



## รายงานการปฏิบัติงานสหกิจศึกษา

การออกแบบและพัฒนาระบบส่วนหลังเพื่อจัดการเนื้อหาเว็บไซต์  
Design and Development of a Backend System for Website  
Content Management

บริษัท แอ็บโซลูท เมเนจเม้นท์ โซลูชันส์ จำกัด  
Absolute Management Solutions Co., Ltd.

โดย

นายดนุพล เรือมอัมพร 6504800026

นายเคารพ 6504800006

รายงานนี้เป็นส่วนหนึ่งของรายวิชา 128-491 สหกิจศึกษาสำหรับนักวิทยาการ  
คอมพิวเตอร์

หลักสูตรวิทยาศาสตรบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์

คณะวิทยาศาสตร์ มหาวิทยาลัยสยาม

ภาคการศึกษาที่ 3 ปีการศึกษา 2567



## รายงานการปฏิบัติงานสหกิจศึกษา

การออกแบบและพัฒนาระบบส่วนหลังเพื่อจัดการเนื้อหาเว็บไซต์  
Design and Development of a Backend System for Website  
Content Management

บริษัท แอ็บโซลูท เมเนจเม้นท์ โซลูชันส์ จำกัด  
Absolute Management Solutions Co., Ltd.

โดย

นายดนุพล เรือมอัมพร 6504800026

นายเคารพ 6504800006

รายงานนี้เป็นส่วนหนึ่งของรายวิชา 128-491 สหกิจศึกษาสำหรับนักวิทยาการ  
คอมพิวเตอร์

หลักสูตรวิทยาศาสตรบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์

คณะวิทยาศาสตร์ มหาวิทยาลัยสยาม

ภาคการศึกษาที่ 3 ปีการศึกษา 2567

หัวข้อโครงการ

การออกแบบและพัฒนาระบบส่วนหลังเพื่อจัดการเนื้อหาเว็บไซต์  
Design and Development of a Backend System for Website  
Content Management

บริษัท แอ็บโซลูท เมเนจเม้นท์ โซลูชันส์ จำกัด

รายชื่อคณะผู้จัดทำ

นาย ดนุพล เรืองอัมพร 6504800026

นาย เคารพ 6504800006

หลักสูตร

วิทยาศาสตร์บัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์

อาจารย์นิเทศ

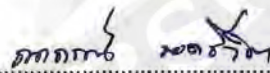
อาจารย์จรรยา แหยมเจริญ

อนุมัติให้โครงการนี้เป็นส่วนหนึ่งของการปฏิบัติสหกิจศึกษาและการศึกษาเชิงบูรณาการกับการทำงาน หลักสูตรวิทยาศาสตร์บัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยสยาม ภาคการศึกษาที่ 3 ปีการศึกษา 2567

คณะกรรมการสอบโครงการ

.....นางนงนุช.....อาจารย์นิเทศ  
( อาจารย์จรรยา แหยมเจริญ )

..........ผู้นิเทศ  
( นางสาวจันทร์สุดา ศิริไกรวัฒน์ )

..........กรรมการกลาง  
( อาจารย์ธนาภรณ์ รอดชีวิต )

..........ผู้ช่วยอธิการบดีและผู้อำนวยการสำนักสหกิจศึกษา  
( ผู้ช่วยศาสตราจารย์ ดร.มารุจ ลิ้มปะวัฒน์ )

## จดหมายนำส่งรายงาน

วันที่ 29 เดือน สิงหาคม พ.ศ. 2568

เรื่อง ขอส่งรายงานการปฏิบัติสหกิจศึกษา  
เรียน อาจารย์นิเทศ หลักสูตรวิทยาศาสตรบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์  
อาจารย์จรรยา แหยมเจริญ

ตามที่ นายดนุพล เรืองอัมพร และ นายเกรพ คณะผู้จัดทำ นักศึกษาหลักสูตรวิทยาศาสตรบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยสยาม ได้ไปปฏิบัติสหกิจศึกษาและการศึกษาเชิงบูรณาการกับการทำงาน ระหว่างวันที่ 19 พฤษภาคม 2568 ถึงวันที่ 29 สิงหาคม 2568 ศึกษา ในตำแหน่ง Front-End Developer ณ บริษัท แอ็บโซลูท เมเนจเม้นท์ โซลูชันส์ จำกัด และได้รับมอบหมายจากผู้นิเทศ (พนักงานที่ปรึกษา) ให้ศึกษา ออกแบบและพัฒนาระบบส่วนหัวสำหรับจัดการเนื้อหาบนเว็บไซต์

บัดนี้การปฏิบัติสหกิจศึกษาและการศึกษาเชิงบูรณาการกับการทำงานได้สิ้นสุดแล้ว คณะผู้จัดทำจึงขอส่งรายงานดังกล่าวมาพร้อมกันนี้จำนวน 1 เล่ม เพื่อขอรับคำปรึกษาต่อไป

จึงเรียนมาเพื่อโปรดพิจารณา

ขอแสดงความนับถือ

ลงชื่อ ..... ดนุพล .....

( นาย ดนุพล เรืองอัมพร )

ลงชื่อ ..... เกรพ .....

( นาย เกรพ )

คณะผู้จัดทำ

## กิตติกรรมประกาศ (Acknowledgement)

การที่ คณะผู้จัดทำ ได้มาปฏิบัติสหกิจศึกษา ในตำแหน่ง Front-End Developer ณ บริษัท แอ็บโซลูท เมเนจเม้นท์ โซลูชันส์ จำกัด ตั้งแต่วันที่ 19 พฤษภาคม 2568 ถึงวันที่ 29 สิงหาคม 2568 ได้สำเร็จลุล่วงตามวัตถุประสงค์ด้วยดี ส่งผลให้คณะผู้จัดทำได้รับความรู้ประสบการณ์การทำงานต่างๆ และความเข้าใจในชีวิตการทำงานจริง ที่เป็นประโยชน์ต่อการเรียนและสามารถนำความรู้ ประสบการณ์ที่ได้ไปใช้ในการประกอบอาชีพในอนาคต ด้วยความอนุเคราะห์อย่างยิ่งจาก บริษัท แอ็บโซลูท เมเนจเม้นท์ โซลูชันส์ จำกัด ที่ให้โอกาสคณะผู้จัดทำเข้ามาปฏิบัติสหกิจศึกษา กรุณาเสียสละเวลาอบรม สอนงาน และช่วยเหลือด้านต่างๆ ตลอดระยะเวลาในการปฏิบัติสหกิจศึกษาในครั้งนี้ จึงขอขอบพระคุณอย่างสูงมา ณ ที่นี้ จากการสนับสนุนหลายฝ่าย ดังนี้

1. คุณ จันทรสุดา ศิริไกรวัฒน์ ( Head Programmer/ผู้นิเทศ )
2. คุณ ชลักร์ แสงศิริรัตน ( DevOps/ผู้นิเทศ )
4. อาจารย์ จรรยา แหยมเจริญ (อาจารย์นิเทศ)

และบุคคลที่ไม่ได้กล่าวนามทุกท่าน ที่ได้ให้คำแนะนำในการจัดทำรายงานสหกิจศึกษานับนี้จนเสร็จสมบูรณ์

คณะผู้จัดทำหวังเป็นอย่างยิ่งว่ารายงานฉบับนี้จะเป็นประโยชน์ผู้ที่สนใจปฏิบัติสหกิจศึกษา เพื่อเป็นแนวทางเบื้องต้นในการทำความเข้าใจและพัฒนาโครงการต่อไป รวมทั้งในการค้นคว้าของผู้สนใจทั่วไปด้วย หากรายงานฉบับนี้มีข้อผิดพลาด ประการใดคณะผู้จัดทำก็ขออภัยมา ณ ที่นี้

دنۇفل رىئوئامفر  
كارف

คณะผู้จัดทำ

29 / สิงหาคม / 2568

ชื่อโครงการ : การออกแบบและพัฒนาระบบส่วนหลังเพื่อจัดการเนื้อหาเว็บไซต์  
หน่วยกิต : 5 หน่วยกิต  
คณะผู้จัดทำ : นายดนุพล เรือมอัมพร 6504800026,  
นายเครพ 6504800006  
อาจารย์ที่ปรึกษา : อาจารย์จรรยา แหยมเจริญ  
ระดับการศึกษา : ปริญญาตรี  
หลักสูตร : วิทยาศาสตร์บัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์  
คณะ : วิทยาศาสตร์  
ภาคการศึกษา/ปีการศึกษา : 3/2567

### บทคัดย่อ

โครงการสหกิจศึกษานี้มุ่งเน้นการออกแบบและพัฒนาส่วนติดต่อผู้ใช้ (Frontend) ของระบบหลังบ้าน (Backend) สำหรับการจัดการเนื้อหาบนเว็บไซต์ โดยใช้เทคโนโลยี Angular Framework ร่วมกับ Tailwind CSS และ Angular Material ระบบที่พัฒนาขึ้นช่วยให้ผู้ดูแลเว็บไซต์สามารถเพิ่มแก้ไข และจัดการเนื้อหาบนเว็บไซต์ได้โดยไม่ต้องมีทักษะการเขียนโปรแกรม โดยสามารถเชื่อมต่อกับ API ของระบบ Backend ได้อย่างมีประสิทธิภาพ นอกจากนี้ ระบบยังรองรับการใช้งานบนอุปกรณ์หลายประเภท (Responsive Design) และมีการออกแบบ UI ที่ใช้งานง่าย (User-Friendly Interface) ผลลัพธ์จากโครงการนี้สามารถนำไปใช้ในการจัดการเว็บไซต์ได้จริง ช่วยลดเวลาและความผิดพลาดในการปรับปรุงข้อมูลบนเว็บไซต์

คำสำคัญ : การจัดการเนื้อหาเว็บไซต์, ระบบส่วนหน้า, ระบบส่วนหลัง, การออกแบบ UX/UI

Project Title: Design and Development of a Backend System for Website Content Management

Credits: 5 Units

By: Mr. Danupol Reangamporn 6504800026  
Mr. Kaorop 6504800006

Advisor: Miss Janya Yamcharoen

Degree: Bachelor of Science

Major: Computer Science

Faculty: Science

Semester/Academic year: 3/2024

### Abstract

This co-operative education project focuses on designing and developing the user interface (Frontend) for the backend system of website content management. Using Angular Framework, Tailwind CSS, and Angular Material, the developed system allows website administrators to add, edit, and manage website content without requiring programming skills. It also integrates effectively with the Backend system's API. Additionally, the system supports multi-device usage (Responsive Design) and features a user-friendly interface (User-Friendly Interface). The outcome of this project can be used for practical website management, reducing time and errors when updating website content.

**Keywords:** website content management, frontend system, backend system, UX/UI design

.....

(Co-op Advisor.)

Approved by

.....

## สารบัญ

|                                                                | หน้า |
|----------------------------------------------------------------|------|
| จดหมายนำส่งรายงาน                                              | ก    |
| กิตติกรรมประกาศ                                                | ข    |
| บทคัดย่อ                                                       | ค    |
| บทคัดย่อภาษาอังกฤษ (Abstract)                                  | ง    |
| บทที่ 1 บทนำ                                                   |      |
| 1.1 ที่มาและความสำคัญของปัญหา.....                             | 1    |
| 1.2 วัตถุประสงค์ของโครงการ.....                                | 1    |
| 1.3 ขอบเขตของโครงการ.....                                      | 2    |
| 1.4 ประโยชน์ที่ได้รับ.....                                     | 2    |
| 1.5 ขั้นตอนและวิธีการดำเนินงาน.....                            | 2    |
| 1.6 ระยะเวลาในการดำเนินงาน.....                                | 3    |
| 1.7 อุปกรณ์และเครื่องมือที่ใช้.....                            | 4    |
| บทที่ 2 การทบทวนเอกสารและวรรณกรรมที่เกี่ยวข้อง                 |      |
| 2.1 Angular .....                                              | 5    |
| 2.2 Visual Studio Code.....                                    | 9    |
| 2.3 HTML.....                                                  | 11   |
| 2.4 CSS.....                                                   | 12   |
| 2.5 SCSS.....                                                  | 13   |
| 2.6 Tailwind CSS.....                                          | 14   |
| 2.7 TypeScript.....                                            | 15   |
| 2.8 Git.....                                                   | 17   |
| 2.9 GitHub.....                                                | 18   |
| 2.10 Git Fork.....                                             | 19   |
| 2.11 Postman.....                                              | 20   |
| บทที่ 3 รายละเอียดการปฏิบัติงาน                                |      |
| 3.1 ชื่อและที่ตั้งสถานประกอบการ.....                           | 22   |
| 3.2 ลักษณะการประกอบการ ผลิตภัณฑ์การให้บริการหลักขององค์กร..... | 23   |

## สารบัญ (ต่อ)

|                                                                          | หน้า      |
|--------------------------------------------------------------------------|-----------|
| 3.4 ตำแหน่งงานและลักษณะงานที่ได้รับมอบหมาย.....                          | 24        |
| 3.4 ชื่อและตำแหน่งงานของพนักงานที่ปรึกษา.....                            | 24        |
| 3.5 ระยะเวลาที่ปฏิบัติงาน.....                                           | 24        |
| <b>บทที่ 4 ผลการปฏิบัติงานโครงการ</b>                                    |           |
| 4.1 รายละเอียดของโครงการ.....                                            | 25        |
| 4.2 เครื่องมือและเทคโนโลยีที่ใช้.....                                    | 25        |
| <b>บทที่ 5 สรุปผลและข้อเสนอแนะ</b>                                       |           |
| 5.1 สรุปผลโครงการ.....                                                   | 30        |
| 5.2 สรุปผลการปฏิบัติสหกิจศึกษา.....                                      | 31        |
| <b>บรรณานุกรม.....</b>                                                   | <b>33</b> |
| <b>ภาคผนวก.....</b>                                                      | <b>34</b> |
| ภาคผนวก ก ภาพขณะปฏิบัติงานสหกิจศึกษา.....                                | 35        |
| <b>ประวัติผู้จัดทำ</b>                                                   |           |
| <b>แบบสรุปโครงการสหกิจศึกษาและการศึกษาเชิงบูรณาการกับการทำงาน (CWIE)</b> |           |

