precondition:** เป็นแอดมิน |
| **Flow of Events:**   1. นำเข้าข้อมูล 2. จัดรูปแบบข้อมูล 3. เก็บเข้าฐานข้อมูล |
| **Postcondition:** จัดเตรียมข้อมูลสำหรับการทำนายและคาดการณ์ได้สำเร็จ |

**3.4 Sequence Diagram**

****

**รูป 3.4 Sequence Diagram สำหรับ UC-01**

**Diagram

Description automatically generated**

**รูป 3.5 Sequence Diagram สำหรับ UC-02**

**Diagram

Description automatically generated**

**รูป 3.6 Sequence Diagram สำหรับ UC-03**

**Diagram

Description automatically generated**

**รูป 3.7 Sequence Diagram สำหรับ UC-04**

**Diagram

Description automatically generated**

**รูป 3.8 Sequence Diagram สำหรับ UC-05**

**Diagram

Description automatically generated**

**รูป 3.9 Sequence Diagram สำหรับ UC-06**

**Diagram

Description automatically generated**

**รูป 3.10 Sequence Diagram สำหรับ UC-07**

**Diagram

Description automatically generated**

**รูป 3.11 Sequence Diagram สำหรับ UC-08**

**Diagram

Description automatically generated**

**รูป 3.12 Sequence Diagram สำหรับ UC-09**

**Diagram

Description automatically generated**

**รูป 3.13 Sequence Diagram สำหรับ UC-10**

**Diagram

Description automatically generated**

**รูป 3.14 Sequence Diagram สำหรับ UC-11**

**Diagram

Description automatically generated**

**รูป 3.15 Sequence Diagram สำหรับ UC-12**

* 1. **การออกแบบฐานข้อมูล**

ระบบฐานข้อมูลของโครงงานนี้ผู้จัดทำได้เลือกใช้ MariaDB ซึ่งเป็นฐานข้อมูลแบบ SQL ที่ถูกพัฒนามาเพื่อสำหรับเก็บข้อมูล โดยฐานข้อมูลของระบบประกอบไปด้วยทั้งหมด 5 ตาราง ดังรูป

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 Graduate

เป็นตารางที่ใช้เก็บข้อมูลบัณฑิตโดยจะมี Attribute ดังนี้

1. student\_id : UUID สำหรับเก็บ id encrypt ของบัณฑิต

2. subject\_id : VARCHAR สำหรับเก็บ id ของรหัสวิชา

3. grade : VARCHAR สำหรับเก็บเกรดแต่ละรายวิชา

4. semester : VARCHAR สำหรับเก็บเทอมที่ลงเรียนรายวิชาของบัณฑิต

5. year : VARCHAR สำหรับเก็บปีที่ลงเรียนรายวิชาของบัณฑิต

6 curriculum : VARCHAR สำหรับเก็บรายชื่อหลักสูตรที่เรียนของบัณฑิต

7. career : VARCHAR สำหรับเก็บรายชื่อตำแหน่งอาชีพแรกของบัณฑิต

3.5.4 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

3.5.5 ตาราง DataCSV

เป็นตารางที่ใช้เก็บข้อมูลบัณฑิตโดยจะมี Attribute ดังนี้

1. data\_id : UUID สำหรับเก็บ id ของไฟล์ CSV

2. name : VARCHAR สำหรับเก็บชื่อของไฟล์ CSV

3. upload\_date : TIMESTAMP สำหรับเก็บเวลาและวันที่อัพโหลดไฟล์ CSV

4. update\_date : TIMESTAMP สำหรับเก็บเวลาและวันที่อัพเดตไฟล์ CSV

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

**บทที่ 4**

**ผลการดำเนินงาน**

ในบทนี้จะอธิบายถึงการทดลองที่ได้ทำในภาคการศึกษานี้ เพื่อแสดงให้เห็นถึงความก้าวหน้า ในโครงงาน