**สารบัญ**

**หน้า**

**บทที่ 1** [**บทนำ** **ผิดพลาด! ไม่ได้กำหนดบุ๊กมาร์ก**](#_Toc134451445)

[**บทที่ 2 เอกสารและงานวิจัยทีเกี่ยวข้อง** 4](#_Toc134451446)

**สารบัญภาพ**

**ภาพที่ หน้า**

**สารบัญตาราง**

**ตารางที่ หน้า**

1.1 แผนการดำเนินงาน 3

2.1 ข้อดีข้อเสียของการเขียนโปรแกรมเชิงวัตถุ 5

**บทที่ 1**

# **บทนำ**

โครงงานพัฒนาระบบจัดการข้อมูลสาขาวิทยาการคอมพิวเตอร์ มหาวิทยาลัยราชภัฏกาญจนบุรีเป็นการพัฒนาระบบขึ้นมาใหม่ ผู้วิจัยได้ทำการศึกษาข้อมูลการจัดทำเว็บแอปพลิเคชันโดยใช้เทคโนโลยี React ในการทำ Frontend หน้าเว็บแอปพลิเคชันและเทคโนโลยี ASP.NET Core ในการทำ Backend สำหรับบริหารจัดการข้อมูล ในการพัฒนาจะมีการแบ่งการทำงานที่เป็นสัดส่วน จัดการฐานข้อมูลด้วยโปรแกรม Microsoft SQL Server Management Studio 19 ตกแต่งหน้าเว็บด้วย Tailwind CSS Framework และทำให้ผู้ใช้ภายนอกสามารถเข้าถึงเว็บไซด์ได้ผ่านเว็บเซิร์ฟเวอร์ Windows Server 2022 โดยในบทความนี้จะกล่าวถึงกระบวนการพัฒนาระบบ มีรายละเอียดดังนี้

**1. ที่มาและความสำคัญของงานวิจัย**

ในอดีตมหาวิทยาลัยราชภัฏกาญจนบุรี มีระบบจัดการข้อมูลที่เกี่ยวกับระบบงานสาขาวิทยาการคอมพิวเตอร์ แต่ในปัจจุบันระบบดังกล่าวได้เกิดข้อผิดพลาด และเกิดการสูญหาย ระบบและข้อมูลเดิมหายไปทำให้ไม่มีระบบในการบริหารจัดการข้อมูล เช่น ข้อมูลนักศึกษา ข้อมูลงานวิจัย ข้อมูลหลักสูตร จึงทำให้เกิดปัญหา คือ ไม่มีแหล่งกระจายข้อมูลข่าวสาร ทั้งนี้เนื่องจากทุกปีต้องมีการรับสมัครนักศึกษาใหม่ผ่านระบบออนไลน์ โดยใช้ข้อมูลของสาขาเป็นสื่อในการประชาสัมพันธ์ต่าง ๆ

ดังนั้นผู้วิจัยจึงมีความต้องการที่จะพัฒนาระบบจัดการบริหารข้อมูลสาขา และให้มีกระบวนการในการทำงานที่เป็นระบบ สามารถแบ่งออกเป็น 5 ระบบ ประกอบไปด้วย 1) ระบบจัดการข้อมูลผู้ใช้ในระบบ 2) ระบบจัดการงานวิจัย 3) ระบบประชาสัมพันธ์ข้อมูลสาขา 4) ระบบจัดการหลักสูตรอบรมระยะสั้น 5) ระบบรายงานข้อมูล ซึ่งจะช่วยลดภาระการทำงาน ลดความซับซ้อนของข้อมูล และเพิ่มประสิทธิภาพในการทำงาน เพื่อเป็นประโยชน์สูงสุดแก่บุคคลทั่วไป

**2. วัตถุประสงค์ของการวิจัย**

2.1 เพื่อพัฒนาระบบบริหารจัดการข้อมูลสาขาวิทยาการคอมพิวเตอร์

2.2 เพื่อประชาสัมพันธ์สาขาวิทยาการคอมพิวเตอร์

**3. ขอบเขตการวิจัย**

3.1 ผู้ใช้งานทั่วไป

1) สืบค้นข้อมูลงานวิจัย

2) ระบบลงทะเบียนเข้าอบรมหลักสูตรระยะสั้น

3) สัดส่วนหลักสูตรสาขาวิทยาการคอมพิวเตอร์ มหาวิทยาลัยราชภัฏกาญจนบุรี

4) แสดงข่าวสารประชาสัมพันธ์ของสาขา

3.2 นักศึกษา

1) ระบบจัดการข้อมูลงานวิจัย

2) ระบบลงทะเบียนเข้าอบรมหลักสูตรระยะสั้น

3) สัดส่วนหลักสูตรสาขาวิทยาการคอมพิวเตอร์ มหาวิทยาลัยราชภัฏกาญจนบุรี

4) แสดงข่าวสารประชาสัมพันธ์ของสาขา

5) สืบค้นข้อมูลงานวิจัย

6) ประวัติการทำงาน

3.3 อาจารย์

1) สืบค้นข้อมูลงานวิจัย

2) ระบบจัดการหลักสูตรอบรมระยะสั้น

3) สัดส่วนหลักสูตรสาขาวิทยาการคอมพิวเตอร์ มหาวิทยาลัยราชภัฏกาญจนบุรี

4) แสดงข่าวสารประชาสัมพันธ์ของสาขา

3.4 ผู้ดูแลระบบ

1) ระบบจัดการข้อมูลผู้ใช้ในระบบ

2) ระบบจัดการประชาสัมพันธ์

3) รายงานข้อมูลผู้ใช้ในระบบ

4) รายงานสัดส่วนการทำงานของศิษย์เก่า

**4. ประโยชน์ที่คาดว่าจะได้รับ**

4.1 ทำให้การจัดการข้อมูลในสาขาวิทยาการคอมพิวเตอร์เป็นระเบียบและจัดการได้ดีขึ้น

4.2 ได้ประชาสัมพันธ์ข่าวสารของสาขาวิทยาการคอมพิวเตอร์

**5. อุปกรณ์ในการทำวิจัย**

5.1 ฮาร์ดแวร์ที่ใช้

1) คอมพิวเตอร์โน้ตบุ๊ก จำนวน 1 เครื่อง

- AMD Ryzen 5 4000 Series

- RAM 16 GB

- SSD 512 GB

5.2 ซอฟต์แวร์ที่ใช้

1) โปรแกรมที่ใช้พัฒนาเว็บแอปพลิเคชัน Visual Studio Code

2) โปรแกรมที่ใช้จัดการฐานข้อมูล Microsoft SQL Server Management Studio 19

3) ภาษาที่ใช้ในการพัฒนาระบบ

- C# โดยใช้ Framework ASP.NET WebAPI

- Typescript โดยใช้ React Library

**6. แผนการดำเนินงาน**

**ตารางที่ 1.1** แผนการดำเนินงาน

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **กิจกรรม** | **ระยะเวลาดำเนินงาน** | | | |
| ธันวาคม 65 | มกราคม 66 | กุมภาพันธ์ 66 | มีนาคม 66 |
| 1.เสนอหัวข้อโครงงาน |  |  |  |  |
| 2.ศึกษาระบบงานและเอกสารที่เกี่ยวข้อง |  |  |  |  |
| 3.การวิเคราะห์ระบบและออกแบบระบบ |  |  |  |  |
| 4.พัฒนาระบบ และ ทดสอบระบบ |  |  |  |  |
| 5.ให้ผู้เชี่ยวชาญประเมินผล |  |  |  |  |
| 6.สรุปผลการดำเนินงาน |  |  |  |  |
| 7.จัดทำรูปเล่ม |  |  |  |  |

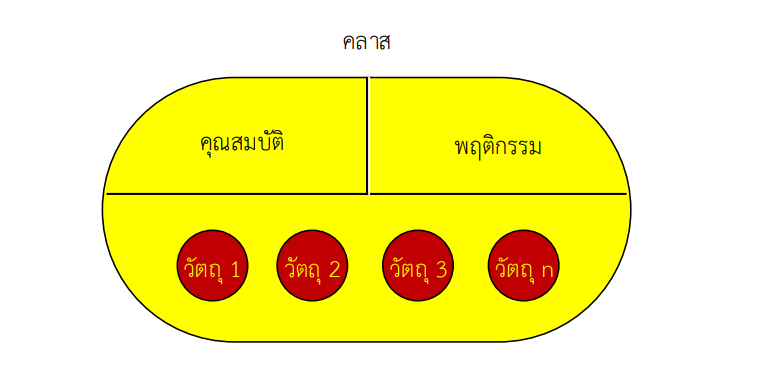
**บทที่ 2**

# **เอกสารและงานทฤษฎีทีเกี่ยวข้อง**

การจัดทำโครงการ โครงงานพัฒนาระบบจัดการข้อมูลสาขาวิทยาการคอมพิวเตอร์ มหาวิทยาลัยราชภัฏกาญจนบุรี ในการพัฒนาระบบจัดการข้อมูลนั้นต้องอาศัยหลักการและทฤษฎีต่าง ๆ รวมทั้งศึกษา แนวทางในการพัฒนาจากงานวิจัยที่เคยมีการพัฒนามาแล้วไปประยุกต์ใช้ในการพัฒนาระบบต่อไป ดังนั้นในบทนี้กล่าวถึง ทฤษฎีที่เกี่ยวข้องกับการพัฒนาระบบงาน โดยมุ่งเน้นการอธิบายทฤษฎีที่สำคัญ และเครื่องมือที่ทันสมัยสำหรับการพัฒนาซอฟต์แวร์โดยมีทั้งหมด 8 หัวข้อดังต่อไปนี้

**1. การเขียนโปรแกรมเชิงวัตถุ (Object Oriented Programming)**

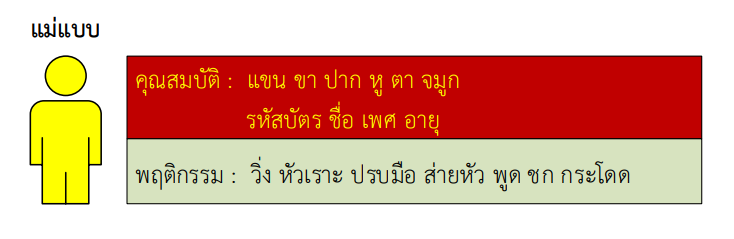
การเขียนโปรแกรมเชิงวัตถุ [1] หมายถึง การมองการเขียนโปรแกรมเป็นลักษณะของวัตถุมากกว่าการกระทำ การเขียนโปรแกรมเชิงวัตถุจะแยกปัญหาหรือระบบงานออกเป็นส่วน เพื่อลดความซับซ้อนให้น้อยลง โดยมุ่งที่จะเพิ่มประสิทธิภาพในการสร้างโปรแกรม ทำให้การพัฒนาโปรแกรมด้วยวิธีนี้ใช้เวลาสั้นลงและต้นทุนต่ำลง โดยการนำชุดคำสั่งที่สร้างไปแล้วกลับมาใช้งานใหม่ได้ และยังหาจุดที่ต้องการแก้ไขในโปรแกรมได้ง่ายขึ้น เป็นวิธีการเขียนโปรแกรมที่นำแนวคิดในความเป็นจริงที่อยู่รอบตัวมาใช้กับการเขียนโปรแกรม

 การเขียนโปรแกรมเชิงวัตถุ มีแนวคิดและองค์ประกอบมาจากความหมายของเชิงวัตถุประกอบไปด้วย 4 ส่วนสำคัญ ได้แก่ คลาส วัตถุ คุณสมบัติ และพฤติกรรม โดยมีรายละเอียดดังภาพที่ 2.1

**ภาพที่ 2.1** ส่วนประกอบของวัตถุ

จากภาพที่ 2.1 ส่วนประกอบของวัตถุ โดยแนวคิดประกอบไปด้วยคลาส ภายใน คลาสนั้นจะประกอบด้วยคุณสมบัติและพฤติกรรม กำหนดคุณลักษณะของวัตถุวัตถุใดที่มีคุณสมบัติ หรือพฤติกรรมเดียวกับคลาส โดยส่วนประกอบของวัตถุมีรายละเอียดดังต่อไปนี้

1.1 คลาส (Class)

 คลาส หมายถึงศัพท์ที่อธิบายโครงร่างหรือพิมพ์เขียวที่กำหนดขอบเขตกลุ่มของวัตถุ มีความหมายและคำจำกัดความกลุ่มของวัตถุกลุ่มใดกลุ่มหนึ่งมีรายละเอียดดังต่อไปนี้

**ภาพที่ 2.2** คลาส

จากภาพที่ 2.2 ตัวอย่างคลาสมนุษย์ เมื่อมองในความหมายขอบเขตของมนุษย์ จะเป็นลักษณะของสิ่งมีชีวิตที่เป็นรูปธรรม เช่น แขน ขา หู ตา จมูก เป็นต้น หรือคุณสมบัติด้านนามธรรม เช่น ชื่อ อายุ เพศ น้ำหนัก ส่วนสูง เป็นต้น ในลักษณะของคลาสนั้นบ่งบอกลักษณะภายในตัวของคลาส และในส่วนที่สองเป็นส่วนของการมองในส่วนของคลาสนั้นสามารถเกิดการกระทำ จากตัวอย่าง จากภาพที่ 2.2 สามารถที่จะกระทำหรือแสดงพฤติกรรมออกมาได้ เช่น วิ่ง หัวเราะ ปรบมือ พูด กระโดด เป็นต้น ซึ่งเมื่อมองในความหมายภาพรวมเห็นได้ว่าเป็นลักษณะของมุมมองที่เป็นนามธรรมที่กำหนดขอบเขตของกลุ่มใดกลุ่มหนึ่ง

1.2 คุณสมบัติ (Properties)

 คุณสมบัติ หมายถึงองค์ประกอบภายในคลาส ใช้ในการกำหนดข้อมูลที่เก็บภายในคลาสหรือส่วนที่กำหนดคุณลักษณะข้อกำหนดของคลาสนั้น แต่ละคลาสจะกำหนดคุณสมบัติที่ซึ่งกำหนดสถานะของวัตถุที่เกิดจากคลาสนั้น มีรายละเอียดดังต่อไปนี้

**ภาพที่ 2.3** คุณสมบัติ

จากภาพที่ 2.3 ตัวอย่างคุณสมบัติที่เกิดจากคลาสมนุษย์ มนุษย์จะมีคุณสมบัติภายในคลาสของมนุษย์นั้น เช่น แขน ขา ชื่อ รหัสบัตร ที่อยู่ อายุ เพศ เป็นต้น คุณสมบัติเป็นองค์ประกอบที่สำคัญของคลาส โดยจะกำหนดคุณลักษณะที่มีของคลาสนั้น คุณสมบัติบางอย่างจะส่งผลต่อเนื่องไปยังพฤติกรรมที่เกิดขึ้นของคลาส

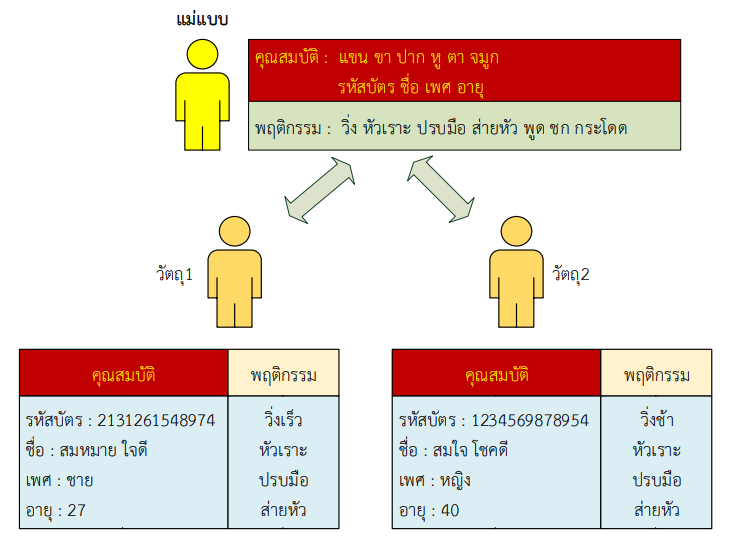
1.3 พฤติกรรม (Behavior)

 พฤติกรรม หรือการกระทำ (Action) หมายถึงองค์ประกอบที่อยู่ภายในคลาสใช้ในการกำหนดการกระทำบางอย่างของคลาส โดยพฤติกรรมเป็นการกำหนดขอบเขตที่สามารถกระทำได้ของคลาส ส่วนมากจะเกิดจากคุณสมบัติที่มีอยู่ในคลาส มีรายละเอียดดังต่อไปนี้

**ภาพที่ 2.4** พฤติกรรม

จากภาพที่ 2.4 ตัวอย่างพฤติกรรมที่เกิดจากคลาสมนุษย์ พฤติกรรมมักเกิดจากคุณสมบัติที่มีภายในคลาส เช่น คุณสมบัติมีขาจะสามารถวิ่งได้ ปากสามารถพูดได้ เป็นต้น พฤติกรรมเป็นกระบวนการที่เกิดขึ้นตั้งแต่นำเข้าข้อมูล ประมวลผลและเกิดผลลัพธ์

1.4 วัตถุ (Object)

 วัตถุ หมายถึงการมองสิ่งที่อยู่รอบข้างเป็นวัตถุเป้าหมาย ภายในวัตถุจะประกอบไปด้วยคุณสมบัติของวัตถุและพฤติกรรมหรือการกระทำ ที่ตอบสนองต่อเหตุการณ์ที่จะทำให้เกิดขึ้นของวัตถุ มีรายละเอียดดังต่อไปนี้

**ภาพที่ 2.5** วัตถุ

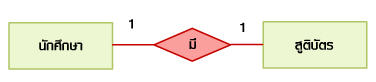
จากภาพที่ 2.5 วัตถุ1 และวัตถุ2 จะมีคุณสมบัติเหมือนกัน คือ รหัสบัตร ชื่อ เพศและอายุ แต่ข้อมูลที่อยู่ภายในคุณสมบัตินี้ จะมีความแตกกต่างกันที่บ่งบอกคุณลักษณะเฉพาะของวัตถุ