## สารบัญตาราง

|                                                        | หน้า |
|--------------------------------------------------------|------|
| ตารางที่ 1.1 ขั้นตอนและระยะเวลาในการดำเนินโครงการ..... | 3    |



## สารบัญรูปภาพ

|                                                                          | หน้า |
|--------------------------------------------------------------------------|------|
| รูปที่ 2.1 สัญลักษณ์ของ Angular.....                                     | 5    |
| รูปที่ 2.2 แสดงตัวอย่างการใช้งาน Angular CLI.....                        | 6    |
| รูปที่ 2.3 การทำงานของ Angular Lifecycle (Angular Life Cycle Hooks)..... | 8    |
| รูปที่ 2.4 สัญลักษณ์ของ Visual Studio Code.....                          | 9    |
| รูปที่ 2.5 ตัวอย่างการใช้งาน Integrated Terminal ใน VS Code.....         | 10   |
| รูปที่ 2.6 ตัวอย่างการใช้งาน Integrated Terminal ใน VS Code.....         | 11   |
| รูปที่ 2.7 สัญลักษณ์ของ HTML.....                                        | 11   |
| รูปที่ 2.8 สัญลักษณ์ของ CSS.....                                         | 12   |
| รูปที่ 2.9 สัญลักษณ์ของ SCSS.....                                        | 13   |
| รูปที่ 2.10 สัญลักษณ์ของ Tailwind CSS.....                               | 14   |
| รูปที่ 2.11 ตัวอย่างของการใช้ Tailwind CSS.....                          | 15   |
| รูปที่ 2.7 สัญลักษณ์ของ TypeScript.....                                  | 15   |
| รูปที่ 2.12 สัญลักษณ์ของ TypeScript.....                                 | 16   |
| รูปที่ 2.13 สัญลักษณ์ของ git version control.....                        | 17   |
| รูปที่ 2.9 สัญลักษณ์ของ GitHub.....                                      | 18   |
| รูปที่ 2.9 สัญลักษณ์ของ Git Fork.....                                    | 19   |
| รูปที่ 2.11 สัญลักษณ์ของ POSTMAN.....                                    | 20   |
| รูปที่ 3.1 แผนที่ตั้ง บริษัท แอ็บโซลูท เมเนจเม้นท์ โซลูชันส์ จำกัด.....  | 22   |
| รูปที่ ก.1 ขณะปฏิบัติงาน.....                                            | 35   |

## บทที่ 1

### บทนำ

#### 1.1 ที่มาและความสำคัญของปัญหา

บริษัท แอ็บโซลูท เมเนจเม้นท์ โซลูชันส์ จำกัด ดำเนินธุรกิจหลักในการให้บริการระบบ ERP (Enterprise Resource Management) ที่ทำงานบนระบบคลาวด์ สำหรับการบริหารงานธุรกิจ ทั้งระบบบริหารจัดการคลังสินค้า, ค้าปลีก, ค้าส่ง, จัดซื้อ, การเงิน, ก่อสร้าง, อสังหาริมทรัพย์, บริหารงานขาย, ฝ่ายบุคคล และสื่อสารทางการตลาด และยังเป็นผู้ให้บริการทางด้านเทคโนโลยีทั้งระบบซอฟต์แวร์ ฮาร์ดแวร์ และอินเทอร์เน็ตทุกสรรพสิ่ง (IoT) โดยมุ่งเน้นการให้คำปรึกษาและพัฒนา ระบบการจัดการทรัพยากรขององค์กร (ERP) ที่สามารถปรับแต่งได้ตามความต้องการเฉพาะของ ลูกค้าในหลากหลายอุตสาหกรรม เช่น การผลิต การค้า และภาคบริการ

ในยุคดิจิทัล องค์กรจำเป็นต้องมีเว็บไซต์ที่สามารถอัปเดตข้อมูลได้อย่างรวดเร็วและแม่นยำ การพัฒนาเว็บไซต์โดยแก้ไขชุดคำสั่ง (Source Code) โดยตรงทำให้กระบวนการล่าช้าและเกิดข้อผิดพลาดได้ง่าย จึงเกิดความจำเป็นในการพัฒนาระบบส่วนหลังหรือหลังบ้าน (Back-Office/Administrator System) เพื่อให้ผู้ดูแลเว็บไซต์สามารถจัดการเนื้อหาได้ด้วยตนเองโดยไม่ต้องเขียนชุดคำสั่ง

โครงการสหกิจศึกษานี้มุ่งเน้นการพัฒนาส่วนติดต่อผู้ใช้ (Frontend) ของระบบหลังบ้าน (Backend) สำหรับผู้ดูแลเว็บไซต์ใช้ในการจัดการเนื้อหาของเว็บไซต์ (Content Management) โดยใช้เทคโนโลยี Angular Framework ร่วมกับ Tailwind CSS และ Angular Material เพื่อออกแบบระบบที่ใช้งานง่าย (User-Friendly) รองรับการใช้งานบนอุปกรณ์ที่หลากหลายแพลตฟอร์ม (Responsive Design) และสามารถเชื่อมโยงกับ API ได้อย่างมีประสิทธิภาพ นอกจากนี้ คณะผู้จัดทำ ยังได้พัฒนาส่วนการเชื่อมต่อข้อมูลไปยังระบบส่วนหลัง เพื่อให้การพัฒนาระบบเป็นไปอย่างสมบูรณ์ และสอดคล้องกันทั้งสองฝั่ง

#### 1.1 วัตถุประสงค์ของโครงการ

เพื่อออกแบบและพัฒนาส่วนติดต่อผู้ใช้ (Frontend) ของระบบหลังบ้าน (Backend) สำหรับการจัดการเนื้อหาบนเว็บไซต์

### 1.3 ขอบเขตของโครงการงาน

- 1.3.1 พัฒนาส่วนติดต่อกับผู้ใช้ (Frontend) ของระบบหลังบ้าน (Backend)
- 1.3.2 ใช้เทคโนโลยี Angular Framework, Tailwind CSS, และ Angular Material
- 1.3.3 พัฒนาฟังก์ชันพื้นฐานสำหรับการจัดการเนื้อหาบนเว็บไซต์ ได้แก่ การเพิ่ม แก้ไข และ ข้อมูลที่จะแสดงบนหน้าเว็บเพจต่างๆ ของเว็บไซต์
- 1.3.4 พัฒนาส่วนเชื่อมต่อกับ Backend API และตรวจสอบการทำงานร่วมกันของระบบบนเว็บไซต์จริง

### 1.4 ประโยชน์ที่ได้รับ

- 1.4.1 ผู้ดูแลเว็บไซต์สามารถจัดการเนื้อหาบนเว็บเพจได้ด้วยตนเองโดยไม่ต้องเขียนชุดคำสั่ง
- 1.4.2 ลดเวลาและความผิดพลาดในการปรับปรุงแก้ไขข้อมูลบนเว็บไซต์
- 1.4.3 ช่วยให้องค์กรมีระบบจัดการเว็บไซต์ที่ทันสมัยและใช้งานได้อย่างมีประสิทธิภาพ

### 1.5 ขั้นตอนและวิธีการดำเนินงาน

#### 1.5.1 การศึกษาเทคโนโลยีที่เกี่ยวข้อง (Technology Learning)

คณะผู้จัดทำได้ศึกษาและฝึกใช้งานเทคโนโลยีหลักที่ใช้ในการพัฒนา ได้แก่ Angular Framework, TypeScript, และ Tailwind CSS รวมถึงการใช้งาน Angular Material เพื่อให้มีความเข้าใจพื้นฐานและสามารถนำไปใช้ในการพัฒนาระบบจริงได้

#### 1.5.2 การศึกษาโครงสร้างระบบและความต้องการ (System Study & Requirement Analysis)

ศึกษาและทำความเข้าใจโครงสร้างระบบหลังบ้าน (Backend) เดิมที่มีอยู่แล้วของบริษัท รวมถึงศึกษาความต้องการของผู้ใช้งานและผู้ดูแลระบบ เพื่อใช้เป็นแนวทางในการพัฒนาส่วนติดต่อผู้ใช้ (Frontend) ให้สามารถรองรับการใช้งานจริงได้อย่างเหมาะสม

#### 1.5.3 การพัฒนาและปรับปรุงส่วนติดต่อผู้ใช้ (Frontend Development)

ดำเนินการพัฒนาและปรับปรุงส่วนติดต่อผู้ใช้บนโครงสร้างเดิม โดยใช้ Angular, TypeScript, Tailwind CSS และ Angular Material เพื่อให้ระบบมีความสมบูรณ์ ใช้งานง่าย และตอบสนองการใช้งานของผู้ดูแลเว็บไซต์ได้อย่างมีประสิทธิภาพ

#### 1.5.4 การเชื่อมต่อและทำงานร่วมกับทีม Backend (Integration & Collaboration)

ทำงานร่วมกับทีม Backend ในการเชื่อมต่อ API และตรวจสอบการรับ-ส่งข้อมูล รวมถึงการแก้ไขปัญหาที่เกิดขึ้นจากการเชื่อมโยงข้อมูลระหว่าง Frontend และ Backend เพื่อให้ระบบทำงานได้ถูกต้องสมบูรณ์

#### 1.5.5 การทดสอบระบบ (Testing & Validation)

ทดสอบระบบหลังบ้านที่พัฒนาขึ้น พร้อมตรวจสอบผลการทำงานกับเว็บไซต์จริง เพื่อให้มั่นใจว่าการจัดการข้อมูลเนื้อหาบนเว็บไซต์ผ่านระบบหลังบ้านสะท้อนผลได้อย่างถูกต้อง ครบถ้วน และตรงตามความต้องการของผู้ใช้งาน

#### 1.5.6 การสรุปผลและจัดทำรายงาน (Documentation & Reporting)

ผู้จัดทำได้สรุปผลการดำเนินงาน ปัญหาและอุปสรรคที่พบ รวมถึงข้อเสนอแนะในการปรับปรุงระบบ พร้อมทั้งจัดทำรายงานสหกิจศึกษาเพื่อนำเสนอแก่มหาวิทยาลัยและสถานประกอบการ

### 1.6 ระยะเวลาในการดำเนินงาน

โครงการสหกิจศึกษานี้ดำเนินการระหว่างวันที่ 19 พฤษภาคม 2568 ถึงวันที่ 29 สิงหาคม 2568 รวมระยะเวลา 15 สัปดาห์ โดยแบ่งการดำเนินงานออกเป็นขั้นตอนดังนี้

ตารางที่ 1.1 ขั้นตอนและระยะเวลาในการดำเนินโครงการ

| ขั้นตอนการดำเนินงาน     | พ.ค. 68 | มิ.ย. 68 | ก.ค. 68 | ส.ค. 68 |
|-------------------------|---------|----------|---------|---------|
| 1.ศึกษาเทคโนโลยี        | ←→      |          |         |         |
| 2.ศึกษาโครงสร้างระบบ    |         | ←→       |         |         |
| 3.พัฒนาและปรับปรุง      |         | ←→       | ←→      |         |
| 4.เชื่อมต่อและทดสอบระบบ |         | ←→       | ←→      |         |
| 5.จัดทำเอกสาร           |         |          |         | ←→      |

## 1.7 อุปกรณ์และเครื่องมือที่ใช้

### 1.7.1 ฮาร์ดแวร์

#### 1.7.1.1 เครื่องคอมพิวเตอร์โน้ตบุค GIGABYTE

- Intel Core i5- 12500H
- Ram 16 GB
- SSD 512 GB
- Windows 11 Home Single Language

#### 1.7.1.2 เครื่องคอมพิวเตอร์โน้ตบุค MSI

- Intel Core i5-12450H
- Ram 24 GB
- SSD 512 GB
- Windows 11 Home Single Language

### 1.7.2 ซอฟต์แวร์

#### 1.7.2.1 ระบบปฏิบัติการ Windows 11

#### 1.7.2.2 Visual Studio Code (VS Code)

#### 1.7.2.3 Angular Framework และ TypeScript

#### 1.7.2.4 Tailwind CSS และ Angular Material

#### 1.7.2.5 Git และ GitHub

#### 1.7.2.6 Git Fork

#### 1.7.2.7 Postman

#### 1.7.2.8 Google Chrome/Edge Developer Tools

## บทที่ 2

### การทบทวนเอกสารและวรรณกรรมที่เกี่ยวข้อง

ในการจัดทำโครงงานสหกิจศึกษาการออกแบบและพัฒนาระบบส่วนหลังเพื่อจัดการเนื้อหาเว็บไซต์ คณะผู้จัดทำได้ทำการศึกษาค้นคว้าแนวคิด ทฤษฎีเทคโนโลยี และเครื่องมือต่าง ๆ เพื่อนำมาประยุกต์ใช้ในการพัฒนาโครงงาน ประกอบด้วย

