



รายงานการปฏิบัติงานสหกิจศึกษา

โครงการบันทึกเวลาทำงานของนักศึกษาฝึกสหกิจศึกษา

สำหรับบริษัท รีเลิร์น โซลูชั่น จำกัด

Project PresentTrack (Intern Attendance Calendar)

For Relearn Solution Co., Ltd

โดย

นางสาวสุจีรา นิตยารส 6504000011

รายงานนี้เป็นส่วนหนึ่งของรายวิชา 155-393 สหกิจศึกษา

หลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยสยาม

ภาคการศึกษา 3 ปีการศึกษา 2567

หัวข้อโครงการ บันทึกเวลาทำงานของนักศึกษาฝึกสหกิจศึกษา สำหรับบริษัท
รีเลิร์น โซลูชั่น จำกัด

PresentTrack (Intern Attendance Calendar) for Relearn
Solution Co., Ltd

รายชื่อผู้จัดทำ นางสาวสุจีรา นิตยารส 6504000011

หลักสูตร หลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์

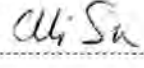
อาจารย์นิเทศ อาจารย์สรายุทธ อินทรเสมา

อนุมัติให้โครงการนี้เป็นส่วนหนึ่งของการปฏิบัติงานสหกิจศึกษาและศึกษา
เชิงบูรณาการกับการทำงาน หลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ มหาวิทยาลัยสยาม ภาคการศึกษาที่ 3 ปีการศึกษา 2567

คณะกรรมการการสอบโครงการ

.....อาจารย์นิเทศ
(อาจารย์สรายุทธ อินทรเสมา)

.....ผู้นิเทศ
(นายชาญชัย ศรีจันทร์)

.....กรรมการกลาง
(ผู้ช่วยศาสตราจารย์วิภาวดี นาคทรัพย์)

.....ผู้ช่วยอธิการบดีและผู้อำนวยการสำนักสหกิจศึกษา
(ผู้ช่วยศาสตราจารย์.ดร.มารุจ ลิ้มปะวัฒน์นะ)

หัวข้อโครงการ

บันทึกเวลาทำงานของนักศึกษาฝึกสหกิจศึกษา สำหรับ
บริษัท รีเลิร์น โซลูชั่น จำกัด

PresentTrack (Intern Attendance Calendar) for
Relearn Solution Co., Ltd

หน่วยกิต

6 หน่วยกิต

ผู้จัดทำ

นางสาวสุจีรา นิตยารส 6504000011

อาจารย์ที่ปรึกษา

อาจารย์สรายุทธ อินทรเสมา

ระดับการศึกษา

ปริญญาตรี

หลักสูตร

วิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์

คณะ

วิศวกรรมศาสตร์

ภาคการศึกษา / ปีการศึกษา

3/2565

บทคัดย่อ

บริษัท รีเลิร์น โซลูชั่น จำกัด (Relearn Solutions Co., Ltd.) เป็นองค์กรด้านเทคโนโลยีและการศึกษา ผู้ดำเนินการหลักของโครงการ Code Camp Thailand และแพลตฟอร์ม Get Code โดยมีพันธกิจสำคัญในการขับเคลื่อนสังคมดิจิทัลผ่านทักษะการเขียนโปรแกรมคอมพิวเตอร์ บริษัทให้บริการครอบคลุมทั้งด้านการพัฒนาบุคลากรผ่านหลักสูตรการเรียนการสอนทั้งรูปแบบออนไลน์และออฟไลน์ ที่เน้นกระบวนการลงมือปฏิบัติจริงภายใต้การดูแลของผู้เชี่ยวชาญที่มีประสบการณ์ เพื่อให้ผู้เรียนสามารถนำทักษะไปประยุกต์ใช้งานได้ด้วยตนเอง ควบคู่ไปกับการให้บริการด้านการพัฒนาซอฟต์แวร์และรับเขียนโค้ดสำหรับผู้ประกอบการที่ต้องการนำเทคโนโลยีไปยกระดับธุรกิจ ทั้งนี้ บริษัทมุ่งมั่นที่จะพัฒนาระบบการเรียนรู้และศักยภาพของบุคลากรอย่างต่อเนื่อง เพื่อส่งมอบผลลัพธ์ที่มีคุณภาพสูงสุดและตอบสนองต่อความต้องการของโลกยุคดิจิทัลได้อย่างมีประสิทธิภาพ