**2. ระบบฐานข้อมูล (Database System)**

ระบบฐานข้อมูล [2] คือ ระบบที่รวบรวมข้อมูลต่าง ๆ ที่เกี่ยวข้องกันเข้าไว้ด้วยกันอย่างมีระบบมีความสัมพันธ์ระหว่างข้อมูลต่าง ๆ ที่ชัดเจน ในระบบฐานข้อมูลจะประกอบด้วยแฟ้มข้อมูลหลายแฟ้มที่มีข้อมูล และเปิดโอกาสให้ผู้ใช้สามารถใช้งานและดูแลรักษาป้องกันข้อมูลเหล่านี้ ได้อย่างมีประสิทธิภาพ โดยมีซอฟต์แวร์ที่เปรียบเสมือนสื่อกลางระหว่าง ผู้ใช้และโปรแกรมต่าง ๆ ที่เกี่ยวข้องกับการใช้ฐานข้อมูล เรียกว่า ระบบจัดการฐานข้อมูล หรือ DBMS (Database Management System) มีหน้าที่ช่วยให้ผู้ใช้เข้าถึงข้อมูลได้ง่าย สะดวกและมีประสิทธิภาพ การเข้าถึงข้อมูลของผู้ใช้อาจเป็นการสร้างฐานข้อมูล การแก้ไขฐานข้อมูล หรือการตั้งคำถามเพื่อให้ได้ข้อมูลมา โดยผู้ใช้ไม่จำเป็นต้องรับรู้เกี่ยวกับรายละเอียดภายในโครงสร้างของฐานข้อมูล

ความสัมพันธ์ [3] ของระบบฐานข้อมูล หมายถึง ความสัมพันธ์ระหว่างเอนทิตี้ เช่น ความสัมพันธ์ระหว่างนักศึกษา และวิชา เป็นลักษณะว่านักศึกษาแต่ละคนเรียนอยู่คณะวิชาหนึ่ง เป็นต้น ความสัมพันธ์แบ่งออกเป็น 3 ประเภทคือ

2.1 ความสัมพันธ์แบบ หนึ่งต่อหนึ่ง (One to one Relationships)

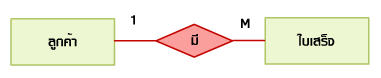
 เป็นการแสดงความสัมพันธ์ของข้อมูล ของเอนทิตี้หนึ่งว่ามีความสัมพันธ์กับข้อมูลในอีกเอนทิตี้หนึ่งในลักษณะ หนึ่ง ต่อ หนึ่ง หรือเขียนได้เป็น 1:1 แสดงว่าข้อมูลเพียงหนึ่งข้อมูลของเอนทิตี้แรกมีความสัมพันธ์กับข้อมูลในเอนทิตี้ที่สองเพียงข้อมูลเดียว ดังรูปภาพที่ 2.6

**ภาพที่ 2.6** ความสัมพันธ์แบบหนึ่งต่อหนึ่ง

จากภาพที่ 2.6 นักศึกษาหนึ่งคนจะมีสูติบัตรเพียงใบเดียวเท่านั้น และสูติบัตรหนึ่งใบจะเป็นของนักศึกษาได้เพียงคนเดียวเท่านั้นเช่นกัน

2.2 ความสัมพันธ์แบบ หนึ่งต่อกลุ่ม (One to many Relationships)

เป็นการแสดงความสัมพันธ์ของข้อมูล ของเอนทิตี้หนึ่งว่ามีความสัมพันธ์กับข้อมูลในอีกเอนทิตี้หนึ่งในลักษณะ หนึ่ง ต่อ กลุ่ม หรือเขียนได้เป็น 1:M แสดงว่าข้อมูลเพียงหนึ่งข้อมูลของเอนทิตี้แรกมีความสัมพันธ์กับข้อมูลในเอนทิตี้ที่สองหลายข้อมูล ดังรูปภาพที่ 2.7

**ภาพที่ 2.7** ความสัมพันธ์แบบหนึ่งต่อกลุ่ม

จากภาพที่ 2.7 ลูกค้าหนึ่งคนมีใบเสร็จได้หลายใบ เนื่องจากลูกค้าหนึ่งคนสามารถซื้อสินค้าได้หลายครั้ง และใบเสร็จหนึ่งใบต้องเป็นของลูกค้าเพียงคนเดียวเท่านั้น

2.3 ความสัมพันธ์แบบ กลุ่มต่อกลุ่ม (Many to Many Relationships)

 เป็นการแสดงความสัมพันธ์ของข้อมูล ของเอนทิตี้หนึ่งว่ามีความสัมพันธ์กับข้อมูลในอีกเอนทิตี้หนึ่งในลักษณะ กลุ่ม ต่อ กลุ่ม หรือเขียนได้เป็น M:M แสดงว่าข้อมูลเพียงหนึ่งข้อมูลของเอนทิตี้แรก มีความสัมพันธ์กับข้อมูลในเอนทิตี้ที่สองหลายข้อมูล และทำนองเดียวกัน ข้อมูลเพียงหนึ่งข้อมูลของเอนทิตี้ที่สองมีความสัมพันธ์ในทางกลับกันกับข้อมูลในเอนทิตี้ที่แรกหลายข้อมูล ดังรูปภาพที่ 2.8

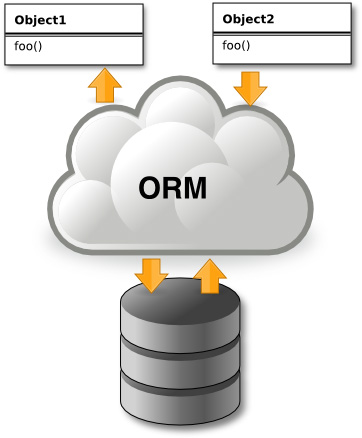
**ภาพที่ 2.8** ความสัมพันธ์แบบกลุ่มต่อกลุ่ม

จากภาพที่ 2.8 นักศึกษาสามารถลงทะเบียนเรียนได้หลายวิชาพร้อมกันในแต่ละครั้ง และในทางตรงกันข้าม ในแต่ละวิชา 1 วิชา จะประกอบด้วยนักศึกษาหลายคนมาลงทะเบียนในวิชานั้น

**3. โออาเอ็ม (ORM)**

ORM (Object / Relational Mapping) [4] หมายถึง การ Map ข้อมูลในตารางข้อมูลของฐานข้อมูลให้อยู่ในรูปของ Object-Oriented Language ซึงจะเป็นการสร้าง Database แบบเสมือนขึ้นให้มาอยู่ในรูป Language Programming ทำให้ไม่ต้องไปยุ่งกับ SQL Statement ซึ่งถ้ามีการเปลี่ยนแปลงแก้ไขต้องแก้ไขที่ตัวโปรแกรมแทน และการกระทำต่าง ๆ ยังคงเป็นแบบ Relational แบบเดิม เช่น มีตารางชื่อ Product จะแปลงมาเป็น Class Product คุณลักษณะ (Attributes) ของตาราง Product ก็จะถูกแปลงมาเป็น Object ใน Class

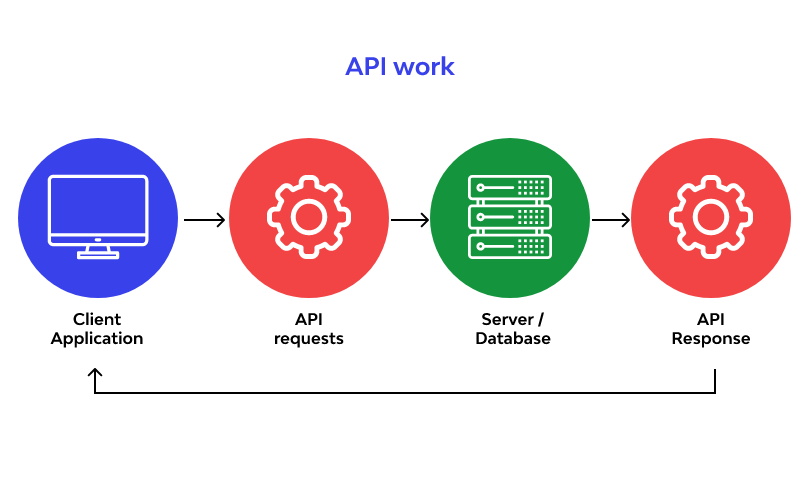
ผู้ใช้ไม่จำเป็นต้องยุ่งในส่วนของ SQL (SELECT, INSERT, UPDATE, DELETE) แต่ผู้ใช้สามารถใช้ Framework มาช่วยจัดการแปลงจากภาษาโปรแกรมมิ่ง ไปเป็น SQL หรือ Database ได้รายละเอียดดังภาพที่ 2.9

**ภาพที่ 2.9** ORM: Object Relational Mapping

จากภาพที่ 2.9 ORM [5] เป็นส่วนที่ใช้สำหรับสลับระหว่าง ข้อมูลที่อยู่ในรูปแบบ Object แปลงไปเป็นข้อมูลที่อยู่ในรูปแบบ ความสัมพันธ์ และดึงข้อมูลที่อยู่ในรูปแบบความสัมพันธ์ กลับไปเป็น ข้อมูลที่อยู่ในรูปแบบ Object และเครื่องมือที่พัฒนา ORM มีให้เลือกใช้หลายรูปแบบโดยที่ไม่จำเป็นต้องสร้างขึ้นใหม่

**4. เอพีไอ (API: Application Programming Interface)**

4.1 API [6]

**** เป็นบริการช่องทางการเชื่อมต่อเพื่อแลกเปลี่ยนข้อมูลจากระบบหนึ่งไปสู่ระบบอื่น ๆ ที่มีความรวดเร็ว ปลอดภัย หน้าที่หลักของ API คือ รับคำสั่งจากฝั่งลูกข่าย (Client ) ซึ่งคือ Application ต่าง ๆ เช่น Web Application, Mobile Application และ Desktop Application เป็นต้น เมื่อฝั่งลูกข่ายส่งคำสั่ง จะเรียกว่าการ Request จากนั้น API จะรับคำสั่งดังกล่าว นำไปประมวลผลและสรุปเป็นข้อมูลที่ตรงกับ Request และส่งข้อมูลเหล่านั้นกลับไปที่ฝั่งลูกข่าย หรือ Application เพื่อนำไปใช้งานต่อไป ดังภาพที่ 2.10

**ภาพที่ 2.10** การทำงานของ API

จากภาพที่ 2.10 API เป็นช่องทางในการเชื่อมต่อเพื่อแลกเปลี่ยนข้อมูลจากระบบหลักไปสู่ระบบ ลูกข่าย อื่นๆ หน้าที่หลักของ API จะคอยรับคำสั่งจากฝั่งลูกข่าย ซึ่งก็คือแอปพลิเคชันต่าง ๆ จะมีการรับคำสั่งไปประมวลผลและสรุปข้อมูลให้ตรงกับ Request ที่ต้องการ ก่อนจะส่งข้อมูลนั้นกลับไปที่ฝั่งลูกข่าย

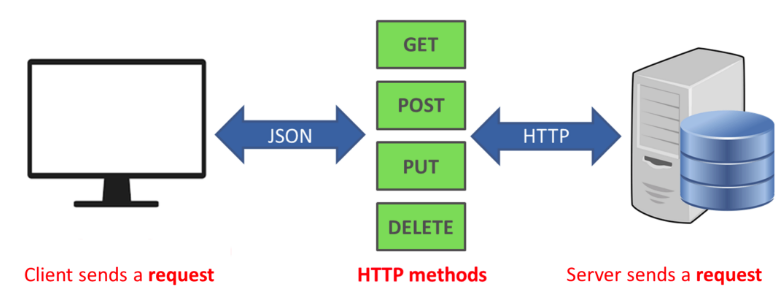
4.2 REST API [7]

REST (Representational State Transfer) เป็นรูปแบบการส่งข้อมูลระหว่าง Server-Client รูปแบบหนึ่งซึ่งอยู่บนพื้นฐานของ HTTP Protocol เป็นการสร้าง Web Service เพื่อแลกเปลี่ยนข้อมูลกันผ่านแอปพลิเคชันวิธีหนึ่ง ซึ่งส่งข้อมูลได้หลายชนิด เช่น Text, XML และ JSON โดยส่วนใหญ่แล้วจะเลือกชนิด JSON ที่สามารถรองรับได้หลายรูปแบบไม่ว่าจะเป็น Browser หรือ Mobile เพื่อระบุและทำงานกับทรัพยากร โดยที่เมธอดที่สำคัญใน REST API ประกอบด้วย

1) GET: ใช้สำหรับเรียกดูข้อมูลทรัพยากร

2) POST: ใช้สำหรับสร้างข้อมูลใหม่ทรัพยากร

3) PUT: ใช้สำหรับอัพเดตข้อมูลของทรัพยากรที่มีอยู่แล้ว

 4) DELETE: ใช้สำหรับลบทรัพยากร

**ภาพที่ 2.11** REST API

จากภาพที่ 2.11 REST API อาศัยรูปแบบของ HTTP Method เช่น GET, POST, PUT และ DELETE ในการทำงาน และจะส่งค่ากลับมาเป็น JSON ส่งผลให้สามารถรับส่งข้อมูลไปมาข้าม Platform ได้

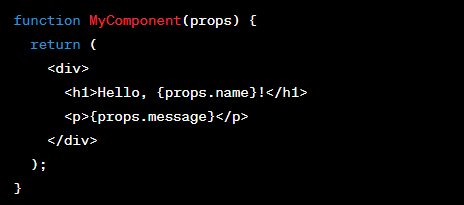
**5. เว็บแอปพลิเคชัน (Web application)**

เว็บแอปพลิเคชัน คือ โปรแกรมประยุกต์ที่จะเข้าถึงด้วยโปรแกรม Internet Browser ซึ่งทำให้ เหมาะสำหรับงานที่ต้องการข้อมูลแบบ Real Time จะพบข้อดีของเว็บแอพพลิเคชัน คือ ข้อมูลต่างๆที่อยู่ในระบบที่มีการไหลเวียนในแบบ Online จึงสามารถโต้ตอบกับผู้ใช้บริการแบบ Real Time ทำให้เกิดความประทับใจ รวมทั้งสามารถใช้งานได้ง่ายโดยไม่จำเป็นต้องติดตั้ง Client Program จะทำให้ไม่ต้อง Upgrade Client Program และสามารถใช้ผ่าน Internet Connection ที่มีความเร็วต่ำส่งผลให้ผู้ใช้สามารถใช้โปรแกรมได้จากทุกที่ ตัวอย่างระบบออนไลน์ที่เหมาะกับเว็บ แอปพลิเคชัน เช่น ระบบการจองสินค้าหรือบริการต่าง ๆ ระบบงานบุคลากร ระบบงานแผนการตลาด ระบบการสั่งซื้อแบบพิเศษ และระบบงานในโรงเรียน เป็นต้น

**6. รีแอคท์ (React)**

React คือ JavaScript Library [8] โดยเป็นเครื่องมือที่ใช้สร้างเว็บสำหรับสร้าง UI (User Interface) หรือ หน้าจอแอปพลิเคชัน จุดเด่นของ React ที่เหมาะนำมาใช้งานคือ ระบบแคชในตัวทำให้หน้าเว็บของผู้พัฒนามีการตอบสนองที่เร็ว และในการเขียน React ผู้พัฒนายังสามารถแยกองค์ประกอบของหน้าเว็บออกเป็นส่วน ๆ เรียกว่า Component แล้วนำมาประกอบกันเป็นหน้าเว็บซึ่งทำให้เราสามารถนำ Component ไปใช้ซ้ำที่อื่นได้ และผู้พัฒนาไม่จำเป็นต้องเขียนขึ้นใหม่ โดยสามารถแบ่งส่วนประกอบหลัก ๆ ของ React ได้ดังต่อไปนี้

6.1 คอมโพเนนท์ (Component)

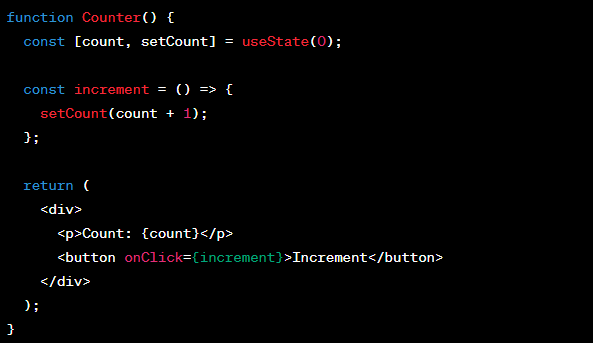
 คอมโพเนนท์ เป็นส่วนที่ใช้สร้างตัวอย่างของ UI ใน React โดยจะเป็นฟังก์ชันหรือคลาสที่จะคืนค่า UI ที่กำหนดรูปแบบและสิ่งที่จะแสดงบนหน้าเว็บ ผู้พัฒนาสามารถสร้างคอมโพเนนท์ใหม่ ๆ และนำมาใช้ซ้ำในแอปพลิเคชันได้

**ภาพที่ 2.12** React Component

จากภาพที่ 2.12 ตัวอย่างของฟังก์ชันคอมโพเนนท์ใน React โดยมีชื่อว่า MyComponent ซึ่งรับพารามิเตอร์ props ภายในฟังก์ชันในส่วนของการแสดงผล ของคอมโพเนนท์นี้มี HTML และ JSX ที่กำหนดรูปแบบและเนื้อหาที่จะแสดงบนหน้าเว็บ

6.2 สเตท (State)

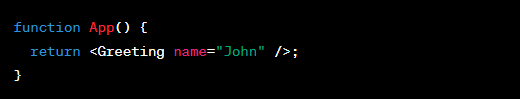
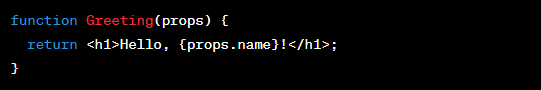
สเตท ใน React เป็นข้อมูลที่ถูกเก็บไว้ภายในคอมโพเนนท์ แต่ละคอมโพเนนท์ สามารถมี สเตทของตัวเองได้โดยอิสระ ซึ่งสเตทนี้สามารถเปลี่ยนแปลงได้และจะทำให้ React รีเรนเดอร์หรืออัปเดตส่วนของ UI ที่เกี่ยวข้อง

**ภาพที่ 2.13** React State

จากภาพที่ 2.13 ตัวอย่างนี้เราใช้ useState hook เพื่อสร้างสเตทที่ชื่อ count และ setCount สำหรับอัปเดตค่า count เมื่อมีการคลิกที่ปุ่มจะเพิ่มค่า count ผ่านฟังก์ชันที่ชื่อ increment

6.3  พรอปส์ (Props: Properties)