#### 2.1 Angular <sup>1</sup>



รูปที่ 2.1 สัญลักษณ์ของ Angular

Angular เป็น JavaScript Framework ที่พัฒนาโดย Google และได้รับความนิยมอย่างแพร่หลายในด้านการพัฒนา Single Page Applications (SPA) โดยเฉพาะในงาน Frontend Development สำหรับแอปพลิเคชันเว็บที่มีความซับซ้อน เนื่องจาก Angular มีฟีเจอร์สำคัญที่ช่วยในการจัดการข้อมูล การสร้างส่วนติดต่อผู้ใช้ (User Interface) ที่ตอบสนองรวดเร็ว รวมถึงการรองรับการทำงานหลายฟังก์ชันในหน้าเดียว เช่น Routing, Forms และ HTTP Requests ทั้งนี้ Angular ได้รับการออกแบบมาให้มีประสิทธิภาพสูงและรองรับการขยายระบบในอนาคต

##### 2.1.1 ทำความเข้าใจเบื้องต้นเกี่ยวกับ Angular

Angular ใช้ภาษา TypeScript ซึ่งถูกพัฒนาโดย Microsoft โดยมีคุณสมบัติที่สำคัญคือ Type Checking ที่ช่วยตรวจสอบชนิดข้อมูล ทำให้ชุดคำสั่งมีความปลอดภัย มีความเสถียร และง่ายต่อการบำรุงรักษา นอกจากนี้ Angular ยังมีสถาปัตยกรรมแบบ Component-based กล่าวคือ UI ถูกแบ่งออกเป็นส่วนประกอบย่อย (Components) ทำให้สามารถพัฒนา จัดการ และนำกลับมาใช้ซ้ำได้อย่างมีประสิทธิภาพ

---

<sup>1</sup> <https://perjerz.medium.com/angular-คืออะไร-เหมาะกับใคร-งานแบบไหน-64a5e33220a0>

อีกหนึ่งคุณสมบัติเด่นของ Angular คือ Two-Way Data Binding ซึ่งช่วยให้การเปลี่ยนแปลงข้อมูลใน Model สะท้อนมายัง View ได้ทันทีและการเปลี่ยนแปลงใน View ก็ส่งผลกลับไปยัง Model โดยอัตโนมัติ รวมถึงการใช้ Dependency Injection (DI) ที่ช่วยให้การจัดการบริการ (Service) ภายในระบบมีความสะดวกและยืดหยุ่น

### 2.1.2 การใช้งาน Angular CLI

Angular CLI (Command Line Interface) เป็นเครื่องมือที่สำคัญสำหรับการพัฒนาแอปพลิเคชัน Angular เพราะมันช่วยให้การสร้างโครงการใหม่ การเพิ่ม component, service, module และการทดสอบระบบทำได้ง่ายและรวดเร็ว ไม่ว่าจะเป็นการติดตั้ง dependencies หรือการสร้าง production build ก็สามารถทำได้ผ่านคำสั่ง CLI

Angular CLI ยังช่วยให้การพัฒนาและทดสอบระบบทำได้ในขั้นตอนเดียว โดยมีคำสั่งที่ช่วยในการเริ่มต้นเซิร์ฟเวอร์ในการทดสอบแอปพลิเคชัน, การสร้าง unit tests และการสร้าง production build ซึ่งทำให้กระบวนการพัฒนาเป็นระบบที่เป็นระเบียบและมีประสิทธิภาพมากยิ่งขึ้น

|                                 |                                                   |
|---------------------------------|---------------------------------------------------|
| <b>ng new my-app</b>            | สร้างโปรเจกต์ Angular ใหม่ในโฟลเดอร์ "my-app"     |
| <b>ng serve</b>                 | รันแอปพลิเคชันในโหมดพัฒนาและเปิดในเบราว์เซอร์     |
| <b>ng g c home</b>              | สร้าง component ใหม่ชื่อ "home"                   |
| <b>ng g s auth</b>              | สร้าง service ใหม่ชื่อ "auth"                     |
| <b>ng g g guard</b>             | สร้าง guard ใหม่ (ใช้สำหรับการป้องกัน route)      |
| <b>ng add @angular/material</b> | ติดตั้ง Angular Material เพื่อใช้งาน UI component |

รูปที่ 2.2 แสดงตัวอย่างการใช้งาน Angular CLI

ตัวอย่างการใช้งานคำสั่ง ng generate component ที่ช่วยในการสร้าง component ใหม่ในโครงการ Angular ทำให้การพัฒนาแอปพลิเคชันเป็นไปอย่างรวดเร็วและเป็นระเบียบ

### 2.1.3 ข้อดีของการใช้ Angular Framework ในการพัฒนาแอปพลิเคชัน

การใช้ Angular ในการพัฒนาแอปพลิเคชันเว็บมีข้อดีหลายประการ เช่น

1. การพัฒนาแบบ Component-based Architecture – ทำให้การจัดการส่วนต่างๆ ของแอปพลิเคชันเป็นไปอย่างมีระเบียบ
2. Two-way Data Binding – ทำให้การปรับปรุงแก้ไขข้อมูลใน UI และ Model เกิดขึ้นได้โดยอัตโนมัติ
3. Modular Development – Angular ช่วยให้สามารถแยกแยะชุดคำสั่งและบริการต่างๆ ออกเป็น module ที่สามารถนำมาใช้ใหม่ได้
4. RxJS Integration – Angular รองรับการใช้งาน Reactive Programming ซึ่งช่วยให้การจัดการข้อมูลที่มีการเปลี่ยนแปลงบ่อยๆ ทำได้ง่าย
5. Community Support – เนื่องจาก Angular เป็น framework ที่ได้รับการสนับสนุนจาก Google และมีชุมชนผู้พัฒนาที่ใหญ่ ทำให้สามารถหาแหล่งข้อมูลและการสนับสนุนได้ง่าย

โดยสรุปแล้ว Angular เป็นเครื่องมือที่มีคุณสมบัติครบถ้วนในการพัฒนาแอปพลิเคชันระดับองค์กร ช่วยให้นักพัฒนาสามารถทำงานได้อย่างมีประสิทธิภาพและสามารถขยายระบบได้อย่างไม่จำกัด

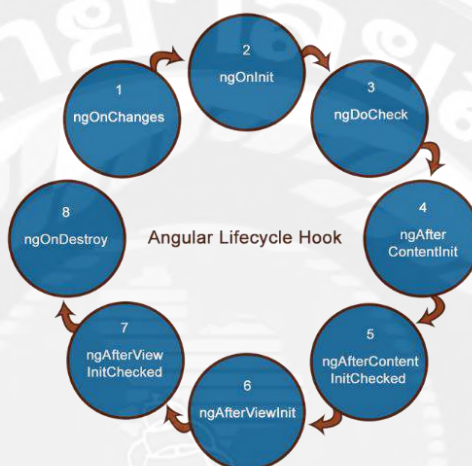
### 2.1.4 การทำงานของ Angular Lifecycle (Angular Life Cycle Hooks)

Angular Lifecycle คือ กระบวนการที่ Angular ใช้ในการควบคุมและจัดการกับสถานะของ component และ directive ต่างๆ ในแอปพลิเคชันในแต่ละช่วงเวลา ตั้งแต่การสร้าง component ใหม่ การปรับปรุงแก้ไขข้อมูล จนถึงการทำลาย (destroy) เมื่อไม่ใช้งานแล้ว นั่นคือสิ่งที่เรียกว่า Lifecycle Hooks

Angular มี lifecycle hooks ที่สำคัญหลายตัว ซึ่งแต่ละตัวจะถูกเรียกใช้ในช่วงเวลาที่ต่างกัน ของการทำงานของ component หรือ directive โดยแต่ละ lifecycle hook จะสามารถจัดการกับการเปลี่ยนแปลงที่เกิดขึ้นในแอปพลิเคชันได้อย่างมีประสิทธิภาพ

### 2.1.5 Lifecycle ของ Component<sup>2</sup>

ใน Angular แต่ละ Component จะมี Lifecycle Hooks ซึ่งเป็นชุดของ method ที่ Angular เรียกใช้งานโดยอัตโนมัติในช่วงเวลาต่างๆ ของการทำงานของ Component ตั้งแต่การสร้าง (creation) การแสดงผล (rendering) การตรวจสอบการเปลี่ยนแปลง (change detection) ไปจนถึงการทำลาย (destroy) Component การเข้าใจลำดับการทำงานนี้ช่วยให้นักพัฒนาสามารถควบคุมการทำงานและจัดการทรัพยากรได้อย่างเหมาะสม



รูปที่ 2.3 การทำงานของ Angular Lifecycle (Angular Life Cycle Hooks)

#### 1. ngOnChanges

เรียกใช้งานเมื่อมีการเปลี่ยนแปลงค่าใน @Input properties ที่ส่งมาจาก Parent Component

#### 2. ngOnInit

เรียกใช้งานครั้งเดียวหลังจาก Angular ได้กำหนดค่าเริ่มต้น (initial value) ให้กับ Component เหมาะสำหรับการทำงานที่ต้องการค่าเริ่มต้น เช่น การเรียก API

#### 3. ngDoCheck

ใช้สำหรับตรวจสอบและตอบสนองต่อการเปลี่ยนแปลงที่ Angular ไม่สามารถตรวจจับได้ อัตโนมัติ จะถูกเรียกทุกครั้งที่ Angular ทำการตรวจสอบ (change detection cycle)

<sup>2</sup> <https://www.angularminds.com/blog/angular-lifecycle-hooks-best-practices>

#### 4. **ngAfterContentInit**

เรียกใช้งานครั้งเดียวหลังจาก Angular โหลดเนื้อหาที่ส่งเข้ามาภายใน (content projection) ของ Component เสร็จสิ้น

#### 5. **ngAfterContentChecked**

เรียกทุกครั้งหลังจาก Angular ตรวจสอบการเปลี่ยนแปลงของเนื้อหาภายใน Component

#### 6. **ngAfterViewInit**

เรียกใช้งานครั้งเดียวหลังจาก Angular สร้างและแสดงผล View ของ Component และ Child Components ทั้งหมดเสร็จสิ้น เหมาะสำหรับการทำงานที่ต้องการ DOM พร้อมใช้งาน

#### 7. **ngAfterViewChecked**

เรียกทุกครั้งที่ Angular ตรวจสอบและรีเฟรช View ของ Component และ Child Components

#### 8. **ngOnDestroy**

เรียกใช้งานก่อนที่ Component จะถูกทำลาย เหมาะสำหรับการทำความสะอาด (cleanup) เช่น การยกเลิกการสมัคร Event, การหยุด Observable หรือการคืนค่าหน่วยความจำ

การใช้งาน Lifecycle Hooks อย่างเหมาะสม จะช่วยให้นักพัฒนาสามารถบริหารจัดการ State ของ Component และทรัพยากรของระบบได้อย่างมีประสิทธิภาพ

## 2.2 Visual Studio Code



รูปที่ 2.4 สัญลักษณ์ของ Visual Studio Code

Visual Studio Code (VS Code) เป็น Text Editor ยอดนิยมนักพัฒนาโดยเฉพาะในสายงาน Web Development เนื่องจากมีคุณสมบัติที่ครบถ้วน เหมาะสมต่อการพัฒนา Angular

ทั้งการรองรับ TypeScript, HTML, CSS, JavaScript และ Node.js ทำให้ VS Code กลายเป็นเครื่องมือหลักที่ช่วยให้การพัฒนาเว็บแอปพลิเคชันทำได้สะดวกและรวดเร็ว

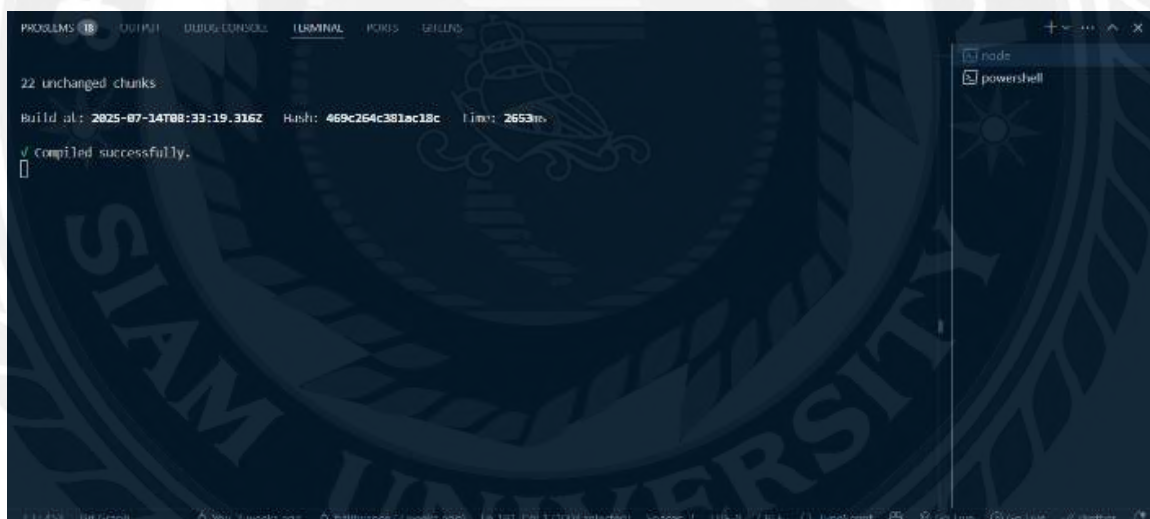
### 2.2.1 ข้อดีของ Visual Studio Code

1. **Lightweight และรวดเร็ว:** ใช้งานสิ้นไหลแม้มันในเครื่องที่ไม่ได้มีสเปกสูง
2. **รองรับหลายภาษา:** ใช้ได้ทั้ง Angular, TypeScript, JavaScript, Python, PHP, HTML, CSS และภาษาอื่น ๆ
3. **Customization:** สามารถติดตั้ง Extensions/Plugins ที่ช่วยให้ทำงานกับ Angular ได้ง่ายขึ้น เช่น Angular Essentials, Prettier, ESLint
4. **Full-stack Support:** มี Integrated Terminal และรองรับ Git อย่างเต็มรูปแบบ