คำสำคัญ : เว็บแอปพลิเคชัน, นักศึกษาฝึกสหกิจศึกษา, ปฏิทินเช็คชื่อ

Project Title: PresentTrack (Internship Attendance Calendar) for
Relearn Solution Co., Ltd

Credits: 6 credits

By: Miss Sujeera Nittayaros

Advisor: Mr. Sarayut Intarasema

Degree: Bachelor of Engineering

Major: Computer Engineering

Faculty: Engineering

Semester / Academic year: 3/2022

Abstract

Relearn Solutions Co., Ltd. is a technology and education organization that serves as the main operator of the Code Camp Thailand project and the Get Code platform. Its primary mission is to drive a digital society through computer programming skills. The company provides comprehensive services, including personnel development through both online and offline training programs. These programs emphasize practical, hands-on learning under the guidance of experienced experts, enabling learners to apply their skills independently. In addition, the company offers software development and coding services for entrepreneurs seeking to leverage technology to enhance their businesses. The organization is committed to continuously developing learning systems and human potential in order to deliver high-quality outcomes and effectively meet the demands of the digital era.

Keywords: Web Application, Interns, Calendar Attendance

Advisor

จดหมายนำส่งรายงาน

วันที่ 4 เดือน กุมภาพันธ์ พ.ศ. 2569

เรื่อง ขอส่งรายงานการปฏิบัติงานสหกิจศึกษา

เรียน อาจารย์นิเทศ หลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์

อาจารย์สรายุทธ อินทรเสมา

ตามที่นางสาวสุจีรา นิตยารส นักศึกษาหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยสยาม ได้ไปปฏิบัติงานสหกิจศึกษาและการศึกษาเชิงบูรณาการกับการทำงานระหว่างวันที่ 26 พฤษภาคม พ.ศ.2568 ถึงวันที่ 5 กันยายน พ.ศ.2568 ในตำแหน่ง Tester ณ บริษัท รีเลิร์น โซลูชั่น จำกัด และได้รับมอบหมายจากผู้นิเทศ (พนักงานที่ปรึกษา) ให้ศึกษาและทำโครงการเรื่องบันทึกเวลาทำงานของนักศึกษาฝึกสหกิจศึกษาสำหรับ บริษัท รีเลิร์น โซลูชั่น จำกัด

บัดนี้การปฏิบัติงานสหกิจศึกษาและการศึกษาเชิงบูรณาการกับการทำงานได้สิ้นสุดแล้ว นางสาวสุจีรา นิตยารส/ผู้จัดทำ จึงขอส่งรายงานดังกล่าวมาพร้อมกันนี้จำนวน 1 เล่ม เพื่อขอรับคำปรึกษาต่อไป

จึงเรียนมาโปรดพิจารณา

ขอแสดงความนับถือ

ลงชื่อ.....

(นางสาวสุจีรา นิตยารส)

ผู้จัดทำ

กิตติกรรมประกาศ (Acknowledgement)

การที่ผู้จัดทำได้มาปฏิบัติสหกิจ ในตำแหน่ง Tester ณ บริษัท รีเลิร์น โซลูชั่น จำกัด ตั้งแต่วันที่ 26 พฤษภาคม 2568 ถึงวันที่ 5 กันยายน 2568 ได้สำเร็จลุล่วงตามวัตถุประสงค์ด้วยดี ส่งผลให้ผู้จัดทำได้รับความรู้ ประสบการณ์การทำงานต่างๆ และความเข้าใจในชีวิตการทำงานจริง ที่เป็นประโยชน์ต่อการเรียนและสามารถนำความรู้ประสบการณ์ที่ได้ไปใช้ในการประกอบอาชีพในอนาคต ด้วยความอนุเคราะห์อย่างยิ่ง จากบริษัท รีเลิร์น โซลูชั่น จำกัด ที่ให้โอกาสผู้จัดทำเข้ามาปฏิบัติสหกิจศึกษา กรุณาเสียสละเวลาอบรม สอนงาน และช่วยเหลือด้านต่างๆ ตลอดระยะเวลาที่ปฏิบัติสหกิจศึกษาในครั้งนี้ จึงขอขอบพระคุณอย่างสูงมา ณ ที่นี้ จากการสนับสนุนหลายฝ่าย ดังนี้