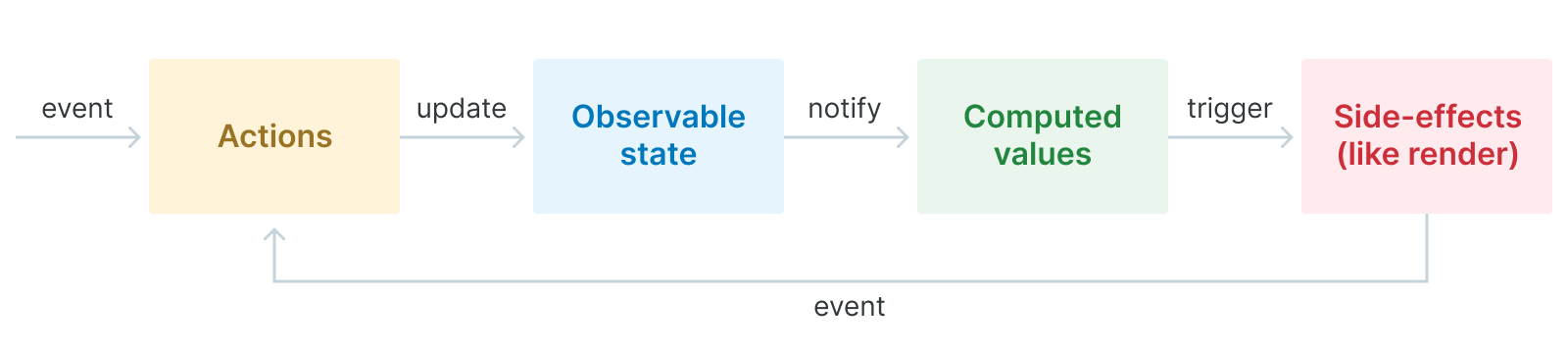
พรอปส์ ใน React คือข้อมูลที่ถูกส่งจากคอมโพเนนท์หนึ่งไปยัง คอมโพเนนท์อื่น โดยใช้ตัวแปรที่สามารถรับค่าได้เป็นพารามิเตอร์ของฟังก์ชันคอมโพเนนท์นั้น ๆ ซึ่งพรอปส์ถูกใช้เพื่อส่งข้อมูลระหว่างคอมโพเนนท์แม่ และคอมโพเนนท์ลูก เพื่อให้คอมโพเนนท์ลูกสามารถใช้ข้อมูลจาก คอมโพเนนท์แม่ได้



**ภาพที่ 2.14** React Props

จากภาพที่ 2.14 ตัวอย่างนี้มีการสร้างฟังก์ชันคอมโพเนนท์ ชื่อ Greeting ที่มีพรอปส์ชื่อ name เพื่อแสดงข้อความ โดยข้อความจะขึ้นอยู่กับค่า name ที่ถูกส่งมาในพรอปส์ของคอมโพเนนท์นี้ และในตัวอย่างนี้เราสร้างฟังก์ชันคอมโพเนนท์ชื่อ App ซึ่งนำคอมโพเนนท์ Greeting มาใช้และส่ง พรอปส์ name ให้กับคอมโพเนนท์ Greeting โดยในตัวอย่างนี้มีการส่งค่า "John" ในพรอปส์ name

6.4 MobX

 เป็นเครื่องมือสำหรับจัดการสเตท [9] (State Management) ที่สามารถนำมาใช้ร่วมกับ React ในการจัดการข้อมูล และสถานะของแอปพลิเคชันอย่างมีประสิทธิภาพและง่ายต่อการใช้งาน MobX ช่วยให้สามารถประกาศตัวแปรแบบ observable ที่จะติดตามการเปลี่ยนแปลงของได้โดยอัตโนมัติ และจะทำการอัปเดตคอมโพเนนท์ที่พึ่งพาข้อมูลนั้น ๆ เมื่อมีการเปลี่ยนแปลงของข้อมูลดังกล่าวเกิดขึ้น นอกจากนี้ยังมีหลักการทำงานเพิ่มเติมอีกหลายอย่างเช่น computed values และ actions ซึ่งช่วยให้สามารถควบคุมและจัดการการเปลี่ยนแปลงข้อมูลได้อย่างมีประสิทธิภาพ

**ภาพที่ 2.15** หลักการทำงานของ MobX

จากภาพที่ 2.15 โดยทั่วไปแล้วแนวคิดของ MobX สามารถอธิบายได้ด้วยหลายส่วนประกอบ ซึ่งเป็นตัวบ่งบอกในกระบวนการจัดการสเตท ดังนี้

1) Actions คือฟังก์ชันหรือเมธอดที่ถูกเรียกใช้เพื่อเปลี่ยนแปลงสเตทหรือข้อมูลในแอปพลิเคชัน สิ่งนี้สามารถทำได้โดยการดำเนินการกับ observable หรือผ่านฟังก์ชันที่ถูกต้องเท่านั้น

2) Observable คือตัวแปรหรือวัตถุที่ถูกติดตามการเปลี่ยนแปลงของ MobX โดยอัตโนมัติ ซึ่งคือสเตทที่ต้องการจัดการในแอปพลิเคชัน เมื่อมีการเปลี่ยนแปลงใด ๆ ที่เกี่ยวข้องกับ observable MobX จะทำการอัปเดตและแจ้งเตือนถึงส่วนที่เกี่ยวข้องให้รีเรนเดอร์หรืออื่น ๆ

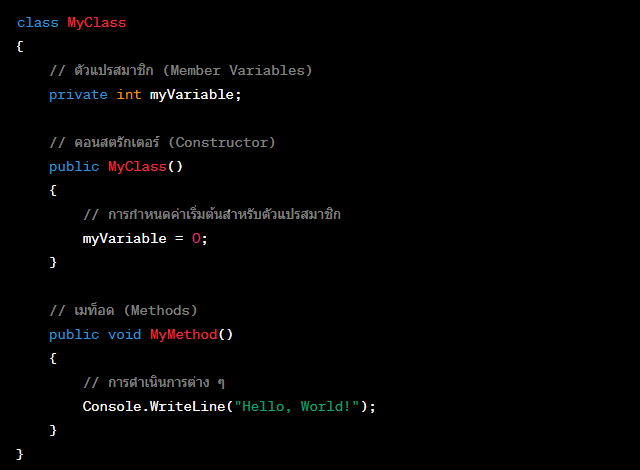
3) Computed Values คือค่าที่คำนวณขึ้นมาจากสเตทอื่น ๆ ในแอปพลิเคชัน โดย MobX จะตรวจสอบการเปลี่ยนแปลงในสเตทที่เกี่ยวข้องและอัปเดตค่า computed values ตามต้องการ

4) Side-Effect เป็นการกระทำที่เกิดขึ้นหลังจากการเปลี่ยนแปลงในสเตท โดยทั่วไปจะเป็นการทำงานเพิ่มเติมที่เกี่ยวข้องกับการแสดงผล หรือการกระทำภายนอก เช่น การรีเรนเดอร์คอมโพเนนท์ React หรือการส่งคำขอไปยังเซิร์ฟเวอร์ เพื่ออัปเดตข้อมูลหรือดึงข้อมูลใหม่ ซึ่งเมื่อเกิดการเปลี่ยนแปลงในสเตทที่เกี่ยวข้อง การกระทำเสริมจะถูกเรียกใช้เพื่อดำเนินการตามที่ต้องการ และอาจมีผลกระทบต่อสเตทหรือข้อมูลอื่น ๆ ในแอปพลิเคชัน

**7.** **ภาษาซีชาร์ป (C# Programming Language)**

ภาษาคอมพิวเตอร์ประเภท Object-Oriented Programming [10] พัฒนาโดย Microsoft โดยมีจุดมุ่งหมายในการวมความสามารถการคำนวณของภาษาซีพลัสพลัส (C++) ด้วยการโปรแกรมง่ายกว่าของ Visual Basic โดยภาษาซีชาร์ปมีพื้นฐานจากภาษาซีพลัสพลัส และเก็บส่วนการทำงานคล้ายกับภาษาจาวา (Java) โครงสร้างของภาษาซีชาร์ป ประกอบด้วยส่วนต่าง ๆ ดังนี้

7.1 คลาส

**** คลาส ในภาษาซีชาร์ปใช้สำหรับกำหนดโครงสร้างและพฤติกรรมของวัตถุที่จะถูกสร้างขึ้นตามแม่แบบที่กำหนดไว้ โดยสามารถขยายและเพิ่มคุณสมบัติตามความต้องการ ดังภาพที่ 2.16

**ภาพที่ 2.16** โครงสร้างคลาสในภาษาซีชาร์ป

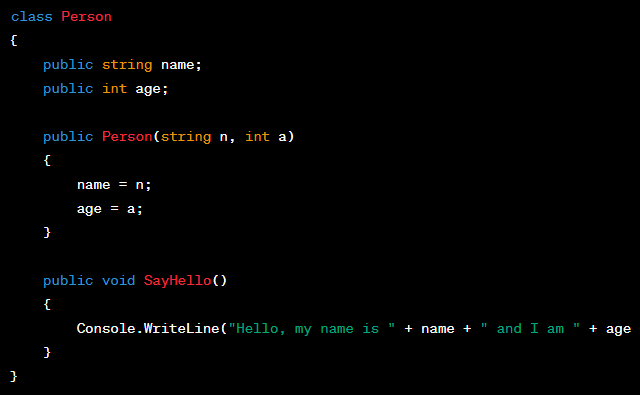
จากภาพที่ 2.16 รายละเอียดโครงสร้างดังนี้

1. MyClass คือ ชื่อของคลาส
2. myVariable คือ ตัวแปรสมาชิกที่เป็นตัวแปรที่ถูกเก็บเฉพาะภายในคลาส
3. MyClass() คือ คอนสตรักเตอร์ (Constructor) ที่ใช้สร้างวัตถุของคลาส ในตัวอย่างนี้จะกำหนดค่าเริ่มต้นของ myVariable เป็น 0
4. MyMethod() คือ เมธอดที่ใช้สำหรับการดำเนินการต่าง ๆ ภายในคลาส

7.2 วัตถุ

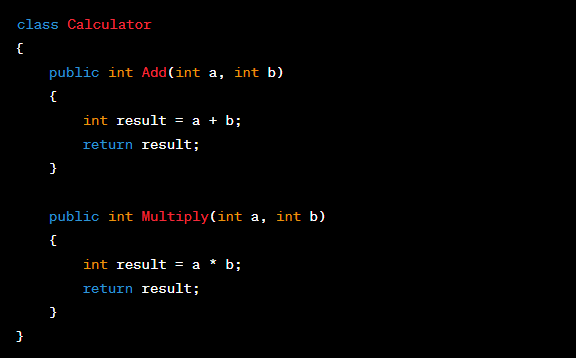
วัตถุเป็นตัวแทนของคลาสที่ถูกสร้างขึ้นจากแม่แบบคลาส โดยมีคุณสมบัติและพฤติกรรมตามที่กำหนดไว้ในคลาสนั้น โดยการสร้างวัตถุจะใช้คีย์เวิร์ด new ตามด้วยชื่อของคลาสและวงเล็บว่างเปล่า () ที่เรียกว่าคอนสตรักเตอร์ของคลาส ตัวอย่างดังภาพที่ 2.17

**ภาพที่ 2.17** การสร้างวัตถุในภาษาซีชาร์ป

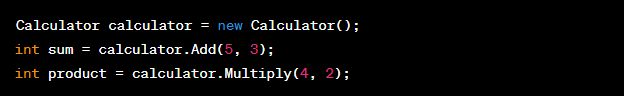


จากภาพที่ 2.17 ตัวอย่างคลาสชื่อ Person ที่มีคุณสมบัติ name และ age และเมธอด SayHello()เมื่อมีสร้างวัตถุของคลาส Person โดยใช้คอนสตรักเตอร์ Person() และกำหนดค่า name และ age ตามลำดับ จากนั้นจะสามารถเรียกใช้เมธอด SayHello() ของวัตถุ person1

7.3 เมธอด

 เมธอด เป็นส่วนที่ใช้ในการกำหนดการกระทำหรือการดำเนินการที่คลาสหรือวัตถุสามารถทำได้ โดยสามารถรับพารามิเตอร์เข้ามาเพื่อประมวลผลและส่งค่ากลับ ดังภาพที่ 2.18

**ภาพที่ 2.18** เมธอดในภาษาซีชาร์ป

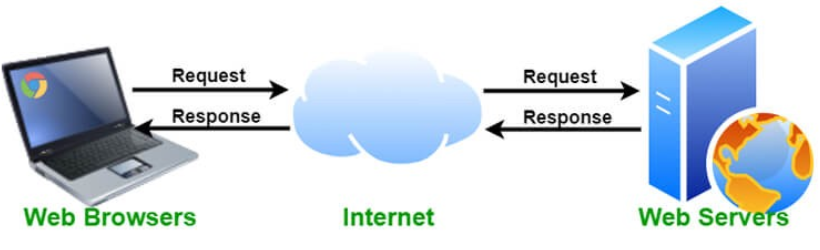
 จากภาพที่ 2.18 ตัวอย่างเมธอดในคลาส Calculator ที่มีเมธอด Add และ Multiply ในตัวอย่างนี้ จะมีคลาส Calculator ที่มีเมธอด Add และ Multiply ซึ่งรับพารามิเตอร์ a และ b เพื่อประมวลผลและส่งค่าผลลัพธ์กลับ โดยสามารถเรียกใช้เมธอดเหล่านี้ได้ดังภาพที่ 2.19

**ภาพที่ 2.19** ตัวอย่างการใช้งานเมธอด

จากภาพที่ 2.19 สร้างวัตถุ calculator จากคลาส Calculator และเรียกใช้เมธอด Add และ Multiply เพื่อทำการบวกและคูณตัวเลข กลับมาเก็บผลลัพธ์ในตัวแปร sum และ product ตามลำดับ

**8. เว็บเซิร์ฟเวอร์ (Web Server)**

เว็บเซิร์ฟเวอร์ [11] คือ เครื่องคอมพิวเตอร์ ที่ติดตั้งโปรแกรมคอมพิวเตอร์ ซึ่งทำหน้าที่ให้บริการข้อมูลแก่ เครื่องคอมพิวเตอร์ที่ขอรับบริการ ในรูปแบบสื่อผสม ผ่านระบบเครือข่าย โดยผู้ใช้งานติดต่อสื่อสารกับเว็บบราวเซอร์ ผ่านทางโปรแกรมเว็บเซิร์ฟเวอร์ และมีการเชื่อมต่อผ่านอินเตอร์เน็ตดังภาพที่ 2.20

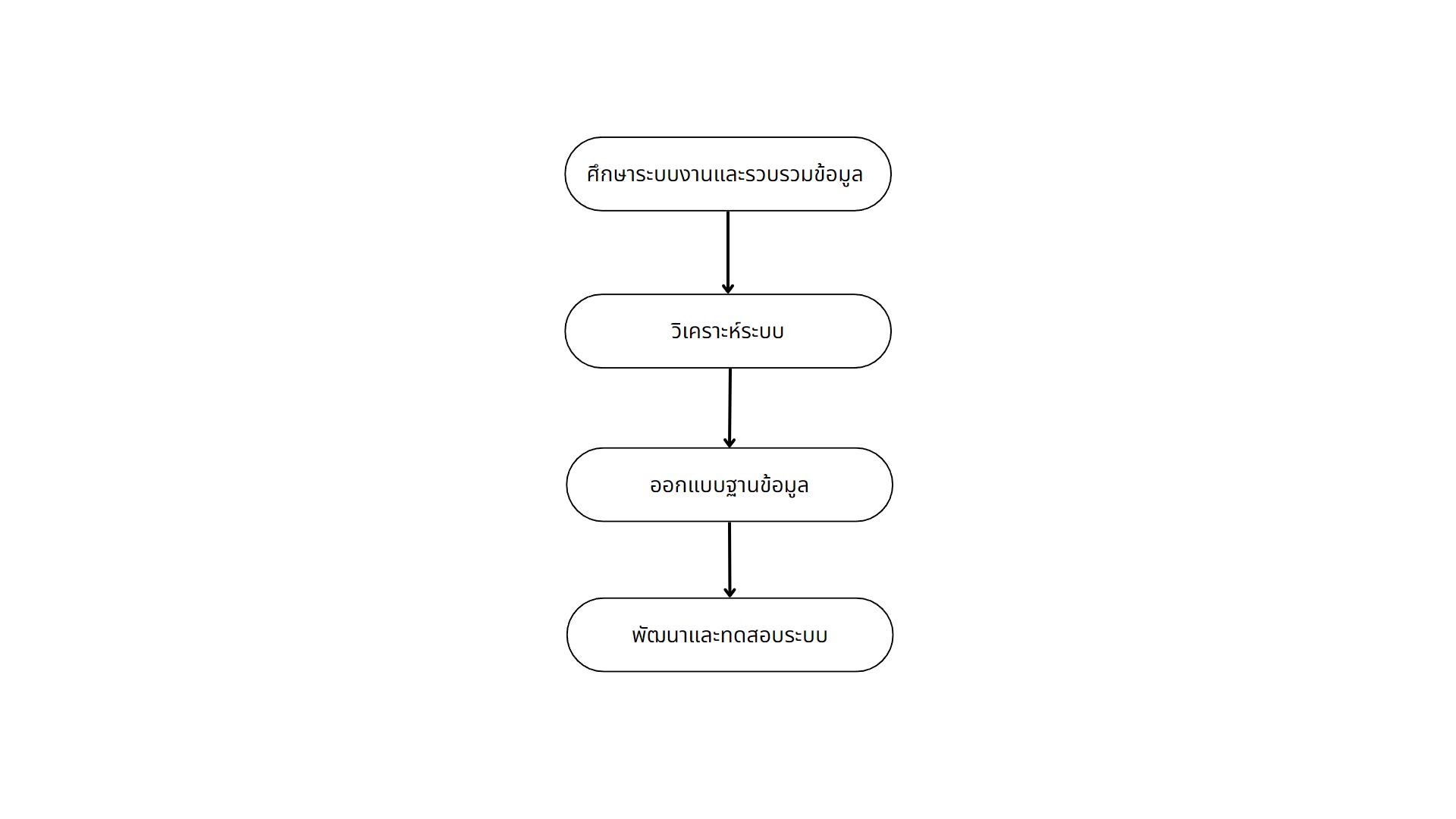


**ภาพที่ 2.20** การทำงานของเว็บเซิร์ฟเวอร์

จากภาพที่ 2.20 ขั้นตอนการทำงานของระบบ โดยเริ่มจากผู้ใช้งานได้ทำการร้องขอบริการด้านข้อมูลกับเซิร์ฟเวอร์ซึ่งผู้ใช้งานสามารถติดต่อสื่อสารผ่านทางระบบอินเตอร์เน็ต และในปัจจุบันสามารถเรียกใช้เว็บเซิร์ฟเวอร์ผ่านทางโทรศัพท์เคลื่อนที่ได้