### 2.2.2 ฟีเจอร์สำคัญของ VS Code สำหรับการพัฒนา Angular

#### 1. Integrated Terminal

ช่วยให้นักพัฒนาสามารถรันคำสั่งจาก Angular CLI ได้โดยตรง เช่น `ng serve`, `ng generate component`, `ng build` โดยไม่ต้องสลับหน้าต่างไปยัง Terminal อื่น ๆ



รูปที่ 2.5 ตัวอย่างการใช้งาน Integrated Terminal ใน VS Code

#### 2. Angular Extensions

VS Code มี Extensions มากมายที่ช่วยอำนวยความสะดวกในการพัฒนา Angular เช่น Angular Snippets, Angular Language Service ซึ่งช่วยแนะนำชุดคำสั่ง ตรวจสอบความถูกต้อง และเพิ่มประสิทธิภาพในการเขียนชุดคำสั่ง



รูปที่ 2.6 ตัวอย่างการใช้งาน Integrated Terminal ใน VS Code

จากคุณสมบัติที่กล่าวมา Visual Studio Code จึงเป็นเครื่องมือที่เหมาะสมอย่างยิ่งสำหรับการพัฒนา Angular โดยผู้จัดทำได้เลือกใช้ VS Code ในการแก้ไขและพัฒนาระบบหลังบ้าน (Admin System) ทางฝั่ง Frontend ซึ่งช่วยให้กระบวนการพัฒนาเป็นไปอย่างมีประสิทธิภาพและตอบสนองความต้องการของโครงการได้อย่างครบถ้วน

## 2.3 HTML



รูปที่ 2.7 สัญลักษณ์ของ HTML

HTML (HyperText Markup Language) เป็นเทคโนโลยีหลักที่ใช้ในการพัฒนาเว็บไซต์ มีบทบาทสำคัญในการกำหนด โครงสร้าง (Structure) ของหน้าเว็บ ในการพัฒนา Angular Application HTML ถูกใช้สำหรับสร้าง View ของแอปพลิเคชัน โดยผสมการทำงานร่วมกับ

Angular Directives และ Binding Syntax เพื่อให้หน้าเว็บสามารถโต้ตอบกับ Data และ Events ได้อย่างมีประสิทธิภาพ

### 2.3.1 การใช้ HTML ใน Angular

ใน Angular แต่ละ Component จะมีไฟล์ HTML สำหรับกำหนดโครงสร้างการแสดงผลของ UI เช่น Layout, Form, Button, Table เป็นต้น HTML ใน Angular สามารถผสมการทำงานกับ

- Property Binding – ใช้ส่งค่าจาก Component ไปยัง View
- Event Binding – ใช้รับเหตุการณ์จาก View ส่งกลับไปยัง Component
- Structural Directives (เช่น \*ngIf, \*ngFor) – ใช้ควบคุมโครงสร้างการแสดงผล

การใช้ HTML ใน Angular จึงไม่ใช่เพียงการกำหนดโครงสร้างคงที่ แต่ยังสามารถสร้าง Dynamic View ที่ตอบสนองต่อข้อมูลและการโต้ตอบของผู้ใช้ได้

## 2.4 CSS



รูปที่ 2.8 สัญลักษณ์ของ CSS

CSS (Cascading Style Sheets) เป็นภาษาที่ใช้สำหรับการกำหนดรูปแบบและการตกแต่ง (Styling) ของ HTML โดยใน Angular นักพัฒนาสามารถใช้ CSS เพื่อควบคุมการแสดงผลของ Component ให้มีความสวยงามและรองรับการใช้งานบนหลายอุปกรณ์ (Responsive Design)

### 2.4.1 การใช้ CSS ภายใน Component Angular

Angular รองรับการใช้ CSS ภายในแต่ละ Component โดยแยกไฟล์ CSS ออกมา เช่น app.component.css เพื่อให้แต่ละ Component มีสไตล์ของตัวเอง และยังมีระบบ View

Encapsulation ที่ช่วยป้องกันไม่ให้ CSS ของ Component หนึ่งไปกระทบกับ Component อื่น ๆ โดยตรง ส่งผลให้การพัฒนามีความเป็นอิสระและง่ายต่อการบำรุงรักษา

#### 2.4.2 การใช้ CSS ร่วมกับ Framework อื่น ๆ

แม้ว่า Angular จะสามารถใช้ CSS มาตรฐานได้โดยตรง แต่ในการพัฒนาระบบจริง นักพัฒนามักเลือกใช้ CSS Frameworks เช่น Tailwind CSS, Bootstrap หรือ Angular Material เพื่อช่วยให้การออกแบบ UI มีความสวยงามและเป็นมาตรฐาน โดยไม่จำเป็นต้องเขียนชุดคำสั่ง CSS ทั้งหมดเอง ซึ่งช่วยประหยัดเวลาและเพิ่มความสม่ำเสมอของงานออกแบบ

### 2.5 SCSS



รูปที่ 2.9 สัญลักษณ์ของ SCSS

SCSS (Sassy CSS) เป็นภาษาที่ขยายมาจาก CSS เพื่อเพิ่มความสามารถในการจัดการสไตล์ให้ง่ายและมีประสิทธิภาพมากยิ่งขึ้น โดย SCSS รองรับฟีเจอร์ต่าง ๆ ที่ช่วยให้การเขียน CSS มีความยืดหยุ่นและเป็นระบบ เช่น Variables, Nesting, Mixins และ Inheritance ซึ่งช่วยให้นักพัฒนาสามารถเขียนชุดคำสั่งสไตล์ที่สะอาดและบำรุงรักษาง่าย

#### 2.5.1 การใช้งาน SCSS ใน Angular

ใน Angular นักพัฒนาสามารถเลือกใช้ SCSS เป็น Preprocessor ของ CSS ได้โดยตรง โดยแต่ละ Component สามารถมีไฟล์ SCSS ของตนเอง เช่น `app.component.scss` ทำให้การจัดการสไตล์ในระดับ Component มีความเป็นอิสระ ไม่รบกวนส่วนอื่น ๆ และยังช่วยให้ชุดคำสั่งสไตล์มีความเป็นระเบียบ

#### 2.5.2 ข้อดีของ SCSS

1. เพิ่มความยืดหยุ่น – สามารถใช้ Variables และ Mixins เพื่อลดการเขียนซ้ำ

2. ชุดคำสั่งสะอาดและเป็นระเบียบ – ด้วย Nesting และ Inheritance ทำให้การเขียน CSS อ่านง่ายขึ้น
3. ควบคุมในระดับ Component – การใช้ SCSS ภายใน Angular Component ช่วยให้สามารถจัดการสไตล์ได้แบบแยกส่วน
4. รองรับการสร้างธีม – สามารถปรับแต่งธีมของระบบได้ง่าย เช่น ใช้ SCSS ร่วมกับ Angular Material เพื่อสร้างธีมที่กำหนดเองได้

สรุปแล้ว SCSS ทำให้การเขียนสไตล์มีความยืดหยุ่นสูง และเมื่อใช้ร่วมกับ Angular จะช่วยให้การพัฒนาเว็บแอปพลิเคชันที่ซับซ้อนสามารถดูแลรักษาได้ง่ายขึ้น

## 2.6 Tailwind CSS



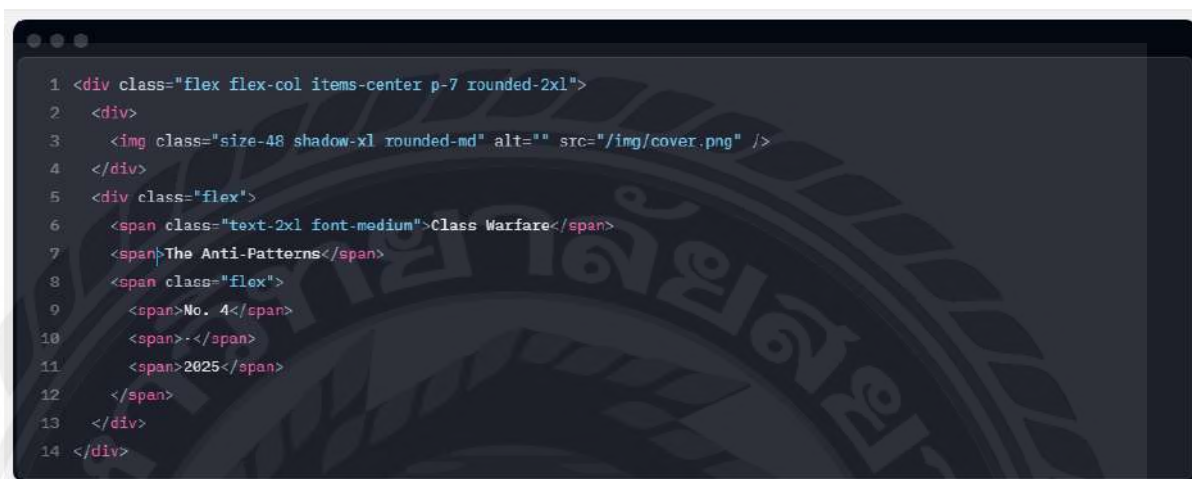
รูปที่ 2.10 สัญลักษณ์ของ Tailwind CSS

Tailwind CSS เป็น Utility-first CSS Framework ที่ได้รับความนิยมอย่างมากในปัจจุบัน เนื่องจากช่วยให้การออกแบบ UI ทำได้รวดเร็ว โดยใช้ Utility Classes ที่เตรียมไว้ เช่น flex, grid, p-4, text-center ทำให้นักพัฒนาไม่จำเป็นต้องเขียน CSS เองทั้งหมดตั้งแต่ต้น และยังสามารถสร้างเว็บไซต์ที่สวยงามได้อย่างมีประสิทธิภาพ

### 2.6.1 ข้อดีของการใช้ Tailwind CSS ใน Angular

1. สร้าง UI ได้รวดเร็ว – ด้วย Utility Classes ที่มีมาให้ ทำให้ไม่ต้องเขียน CSS เอง
2. ปรับแต่งได้ง่าย – สามารถปรับค่า Tailwind Configuration เช่น สี (Colors), ระยะห่าง (Spacing), ฟอนต์ (Font Size) ให้เหมาะสมกับโปรเจกต์
3. รองรับ Responsive Design – Tailwind มีคลาสสำหรับปรับการแสดงผลตามขนาดหน้าจอ เช่น sm:, md:, lg:

4. ทำงานร่วมกับ SCSS ได้ – สามารถใช้ Tailwind ควบคู่กับ SCSS เพื่อจัดการสไตล์ให้มี  
ความยืดหยุ่นและเป็นระเบียบมากขึ้น



รูปที่ 2.11 ตัวอย่างของการใช้ Tailwind CSS

## 2.7 TypeScript<sup>3</sup>



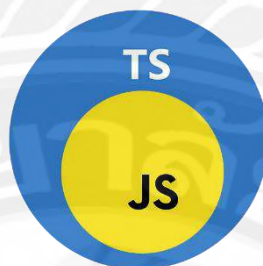
รูปที่ 2.7 สัญลักษณ์ของ TypeScript

TypeScript เป็นภาษาที่ถูกออกแบบและพัฒนาโดย Microsoft เพื่อเพิ่มประสิทธิภาพในการเขียนโปรแกรมด้วย JavaScript โดย TypeScript เป็น Superset ของ JavaScript หมายความว่า JavaScript ทำอะไรได้ TypeScript ก็สามารถทำได้เช่นกัน แต่เพิ่มเติมด้วยความสามารถใหม่ ๆ เช่น Static Typing, Interfaces, Classes, Decorators เป็นต้น ชุดคำสั่งที่เขียนด้วย TypeScript จะต้องถูก คอมไพล์ (Compile) ให้เป็น JavaScript ก่อนนำไปใช้งาน เนื่องจาก Browser เข้าใจ

<sup>3</sup> <https://kongruksiam.medium.com/รู้จักกับ-typescript-3e67f0db2897>

เฉพาะภาษา JavaScript เท่านั้น ดังนั้นนักพัฒนาสามารถใช้ TypeScript เพื่อเขียนชุดคำสั่งที่มีความเสถียร และให้ Compiler แปลงชุดคำสั่งเป็น JavaScript เวอร์ชันที่รองรับกับ Browser ที่ต้องการได้

### 2.7.1 ทำไมต้องใช้ TypeScript



รูปที่ 2.12 สัญลักษณ์ของ TypeScript

1. **Superset ของ JavaScript** – สามารถใช้ทุกฟีเจอร์ของ JavaScript ได้ และยังเพิ่มความสามารถใหม่ ๆ เข้ามา
2. **Static Type** – กำหนดประเภทข้อมูลได้ชัดเจน ป้องกันข้อผิดพลาดที่เกิดจาก Dynamic Typing ของ JavaScript เช่น ตัวแปร number จะไม่สามารถเก็บค่าเป็น string ได้
3. **พัฒนาชุดคำสั่งได้ปลอดภัยและแม่นยำขึ้น** – ลดโอกาสการเกิดบั๊กในระบบ โดยเฉพาะโครงการขนาดใหญ่ที่มีทีมหลายคนทำงานร่วมกัน