1. นางสาวศศิชา ศิริบุรณ
2. นายชาญชัย ศรีจันทร์
3. อาจารย์สรายุทธ อินทรเสมา (อาจารย์นิเทศ)

และบุคคลที่ไม่ได้กล่าวชื่อนามทุกท่าน ที่ได้ให้คำแนะนำช่วยเหลือในการจัดทำรายงานสหกิจศึกษาฉบับนี้จนเสร็จสมบูรณ์

ผู้จัดทำหวังเป็นอย่างยิ่งว่ารายงานฉบับนี้จะเป็นประโยชน์ต่อ บริษัท รีเลิร์น โซลูชั่น จำกัด และผู้สนใจปฏิบัติสหกิจศึกษาของบริษัทเพื่อเป็นแนวทางเบื้องต้นในการทำความเข้าใจและพัฒนาโครงการต่อไป รวมทั้งในการค้นคว้าของผู้สนใจทั่วไปด้วย หากรายงานฉบับนี้มีข้อผิดพลาดประการใดผู้จัดทำ ก็ขออภัยมา ณ ที่นี้

สุจิรา นิตยารส

ผู้จัดทำ

4 กุมภาพันธ์ 2569

สารบัญ

	หน้า
จดหมายนำส่งรายงาน	ก
กิตติกรรมประกาศ	ข
บทคัดย่อ	ค
Abstract	ง
บทที่ 1 บทนำ	1
1.1 ข้อมูลของสถานประกอบการและลักษณะงานที่ได้รับมอบหมาย	1
1.2 ที่มาและความสำคัญของโครงการ	1
1.3 วัตถุประสงค์ของโครงการ	1
1.4 ขอบเขตของโครงการ	2
1.5 ประโยชน์ที่คาดว่าจะได้รับ	2
บทที่ 2 การทบทวนเอกสารงานวิจัย/วรรณกรรมที่เกี่ยวข้อง (Literature)	3
2.1 ทฤษฎีเกี่ยวกับการพัฒนาเว็บแอปพลิเคชัน	3
2.2 เว็บแอปพลิเคชันปฏิทินเช็คอินนักศึกษา	8
บทที่ 3 รายละเอียดการปฏิบัติงาน	14
3.1 รายละเอียดการปฏิบัติงานตามโครงการ	14
3.2 รายละเอียดการปฏิบัติงานตามที่ได้รับมอบหมาย	15
3.3 รูปแบบการจัดการองค์กรและการบริหารขององค์กร	16
3.4 ตำแหน่งและลักษณะงานที่ได้รับมอบหมาย	17

สารบัญ (ต่อ)

หน้า

3.5 ชื่อและตำแหน่งของพนักงานที่ปรึกษา	17
3.6 ระยะเวลาในการปฏิบัติสหกิจศึกษา	18
3.7 ขั้นตอนและการดำเนินงาน	18
3.8 อุปกรณ์และเครื่องมือที่ใช้	19
บทที่ 4 ผลการปฏิบัติงาน	20
4.1 ผลการปฏิบัติงานตามโครงการ	20
4.2 ผลการปฏิบัติงานตามที่ได้รับมอบหมาย	24
บทที่ 5 สรุปผลและข้อเสนอแนะ	31
5.1 สรุปผลโครงการ	31
5.2 สรุปผลการปฏิบัติสหกิจศึกษา	32
บรรณานุกรม	33
ภาคผนวก	
ภาคผนวก ก หนังสือยินยอมให้เผยแพร่รายงาน/โครงการสหกิจศึกษา	36
ภาคผนวก ข ภาพการนิเทศงานของอาจารย์	38
ภาคผนวก ค ภาพขณะปฏิบัติงาน	40
ภาคผนวก ง ภาพการนำเสนอโครงการ	43
ภาคผนวก จ การตรวจสอบการลอกเลียนวรรณกรรมทางวิชาการ	45
แบบสรุปโครงการสหกิจศึกษาและการศึกษาเชิงบูรณาการกับการทำงาน (CWIE) มหาวิทยาลัยสยาม	