**บทที่ 3**

**วิธีการดำเนินงาน**

 การจัดทำโครงงานจะต้องมีวิธีการดำเนินงานที่เป็นขั้นตอนเพื่อที่จะทำให้เกิดผลลัพธ์ที่มีประสิทธิภาพมากที่สุด ตรงกับขอบเขตงานที่ได้ตั้งเป้าหมายไว้ ซึ่งได้นำทฤษฎี SDLC (SYSTEM DEVELOPMENT LIFE CYCLE) โดยในการพัฒนานั้นจะมีการกำหนดให้เป็นไปในทิศทางเดียวกัน และกำหนดขั้นตอนที่เป็นแนวทางในการวิเคราห์ระบบโดยพยายามให้มีข้อบกพร่องน้อยที่สุดเพราะงานการวิเคราะห์ระบบในปัจจุบันมีความซันซ้อนของงานมากกว่าสมัยก่อนนักวิเคราะห์ระบบจึงต้องมาตรฐานในการพัฒนาระบบงานดังกล่าว จึงได้มีการคิดค้นวงจรการพัฒนาระบบขึ้นเรียกว่า วงจรการพัฒนาระบบงาน และมีนำทฤษฎีมาประยุกต์ใช้ โดยมีรายละเอียดดังภาพที่ 3.1

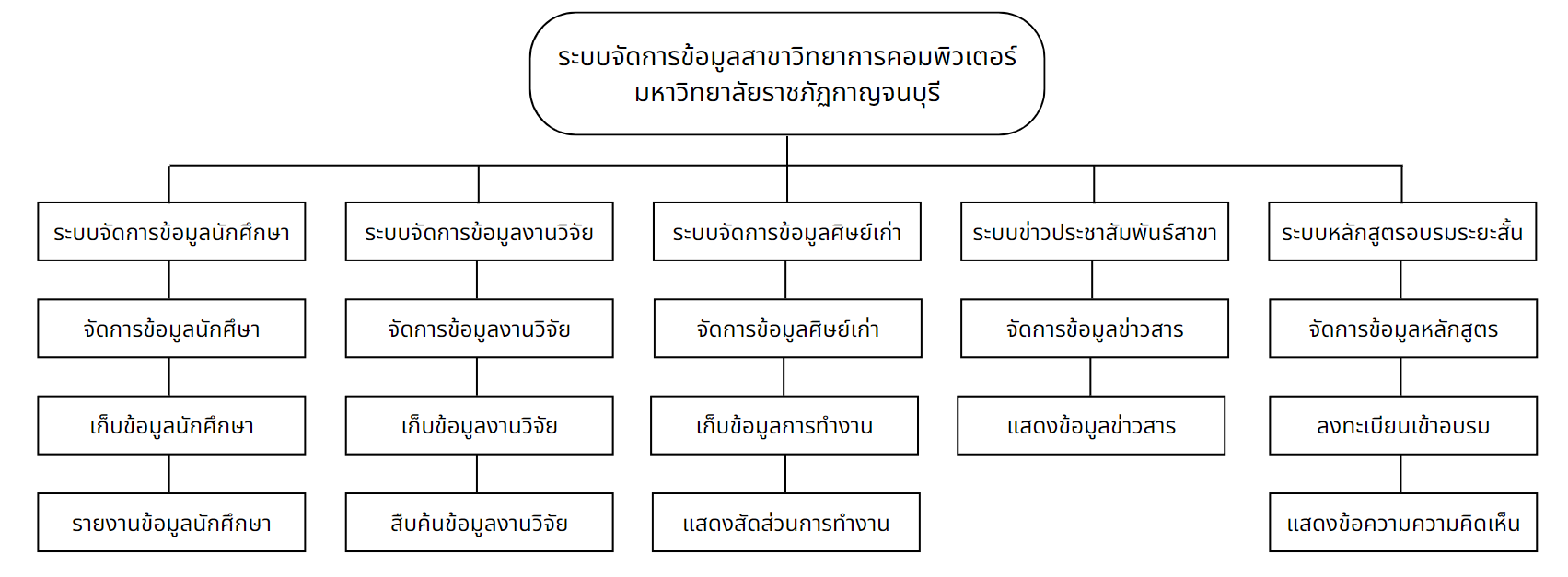
**ภาพที่ 3.1** ขั้นตอนของการดําเนินงาน

จากภาพที่ 3.1 ขั้นตอนการดำเนินงาน 4 ขั้นตอน ประกอบไปด้วย 1) ศึกษาระบบงานและรวบรวมข้อมูล เพื่อหาข้อมูลที่จำเป็นในการจัดทำโครงงาน โดยการค้นหาทางสื่อดิจิทัลและสอบถามข้อมูลจากผู้ที่เกี่ยงข้อง 2) วิเคราะห์ระบบ เพื่อนำมาออกแบบโดยการนำข้อมูลที่ได้มาทำการวิเคราะห์และออกแบบให้ตรงกับความต้องการของขอบเขตงาน 3) ออกแบบฐานข้อมูล เพื่อลดการเก็บข้อมูลที่ซ้ำซ้อนและแก้ไขปัญหาต่าง ๆ และ 4) พัฒนาและทดสอยระบบ เพื่อการพัฒนาที่มีประสิทธิภาพให้ได้ระบบที่ถูกต้องและตอบโจทย์ความต้องการของผู้ใช้งาน จากขั้นตอนทั้ง 4 ข้อ ที่กล่าวมาข้างต้นจะมีรายละเอียดดังต่อไปนี้

**1. การศึกษาระบบงานและรวบรวมข้อมูล**

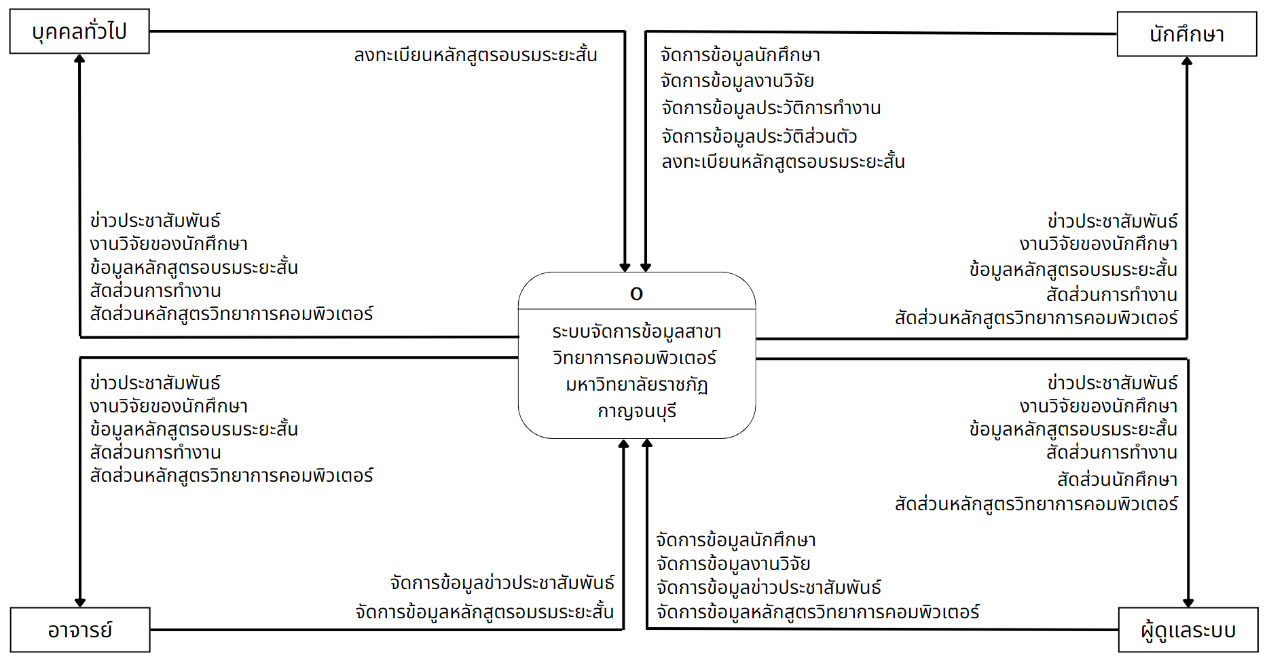
จากการศึกษาจากเอกสารระบบงานเดิม ซึ่งมีขอบเขตที่ครอบคลุม เช่น ระบบจัดการนักศึกษาระบบจัดการงานวิจัย และระบบประชาสัมพันธ์ เป็นต้น และผู้จัดทำได้กำหนดขอบเขตเพิ่มเติม เช่นระบบจัดการหลักสูตรอบรมระยะสั้น เป็นต้น นอกจากนี้เพื่อให้ซอฟท์แวร์มีความทันสมัย จะต้องศึกษาการพัฒนาเว็บแอปพลิเคชันด้วยเทคโนโลยีต่าง ๆ ในปัจจุบัน เพื่อหาข้อมูลทั้งข้อดีและข้อเสียสำหรับนำมาพัฒนาตัวระบบ เทคโนโลยีที่ใช้ เช่น ASP.NET Core สำหรับทำ API ในการบริหารข้อมูลในระบบ พัฒนาส่วนแสดงผลหน้าเว็บ React Library และจัดเก็บข้อมูลลงฐานข้อมูลMicrosoft SQL Server เป็นต้น

**2. การวิเคราะห์ระบบงาน**

 การวิเคราะห์ระบบเพื่อใช้ในการพัฒนาระบบจัดการข้อมูลสาขาวิทยาการคอมพิวเตอร์มหาวิทยาลัยราชภัฏกาญจนบุรี ซึ่งได้นําปัญหาและความต้องการของผู้ใช้งานมาวิเคราะห์ โดยระบบงานจะจัดโครงสร้างตามกลุ่มเนื้อหา จึงเลือกใช้โครงสร้างแบบลำดับขั้น เพื่อให้เข้าถึงข้อมูลต่าง ๆ ได้ง่ายขึ้น โดยจะมีหน้าหลักเป็นจุดเริ่มต้น และจุดร่วมจุดเดียวที่จะนำไปสู่การเชื่อมโยงเนื้อหาเป็นลำดับจากบนลงล่าง เพื่อให้เข้าใจการทำงานของระบบได้ง่ายมากยิ่งขึ้น ดังภาพที่ 3.2

**ภาพที่ 3.2**  ระบบจัดการข้อมูลสาขาวิทยาการคอมพิวเตอร์

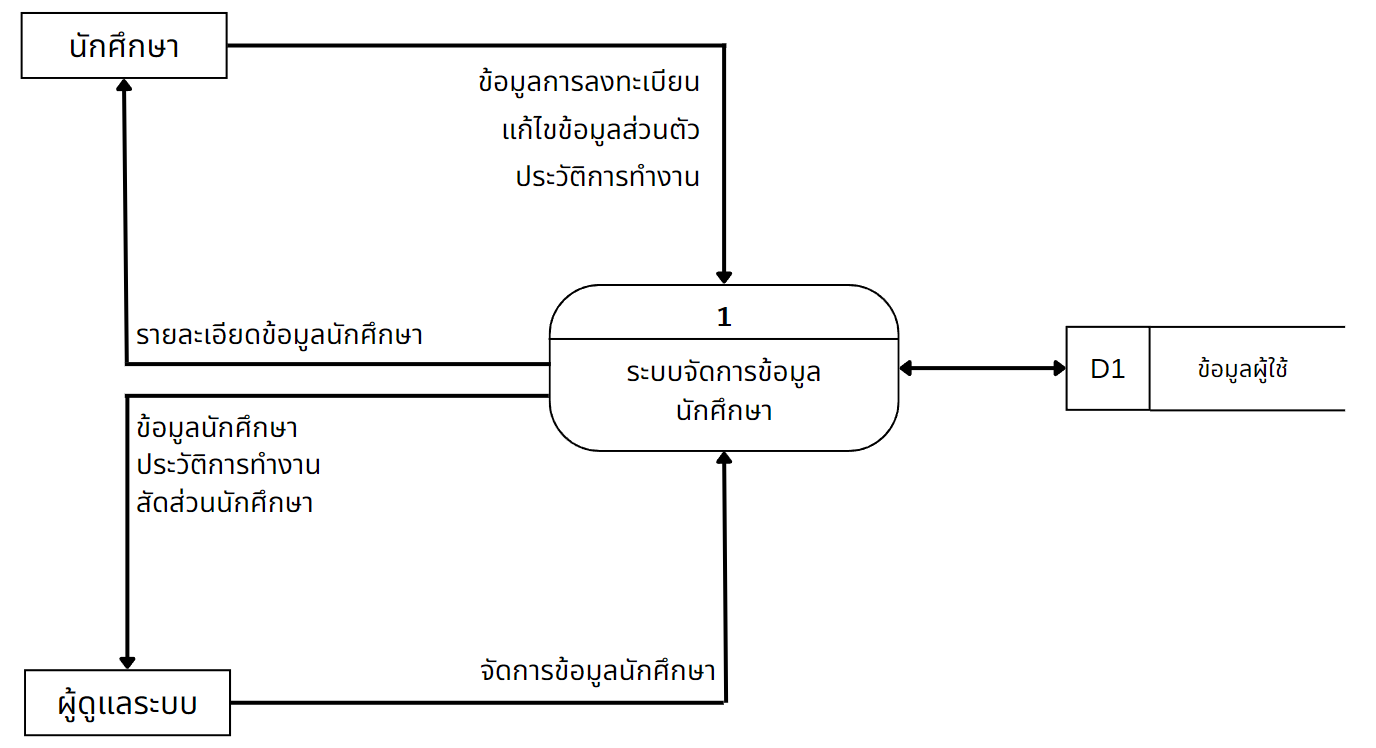
จากภาพที่ 3.2 การทำงานแบ่งออกเป็น 5 ส่วน ได้แก่ 1) ระบบจัดการข้อมูลนักศึกษา ในส่วนนี้จะทำงานเกี่ยวกับข้อมูลนักศึกษา 2) ระบบจัดการข้อมูลงานวิจัย ส่วนนี้จะเกี่ยวกับข้อมูลงานวิจัยเพื่อเป็นผลงานของนักศึกษา 3) ระบบจัดการข้อมูลศิษย์เก่า ส่วนนี้จะเกี่ยวกับการเก็บข้อมูลการเข้าทำงานนักศึกษา เพื่อเป็นข้อมูลในการประชาสัมพันธ์ 4) ระบบข่าวประชาสัมพันธ์สาขา ส่วนนี้จะเกี่ยวกับการจัดการข้อมูลในข่าวประชาสัมพันธ์ในสาขา เช่น การเปิดรับนักศึกษาใหม่ เป็นต้น และ 5) ระบบหลักสูตรอบรมระยะสั้น ส่วนนี้จะเกี่ยวกับการจัดอบรมหลักสูตรระยะสั้น เพื่อเป็นช่องทางในการรับลงทะเบียนผู้เข้าอบรม จากที่กล่าวมาจึงออกแบบแผนภาพบริบท (Context Diagram) ดังนี้

 2.1 แผนภาพบริบทระดับที่ 0

**ภาพที่ 3.3** แผนภาพบริบทระบบจัดการข้อมูลสาขาวิทยาการคอมพิวเตอร์

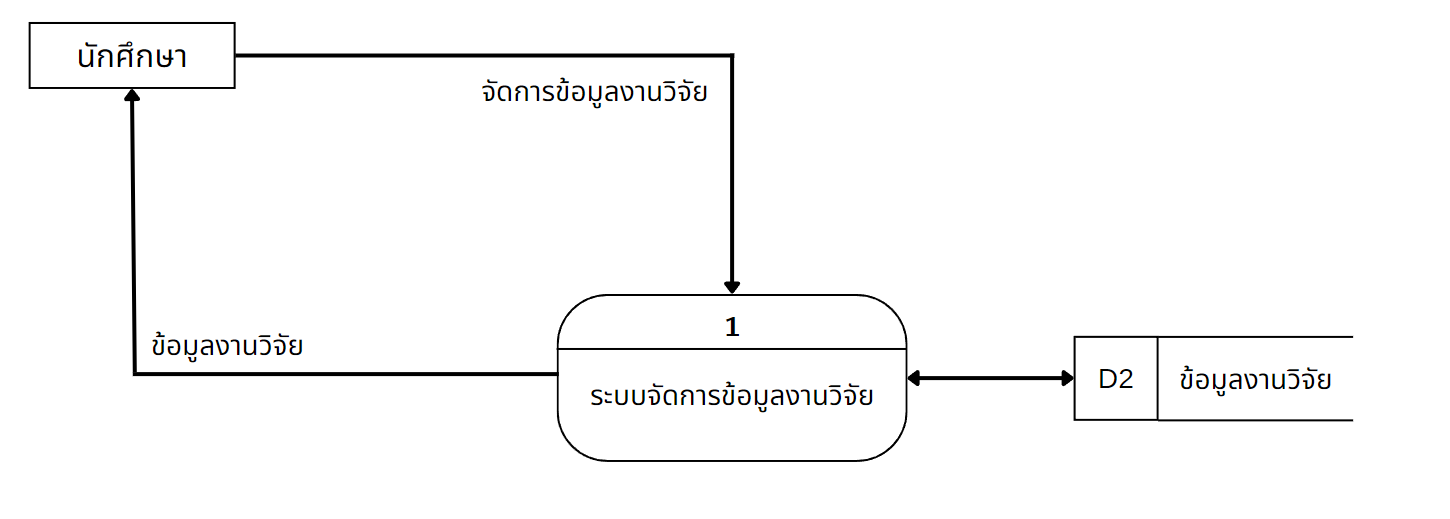
จากภาพที่ 3.3 แสดงรายละเอียดการทำงานกับระบบของผู้ใช้งาน การติดต่อกับฐานข้อมูล มีการติดต่อจากผู้ใช้งานระบบ 4 ทาง เช่น บุคคลทั่วไปสามารถเรียกดูข้อมูลงานวิจัยของนักศึกษา ประวัติการทำงานและข้อมูลข่าวสารได้ นักศึกษาสามารถจัดการข้อมูลงานวิจัย จัดการประวัติการทำงานได้ อาจารย์สามารถจัดการข้อมูลหลักสูตรอบรมระยะสั้นได้ และผู้ดูแลสามารถจัดการข้อมูลนักศึกษา จัดการข้อมูลงานวิจัยของนักศึกษาได้ เป็นต้น