### 2.7.2 ข้อดีของการใช้ TypeScript ใน Angular

1. **การจัดการประเภทข้อมูล (Type Safety)**  
กำหนดชนิดของตัวแปรได้ชัดเจน ตรวจสอบข้อผิดพลาดได้ตั้งแต่ขั้นตอนการเขียนชุดคำสั่ง
2. **การสนับสนุนจาก IDE และ Autocompletion**  
ทำงานร่วมกับ IDE เช่น Visual Studio Code ได้อย่างเต็มประสิทธิภาพ มีระบบ IntelliSense, Autocomplete, และ Error Checking
3. **การเขียนชุดคำสั่งที่มีความชัดเจน**  
การกำหนดประเภทข้อมูลที่ชัดเจนทำให้ชุดคำสั่งมีโครงสร้างที่อ่านง่ายและบำรุงรักษาได้ง่าย
4. **รองรับการเขียนเชิงวัตถุ (Object-Oriented Programming)**

สนับสนุน Class, Inheritance, Interfaces, และ Encapsulation ซึ่งเป็นแนวคิดหลักของ OOP

### 2.7.3 ฟีเจอร์หลักของ TypeScript ที่ใช้ใน Angular

#### 1. การประกาศตัวแปรและประเภทข้อมูล (Type Annotations)

เช่น `let username: string; let age: number;` ช่วยให้ชุดคำสั่งปลอดภัยจากข้อผิดพลาดในการใช้ข้อมูล

#### 2. Interfaces

ใช้ในการกำหนดโครงสร้างของอ็อบเจกต์ เช่น การบังคับว่าข้อมูลผู้ใช้นี้ต้องมี id, name, email

#### 3. Classes และ Inheritance

ช่วยสร้างโครงสร้างที่ชัดเจน และสามารถสืบทอดคุณสมบัติจากคลาสอื่น ๆ ได้

#### 4. Generics

ช่วยให้สามารถสร้างฟังก์ชันหรือคลาสที่รองรับหลายชนิดข้อมูลโดยไม่ต้องเขียนซ้ำ

#### 5. Decorators

ฟีเจอร์สำคัญของ Angular ที่ใช้กำหนด Metadata ให้กับคลาส เช่น `@Component`, `@Injectable` ทำให้ Angular รู้จักว่าคลาสใดเป็น Component หรือ Service

## 2.8 Git<sup>4</sup>



รูปที่ 2.13 สัญลักษณ์ของ git version control

Git เป็นระบบควบคุมเวอร์ชัน (Version Control System: VCS) แบบกระจายศูนย์ (Distributed) ที่ได้รับความนิยมมากที่สุดในปัจจุบัน ใช้สำหรับติดตามการเปลี่ยนแปลงของชุดคำสั่ง

<sup>4</sup> <https://stackpython.medium.com/ทำความเข้าใจ-git-github-พร้อมการใช้งานร่วมกับ-vs-code-เบื้องต้น-f848f41a39e9>

ในโปรเจกต์ ทำให้นักพัฒนาสามารถตรวจสอบประวัติการแก้ไข และย้อนกลับไปยังสถานะก่อนหน้าได้อย่างง่ายดาย อีกทั้งยังช่วยให้การทำงานร่วมกันในทีมเป็นระบบและมีประสิทธิภาพ

### 2.8.1 ข้อดีของการใช้ Git

- สามารถเก็บและติดตามประวัติการเปลี่ยนแปลงของชุดคำสั่ง (Commit History)
- รองรับการทำงานแบบร่วมกันหลายคน โดยไม่กระทบกัน เนื่องจากแต่ละคนสามารถพัฒนาใน Branch ของตนเอง
- ทุกการเปลี่ยนแปลงจะถูกบันทึกไว้ใน Commit ทำให้สามารถย้อนกลับไปยังเวอร์ชันก่อนหน้าได้ง่าย
- ช่วยให้การรวมชุดคำสั่ง (Merge) จากหลายแหล่งเป็นไปอย่างมีประสิทธิภาพ

ในการทำโครงงานสหกิจศึกษาครั้งนี้ ผู้จัดทำได้ใช้ Git เป็นเครื่องมือหลักในการควบคุมเวอร์ชันชุดคำสั่งของระบบหลังบ้าน (Admin System) ฝั่ง Frontend และใช้ร่วมกับทีม Backend เพื่อพัฒนาและทดสอบระบบร่วมกัน

## 2.9 GitHub



รูปที่ 2.9 สัญลักษณ์ของ GitHub

GitHub เป็นบริการ Hosting สำหรับ Git Repository ที่ช่วยให้การเก็บ แบ่งปัน และทำงานร่วมกันบนชุดคำสั่งเป็นไปอย่างสะดวก GitHub มีฟีเจอร์สำคัญ เช่น Pull Requests, Issues และ Project Boards ที่ช่วยให้ทีมงานสามารถจัดการโปรเจกต์และทำงานร่วมกันได้อย่างมีประสิทธิภาพ

### 2.9.1 การใช้งาน GitHub สำหรับการทำงานร่วมกัน

- **การ Fork Repository**

ช่วยให้นักพัฒนาสามารถคัดลอกโปรเจกต์ไปยังพื้นที่ของตนเองเพื่อพัฒนาเพิ่มเติมได้ โดยไม่กระทบกับ Repository หลัก

- การสร้าง Pull Request (PR)

ใช้สำหรับเสนอการเปลี่ยนแปลงจาก Branch หรือ Fork กลับไปยัง Repository หลัก โดยทีมงานสามารถตรวจสอบ (Review) ก่อนรวมชุดคำสั่งได้

- การสร้าง Issues

ใช้สำหรับรายงานบั๊ก ติดตามปัญหา หรือเสนอฟีเจอร์ใหม่ ๆ ในโปรเจกต์

- การใช้ Branches

ช่วยให้การพัฒนาสามารถแยกฟีเจอร์หรือโมดูลออกจากกันได้อย่างเป็นอิสระ โดยไม่กระทบกับ Main Branch ของโปรเจกต์หลัก

การใช้ Git และ GitHub ในโครงการนี้ ช่วยให้การพัฒนาระบบหลังบ้านฝั่ง Frontend มีความเป็นระบบมากขึ้น ทั้งในด้านการควบคุมเวอร์ชัน การจัดการชุดคำสั่ง และการทำงานร่วมกับทีม Backend ทำให้สามารถพัฒนาระบบได้อย่างมีประสิทธิภาพและลดความผิดพลาดที่อาจเกิดขึ้น

## 2.10 Git Fork



รูปที่ 2.9 สัญลักษณ์ของ Git Fork

Git Fork คือกระบวนการคัดลอก (Copy) Repository จาก GitHub ของผู้อื่นมายังบัญชีของเราเอง เพื่อให้สามารถแก้ไขหรือพัฒนาต่อยอดได้โดยไม่กระทบกับ Repository ต้นฉบับ การ Fork มักใช้ในกรณีที่ต้องการพัฒนาร่วมกันในโปรเจกต์ Open-source หรือการทำงานในทีมที่ต้องการแยกพื้นที่การพัฒนากัน

### 2.10.1 ประโยชน์ของการใช้ Git Fork

- สามารถทดลองพัฒนาและปรับปรุงชุดคำสั่งได้ โดยไม่กระทบกับ Repository หลัก

- รองรับการทำงานร่วมกันในลักษณะ Collaborative Development ผ่านการสร้าง Pull Request (PR)
- ช่วยให้นักพัฒนาสามารถมีพื้นที่การทำงานของตนเอง แต่ยังสามารถส่งชุดคำสั่งกลับไปรวมกับ Repository ต้นฉบับได้

ในโครงการนี้ Git Fork ช่วยให้การพัฒนาชุดคำสั่งฝั่ง Frontend สามารถแยกสาขาการทำงานออกจากทีม Backend ได้อย่างอิสระ ก่อนจะรวมเข้ามายัง Repository หลัก ทำให้กระบวนการพัฒนามีความเป็นระบบและลดความผิดพลาดในการรวมชุดคำสั่ง

## 2.11 Postman



รูปที่ 2.11 สัญลักษณ์ของ POSTMAN

Postman เป็นเครื่องมือที่ได้รับความนิยมในการทดสอบ API (Application Programming Interface) โดยเฉพาะในการพัฒนาเว็บแอปพลิเคชันที่มีการสื่อสารระหว่าง Frontend และ Backend Postman ช่วยให้นักพัฒนาสามารถส่งคำขอ (Request) และตรวจสอบการตอบกลับ (Response) ได้อย่างสะดวก

### 2.11.1 ฟีเจอร์หลักของ Postman

1. **API Testing** – สามารถส่ง Request แบบ GET, POST, PUT, DELETE เพื่อตรวจสอบการทำงานของ API
2. **Collection** – จัดกลุ่ม API ที่ต้องการทดสอบไว้เป็นหมวดหมู่ เพื่อให้การทดสอบเป็นระบบและสามารถนำกลับมาใช้ซ้ำได้
3. **Environment Variables** – กำหนดค่าตัวแปร เช่น Base URL หรือ Token สำหรับใช้ในหลาย ๆ Request ได้
4. **Mock Server** – สร้าง API จำลองเพื่อใช้ทดสอบระบบได้ แม้ Backend จะยังพัฒนาไม่เสร็จสมบูรณ์

### 2.11.2 ประโยชน์ของการใช้ Postman

- ช่วยตรวจสอบความถูกต้องของ API ที่เชื่อมต่อระหว่าง Angular (Frontend) และระบบ Backend
- ลดเวลาในการ Debugging และทำให้การพัฒนา Frontend สามารถทดสอบได้แม้ Backend จะยังอยู่ระหว่างการพัฒนา
- ช่วยยืนยันว่าโครงสร้างข้อมูล (Request/Response) ถูกต้องตามที่ออกแบบไว้



### บทที่ 3

#### รายละเอียดการปฏิบัติงาน

##### 3.1 ชื่อและที่ตั้งสถานประกอบการ

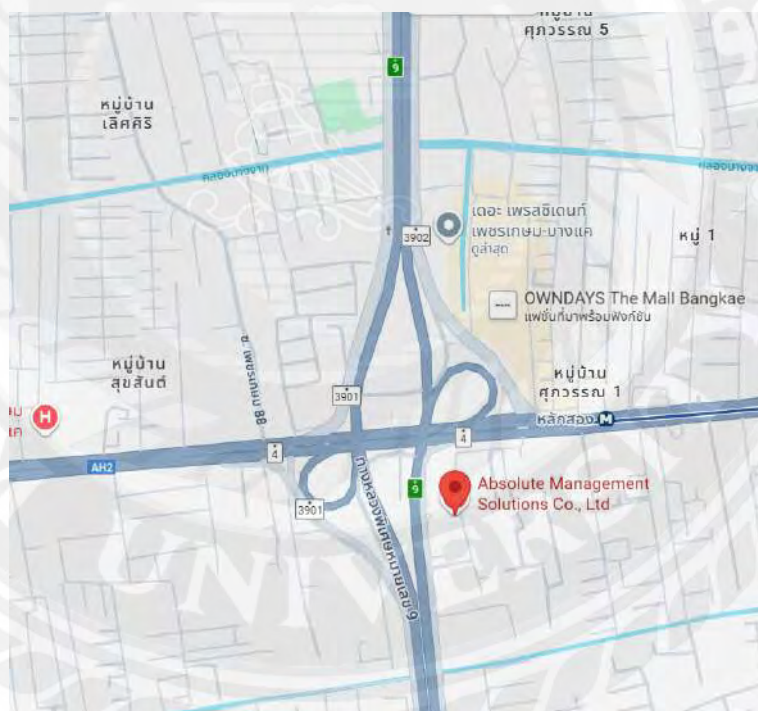
ชื่อสถานประกอบการ : บริษัท แอ็บโซลูท เมเนจเม้นท์ โซลูชันส์ จำกัด  
(Absolute Management Solutions Co., Ltd.)

ที่ตั้ง : 22 ซอย เพชรเกษม 47/2 แขวงบางแค บางแค  
กรุงเทพมหานคร 10160

โทรศัพท์ : 02-026-3200

E-mail : support@absolute.co.th

เว็บไซต์ : <https://absolute.co.th>



รูปที่ 3.1 แผนที่ตั้ง บริษัท แอ็บโซลูท เมเนจเม้นท์ โซลูชันส์ จำกัด

### 3.2 ลักษณะการประกอบการ ผลิตภัณฑ์การให้บริการหลักขององค์กร

บริษัท แอ็บโซลูท แมนเนจเม้นท์ โซลูชันส์ จำกัด เป็นบริษัทที่ประกอบธุรกิจพัฒนาซอฟต์แวร์บริหารธุรกิจ (ERP) โปรแกรมบริหารธุรกิจผ่านระบบคลาวด์ (CLOUD ERP) มาพร้อม Solutions ทางธุรกิจที่ครอบคลุมมากที่สุดตั้งแต่ระบบขายสินค้าหน้าร้าน, คลังสินค้า, จัดซื้อ เป็นต้น โดยเชื่อว่า Cloud จะเป็นเทคโนโลยีที่เปิดโอกาสให้ SMEs สามารถมีระบบบริหารจัดการธุรกิจ (ERP) ที่ดีได้ ซึ่งเป็นจุดกำเนิดของระบบ Absolute ERP (ISoLn) ที่ถือเป็นระบบบริหารจัดการธุรกิจออนไลน์บน Cloud รายแรกของไทย

ABSOLUTE ชนะเลิศรางวัล True Cloud Winner Project ABSOLUTE ได้รับการรับรองจาก SOFTWARE PARK (THAILAND) ว่าเป็น ERP on Cloud รายแรกของไทย ผ่านการประกวดจากโครงการ "True Cloud Winner Project" ซึ่งถือเป็นโครงการประกวด Cloud Software ที่จัดขึ้นเป็นครั้งแรกของประเทศไทย