สารบัญตาราง

หน้า

ตารางที่ 3.1 ระยะเวลาในการฝึกสกิจศึกษา	1
--	---



สารบัญรูปภาพ

หน้า

รูปที่ 2.1 องค์ประกอบพื้นฐานในการพัฒนาส่วนติดต่อผู้ใช้งาน: HTML5 JavaScript และ CSS3	3
รูปที่ 2.2 องค์ประกอบพื้นฐานในการพัฒนาส่วนติดต่อผู้ใช้งาน: Python	3
รูปที่ 2.3 รูปโปรแกรม Docker	4
รูปที่ 2.4 เครื่องมือ Kubernetes สำหรับบริหารจัดการคอนเทนเนอร์และกระบวนการนำส่งซอฟต์แวร์	4
รูปที่ 2.5 เครื่องมือ GitLab สำหรับบริหารจัดการคอนเทนเนอร์และกระบวนการนำส่งซอฟต์แวร์	5
รูปที่ 2.6 เครื่องมือ Trivy สำหรับการตรวจสอบช่องโหว่และการจัดการนโยบายความปลอดภัย	5
รูปที่ 2.7 เครื่องมือ Kube-bench สำหรับการตรวจสอบช่องโหว่และการจัดการนโยบายความปลอดภัย	6
รูปที่ 2.8 เครื่องมือ Open Policy Agent สำหรับการตรวจสอบช่องโหว่และการจัดการนโยบายความปลอดภัย	6
รูปที่ 2.9 เครื่องมือ OWASP ZAP สำหรับการตรวจสอบช่องโหว่และการจัดการนโยบายความปลอดภัย	6
รูปที่ 2.10 เครื่องมือ Prometheus & Grafana สำหรับการเฝ้าระวังระบบและการวิเคราะห์ข้อมูลจราจรคอมพิวเตอร์	7
รูปที่ 2.11 เครื่องมือ ELK Stack สำหรับการเฝ้าระวังระบบและการวิเคราะห์ข้อมูลจราจรคอมพิวเตอร์	7
รูปที่ 2.12 หน้าเว็บแอปพลิเคชันปฏิทินเช็กชื่อนักศึกษา	8
รูปที่ 2.13 หน้าเว็บเบราว์เซอร์	10

สารบัญรูปภาพ (ต่อ)

	หน้า
รูปที่ 2.14 หน้า Login/Register	10
รูปที่ 2.15 หน้า celendar แสดงปฏิทินของเดือน	11
รูปที่ 2.16 หน้า dashbord แสดงกราฟวงกลมและกราฟแท่ง	12
รูปที่ 2.17 หน้า Profile กรอกข้อมูลในแบบฟอร์ม	12
รูปที่ 3.1 แผนทื่อาคารวรรณสรณ์	14
รูปที่ 3.2 รูปแบบการจัดการองค์กร	16
รูปที่ 4.1 ภาพการทดสอบตาม Requirements	20
รูปที่ 4.2 ภาพการเขียน Use Case ตาม Requirements	21
รูปที่ 4.3 ภาพการบันทึกข้อผิดพลาดของระบบ (Bug Reporting) ด้วยเครื่องมือ Jam	21
รูปที่ 4.4 ภาพ source code ผ่านโปรแกรม Visual Studio Code	22
รูปที่ 4.5 หน้าจอแสดงผลการทำงานของสคริปต์ทดสอบอัตโนมัติ	23
รูปที่ 4.6 ตัวอย่างชุดคำสั่ง สำหรับการทดสอบฟังก์ชันพิมพ์รายงาน	23
รูปที่ 4.7 ผลลัพธ์การรันทดสอบผ่าน Playwright (Test Passed)	23
รูปที่ 4.8 หน้าระบบ Login/Register	24
รูปที่ 4.9 หน้าระบบบันทึกเวลาเข้าทำงาน	24
รูปที่ 4.10 หน้า Dashbord สรุปผลการบันทึกเวลาเข้าทำงาน	25
รูปที่ 4.11 หน้าบันทึกข้อมูล Profile ของผู้ใช้งาน	25
รูปที่ 4.12 ภาพ source code database การทำงานของส่วนหลังบ้าน	26
รูปที่ 4.13 ภาพ source code การเชื่อมต่อ Frontend และ Backend	26
รูปที่ 4.14 ภาพระบบโปรแกรม Docker	27