2.2 แผนภาพบริบทระดับที่ 1

 1) ระบบจัดการข้อมูลนักศึกษา

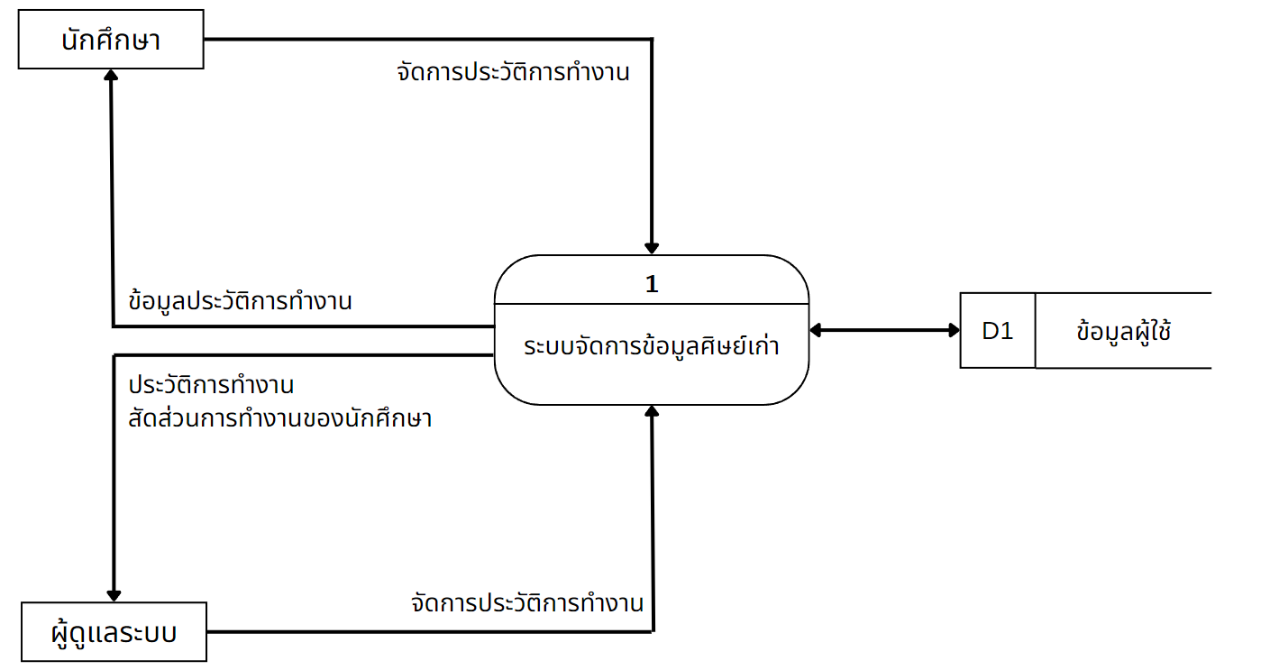
**ภาพที่ 3.4**  แผนภาพบริบทของระบบจัดการข้อมูลนักศึกษา

จากภาพที่ 3.3 เป็นแผนภาพการทำงานของระบบจัดการนักศึกษา โดยผู้ที่เกี่ยวข้องกับระบบจะเป็นนักศึกษาและผู้ดูแลระบบ โดยที่นักศึกษาสามารถลงทะเบียนเข้าสู่ระบบ และจัดการข้อมูลของตัวเอง จากนั้นระบบจะแสดงข้อมูลสัดส่วนของนักศึกษา ข้อมูลนักศึกษาและประวัติการทำงานให้ผู้ดูแลระบบ จึงสามารถจัดการข้อมูลของนักศึกษาได้

**** 2) ระบบจัดการข้อมูลงานวิจัย

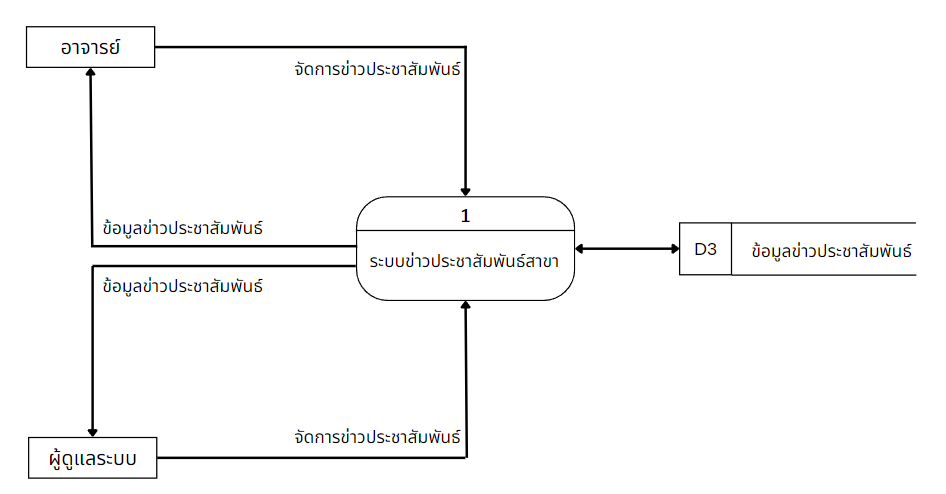
**ภาพที่ 3.5** แผนภาพบริบทของระบบจัดการข้อมูลงานวิจัย

จากภาพที่ 3.5 เป็นแผนภาพการทำงานของระบบจัดการข้อมูลงานวิจัยโดยผู้ที่เกี่ยวข้องกับระบบจะเป็นนักศึกษา โดยระบบจะแสดงข้อมูลของงานวิจัยจากฐานข้อมูล จากนั้นนักศึกษาสามารถจัดการของานวิจัยของตัวเองได้

3) ระบบจัดการข้อมูลศิษย์เก่า

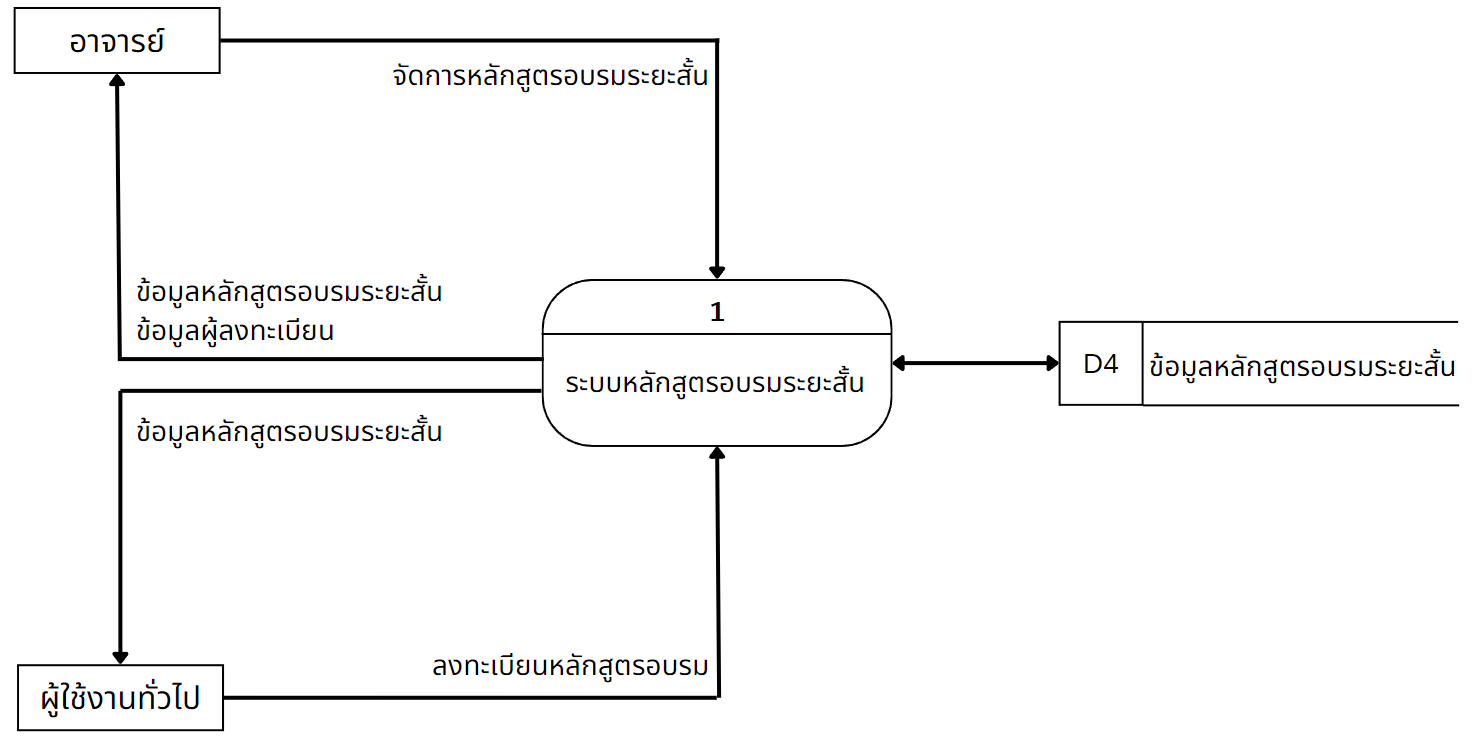
**ภาพที่ 3.6** แผนภาพบริบทของระบบจัดการข้อมูลศิษย์เก่า

จากภาพที่ 3.6 เป็นแผนภาพการทำงานของระบบจัดการข้อมูลศิษย์เก่า โดยผู้ที่เกี่ยวข้องกับระบบจะเป็นนักศึกษากับผู้ดูแลระบบ โดยระบบจะแสดงข้อมูลของประวัติการทำงานจากฐานข้อมูล และนักศึกษาสามารถจัดการประวัติการทำงานของตัวเองได้ จากนั้นระบบจะแสดงสัดส่วนการทำงานให้กับผู้ดูแลระบบ

 4) ระบบข่าวประชาสัมพันธ์สาขา

**ภาพที่ 3.7** แผนภาพบริบทของระบบข่าวประชาสัมพันธ์สาขา

จากภาพที่ 3.7 เป็นแผนภาพการทำงานของระบบข่าวประชาสัมพันธ์สาขาโดยมีผู้ที่เกี่ยวข้องกับระบบได้แก่ อาจารย์ และผู้ดูแลระบบ โดยระบบจะแสดงข้อมูลข่าวประชาสัมพันธ์ ให้กับอาจารย์และผู้ดูแลระบบ จึงสามารถจัดการข้อมูลข่าวประชาสัมพันธ์ได้

 5) ระบบหลักสูตรอบรมระยะสั้น

**ภาพที่ 3.8** แผนภาพบริบทของระบบหลักสูตรอบรมระยะสั้น

จากภาพที่ 3.8 เป็นแผนภาพการทำงานของระบบหลักสูตรอบรมระยะสั้นโดยมีผู้ที่เกี่ยวข้องกับระบบได้แก่ อาจารย์ และผู้ใช้งานทั่วไป โดยระบบจะแสดงข้อมูลหลักสูตรอบรมระยะสั้น ให้กับอาจารย์และผู้ใช้งานทั่วไป จากนั้นผู้ใช้งานทั่วไปสามารถลงทะเบียนเข้าสู่หลักสูตรอบรมได้ และอาจารย์สามารถจัดการข้อมูลสูตรของตัวเองได้

**3. การออกแบบฐานข้อมูล**

การออกแบบฐานข้อมูลของระบบสาขาวิทยาการคอมพิวเตอร์ ใช้หลักการออกแบบจําลองข้อมูลเชิงสัมพันธ์ (Relational Data Model) โดยจัดข้อมูลให้อยู่ในรูปของตารางที่มีระบบคล้ายแฟ้ม โดยที่ข้อมูลแต่ละแถวของตารางจะแทนเรคคอร์ด ส่วนข้อมูลแนวนอนจะแทนคอลัมน์ ซึ่งเป็นขอบเขตของข้อมูล โดยที่ตารางแต่ละตารางที่สร้างขึ้นจะเป็นอิสระ ดังนั้นผู้ออกแบบฐานข้อมูลจะต้องมีการวางแผนถึงตารางข้อมูลที่จำเป็นต้องใช้ตามหลักการพัฒนาระบบ โดยยึดหลักทฤษฎีของวงจรการพัฒนา (System Development Life Cycle: SDLC) โดยมีรายละเอียดดังต่อไปนี้

3.1 พจนานุกรมฐานข้อมูล

พจนานุกรมข้อมูล (Data Dictionary) คือ การแสดงรายละเอียดของตารางข้อมูลต่าง ๆ ในฐานข้อมูลทำให้สามารถค้นหารายละเอียดที่ต้องการได้สะดวกมากยิ่งขึ้น และได้จัดทำเป็นตารางทั้งหมด 24 ตาราง ดังนี้

**ตารางที่ 3.1** ผู้ใช้งาน

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ชื่อแฟ้มข้อมูล: AspNetUsers (ข้อมูลผู้งาน)** | | | | |
| **ลำดับ** | **ชื่อ** | **คุณสมบัติ** | **คำอธิบาย** | **หมายเหตุ** |
| 1 | Id | nvarchar | รหัสผู้ใช้งาน | (PK) |
| 2 | UserName | nvarchar | ชื่อผู้ใช้งาน |  |
| 3 | Email | nvarchar | อีเมล |  |
| 4 | PasswordHash | nvarchar | รหัสผ่านที่ถูกแฮช |  |
| 5 | FullName | nvarchar | ชื่อ-นามาสกุล |  |
| 6 | Bio | nvarchar | ประวัติส่วนตัว |  |
| 7 | Image | nvarchar | รูปภาพประจำตัว |  |
| 8 | IsRole | int | ประเภทผู้ใช้งาน |  |
| 9 | IsUsed | bit | สถานการณ์ใช้งาน |  |
| 10 | CreatedAt | datetime2 | วันที่สร้างข้อมูล |  |

**ตารางที่ 3.2** ข้อมูลผู้ใช้ของนักศึกษา

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ชื่อแฟ้มข้อมูล: UserStudent (ข้อมูลผู้ใช้ของนักศึกษา)** | | | | |
| **ลำดับ** | **ชื่อ** | **คุณสมบัติ** | **คำอธิบาย** | **หมายเหตุ** |
| 1 | Id | nvarchar | รหัสผู้ใช้งาน | (FK) |
| 2 | YearEdu | nvarchar | ปีการศึกษา |  |
| 3 | OldEdu | nvarchar | วุฒิการศึกษาเดิม |  |
| 4 | Address | nvarchar | ที่อยู่ |  |

**ตารางที่ 3.3** ข้อมูลผู้ใช้ของอาจารย์

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ชื่อแฟ้มข้อมูล: UserLecturer (ข้อมูลผู้ใช้ของอาจารย์)** | | | | |
| **ลำดับ** | **ชื่อ** | **คุณสมบัติ** | **คำอธิบาย** | **หมายเหตุ** |
| 1 | Id | nvarchar | รหัสผู้ใช้งาน | (FK) |
| 2 | Expert | nvarchar | ความชำนาญ |  |
| 3 | LvEdu | nvarchar | วุฒิการศึกษา |  |
| 4 | Program | nvarchar | สาขาที่สังกัด |  |
| 5 | Position | nvarchar | ตำแหน่งงาน |  |
| 6 | Contact | nvarchar | ช่องทางติดต่อ |  |

**ตารางที่ 3.4** ประวัติการทำงานของนักศึกษา

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ชื่อแฟ้มข้อมูล: JobHistory (ข้อมูลประวัติการทำงานของนักศึกษา)** | | | | |
| **ลำดับ** | **ชื่อ** | **คุณสมบัติ** | **คำอธิบาย** | **หมายเหตุ** |
| 1 | Id | nvarchar | รหัสประวัติการทำงาน | (FK) |
| 2 | JobName | nvarchar | ชื่องาน |  |
| 3 | Position | nvarchar | ตำแหน่งงาน |  |
| 4 | Date | datetime2 | วันที่เริ่มทำงาน |  |
| 5 | Company | nvarchar | ชื่อบริษัท |  |
| 6 | Description | nvarchar | รายละเอียด |  |
| 7 | CreatedAt | datetime2 | วันที่สร้างข้อมูล |  |
| 8 | IsUsed | bit | สถานะการใช้งาน |  |

**ตารางที่ 3.5** บทบาท

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ชื่อแฟ้มข้อมูล: AspNetRoles (ข้อมูลบทบาทผู้ใช้งาน)** | | | | |
| **ลำดับ** | **ชื่อ** | **คุณสมบัติ** | **คำอธิบาย** | **หมายเหตุ** |
| 1 | Id | nvarchar | รหัสบทบาท | (PK) |
| 2 | Name | nvarchar | ชื่อบทบาท |  |

**ตารางที่ 3.6** ผู้ใช้งานและบทบาท

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ชื่อแฟ้มข้อมูล: AspNetUserRoles (ข้อมูลผู้ใช้งานและบทบาท)** | | | | |
| **ลำดับ** | **ชื่อ** | **คุณสมบัติ** | **คำอธิบาย** | **หมายเหตุ** |
| 1 | UserId | nvarchar | รหัสผู้ใช้งาน | (PK) |
| 2 | RoleId | nvarchar | รหัสบทบาท | (FK) |

**ตารางที่ 3.7** ข่าวประชาสัมพันธ์

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ชื่อแฟ้มข้อมูล: Newses (ข้อมูลข่าวประชาสัมพันธ์)** | | | | |
| **ลำดับ** | **ชื่อ** | **คุณสมบัติ** | **คำอธิบาย** | **หมายเหตุ** |
| 1 | Id | nvarchar | รหัสข่าวประชาสัมพันธ์ | (PK) |
| 2 | AuthorId | nvarchar | รหัสผู้เขียน | (FK) |
| 3 | Title | nvarchar | หัวข้อ |  |
| 4 | SubTitle | nvarchar | คำบรรยาย |  |
| 5 | Body | nvarchar | เนื้อหา |  |
| 6 | IsHidden | bit | สถานะการซ่อนข้อมูล |  |
| 7 | CreatedAt | datetime2 | วันที่สร้างข้อมูล |  |

**ตารางที่ 3.8** รูปภาพข่าวประชาสัมพันธ์

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ชื่อแฟ้มข้อมูล: NewsPhotos (ข้อมูลรูปภาพข่าวประชาสัมพันธ์)** | | | | |
| **ลำดับ** | **ชื่อ** | **คุณสมบัติ** | **คำอธิบาย** | **หมายเหตุ** |
| 1 | Id | nvarchar | รหัสรูปภาพ | (PK) |
| 2 | NewsId | nvarchar | รหัสข่าวประชาสัมพันธ์ | (FK) |
| 3 | Url | nvarchar | ที่อยู่รูปภาพ |  |
| 4 | IsMain | bit | สถานะรูปหลัก |  |
| 5 | CreatedAt | datetime2 | วันที่สร้างข้อมูล |  |