จุดเริ่มต้นของทีม "บริการลูกค้า" จัดตั้งทีม C ก้าวเข้าสู่กลุ่มธุรกิจรับเหมาก่อสร้าง ปี พ.ศ. 2555 พัฒนาระบบบริหารงานก่อสร้าง และระบบบริหารงานขายเบื่อง จากกลุ่มธุรกิจรับเหมาก่อสร้าง เป็นธุรกิจที่ต้องใช้เงินลงทุนค่อนข้างสูง และมีปัจจัยที่สำคัญในการอยู่รอดคือ การควบคุมต้นทุน, คุณภาพ และเวลา ทางบริษัทฯ จึงเล็งเห็นโอกาสในการเจาะตลาดกลุ่มธุรกิจนี้ และได้พัฒนาระบบบริหารงานก่อสร้าง (Construction) ด้วยกระแสดความต้องการระบบ CRM ที่เพิ่มขึ้นอย่างมาก บริษัท จึงได้พัฒนาระบบบริหารงานขาย (AMS) และเปิดให้บริการแก่ลูกค้าทั่วไปอีกด้วย (CUSTOMER SUPP) ต่อเนื่องไปที่ธุรกิจอสังหาริมทรัพย์ ปี พ.ศ. 2556 พัฒนาระบบขายอสังหาริมทรัพย์ ระบบทรัพยากรบุคคล พัฒนาระบบขายอสังหาริมทรัพย์ (Real-estate) สำหรับบริหารจัดการ งานขาย บ้าน และคอนโด เป็นต้น

โดยรองรับตั้งแต่การออกไปเสนอราคา, ใบจอง และสัญญา เป็นต้น ไปจนถึงการโอนกรรมสิทธิ์ และการรับประกันบ้าน จากประสบการณ์ตรงของลูกค้าที่ประสบปัญหาในการบันทึกเวลาเข้า-ออกของพนักงานรายวัน บริษัทฯ จึงได้พัฒนาระบบทรัพยากรบุคคล (HR) ที่ใช้การสแกนใบหน้า (Face Recognition) ในการบันทึกเวลาเข้า-ออก เพื่อป้องกันปัญหาการทุจริตและแก้ปัญหาด้านเวลานี้ขึ้นมาตามของตนเอง ORT เพื่อเพิ่มคุณภาพการให้บริการตามฐานลูกค้าที่เติบโตขึ้นกว่าเท่าตัว

ผลิตภัณฑ์และบริการหลักของบริษัทครอบคลุมธุรกิจต่อไปนี้

- คลังสินค้า
- ค้าปลีก
- ค้าส่ง

- จัดซื้อ
- การเงิน
- ก่อสร้าง
- อสังหาริมทรัพย์
- บริหารงานขาย
- ฝ่ายบุคคล

### 3.3 ตำแหน่งงานและลักษณะงานที่ได้รับมอบหมาย

ตำแหน่งงานที่คณะผู้จัดทำได้รับหมาย คือ Front-End Developer โดยระหว่างปฏิบัติงาน สหกิจศึกษาที่สถานประกอบการได้รับมอบหมายให้ปฏิบัติงาน โดยมีลักษณะงานดังนี้

- พัฒนาและปรับปรุงส่วนติดต่อผู้ใช้ (Frontend) ของระบบหลังบ้าน (Back-Office/Administrator System) ด้วย Angular Framework
- เชื่อมต่อ API ร่วมกับทีม Backend และทดสอบการทำงานร่วมกัน
- ตรวจสอบและทดสอบระบบจริง (Production Site) เพื่อให้มั่นใจว่าการจัดการข้อมูลผ่านระบบหลังบ้านสะท้อนผลได้ถูกต้อง
- ศึกษาและฝึกใช้งานเทคโนโลยีที่ใช้จริงในการทำงาน เช่น TypeScript, Tailwind CSS, Angular Material, Git/GitHub และ Postman
- แก้ไขปัญหาที่เกิดขึ้นระหว่างการพัฒนาและรายงานผลการทำงานให้ทีมพัฒนา

### 3.4 ชื่อและตำแหน่งของพนักงานที่ปรึกษา

|               |              |                         |
|---------------|--------------|-------------------------|
| คุณจันทร์สุดา | ศิริไกรวัฒน์ | ตำแหน่ง Head Programmer |
| คุณชลักร์     | แสงศิริรัตน์ | ตำแหน่ง DevOps          |

### 3.5 ระยะเวลาที่ปฏิบัติงาน

ได้เข้าปฏิบัติงานสหกิจศึกษา ณ บริษัท แอ็บโซลูท เมเนจเม้นท์ โซลูชั่นส์ จำกัดตั้งแต่วันที่ 19 พฤษภาคม พ.ศ. 2568 ถึงวันที่ 29 สิงหาคม พ.ศ. 2568

## บทที่ 4

### ผลการปฏิบัติงานโครงการ

#### 4.1 รายละเอียดของโครงการ

โครงการสหกิจศึกษาที่คณะผู้จัดทำได้รับมอบหมายคือ “การออกแบบและพัฒนาระบบส่วนหลังเพื่อจัดการเนื้อหาเว็บไซต์” โดยมีเป้าหมายเพื่อให้ผู้ดูแลเว็บไซต์สามารถเพิ่ม แก้ไข และจัดการเนื้อหาได้ด้วยตนเอง โดยไม่จำเป็นต้องมีทักษะด้านการเขียนโปรแกรมคอมพิวเตอร์

การพัฒนาโครงการนี้มุ่งเน้นที่การสร้าง ส่วนติดต่อผู้ใช้ (Frontend) ของระบบหลังบ้าน (Backend) โดยใช้ Angular Framework ร่วมกับ TypeScript, Tailwind CSS และ Angular Material สำหรับการออกแบบและปรับแต่ง UI รวมถึงการทำงานร่วมกับฝั่ง Backend เพื่อเชื่อมต่อ API และทดสอบการทำงานร่วมกัน

#### 4.2 เครื่องมือและเทคโนโลยีที่ใช้

- **Angular Framework:** สำหรับพัฒนา Frontend ของระบบหลังบ้าน
- **TypeScript:** เพิ่มความถูกต้องของชุดคำสั่งด้วยการตรวจสอบประเภทข้อมูล
- **Tailwind CSS:** สำหรับจัดการรูปแบบการแสดงผลของหน้าเว็บ
- **Angular Material:** สำหรับคอมโพเนนต์ UI ที่มีมาตรฐานและใช้งานสะดวก
- **Git / GitHub:** สำหรับการควบคุมเวอร์ชันและการทำงานร่วมกับทีมพัฒนา
- **Postman:** สำหรับทดสอบ API ของฝั่ง Backend
- **Visual Studio Code (VS Code):** สำหรับเขียนและแก้ไขชุดคำสั่ง

##### 4.2.1 ศึกษาเครื่องมือในการทำงาน

ในขั้นตอนนี้คณะผู้จัดทำได้ศึกษาเกี่ยวกับวิธีใช้งาน Angular Framework สำหรับการพัฒนาเว็บแอปพลิเคชัน โดยเริ่มตั้งแต่การสร้างโปรเจกต์ใหม่ การจัดการโครงสร้างไฟล์ ไปจนถึงการพัฒนา Component และการเชื่อมโยงข้อมูล (Data Binding) เพื่อสร้างส่วนติดต่อผู้ใช้ที่มีความสามารถในการโต้ตอบกับผู้ใช้จริง

นอกจากนี้ยังได้ศึกษา ภาษา TypeScript ซึ่งเป็นภาษาหลักที่ใช้ในการพัฒนา Angular โดยเน้นการกำหนดชนิดข้อมูล (Type Checking) การใช้ Class, Interface และการเขียนโปรแกรมเชิงวัตถุ (OOP) เพื่อเพิ่มความถูกต้องและความเป็นระเบียบของชุดคำสั่ง

ด้านการออกแบบคณะผู้จัดทำได้เรียนรู้การใช้งาน Tailwind CSS ซึ่งเป็น Utility-first CSS Framework ที่ช่วยให้การจัดการ Layout และการทำ Responsive Design มีความสะดวกและรวดเร็ว รวมถึงการใช้ Angular Material เพื่อสร้างคอมโพเนนต์มาตรฐาน เช่น Table, Dialog และ Form Controls เพื่อยกระดับประสบการณ์ผู้ใช้ (UX/UI)

สำหรับการจัดการเวอร์ชันชุดคำสั่ง ผู้จัดทำได้ศึกษาและใช้งาน Git และ GitHub โดยเรียนรู้คำสั่งพื้นฐาน เช่น commit, push, pull และ branch รวมถึงการทำงานร่วมกันในทีมพัฒนา ขณะเดียวกันยังได้ใช้ Postman ในการทดสอบ API เพื่อยืนยันความถูกต้องของการเชื่อมต่อระหว่าง Frontend และ Backend

เครื่องมือหลักที่ใช้ในการพัฒนาคือ Visual Studio Code (VS Code) โดยคณะผู้จัดทำได้ศึกษาและใช้งาน Extensions ต่างๆ เช่น Angular Essentials, Prettier และ ESLint รวมถึงการใช้ Integrated Terminal เพื่อรันคำสั่ง Angular CLI โดยตรงภายในโปรแกรม

จากการศึกษาข้างต้นคณะผู้จัดทำสามารถทำความเข้าใจและใช้งานเครื่องมือสำคัญที่เกี่ยวข้องกับการพัฒนา Frontend ได้อย่างมีประสิทธิภาพ และนำไปประยุกต์ใช้ในการพัฒนาระบบหลังบ้านตามที่ได้รับมอบหมาย

#### 4.2.2 พัฒนาและปรับปรุงระบบ (Development & Implementation)

ในขั้นตอนนี้ ผู้จัดทำได้ดำเนินการพัฒนาและปรับปรุงระบบหลังบ้าน (Back-Office/Administrator System) โดยมุ่งเน้นที่การพัฒนา ส่วนติดต่อผู้ใช้ (Frontend) ด้วย Angular Framework และการทำงานร่วมกับทีม Backend โดยมีรายละเอียดดังนี้

##### 1. การพัฒนาโครงสร้างและส่วนติดต่อผู้ใช้ (UI Structure & Components)

คณะผู้จัดทำได้พัฒนา Component หลักของระบบ เช่น หน้า Dashboard, Content Management, User Management และหน้าแบบฟอร์มต่าง ๆ โดยใช้ Angular ในการจัดการโครงสร้าง และ Tailwind CSS ร่วมกับ Angular Material ในการออกแบบ UI ให้สวยงามและใช้งานง่าย

##### 2. การเชื่อมต่อกับ API (API Integration)

ได้ทำการเชื่อมต่อกับ REST API ที่ทีม Backend พัฒนาขึ้น เพื่อให้ระบบสามารถดึงและส่งข้อมูลได้จริง โดยใช้บริการ Http Client ของ Angular และทำการทดสอบการเรียกใช้งาน API ผ่าน Postman เพื่อให้มั่นใจว่าข้อมูลสามารถเชื่อมโยงและปรับปรุงแก้ไขได้อย่างถูกต้อง

### 3. การจัดการข้อมูล (Data Binding & State Management)

ในการพัฒนาคณะผู้จัดทำใช้ Two-way Data Binding ของ Angular เพื่อให้ข้อมูลในฟอร์ม และ UI เชื่อมโยงกับข้อมูลจาก API ได้อย่างอัตโนมัติ และใช้ Service ของ Angular ในการจัดการข้อมูลระหว่าง Component ต่าง ๆ

### 4. การทดสอบและปรับปรุงระบบ (Testing & Debugging)

หลังจากการพัฒนาแต่ละส่วนคณะผู้จัดทำได้ทดสอบการทำงานของระบบด้วยการรันโปรแกรมผ่าน Angular CLI และทดสอบการใช้งานบน Browser จริง เพื่อหาข้อผิดพลาด (Bug) ที่เกิดขึ้น เช่น การแสดงผล UI ที่ไม่ตรงตามที่ออกแบบ หรือข้อมูลไม่ปรับปรุงแก้ไขตามที่ต้องการ จากนั้นทำการแก้ไขให้ระบบสามารถทำงานได้สมบูรณ์

### 5. การปรับปรุงการใช้งาน (Improvement & Optimization)

เพื่อให้ระบบมีความเสถียรและใช้งานได้อย่างราบรื่นคณะผู้จัดทำได้ปรับปรุงประสิทธิภาพของ UI เช่น การทำ Responsive Design ให้รองรับการใช้งานทั้งบนคอมพิวเตอร์และอุปกรณ์พกพา รวมถึงการปรับโครงสร้างชุดคำสั่งให้เป็นระเบียบและง่ายต่อการบำรุงรักษาในอนาคต

## 4.3 ปัญหาที่เจอระหว่างสหกิจศึกษา

ระหว่างการปฏิบัติสหกิจศึกษาคณะผู้จัดทำได้พบเจอปัญหาและอุปสรรคหลายประการ ทั้งในด้านเทคนิคที่เกี่ยวข้องกับการพัฒนาโปรแกรม และด้านการทำงานร่วมกับทีม ซึ่งปัญหาเหล่านี้สร้างความท้าทาย แต่ก็ยังเป็นโอกาสสำคัญที่ช่วยให้คณะผู้จัดทำได้เรียนรู้วิธีแก้ไขและพัฒนาทักษะใหม่ๆ โดยสามารถสรุปปัญหาหลักที่พบเจอได้ดังนี้

### 4.3.1 การเรียนรู้เทคโนโลยีใหม่ในช่วงแรก

คณะผู้จัดทำไม่มีประสบการณ์ในการใช้ Angular, TypeScript, Tailwind CSS และ Postman เป็นต้น ทำให้ช่วงแรกต้องใช้เวลาในการเรียนรู้และทดลองใช้งานจริง จึงเกิดความล่าช้าในบางขั้นตอนของการ