สารบัญรูปภาพ (ต่อ)

	หน้า
รูปที่ 4.15 ภาพระบบโปรแกรม Docker ที่ใช้เครื่องมือ kubernetes	27
รูปที่ 4.16 ภาพเครื่องมือ GitLab ที่ทำการ Build Scan และ deploy	27
รูปที่ 4.17 ภาพสแกนหาช่องโหว่ด้วยเครื่องมือ Trivy	28
รูปที่ 4.18 ภาพผลการสแกนทดสอบช่องโหว่	28
รูปที่ 4.19 ภาพการติดตั้งเครื่องมือ	29
รูปที่ 4.20 ภาพการทดสอบการใช้เครื่องมือ OPA	29
รูปที่ 4.21 ภาพเครื่องมือ OWAPS ZAP สแกนหาช่องโหว่ของเว็บแอปพลิเคชัน	30

บทที่ 1

บทนำ

1. ข้อมูลของสถานประกอบการ

บริษัท รีเลิร์น โซลูชัน จำกัด (Relearn Solutions Co., Ltd.) เป็นองค์กรด้านเทคโนโลยีและการศึกษา ผู้ดำเนินการหลักของโครงการ Code Camp Thailand และแพลตฟอร์ม Get Code โดยมีพันธกิจสำคัญในการขับเคลื่อนสังคมดิจิทัลผ่านทักษะการเขียนโปรแกรมคอมพิวเตอร์ บริษัทให้บริการครอบคลุมทั้งด้านการพัฒนาบุคลากรผ่านหลักสูตรการเรียนการสอนทั้งรูปแบบออนไลน์และออฟไลน์ ที่เน้นกระบวนการลงมือปฏิบัติจริงภายใต้การดูแลของผู้เชี่ยวชาญที่มีประสบการณ์ เพื่อให้ผู้เรียนสามารถนำทักษะไปประยุกต์ใช้งานได้ด้วยตนเอง ควบคู่ไปกับการให้บริการด้านการพัฒนาซอฟต์แวร์และรับเขียนโค้ดสำหรับผู้ประกอบการที่ต้องการนำเทคโนโลยีไปยกระดับธุรกิจ

2. ที่มาและความสำคัญของโครงการ

ปัจจุบัน บริษัท รีเลิร์น โซลูชัน จำกัด มีระบบบันทึกเวลาเข้า-ออกงานของพนักงานประจำด้วยการสแกนลายนิ้วมือและใบหน้า อย่างไรก็ตาม ระบบดังกล่าวยังไม่รองรับการใช้งานสำหรับนักศึกษาฝึกงาน (Intern) ส่งผลให้การตรวจสอบการเข้างานเกิดความล่าช้าและไม่สะดวก เนื่องจากต้องใช้วิธีการขานชื่อรายบุคคล ซึ่งใช้เวลานานโดยเฉพาะเมื่อมีนักศึกษาฝึกงานจำนวนมาก

เพื่อแก้ไขปัญหาดังกล่าว โครงการ PresenTrack (Intern Attendance Calendar) จึงถูกพัฒนาขึ้นเป็นระบบบันทึกเวลาทำงานของนักศึกษาฝึกสหกิจศึกษา ระบบสามารถบันทึกและตรวจสอบการเข้าชื่อย้อนหลังได้ แต่ไม่สามารถบันทึกหรือตรวจสอบล่วงหน้าได้ ผู้ใช้งานสามารถดูสถิติการเข้าทำงานผ่านกราฟวงกลมสำหรับสรุปผลรายเดือน และกราฟแท่งสำหรับสรุปผลรายสัปดาห์ การเข้าชื่อย้อนหลังครอบคลุมกรณีที่นักศึกษาฝึกสหกิจศึกษาเข้าทำงานแต่ลืมบันทึกเวลาเข้าทำงาน ระบบนี้ช่วยลดข้อผิดพลาด เพิ่มความสะดวกและประสิทธิภาพในการบริหารจัดการพนักงานนักศึกษาฝึกสหกิจศึกษา