**ตารางที่ 3.9** งานวิจัยของนักศึกษา

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ชื่อแฟ้มข้อมูล: Projects (ข้อมูลงานวิจัยของนักศึกษา)** | | | | |
| **ลำดับ** | **ชื่อ** | **คุณสมบัติ** | **คำอธิบาย** | **หมายเหตุ** |
| 1 | Id | nvarchar | รหัสงานวิจัย | (PK) |
| 2 | StudentId | nvarchar | รหัสผู้ใช้งานของนักศึกษา | (FK) |
| 3 | NameTH | nvarchar | ชื่องานวิจัยภาษาไทย |  |
| 4 | NameEN | nvarchar | ชื่องานวิจัยภาษาอังกฤษ |  |
| 5 | Description | nvarchar | รายละเอียด |  |
| 6 | Image | nvarchar | รูปภาพงานวิจัย |  |
| 7 | PDF | nvarchar | ที่อยู่ของเอกสารประกอบงานวิจัย |  |
| 8 | VideoUrl | nvarchar | ที่อยู่วิดีโอนำเสนองานวิจัย |  |
| 9 | WebUrl | nvarchar | ลิงค์เว็บของงานวิจัย |  |
| 10 | GithubUrl | nvarchar | ที่อยู่ของกิตฮับ (Github) |  |
| 11 | KeyWords | nvarchar | คำสำคัญที่เกี่ยวข้อง |  |
| 12 | IsuUsed | bit | สถานการณ์ใช้งาน |  |
| 13 | CreatedAt | datetime2 | วันที่สร้างข้อมูล |  |

**ตารางที่ 3.10** อาจารย์ที่ปรึกษางานวิจัย

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ชื่อแฟ้มข้อมูล: Consultants (ข้อมูลอาจารย์ที่ปรึกษาของงานวิจัย)** | | | | |
| **ลำดับ** | **ชื่อ** | **คุณสมบัติ** | **คำอธิบาย** | **หมายเหตุ** |
| 1 | Id | nvarchar | รหัสอาจารย์ที่ปรึกษา | (PK) |
| 2 | ProjectId | nvarchar | รหัสงานวิจัย | (FK) |
| 3 | LecturerName | nvarchar | ชื่ออาจารย์ที่ปรึกษา |  |

**ตารางที่ 3.11** หลักสูตรอบรมระยะสั้น

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ชื่อแฟ้มข้อมูล: Courses (ข้อมูลหลักสูตรอบรมระยะสั้น)** | | | | |
| **ลำดับ** | **ชื่อ** | **คุณสมบัติ** | **คำอธิบาย** | **หมายเหตุ** |
| 1 | Id | nvarchar | รหัสหลักสูตรอบรมระยะสั้น | (PK) |
| 2 | LecturerId | nvarchar | รหัสอาจารย์ | (FK) |
| 3 | Title | nvarchar | ชื่อหลักสูตรอบรมระยะสั้น |  |
| 4 | Description | nvarchar | รายละเอียด |  |
| 5 | IsUsed | bit | สถานการณ์ใช้งาน |  |
| 6 | CreatedAt | datetime2 | วันที่สร้างข้อมูล |  |

**ตารางที่ 3.12** รูปภาพหลักสูตรอบรมระยะสั้น

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ชื่อแฟ้มข้อมูล: CoursePhoto (ข้อมูลรูปภาพหลักสูตรอบรมระยะสั้น)** | | | | |
| **ลำดับ** | **ชื่อ** | **คุณสมบัติ** | **คำอธิบาย** | **หมายเหตุ** |
| 1 | Id | nvarchar | รหัสรูปภาพ | (PK) |
| 2 | CourseId | nvarchar | รหัสหลักสูตรอบรมระยะสั้น | (FK) |
| 3 | Url | nvarchar | ที่อยู่ของรูปภาพ |  |
| 4 | IsMain | bit | สถานะรูปภาพหลัก |  |

**ตารางที่ 3.13** รุ่นของหลักสูตรอบรมระยะสั้น

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ชื่อแฟ้มข้อมูล: Generations (ข้อมูลรุ่นของหลักสูตรอบรมระยะสั้น)** | | | | |
| **ลำดับ** | **ชื่อ** | **คุณสมบัติ** | **คำอธิบาย** | **หมายเหตุ** |
| 1 | Id | nvarchar | รหัสรุ่น | (PK) |
| 2 | CourseId | nvarchar | รหัสหลักสูตรอบรมระยะสั้น | (FK) |
| 3 | SubTitle | nvarchar | คำบรรยาย |  |
| 4 | Description | nvarchar | รายละเอียด |  |
| 5 | StartDate | datetime2 | วันที่เริ่มต้น |  |
| 6 | EndDate | datetime2 | วันที่สิ้นสุด |  |
| 7 | GenPhto | nvarchar | ที่อยู่รูปภาพประจำรุ่น |  |
| 8 | Quantity | int | จำนวนรับลงทะเบียน |  |
| **ลำดับ** | **ชื่อ** | **คุณสมบัติ** | **คำอธิบาย** | **หมายเหตุ** |
| 9 | IsCancelled | bit | สถานะยกเลิกรุ่น |  |
| 10 | CreatedAt | datetime2 | วันที่สร้างข้อมูล |  |

**ตารางที่ 3.14** ลงทะเบียนหลักสูตรอบรมระยะสั้น

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ชื่อแฟ้มข้อมูล: CourseAttendees (ข้อมูลลงทะเบียนหลักสูตรอบรมระยะสั้น)** | | | | |
| **ลำดับ** | **ชื่อ** | **คุณสมบัติ** | **คำอธิบาย** | **หมายเหตุ** |
| 1 | AppUserId | nvarchar | รหัสผู้ใช้งาน | (PK) |
| 2 | GenerationId | nvarchar | รหัสรุ่นของหลักสูตรอบรม | (FK) |
| 3 | CreatedAt | datetime2 | วันที่ลงทะเบียน |  |

**ตารางที่ 3.15** แสดงความคิดเห็นในหลักสูตรอบรมระยะสั้น

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ชื่อแฟ้มข้อมูล: CourseComments (ข้อมูลแสดงความคิดเห็นในหลักสูตรอบรมระยะสั้น)** | | | | |
| **ลำดับ** | **ชื่อ** | **คุณสมบัติ** | **คำอธิบาย** | **หมายเหตุ** |
| 1 | Id | nvarchar | รหัสความคิดเห็น | (PK) |
| 2 | AuthorId | nvarchar | รหัสผู้เขียน | (FK) |
| 3 | GenerationId | nvarchar | รหัสรุ่นของหลักสูตรอบรม | (FK) |
| 4 | Body | nvarchar | เนื้อหา |  |
| 5 | CreatedAt | datetime2 | วันที่แสดงความคิดเห็น |  |

**ตารางที่ 3.16** หลักสูตรวิทยาการคอมพิวเตอร์

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ชื่อแฟ้มข้อมูล: Syllabuses (ข้อมูลหลักสูตรวิทยาการคอมพิวเตอร์)** | | | | |
| **ลำดับ** | **ชื่อ** | **คุณสมบัติ** | **คำอธิบาย** | **หมายเหตุ** |
| 1 | Id | nvarchar | รหัสหลักสูตร | (PK) |
| 2 | NameTH | nvarchar | ชื่อหลักสูตรภาษาไทย |  |
| 3 | NameEN | nvarchar | ชื่อหลักสูตรอังกฤษ |  |
| 4 | Year | nvarchar | ปีการศึกษา |  |
| 5 | DegreeTH | nvarchar | ชื่อปริญญาภาษาไทย |  |
| 6 | DegreeEN | nvarchar | ชื่อปริญญาภาษาอังกฤษ |  |
| 7 | PDF | nvarchar | ที่อยู่เอกสารหลักสูตร |  |
| 8 | IsUsed | bit | สถานการณ์ใช้งาน |  |
| **ลำดับ** | **ชื่อ** | **คุณสมบัติ** | **คำอธิบาย** | **หมายเหตุ** |
| 9 | Hidden | bit | สถานะการซ่อนข้อมูล |  |
| 10 | CreatedAt | datetime2 | วันที่สร้างข้อมูล |  |

**ตารางที่ 3.17** วิชาในหลักสูตรวิทยาการคอมพิวเตอร์

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ชื่อแฟ้มข้อมูล: Subjects (ข้อมูลวิชาในหลักสูตรวิทยาการคอมพิวเตอร์)** | | | | |
| **ลำดับ** | **ชื่อ** | **คุณสมบัติ** | **คำอธิบาย** | **หมายเหตุ** |
| 1 | Id | nvarchar | รหัสวิชา | (PK) |
| 2 | SyllabusId | nvarchar | รหัสหลักสูตร | (FK) |
| 3 | Name | nvarchar | ชื่อวิชา |  |
| 4 | Credit | real | หน่วยกิต |  |
| 5 | SubjectCategory | int | ประเภทของวิชา |  |

**ตารางที่ 3.18** วัตถุประสงค์ของหลักสูตรวิทยาการคอมพิวเตอร์

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ชื่อแฟ้มข้อมูล: Objectives (ข้อมูลวัตถุประสงค์ของหลักสูตรวิทยาการคอมพิวเตอร์)** | | | | |
| **ลำดับ** | **ชื่อ** | **คุณสมบัติ** | **คำอธิบาย** | **หมายเหตุ** |
| 1 | Id | nvarchar | รหัสวิชา | (PK) |
| 2 | SyllabusId | nvarchar | รหัสหลักสูตร | (FK) |
| 3 | Sentence | nvarchar | วัตถุประสงค์ |  |

**ตารางที่ 3.19**  สายอาชีพในหลักสูตรวิทยาการคอมพิวเตอร์

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ชื่อแฟ้มข้อมูล: Occupations (ข้อมูลสายอาชีพในหลักสูตรวิทยาการคอมพิวเตอร์)** | | | | |
| **ลำดับ** | **ชื่อ** | **คุณสมบัติ** | **คำอธิบาย** | **หมายเหตุ** |
| 1 | Id | nvarchar | รหัสวิชา | (PK) |
| 2 | SyllabusId | nvarchar | รหัสหลักสูตร | (FK) |
| 3 | Sentence | nvarchar | อาชีพที่สามารถประกอบได้ |  |

**ตารางที่ 3.20** ภาพประชาสัมพันธ์ของสาขา

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ชื่อแฟ้มข้อมูล: HomePhotos (ข้อมูลภาพประชาสัมพันธ์ของสาขา)** | | | | |
| **ลำดับ** | **ชื่อ** | **คุณสมบัติ** | **คำอธิบาย** | **หมายเหตุ** |
| 1 | Id | nvarchar | รหัสภาพประชาสัมพันธ์ | (PK) |
| **ลำดับ** | **ชื่อ** | **คุณสมบัติ** | **คำอธิบาย** | **หมายเหตุ** |
| 2 | Title | nvarchar | หัวข้อ |  |
| 3 | Url | nvarchar | ที่อยู่ของรูปภาพ |  |
| 4 | IsEnable | bit | สถานการณ์แสดงผล |  |
| 5 | IsUsed | bit | สถานการณ์ใช้งาน |  |
| 6 | CreatedAt | datetime2 | วันที่สร้างข้อมูล |  |

**ตารางที่ 3.21** อาจารย์ประจำสาขา

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ชื่อแฟ้มข้อมูล: Lecturers (ข้อมูลอาจารย์ประจำสาขา)** | | | | |
| **ลำดับ** | **ชื่อ** | **คุณสมบัติ** | **คำอธิบาย** | **หมายเหตุ** |
| 1 | Id | nvarchar | รหัสอาจารย์ | (PK) |
| 2 | FullName | nvarchar | ชื่อ-นามสกุล |  |
| 3 | Prefix | nvarchar | คำนำหน้าชื่อ |  |
| 4 | Position | nvarchar | ตำแหน่งการทำงาน |  |
| 5 | Image | nvarchar | รูปภาพประจำตัว |  |
| 6 | Expert | nvarchar | ความชำนาญ |  |
| 7 | Hidden | bit | สถานะการซ่อนข้อมูล |  |
| 8 | CreatedAt | datetime2 | วันที่สร้างข้อมูล |  |

**ตารางที่ 3.22** ตั้งค่าของระบบ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ชื่อแฟ้มข้อมูล: SystemSettings (ข้อมูลการตั้งค่าของระบบ)** | | | | |
| **ดับ** | **ชื่อ** | **คุณสมบัติ** | **คำอธิบาย** | **หมายเหตุ** |
| 1 | Id | nvarchar | รหัสตั้งค่า | (PK) |
| 2 | WebName | nvarchar | ชื่อเว็บ |  |
| 3 | RegisterUrl | nvarchar | ลิงค์ลงทะเบียนเรียน |  |
| 4 | KruUrl | nvarchar | ลิงค์เว็บมหาวิทยาลัย |  |
| 5 | PageFacebook | nvarchar | ลิงค์เพจเฟสบุ๊ค |  |
| 6 | Logo | nvarchar | รูปภาพสัญลักษณ์ |  |
| 7 | VideoUrl | nvarchar | ลิงค์วิดีโอแนะนำสาขา |  |
| 8 | Location | nvarchar | พิกัดสถานที่ของสาขา |  |

**ตารางที่ 3.23** วิชาที่น่าสนใจในสาขา

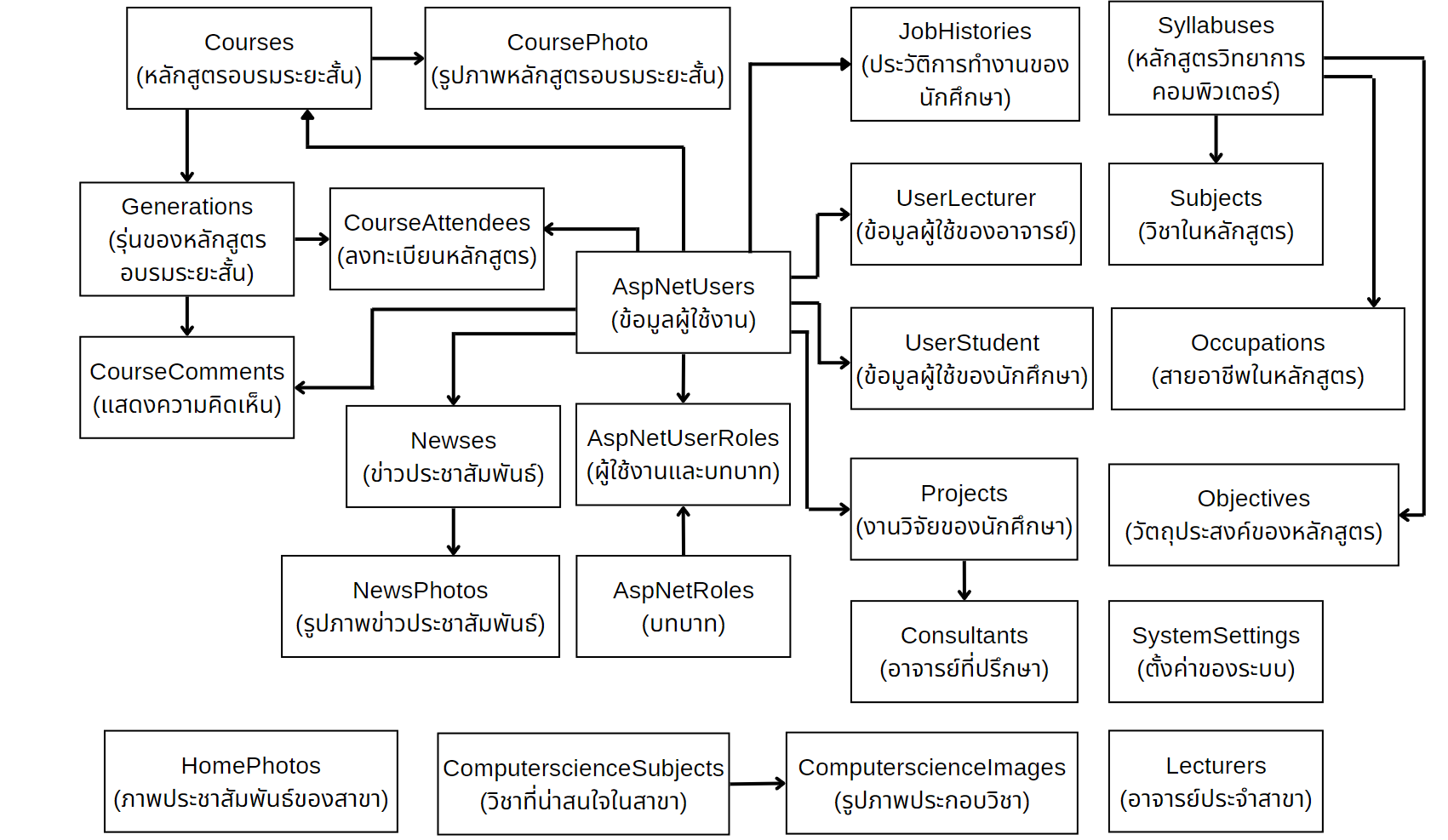
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ชื่อแฟ้มข้อมูล: ComputerScienceSubjects (ข้อมูลวิชาที่น่าสนใจในสาขา)** | | | | |
| **ลำดับ** | **ชื่อ** | **คุณสมบัติ** | **คำอธิบาย** | **หมายเหตุ** |
| 1 | Id | nvarchar | รหัสวิชา | (PK) |
| 2 | Icon | nvarchar | รูปภาพประจำวิชา |  |
| 3 | SubjectName | nvarchar | ชื่อวิชา |  |
| 4 | SubTitle | nvarchar | คำบรรยาย |  |
| 5 | Description | nvarchar | รายละเอียด |  |
| 6 | CreatedAt | datetime2 | วันที่สร้างข้อมูล |  |

**ตารางที่ 3.24** รูปภาพประกอบวิชาที่น่าสนใจในสาขา

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ชื่อแฟ้มข้อมูล: ComputerScienceImages (ข้อมูลรูปภาพประกอบวิชาที่น่าสนใจในสาขา)** | | | | |
| **ลำดับ** | **ชื่อ** | **คุณสมบัติ** | **คำอธิบาย** | **หมายเหตุ** |
| 1 | Id | nvarchar | รหัสรูปภาพ | (PK) |
| 2 | ComputerScienceSubjectId | nvarchar | รหัสวิชา | (FK) |
| 3 | URL | nvarchar | ที่อยู่ของรูปภาพ |  |

3.2 คลาสไดอะแกรม

จากตารางฐานข้อมูลที่ได้กล่าวมาข้างต้นจะใช้ในการเก็บข้อมูลภายในระบบ โดยหัวข้อนี้จะบอกความสัมพันธ์ของระบบฐานข้อมูลโดยใช้คลาสไดอะแกรม (Class Diagram) ซึ่งเป็นแผนภาพแสดงความสัมพันธ์ ดังภาพที่ 3.9