#### วิธีแก้ปัญหา

คณะผู้จัดทำได้แบ่งเวลาในการเรียนรู้เพิ่มเติมนอกเวลาทำงาน และขอคำแนะนำจากพนักงานในทีมที่มีประสบการณ์ ซึ่งช่วยให้สามารถเข้าใจการทำงานของเทคโนโลยีใหม่ ๆ ได้รวดเร็วขึ้น

### 4.3.2 การจัดการ CSS และการออกแบบ UI

ในการพัฒนา Frontend ด้วย Tailwind CSS ร่วมกับ Angular Material พบว่ามีความซับซ้อนในบางส่วน โดยเฉพาะการจัดการสไตล์ของคอมโพเนนต์ที่ทับซ้อนกัน ทำให้บางครั้ง UI ไม่แสดงผลตรงตามที่ต้องการ

#### วิธีแก้ปัญหา

คณะผู้จัดทำได้ศึกษาเพิ่มเติมเกี่ยวกับการ Override Class ของ Tailwind และการใช้ Custom Theme ของ Angular Material รวมถึงการปรับโครงสร้าง CSS ให้เหมาะสมมากขึ้น ส่งผลให้ UI สามารถแสดงผลได้ตรงตามที่ต้องการ

### 4.3.3 การรวมชุดคำสั่ง (Code Integration)

ในระหว่างการทำงานร่วมกับทีม มีปัญหาในการ Merge Code ผ่าน Git/GitHub เนื่องจากบางครั้งมีการแก้ไขไฟล์เดียวกันพร้อมกัน ทำให้เกิด Conflict ซึ่งส่งผลให้ต้องใช้เวลาในการแก้ไข

#### วิธีแก้ปัญหา

คณะผู้จัดทำได้เรียนรู้การใช้ Branch แยกตามฟีเจอร์ เพื่อป้องกันการชนกันของชุดคำสั่ง และฝึกการแก้ไข Conflict ด้วย Git รวมถึงการสื่อสารกับทีมอย่างสม่ำเสมอ เพื่อให้การรวมชุดคำสั่งเป็นไปอย่างราบรื่น

## 4.5 ผลลัพธ์ของโครงการ

จากการพัฒนาระบบหลังบ้าน (Back-Office/Administrator System) คณะผู้จัดทำสามารถสร้างและปรับปรุงระบบส่วนติดต่อผู้ใช้ (Frontend) ให้สามารถใช้งานได้จริง โดยมีผลลัพธ์ที่สำคัญดังนี้

### 1. ระบบจัดการเนื้อหาเว็บไซต์ (Content Management System)

ผู้ดูแลเว็บไซต์สามารถเพิ่ม แก้ไข และลบข้อมูลเนื้อหาบนเว็บไซต์ได้ด้วยตนเอง โดยไม่ต้องแก้ไขชุดคำสั่งโปรแกรมหรือติดต่อฝ่ายพัฒนาโดยตรง

### 2. ส่วนติดต่อผู้ใช้ที่ใช้งานง่าย (User-Friendly Interface)

การใช้ Angular Material ร่วมกับ Tailwind CSS ทำให้ระบบมี UI ที่ทันสมัย ใช้งานง่าย และเป็นมิตรต่อผู้ใช้ แม้ผู้ดูแลเว็บไซต์ที่ไม่มีพื้นฐานด้านการพัฒนาซอฟต์แวร์ก็สามารถใช้งานได้สะดวก

### 3. การเชื่อมต่อกับ API ได้อย่างมีประสิทธิภาพ

ระบบสามารถเชื่อมโยงและแลกเปลี่ยนข้อมูลกับ API ของ Backend ได้อย่างถูกต้องและเสถียร ทำให้ข้อมูลที่ปรับปรุงจาก Admin System แสดงผลบนหน้าเว็บไซต์จริงได้ทันที

### 4. รองรับการใช้งานบนหลายอุปกรณ์ (Responsive Design)

ระบบถูกออกแบบให้สามารถใช้งานได้ทั้งบนคอมพิวเตอร์ แท็บเล็ต และสมาร์ทโฟน เพื่อให้ผู้ดูแลเว็บไซต์สามารถจัดการข้อมูลได้ทุกที่ทุกเวลา

### 5. ชุดคำสั่งที่เป็นระบบและง่ายต่อการบำรุงรักษา (Maintainable Codebase)

ด้วยการใช้ TypeScript และการพัฒนาเชิง Component ของ Angular ทำให้ชุดคำสั่งมีความเป็นระเบียบ ง่ายต่อการปรับปรุงและขยายระบบในอนาคต

จึงสรุปได้ว่าการดำเนินโครงการสหกิจศึกษาเรื่อง “การออกแบบและพัฒนาระบบส่วนหลังเพื่อจัดการเนื้อหาเว็บไซต์” ทำให้คณะผู้จัดทำได้พัฒนาระบบส่วนติดต่อผู้ใช้ (Frontend) ด้วย Angular Framework ร่วมกับ TypeScript, Tailwind CSS และ Angular Material ตลอดจนได้ฝึกการเชื่อมต่อและทดสอบการทำงานกับ API จากทีม Backend ผลลัพธ์ของโครงการสามารถนำไปใช้งานได้จริง ผู้ดูแลเว็บไซต์สามารถเพิ่ม แก้ไข และจัดการข้อมูลได้อย่างสะดวกและมีประสิทธิภาพ นอกจากนี้ผู้จัดทำยังได้รับประสบการณ์ตรงในการแก้ไขปัญหา การทำงานร่วมกับทีม และการประยุกต์ใช้เทคโนโลยีใหม่ ๆ ซึ่งจะเป็นประโยชน์ต่อการทำงานในอนาคต

## บทที่ 5

### สรุปผลและข้อเสนอแนะ

#### 5.1 สรุปผลโครงการ

โครงการสหกิจศึกษาเรื่อง “การออกแบบและพัฒนาระบบส่วนหลังเพื่อจัดการเนื้อหาเว็บไซต์” มีวัตถุประสงค์เพื่อพัฒนาส่วนติดต่อผู้ใช้ (Frontend) ของระบบหลังบ้าน (Backend) โดยใช้ Angular Framework ร่วมกับ TypeScript, Tailwind CSS และ Angular Material ผลลัพธ์ที่ได้คือระบบที่ผู้ดูแลเว็บไซต์สามารถเพิ่ม แก้ไข และลบเนื้อหาได้ด้วยตนเอง โดยไม่ต้องพึ่งพนักพัฒนาโดยตรงและไม่จำเป็นต้องมีทักษะด้านการเขียนโปรแกรมคอมพิวเตอร์ อีกทั้งระบบยังมีการเชื่อมต่อกับ API ฝั่ง Backend ทำให้ข้อมูลที่จัดการจาก Admin System สามารถทำการส่งข้อมูลที่ต้องการจัดการหรือปรับปรุงแก้ไขไปยังเว็บไซต์จริงได้อย่างถูกต้อง

นอกจากนี้ ระบบที่พัฒนายังรองรับการใช้งานบนหลายอุปกรณ์ (Responsive Design) มี UI ที่ใช้งานง่ายและเป็นมิตรต่อผู้ใช้ (User-Friendly) และสามารถต่อยอดเพื่อเพิ่มฟังก์ชันใหม่ ๆ ได้ในอนาคต

##### 5.1.1 ข้อจำกัดหรือปัญหาของโครงการ

5.1.1.1 คณะผู้จัดทำมีเวลาจำกัดในการเรียนรู้เทคโนโลยีใหม่ เช่น Angular, TypeScript และ Tailwind CSS ทำให้บางฟังก์ชันไม่สามารถพัฒนาได้เต็มประสิทธิภาพ

5.1.1.2 การเชื่อมต่อกับ API ฝั่ง Backend บางครั้งเกิดปัญหาด้านรูปแบบข้อมูล (Data Format) และข้อผิดพลาดในการสื่อสารระหว่างระบบ

5.1.1.3 การใช้ Tailwind CSS ร่วมกับ Angular Material ทำให้เกิดความซับซ้อนในการจัดการ UI และต้องมีการปรับแต่งเพิ่มเติม

##### 5.1.2 ข้อเสนอแนะ

5.1.2.1 ควรมีการวางแผนและกำหนดโครงสร้างการพัฒนาระบบให้ชัดเจนตั้งแต่เริ่มต้น เพื่อป้องกันปัญหาที่อาจเกิดขึ้นระหว่างการรวมโค้ด

5.1.2.2 ควรปรับปรุงคู่มือการใช้งาน (User Manual) เพื่อช่วยให้ผู้ดูแลเว็บไซต์สามารถใช้งานระบบได้อย่างสะดวกมากยิ่งขึ้น

5.1.2.3 ควรมีการทดสอบระบบร่วมกับผู้ใช้งานจริง (User Testing) เพื่อปรับปรุง UX/UI ให้ตอบโจทย์ความต้องการได้ดียิ่งขึ้น

## 5.2 สรุปผลการปฏิบัติงานสหกิจ

### 5.2.1 ข้อดีของการปฏิบัติงานสหกิจ

จากที่ได้มาปฏิบัติสหกิจศึกษาทางคณะผู้จัดทำได้รับความรู้จากการปฏิบัติงานในด้าน Frontend ได้เรียนรู้และฝึกปฏิบัติจริงเกี่ยวกับการพัฒนา Frontend ด้วย Angular, TypeScript, Tailwind CSS และ Angular Material พัฒนาทักษะการใช้งานเครื่องมือในการทำงานจริง เช่น Git/GitHub สำหรับควบคุมเวอร์ชัน และ Postman สำหรับทดสอบ API ได้ประสบการณ์ในการทำงานร่วมกับทีมพัฒนา โดยเฉพาะการทำงานกับทีม Backend ทำให้เข้าใจขั้นตอนการพัฒนาระบบแบบ Full-stack มากขึ้น ฝึกทักษะ Soft Skills เช่น การทำงานเป็นทีม การสื่อสาร การแก้ปัญหาเฉพาะหน้า และการปรับตัวเข้ากับสภาพแวดล้อมการทำงานจริง และได้เรียนรู้กระบวนการทำงานขององค์กรจริง ซึ่งเป็นพื้นฐานสำคัญในการก้าวเข้าสู่สายอาชีพด้านการพัฒนาเว็บไซต์ในอนาคต

### 5.2.2 ปัญหาที่พบของการปฏิบัติงานสหกิจ

ระหว่างการปฏิบัติงานสหกิจศึกษาคณะผู้จัดทำพบปัญหาและอุปสรรคหลายประการ โดยเฉพาะในช่วงแรกที่ต้องปรับตัวให้เข้ากับการทำงานจริง ปัญหาสำคัญคือการเรียนรู้เทคโนโลยีใหม่ที่ไม่เคยใช้งานมาก่อน เช่น Angular, TypeScript, Tailwind CSS และ Postman ซึ่งทำให้ต้องใช้เวลาในการศึกษาและทดลองด้วยตนเองจนกว่าจะเข้าใจและสามารถนำมาใช้ได้อย่างถูกต้อง นอกจากนี้ยังพบปัญหาในการจัดการด้านการออกแบบส่วนติดต่อผู้ใช้ เนื่องจากการใช้ Tailwind CSS ร่วมกับ Angular Material มีความซับซ้อน ทำให้บางครั้งการแสดงผลไม่ตรงตามที่ออกแบบไว้

และปัญหาในการการทำงานร่วมกับทีมผ่าน Git/GitHub ซึ่งมักเกิดการชนกันของชุดคำสั่ง (Conflict) เมื่อต้องรวมชุดคำสั่งของหลายคนเข้าด้วยกัน ทำให้ต้องใช้เวลาในการแก้ไขและเรียนรู้วิธีการจัดการ Branch ให้เหมาะสม รวมถึงในบางช่วงยังมีปัญหาเรื่องการลำดับความสำคัญของงาน ทำให้งานบางส่วนล่าช้า อย่างไรก็ตาม ปัญหาเหล่านี้ได้กลายเป็นโอกาสให้ผู้จัดทำได้เรียนรู้วิธีการแก้ไขปัญหาเฉพาะหน้า ฝึกการทำงานร่วมกับทีม และพัฒนาทักษะด้านการสื่อสารในการทำงานจริงมากขึ้น

### 5.2.3 ข้อเสนอแนะ

สำหรับนักศึกษารุ่นต่อไปที่จะต้องไปปฏิบัติงานสหกิจศึกษา ควรมีการเตรียมในด้านต่อไปนี้

5.2.3.1 ศึกษา Angular Framework และ TypeScript เพื่อให้สามารถพัฒนา Frontend ได้ อย่างมีประสิทธิภาพ

5.2.3.2 ฝึกการใช้ Tailwind CSS และ Angular Material เพื่อจัดการ UI และ

สร้างระบบที่รองรับ Responsive Design

5.2.3.3 ทำความเข้าใจกับ Git/Github/Git fork ในการควบคุมเวอร์ชันและการทำงานร่วมกับทีมพัฒนา

5.2.3.3 ฝึกการใช้ Postman เบื้องต้น สำหรับทดสอบ API และตรวจสอบการเชื่อมต่อระหว่าง Frontend กับ Backend

5.2.3.4 ฝึกการจัดการเวลาและการทำงานเป็นทีม เพื่อปรับตัวให้เข้ากับการทำงานจริงในองค์กรได้รวดเร็ว



## บรรณานุกรม

คงรักษ์สยาม. (2563, 13 พฤษภาคม). รู้จักกับ TypeScript. Medium.

<https://kongruksiam.medium.com/รู้จักกับ-typescript-3e67f0db2897>

เพชรเจ็ช. (2566, 16 กรกฎาคม). Angular คืออะไร? เหมาะกับใคร งานแบบไหน. Medium.

<https://perjerz.medium.com/angular-คืออะไร-เหมาะกับใคร-งานแบบไหน-64a5e33220a0>

สแต่คไพรอน. (2564, 26 กุมภาพันธ์). ทำความรู้จัก Git / GitHub พร้อมการใช้งานร่วมกับ VS Code เบื้องต้น. Medium. <https://stackpython.medium.com/ทำความรู้จัก-git-github-พร้อมการใช้งานร่วมกับ-vs-code-เบื้องต้น-f848f41a39e9>

แองกูลาร์มายด์ส. (2565, 11 กรกฎาคม). Angular Lifecycle Hooks Best Practices. [เว็บไซต์].

<https://www.angularminds.com/blog/angular-lifecycle-hooks-best-practices>

ภาคผนวก

ภาคผนวก ก.  
รูปภาพขณะปฏิบัติงานสหกิจ



รูปที่ ก.1 ขณะปฏิบัติงาน

## ประวัติคณะผู้จัดทำ



รหัสนักศึกษา : 6504800026  
ชื่อ - นามสกุล : นาย ดนุพล เรืองอัมพร  
คณะ : วิทยาศาสตร์  
สาขาวิชา : วิทยาการคอมพิวเตอร์  
ที่อยู่ : อาคารธรรมรักษ์ 1 1058 ซอย 102/4  
ถนน เพชรเกษม บางแคเหนือ บางแค  
กรุงเทพฯ 10160  
ผลงาน : พัฒนาเว็บไซต์สำหรับแนะนำตัวเอง  
โดยใช้ HTML, CSS, และ JavaScript,  
React และ Vite  
Hero Search Website พัฒนาเว็บไซต์  
สำหรับค้นหาข้อมูลเกี่ยวกับ Super  
Hero โดยใช้ HTML, CSS, และ  
JavaScript



รหัสนักศึกษา : 6504800006  
ชื่อ - นามสกุล : นาย เคารพ -  
คณะ : วิทยาศาสตร์  
สาขาวิชา : วิทยาการคอมพิวเตอร์  
ที่อยู่ : 45 ซอยราชมนตรี 6 แขวงคลองขวาง  
เขตภาษีเจริญ 10160  
ผลงาน : พัฒนา เว็บไซต์สำหรับค้นหาข้อมูล  
เกี่ยวกับ หนังสือ โดยใช้ PHP และ MYSQL,  
XAMMP  
เว็บเพิ่มวิชาเรียน พัฒนา เว็บไซต์  
สำหรับลงวิชาและเพิ่มวิชาเรียน โดยใช้  
PHP, MySQL และ XAMMP



แบบสรุปรองงานสหกิจศึกษาและการศึกษาเชิงบูรณาการกับการทำงาน (CWIE)  
มหาวิทยาลัยสยาม

ข้อมูลของนักศึกษา

1. ชื่อ-สกุล : นาย ดนุพล เรืองอัมพร  
นาย เคารพ -
2. สาขาวิชา/คณะ : สาขาวิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์
3. E-mail นักศึกษา : kaolo.jai@siam.edu  
danupol.rea@siam.edu
4. ชื่อโครงการ/ผลงาน : การออกแบบและพัฒนาระบบหลังบ้าน เพื่อจัดการเนื้อหาเว็บไซต์
5. ชื่อสถานประกอบการ : บริษัท แอ็บโซลูท เมเนจเม้นท์ โซลูชันส์ จำกัด
6. ที่อยู่สถานประกอบการ : 22 ซอย เพชรเกษม 47/2 แขวงบางแค บางแค กรุงเทพมหานคร  
10160
7. ระยะเวลาปฏิบัติงาน : 19 พฤษภาคม 2568 ถึง 29 สิงหาคม 2568
8. ผู้ให้ผลงานในสถานประกอบการ (พนักงานพี่เลี้ยง)  
ชื่อ – สกุล : นางสาวจันทร์สุดา ศิริไกรวัฒน์, นายชลักร์ แสงศิริรัตน  
ตำแหน่ง : Head Programmer, DevOps

## ข้อมูลโครงการ/ผลงาน

1. โครงการ/ผลงาน/งานประจำ ได้รับการจัดระบบการทำงานที่เหมาะสมจากสถานประกอบการ ทั้งลักษณะงานและระยะเวลา มีการจัดระบบพี่เลี้ยงสอนงาน

โครงการ “การออกแบบและพัฒนาระบบหลังบ้านเพื่อจัดการเนื้อหาเว็บไซต์” ได้รับการสนับสนุนอย่างดีจาก บริษัท แอ็บโซลูท เมเนจเม้นท์ โซลูชันส์ จำกัด ซึ่งมีการจัดระบบการทำงานที่ชัดเจนเหมาะสมกับลักษณะงานและช่วงเวลาในการฝึกปฏิบัติ โดยมีการกำหนดขั้นตอนการเรียนรู้และการทำงานในแต่ละช่วงเวลาอย่างเป็นระบบ เช่น การศึกษาระบบ, การพัฒนา, การเชื่อมต่อ API, และการทดสอบระบบจริง

นอกจากนี้ สถานประกอบการยังจัดให้มี พี่เลี้ยง (Senior Programmer และ Lead Programmer) เพื่อให้คำแนะนำและสอนงานอย่างใกล้ชิด ทั้งในด้านเทคนิคและกระบวนการพัฒนาโครงการจริง ช่วยให้ผู้จัดทำเข้าใจหลักการทำงานของระบบหลังบ้านได้ลึกซึ้งขึ้น และสามารถแก้ไขปัญหาที่เกิดขึ้นในระหว่างการพัฒนาได้อย่างถูกต้องและมีประสิทธิภาพ

โครงการนี้มีเป้าหมายเพื่อพัฒนาระบบหลังบ้าน (Admin Panel) ที่สามารถจัดการข้อมูลบนเว็บไซต์ได้โดยตรง โดยเชื่อมต่อกับ API จากฝั่ง Backend เพื่อให้ผู้ดูแลระบบสามารถเพิ่ม แก้ไข หรือลบข้อมูล และอัปโหลดรูปภาพได้ทันทีบนเว็บไซต์จริง

2. การดำเนินงานมีความถูกต้อง มีระเบียบแบบแผนและทำให้ผู้จัดทำมีโอกาสประยุกต์ใช้วิชาความรู้/ทักษะตามที่ได้เรียนมา โดยใช้ความรู้ทักษะในการศึกษากระบวนการ การวิเคราะห์ และการแก้ปัญหา หรือสร้างแนวทางใหม่

โครงการ “การออกแบบและพัฒนาระบบหลังบ้านเพื่อจัดการเนื้อหาเว็บไซต์” ถูกดำเนินงานอย่างมีแบบแผนและเป็นระบบตั้งแต่เริ่มต้น โดยใช้กระบวนการพัฒนาโปรแกรมอย่างเป็นขั้นตอน (จากการศึกษาเทคโนโลยี พัฒนา เชื่อมต่อ ทดสอบ จัดทำเอกสาร)

การประยุกต์ใช้ความรู้ทางวิชาการ:

- ผู้จัดทำได้นำความรู้จากสาขาวิทยาการคอมพิวเตอร์มาใช้ในการพัฒนาระบบ เช่น ความรู้ด้านโครงสร้างข้อมูล, การพัฒนาเว็บแอปพลิเคชัน, และการออกแบบส่วนติดต่อผู้ใช้ (UI/UX)
- ใช้เครื่องมือและเทคโนโลยีที่ได้เรียนและศึกษาด้วยตนเอง เช่น Angular Framework, Tailwind CSS, Angular Material, และ Git เพื่อจัดการเวอร์ชันของโค้ด
- มีการทำงานร่วมกับทีม Backend ในการเชื่อมต่อ API และตรวจสอบการทำงานของระบบ เพื่อให้ข้อมูลในระบบหลังบ้านสามารถอัปเดตไปยังหน้าเว็บไซต์จริงได้อย่างถูกต้อง

การแก้ปัญหาและแนวทางใหม่:

- แก้ไขปัญหาที่เกิดจากการรวมโค้ด (merge conflict) ด้วยการใช้ branch แยก feature และเรียนรู้การ resolve conflict ผ่าน GitHub
- ปรับโครงสร้าง CSS และใช้ custom theme ของ Angular Material เพื่อแก้ปัญหาการแสดงผล UI ที่ซับซ้อน
- พัฒนาแนวทางใหม่ในการจัดการข้อมูล เช่น ระบบ Export ข้อมูลเป็นไฟล์ Excel, ระบบตรวจสอบความถูกต้องของข้อมูล (Validation) และระบบแสดงผล Responsive ให้รองรับทั้ง Desktop และ Mobile

### 3. เป็นโครงการ/ผลงานที่นำไปใช้ประโยชน์ได้อย่างเป็นรูปธรรมในสถานประกอบการ

หมายเหตุ: - หากเป็นงานประจำต้องสามารถนำไปพัฒนาองค์กร/หน่วยงานได้อย่างชัดเจน อาทิ ลดเวลาในการทำงานประจำ/ลดต้นทุนค่าใช้จ่าย

โครงการมีการสร้างความคิดสร้างสรรค์ให้กับสถานประกอบการในระหว่างปฏิบัติสหกิจศึกษาและการศึกษาเชิงบูรณาการกับการทำงาน หรือมีการยื่นจดคุ้มครองทรัพย์สินทางปัญญาหรือไม่ ถ้ามีโปรดอธิบายจากการพัฒนาโครงการ “การออกแบบและพัฒนาระบบหลังบ้านเพื่อจัดการเนื้อหาเว็บไซต์” สามารถนำไปใช้ประโยชน์ในสถานประกอบการได้อย่างชัดเจน โดยช่วยเสริมสร้างประสิทธิภาพในการบริหารจัดการข้อมูลภายในเว็บไซต์ และเป็นประโยชน์ต่อการดำเนินงานขององค์กร ดังนี้

การพัฒนาองค์กรและลดเวลาในการทำงาน:

- โครงการนี้ช่วยให้สถานประกอบการสามารถจัดการข้อมูลบนหน้าเว็บไซต์ได้ด้วยตนเอง โดยไม่ต้องพึ่งนักพัฒนาโปรแกรมทุกครั้งที่มีการแก้ไข เพิ่ม หรือลบข้อมูล ทำให้ลดระยะเวลาในการปรับปรุงข้อมูลและอัปเดตเว็บไซต์ได้อย่างรวดเร็ว
- ระบบที่พัฒนาขึ้นมีการเชื่อมต่อข้อมูลผ่าน API กับฝั่ง Backend ทำให้เมื่อมีการปรับเปลี่ยนข้อมูลในระบบหลังบ้าน ข้อมูลจะอัปเดตบนเว็บไซต์จริงได้ทันที ช่วยให้กระบวนการทำงานมีความคล่องตัวมากขึ้น

การลดต้นทุนค่าใช้จ่าย:

- ระบบหลังบ้านที่พัฒนาช่วยลดค่าใช้จ่ายในการดูแลเว็บไซต์ระยะยาว เพราะผู้ดูแลทั่วไปสามารถใช้งานได้โดยไม่ต้องจ้างบุคลากรเพิ่มเติม
- ระบบมีการตรวจสอบความถูกต้องของข้อมูล (Validation) เพื่อลดความผิดพลาดในการกรอกหรือแก้ไขข้อมูล ลดค่าใช้จ่ายจากการแก้ไขข้อมูลผิดพลาดในภายหลัง

ความคิดสร้างสรรค์และการพัฒนากระบวนการทำงาน:

- โครงการนี้นำเสนอแนวทางใหม่ในการพัฒนาเว็บแอปพลิเคชัน โดยใช้เทคโนโลยี Angular Framework และ Tailwind CSS ซึ่งช่วยให้ระบบมีความทันสมัย ใช้งานง่าย และรองรับการแสดงผลทั้งบน Desktop และ Mobile
- แนวคิดดังกล่าวสามารถต่อยอดไปพัฒนาเป็นระบบบริหารจัดการเว็บไซต์อัตโนมัติ (Web CMS) ที่เหมาะกับองค์กรอื่น ๆ ได้ ช่วยสร้างคุณค่าใหม่ให้กับบริษัทในอนาคต

การจดคุ้มครองทรัพย์สินทางปัญญา:

ในขณะนี้ โครงการยังไม่ได้ดำเนินการยื่นจดคุ้มครองทรัพย์สินทางปัญญา แต่แนวคิดและผลการพัฒนาโครงการสามารถต่อยอดไปสู่การสร้างซอฟต์แวร์หรือโมดูลบริหารจัดการข้อมูลเว็บไซต์เชิงพาณิชย์ ที่มีศักยภาพในการพัฒนาเป็นผลิตภัณฑ์ของบริษัท และสามารถนำไปจดคุ้มครองในอนาคตได้



(<https://bit.ly/4p4LaHX>)

การออกแบบและพัฒนาระบบส่วนหลังเพื่อจัดการเนื้อหาเว็บไซต์  
Design and Development of a Backend System for Website Content Management  
บริษัท แอ็บโซลูท เมเนจเม้นท์ โซลูชันส์ จำกัด  
Absolute Management Solutions Co., Ltd.

โดย

|          |            |            |
|----------|------------|------------|
| นายธนุพล | เรียมอัมพร | 6504800026 |
| นายเครพ  |            | 6504800006 |

รายงานนี้เป็นส่วนหนึ่งของรายวิชา 128-491 สหกิจศึกษาสำหรับนักวิทยาการคอมพิวเตอร์  
หลักสูตรวิทยาศาสตรบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์  
คณะวิทยาศาสตร์ มหาวิทยาลัยสยาม  
ภาคการศึกษาที่ 3 ปีการศึกษา 2567