**ภาพที่ 3.9** ความสัมพันธ์ของระบบฐานข้อมูล

จากภาพที่ 3.9 เป็นความสัมพันธ์ระหว่างคลาส โดยมีรายละเอียดดังตารางที่ 3.25

**ตารางที่ 3.25** ตารางความสัมพันธ์ของคลาสไดอะแกรม

| Class Diagram | ความสัมพันธ์ |
| --- | --- |
| AspNetRoles กับ AspNetUserRoles | ข้อมูลบาทบาทหนึ่งบทบาทสามารถนำไปใช้กับผู้ใช้ได้หลายผู้ใช้ เป็นความสัมพันธ์แบบหนึ่งต่อกลุ่ม |
| AspNetUsers กับ AspNetUserRoles | ผู้ใช้งานหนึ่งคนสามารถมีบทบาทได้หลายบทบาท เป็นความสัมพันธ์แบบหนึ่งต่อกลุ่ม |
| AspNetUsers กับ Newses | ผู้ใช้งานหนึ่งคนสามารถเขียนข่าวประชาสัมพันธ์ได้หลายข่าว เป็นความสัมพันธ์แบบหนึ่งต่อกลุ่ม |
| Newses กับ NewsPhotos | ข่าวประชาสัมพันธ์หนึ่งข่าวสามารถมีรูปภาพประกอบข่าวได้หลายรูป เป็นความสัมพันธ์แบบหนึ่งต่อกลุ่ม |
| AspNetUsers กับ JobHistories | ผู้ใช้งานประเภทนักศึกษาหนึ่งคนสามารถกรอกประวัติของตัวได้ เป็นความสัมพันธ์แบบหนึ่งต่อหนึ่ง |
| AspNetUsers กับ UserStudent | ผู้ใช้งานประเภทนักศึกษาหนึ่งคนสามารถมีข้อมูลส่วนของนักศึกษาเพิ่มเติม เป็นความสัมพันธ์แบบหนึ่งต่อหนึ่ง |
| AspNetUsers กับ UserLecturer | ผู้ใช้งานประเภทอาจารย์หนึ่งคนสามารถมีข้อมูลส่วนของอาจารย์เพิ่มเติม เป็นความสัมพันธ์แบบหนึ่งต่อหนึ่ง |
| AspNetUsers กับ Projects | ผู้ใช้งานหนึ่งคนสามารถมีข้อมูลงานวิจัยได้หลายงาน เป็นความสัมพันธ์แบบหนึ่งต่อกลุ่ม |
| Projects กับ Consultants | งานวิจัยหนึ่งงานสามารถมีข้อมูลอาจารย์ที่ปรึกษาได้หลายคน เป็นความสัมพันธ์แบบหนึ่งต่อกลุ่ม |
| Courses กับ CoursePhoto | หลักสูตรอบรมระยะสั้นหนึ่งหลักสูตรสามารถมีภาพประกอบหลักสูตรได้หลายรูป เป็นความสัมพันธ์แบบหนึ่งต่อกลุ่ม |
| Courses กับ Generations | หลักสูตรอบรมระยะสั้นหนึ่งหลักสูตรสามารถเปิดหลักรอบการอบรมได้หลายรอบ เป็นความสัมพันธ์แบบหนึ่งต่อกลุ่ม |
| Generations กับ CourseAttendees | รอบการเปิดหลักสูตรอบรมระยะสั้นหนึ่งรอบ สามารถมีผู้เข้าร่วมอบรมได้หลายคน เป็นความสัมพันธ์แบบหนึ่งต่อกลุ่ม |
| AspNetUsers กับ CourseAttendees | ผู้ใช้งานหนึ่งคนสามารถเข้าร่วมรอบหลักสูตรที่เปิดอบรมได้หลายหลักสูตร เป็นความสัมพันธ์แบบหนึ่งต่อกลุ่ม |
| Generations กับ CourseComments | รอบการเปิดหลักสูตรอบรมระยะสั้นหนึ่งรอบ สามารถมีข้อความการแสดงความคิดเห็นได้หลายข้อความ เป็นความสัมพันธ์แบบหนึ่งต่อกลุ่ม |
| AspNetCourse กับ CourseComments | ผู้ใช้งานหนึ่งคนสามารถแสดงข้อความคิดเห็นภายใน รอบหลักสูตรนั้นได้หลายข้อความ เป็นความสัมพันธ์แบบหนึ่งต่อกลุ่ม |
| Syllabuses กับ Subjects | หลักสูตรวิทยากาคอมพิวเตอร์หนึ่งหลักสูตร สามารถมีข้อมูลรายวิชาได้หลายวิชา เป็นความสัมพันธ์แบบหนึ่งต่อกลุ่ม |
| Syllabuses กับ Occupations | หลักสูตรวิทยากาคอมพิวเตอร์หนึ่งหลักสูตร สามารถมีข้อมูลสายอาชีพที่ประกอบได้หลายอาชีพ เป็นความสัมพันธ์แบบหนึ่งต่อกลุ่ม |
| Syllabuses กับ Objectives | หลักสูตรวิทยากาคอมพิวเตอร์หนึ่งหลักสูตร สามารถมีข้อมูลวัตถุประสงค์ของหลักสูตรได้หลายข้อ เป็นความสัมพันธ์แบบหนึ่งต่อกลุ่ม |
| ComputerscienceSubjects กับ ComputerscienceImages | วิชาที่น่าสนใจหนึ่งวิชา สามารถมีรูปภาพประกอบวิชาได้หลายรูป เป็นความสัมพันธ์แบบหนึ่งต่อกลุ่ม |

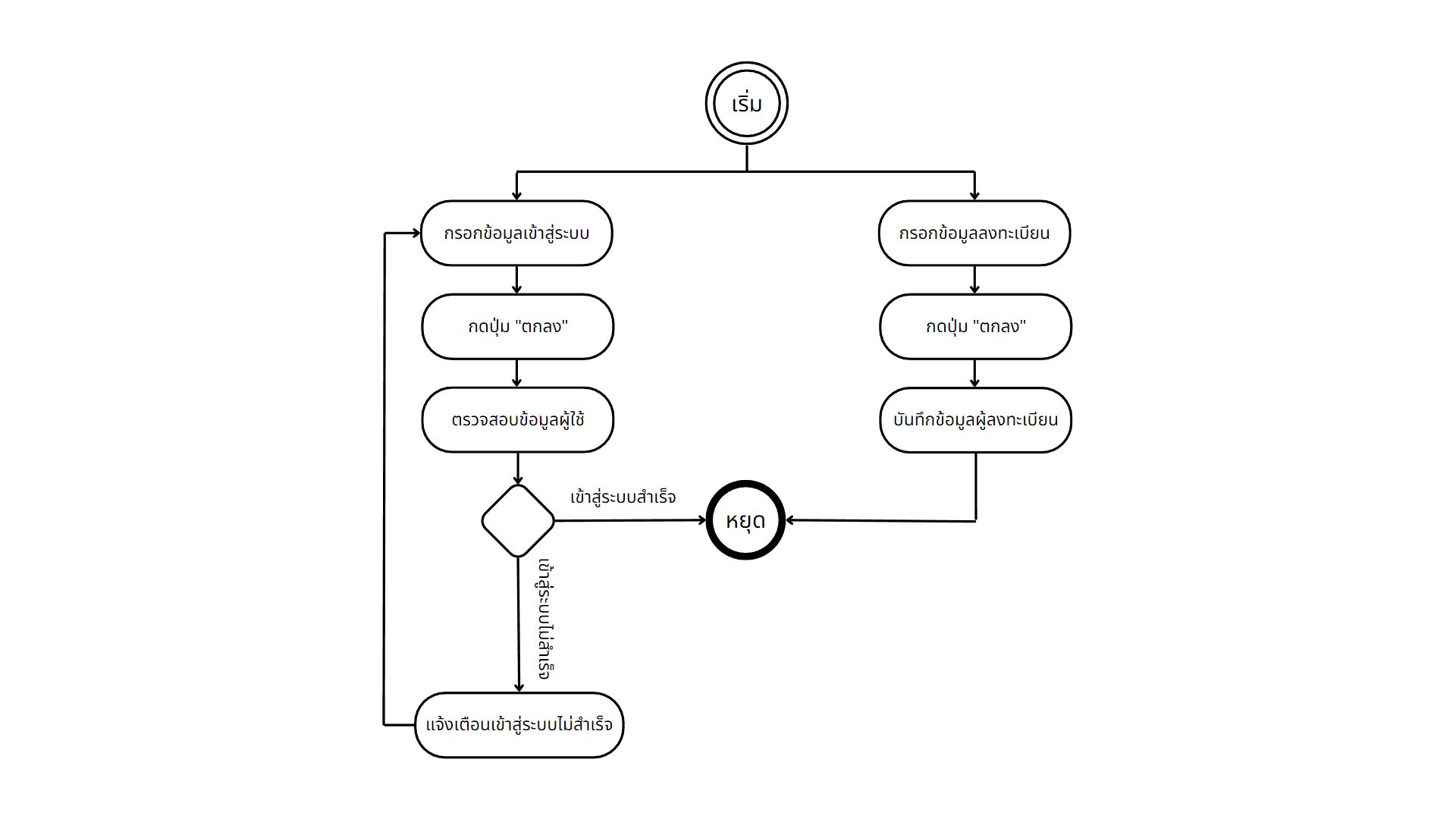
**4. พัฒนาและทดสอบระบบ**

ในการพัฒนาระบบทั้งหมดได้มีการแบ่งการทำงานเป็นระบบย่อยในส่วนนี้จะอธิบายการทำงานของแต่ละส่วนว่ามีการทำงานอะไรบ้าง โดยมีระบบย่อยอยู่ 6 ระบบ ได้แก่ 1) ระบบบริการสมาชิก 2) ระบบจัดการข่าวประชาสัมพันธ์ 3) ระบบลงทะเบียนหลักสูตรอบรมระยะสั้น 4) ระบบแสดงความคิดเห็นหลักสูตร 5) ระบบจัดการงานวิจัยนักศึกษา 6) ระบบจัดการประวัติทำงานนักศึกษา โดยมีรายละเอียดดังนี้

4.1 แผนภาพกิจกรรม

แผนภาพกิจกรรม (Activity Diagram) คือ แผนภาพที่ใช้อธิบายกิจกรรมที่เกิดขึ้นในลักษณะกระแสการไหลของการทำงาน โดยมีรายละเอียดดังนี้

4.1.1 ระบบบริการสมาชิก

 โดยมีแผนภาพกิจกรรมดังภาพที่ 3.10

**ภาพที่ 3.10** แผนภาพกิจกรรมของระบบบริการสมาชิก

จากภาพที่ 3.10 จะเป็นแผนภาพกิจกรรมของระบบบริการสมาชิก โดยจะแบ่งออกเป็น 2 ส่วนได้แก่ เข้าสู่ระบบ และลงทะเบียน โดยจะมีขั้นตอนการทำงานดังนี้

1) เข้าสู่ระบบ

1. กรอกข้อมูลเข้าสู่ระบบ

2. กดปุ่ม “ตกลง”

3. ตรวจสอบข้อมูลผู้ใช้

4. เข้าสู่ระบบ

2) ลงทะเบียน

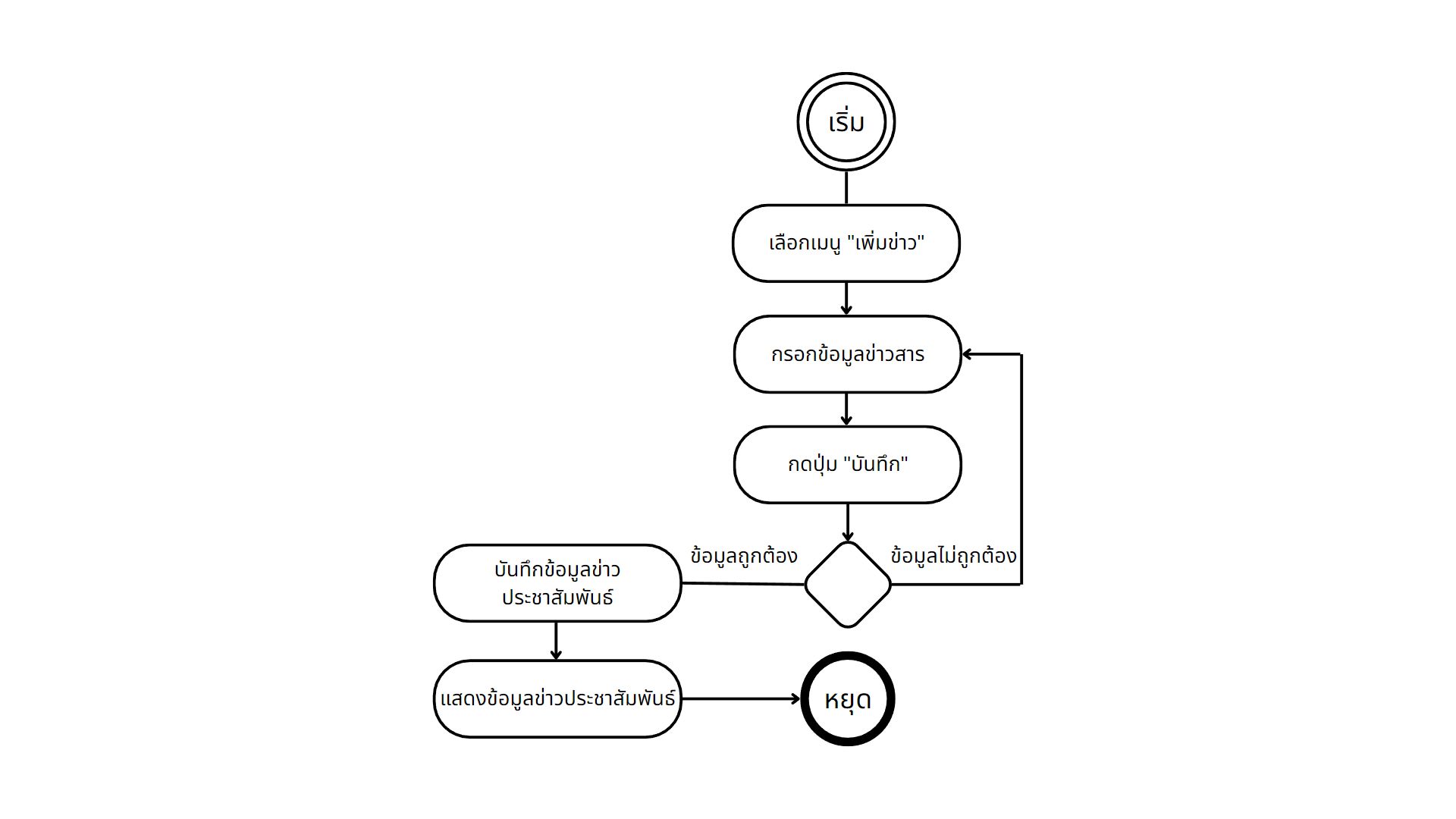
1. กรอกข้อมูลลงทะเบียน

2. กดปุ่ม “ตกลง”

3. ระบบทำการบันทึกข้อมูลผู้ลงทะเบียน

4. เข้าสู่ระบบ

4.1.2 ระบบจัดการข่าวประชาสัมพันธ์

 โดยมีแผนภาพกิจกรรมดังภาพที่ 3.11

**ภาพที่ 3.11** แผนภาพกิจกรรมของระบบจัดการข่าวประชาสัมพันธ์

จากภาพที่ 3.11 จะเป็นแผนภาพกิจกรรมของระบบจัดการข่าวประชาสัมพันธ์โดยจะมีขั้นตอนการทำงานดังนี้

1) จากหน้าจอแสดงข่าวประชาสัมพันธ์ เลือกเมนู “เพิ่มข่าว”

2) กรอกข้อมูลข่าวประชาสัมพันธ์

3) กดปุ่ม “บันทึก”

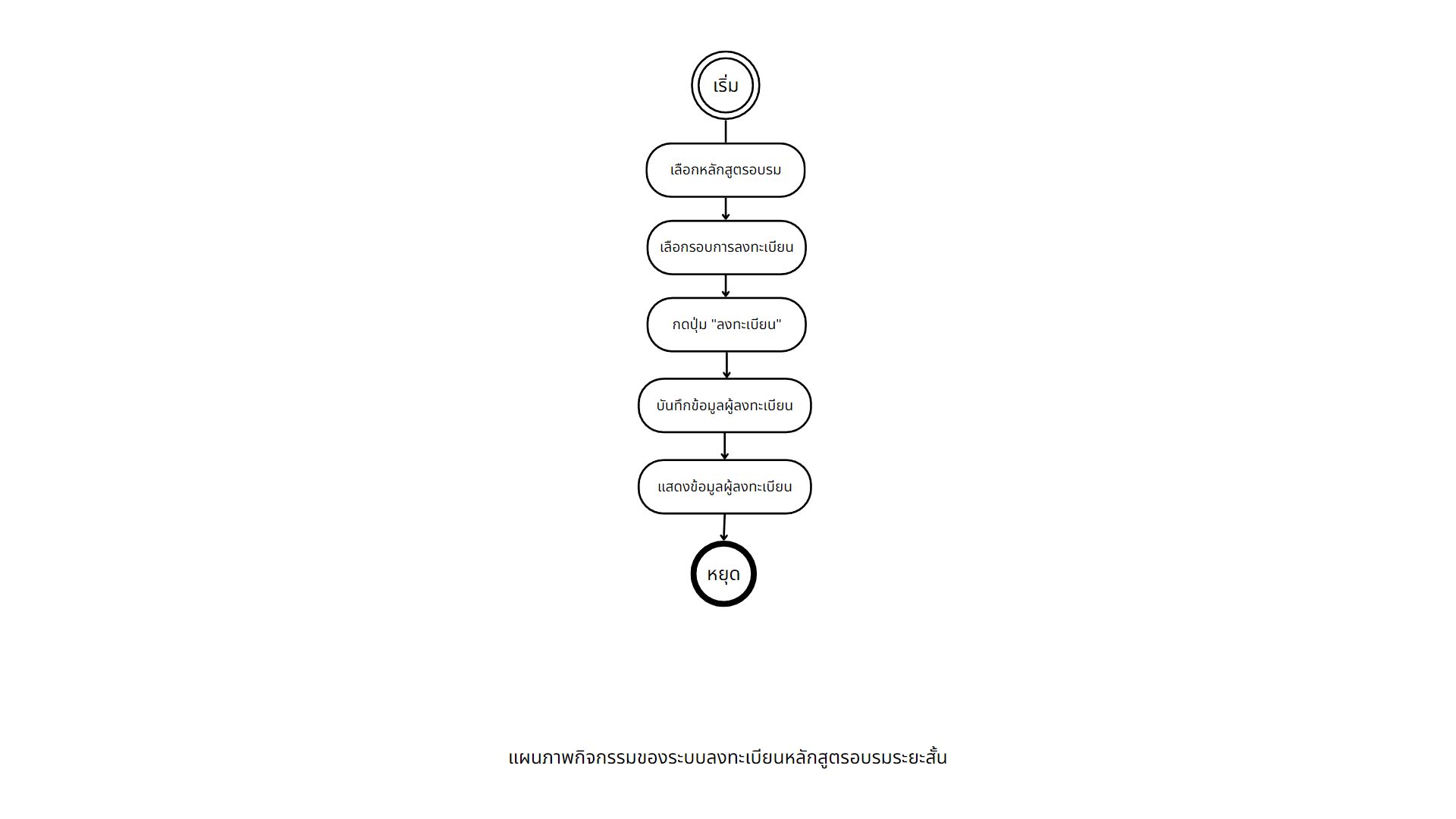
4) ระบบตรวจสอบความถูกต้อง

5) ระบบทำการบันทึกข้อมูล

6) แสดงข้อมูลข่าวประชาสัมพันธ์

4.1.3 ระบบลงทะเบียนหลักสูตรอบรมระยะสั้น

โดยมีแผนภาพกิจกรรมดังภาพที่ 3.12

**ภาพที่ 3.12** แผนภาพกิจกรรมของระบบลงทะเบียนหลักสูตรอบรมระยะสั้น

จากภาพที่ 3.12 จะเป็นแผนภาพกิจกรรมของระบบลงทะเบียนหลักสูตรอบรมระยะสั้นประชาสัมพันธ์ โดยจะมีขั้นตอนการทำงานดังนี้

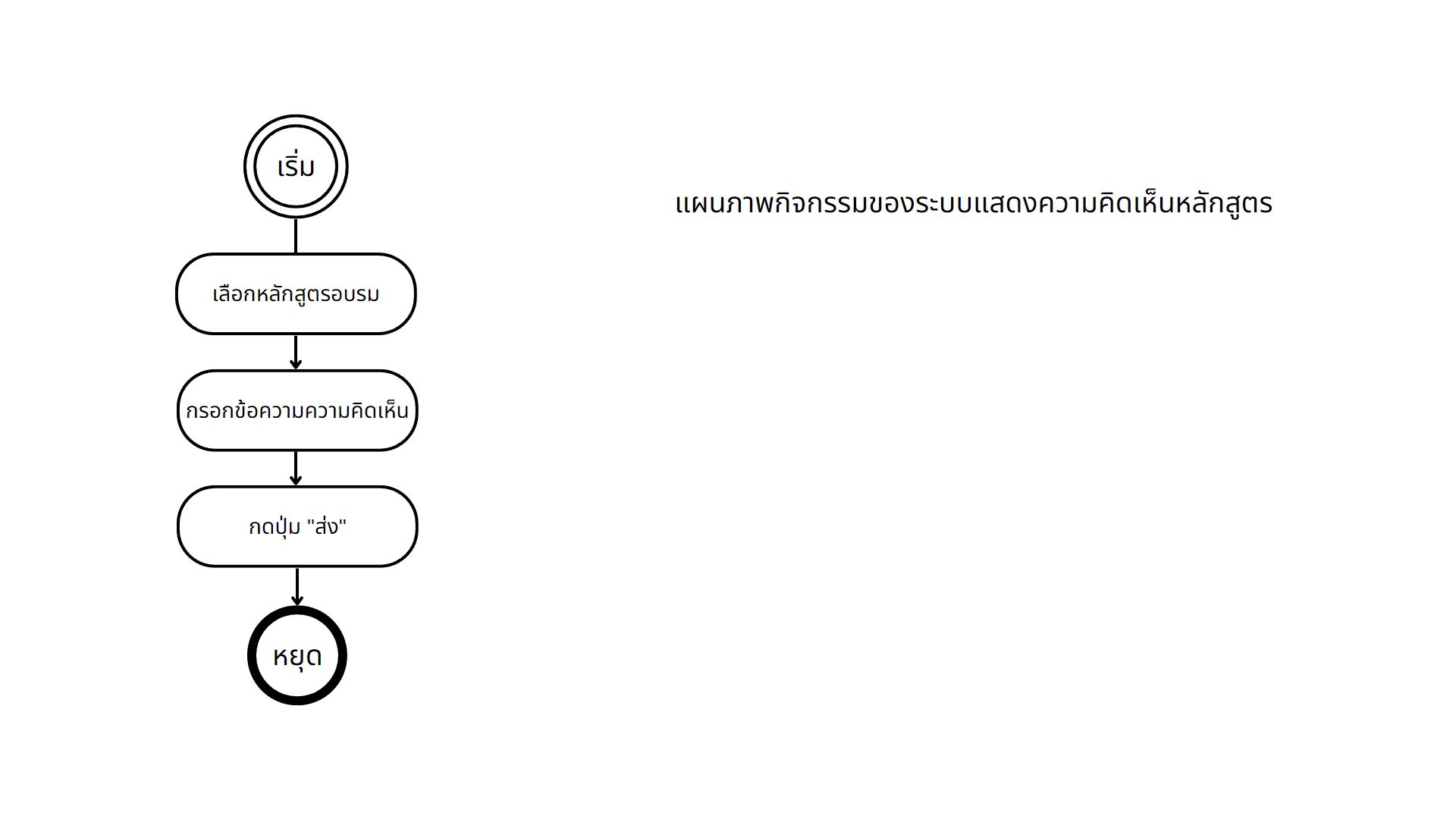
1) เลือกหลักสูตรอบรม

2) เลือกรอบการลงทะเบียน

3) กดปุ่ม “ลงทะเบียน”

4) ระบบทำการบันทึกข้อมูลลงทะเบียน

5) ระบบแสดงข้อมูลการลงทะเบียน

 4.1.4 ระบบแสดงความคิดเห็นหลักสูตร โดยมีแผนภาพกิจกรรมดังภาพที่ 3.13

**ภาพที่ 3.13** แผนภาพกิจกรรมของระบบแสดงความคิดเห็นหลักสูตร

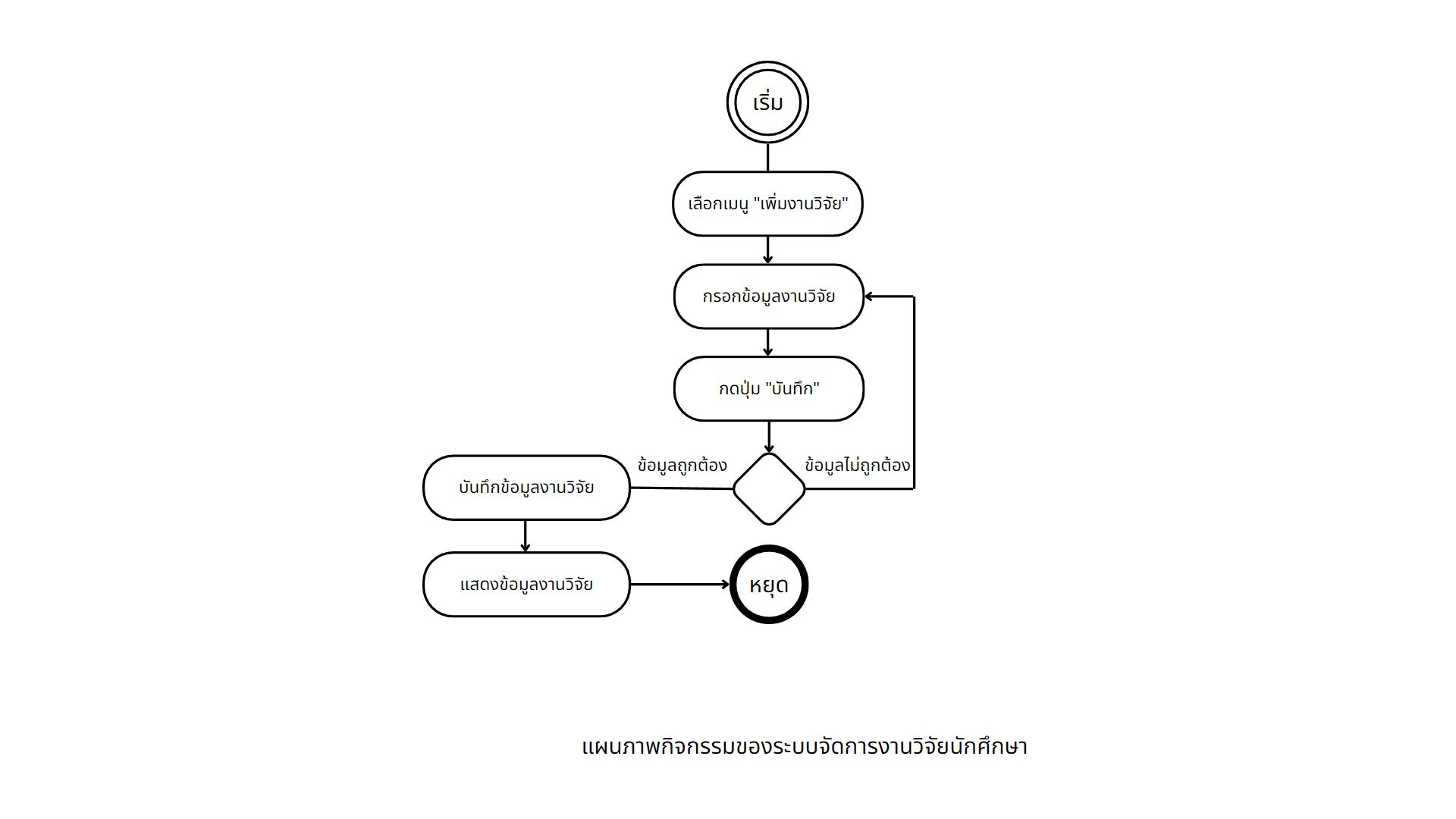
จากภาพที่ 3.13 จะเป็นแผนภาพกิจกรรมของระบบแสดงความคิดเห็นหลักสูตร โดยจะมีขั้นตอนการทำงานดังนี้

1) เลือกหลักสูตรอบรม

2) กรอกข้อความ ความคิดเห็น

3) กดปุ่ม “ส่ง”

4.1.5 ระบบจัดการงานวิจัยนักศึกษา

 โดยมีแผนภาพกิจกรรมดังภาพที่ 3.14

**ภาพที่ 3.14** แผนภาพกิจกรรมของระบบจัดการงานวิจัยนักศึกษา

จากภาพที่ 3.14 จะเป็นแผนภาพกิจกรรมของระบบจัดการงานวิจัยนักศึกษาโดยจะมีขั้นตอนการทำงานดังนี้

1) เลือกเมนู “เพิ่มงานวิจัย”

2) กรอกข้อมูลงานวิจัย

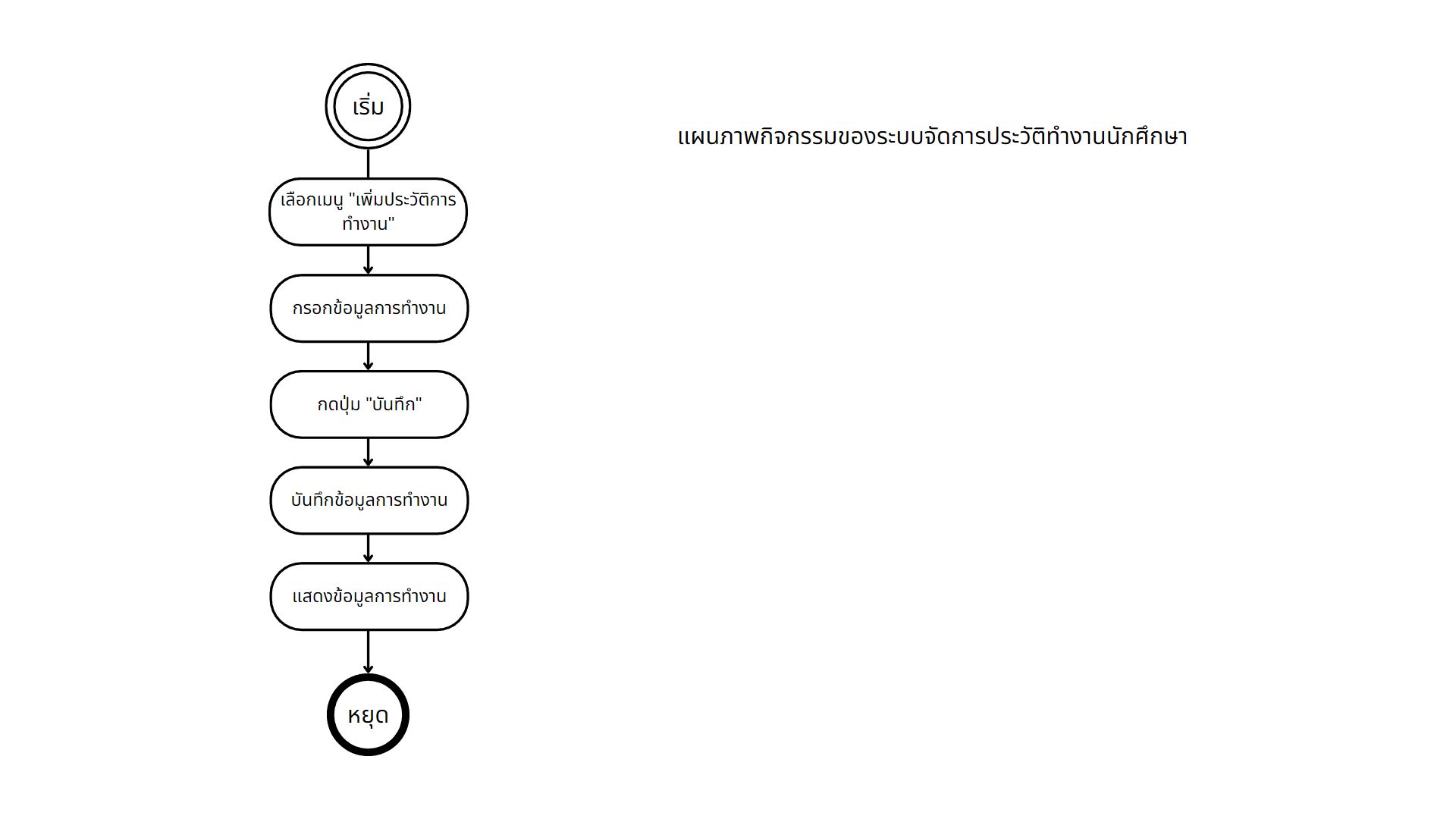
3) กดปุ่ม “บันทึก”

4) ระบบตรวจสอบความถูกต้อง

5) ระบบทำการบันทึกข้อมูล

6) แสดงรายละเอียดข้อมูลงานวิจัย

4.1.6 ระบบจัดการประวัติทำงานนักศึกษา โดยมีแผนภาพกิจกรรมดังภาพที่ 3.15

**ภาพที่ 3.15** แผนภาพกิจกรรมของระบบจัดการประวัติทำงานนักศึกษา

จากภาพที่ 3.15 จะเป็นแผนภาพกิจกรรมของระบบจัดการประวัติทำงานนักศึกษา โดยจะมีขั้นตอนการทำงานดังนี้

1) เลือกเมนู “เพิ่มประวัติการทำงาน”

2) กรอกข้อมูลการทำงาน

3) กดปุ่ม “บันทึก”

4) ระบบทำการบันทึกข้อมูล

5) แสดงข้อมูลการทำงาน

**5) สรุป**

จากขั้นตอนและวิธีการดำเนินงานที่ได้กล่าวมา ผู้จัดทำได้เริ่มศึกษาข้อมูลที่จำเป็นในการพัฒนา จากเอกสารของระบบงานเก่าเดิม และความต้องการของผู้ใช้งาน โดยจาการวิเคราะห์และออกแบบ ได้แก่ ระบบประชาสัมพันธ์ข่าวสารข้อมูลในสาขา ระบบจัดข้อมูลงานวิจัยของนักศึกษา และระบบจัดหลักสูตรอบรมระยะสั้น เป็นต้น จากนั้นจะเป็นขั้นตอนการออกแบบระบบฐานข้อมูล ซึ่งจะมีตารางข้อมูลทั้งหมด 24 ตาราง และมีตารางหลักเป็นตารางข้อมูลผู้ใช้งาน ตารางข้อมูลงานวิจัย ตารางข้อมูลข่าวประชาสัมพันธ์ และตารางข้อมูลหลักสูตรอบรมระยะสั้น เป็นต้น โดยจะออกแบบให้มีความสัมพันธ์กันเพื่อลดความซับซ้อนของข้อมูลเพื่อนำไปพัฒนาซอฟท์แวร์ และระบบที่ได้จะมีความทันสมัยและประสิทธิภาพที่ตรงตามความต้องการของผู้ใช้งาน

**บทที่ 4**

**ผลการดำเนินงาน**

**บรรณานุกรม**

[1] เอกราช ธรรมษา. (2559). **การเขียนโปรแกรมเชิงวัตถุ.** <http://www.com-> tech.ubru.ac.th/~ekarach/book/pcp/09\_OOP.pdf.

[2] Mindphp.com. (2565). **ระบบฐานข้อมูลคืออะไร.** <https://www.mindphp.com/คู่มือ/73-คืออะไร/2055-database-คืออะไร23.html>.

[3] **Thaksin University**. (2563). **ความสัมพันธ์ของระบบฐานข้อมูล.** https://cst.tsu.ac.th/courseonline/course/computer\_it/database/relationship.html.

[4] ณัฐดนัย อุปทอง. (2563**). ความหมายของ ORM.** <https://www.mindphp.com/คู่มือ/73-คืออะไร/3247-orm-doctrine.html>.

[5] Box Single. (2556). **ORM (Object Relational Mapping).** <https://www.boxsingle.com/3/ORM-(Object-Relational-Mapping)-คือ>

[6] Hayatee e-pong. (2565). **API คืออะไร.** <https://www.mindphp.com/คู่มือ/73-คืออะไร/8082-> what-is-an-api.html.

[7] Settawat Janpuk. (2562). **ทำความรู้จักกับ REST API.** <https://medium.com/@settawatjanpuk/beginner-มาทำความรู้จักกับ-rest-api-กันดีกว่า-มีตัวอย่าง-c855d93beaf3>.

[8] Mos Programmer. (2562). **รู้จัก React.** <https://mosprogramer.medium.com/มารู้จัก-react-กันเถอะ-eb04ab8b8ec1>.

[9] hilla. (2565). **Application State Management With MobX.** https://hilla.dev/docs/tutorials/in-depth-course/application-state-management-with-mobx

[10] Mindphp.com. (2560). **ซีชาร์ป คือ ภาษาคอมพิวเตอร์ ซึ่งพัฒนามาจากภาษา C++.** https://www.mindphp.com/คู่มือ/73-คืออะไร/2184-c-ชาร์ป-คืออะไร.html.

[11] Mindphp.com. (2560). **Web Server คืออะไร.** <https://www.mindphp.com/คู่มือ/73-คืออะไร/2053-web-server-คืออะไร.html>.