3. วัตถุประสงค์ของโครงการ

- 3.1.1 เพื่อพัฒนาระบบบันทึกเวลาทำงานของนักศึกษาฝึกสหกิจศึกษา
- 3.1.2 เพื่อให้ผู้ใช้งานสามารถบันทึกและตรวจสอบข้อมูลย้อนหลัง
- 3.1.3 เพื่อเพิ่มประสิทธิภาพและความสะดวกในการบริหารจัดการฝึกสหกิจศึกษา

4. ขอบเขตของโครงการ

- 4.1.1 พัฒนาระบบบันทึกเวลาทำงานนักศึกษาฝึกสหกิจศึกษา
- 4.1.2 บันทึกเวลาเข้าทำงานย้อนหลัง แต่ไม่สามารถบันทึกเวลาเข้าทำงานล่วงหน้าได้
- 4.1.3 แสดงสรุปสถิติการเข้างานในรูปแบบกราฟ และสามารถดูข้อมูลย้อนหลังได้

5. ประโยชน์ที่คาดว่าจะได้รับ

- 5.1.1 เพิ่มความสะดวกในการบันทึกเวลาเข้าทำงานและลดความซับซ้อน
- 5.1.2 ลดปัญหาข้อมูลผิดพลาดหรือสูญหาย โดยสามารถติดตามและบันทึกข้อมูลย้อนหลัง
- 5.1.3 ช่วยให้สรุปผลได้อย่างรวดเร็ว ผ่านการแสดงสถิติในรูปแบบกราฟที่เข้าใจง่าย



บทที่ 2

การทบทวนเอกสารและวรรณกรรมที่เกี่ยวข้อง

ในการพัฒนาโครงงานบันทึกเวลาทำงานของนักศึกษาฝึกสหกิจศึกษาผู้จัดทำได้ศึกษา ทฤษฎีและเทคโนโลยีที่เกี่ยวข้องเพื่อให้ระบบทำงานได้อย่างมีประสิทธิภาพ มีความปลอดภัย และรองรับการขยายตัวในอนาคต

2.1 ทฤษฎีเกี่ยวกับการพัฒนาเว็บแอปพลิเคชัน (Web Application Development)

2.1.1 ส่วนหน้าบ้าน (Frontend)



รูปที่ 2.1 องค์ประกอบพื้นฐานในการพัฒนาส่วนติดต่อผู้ใช้งาน: HTML5 JavaScript และ CSS3

ทำหน้าที่เป็นส่วนติดต่อกับผู้ใช้งาน (User Interface) เพื่อรับข้อมูลและแสดงผล โดยใช้เทคโนโลยีหลักคือ

- HTML สำหรับวางโครงสร้าง
- CSS สำหรับตกแต่งความสวยงาม
- JavaScript สำหรับจัดการการทำงานโต้ตอบกับผู้ใช้ เช่น การกดปุ่มเช็คชื่อ หรือการแสดงผลปฏิทิน

2.1.2 ส่วนหลังบ้าน (Backend)



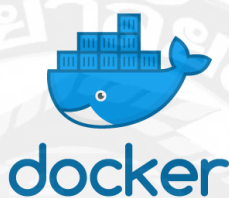
รูปที่ 2.2 องค์ประกอบพื้นฐานในการพัฒนาส่วนติดต่อผู้ใช้งาน: Python

ทำหน้าที่ประมวลผลข้อมูล จัดการตรรกะของระบบ (Business Logic) และเชื่อมต่อกับฐานข้อมูล โครงสร้างนี้เลือกใช้ภาษา Python ร่วมกับเฟรมเวิร์ก Flask ซึ่งเป็น Micro-framework ที่มีความยืดหยุ่นสูง โดยมีการสื่อสารระหว่าง Frontend และ Backend ผ่านรูปแบบ REST API

ฐานข้อมูล (Database) ในระยะเริ่มต้นและเพื่อความคล่องตัวในการพัฒนา โครงงานนี้ใช้การจัดเก็บข้อมูลในรูปแบบไฟล์ JSON (db.json) เพื่อบันทึกข้อมูลผู้ใช้งานและประวัติการเข้างาน ซึ่งง่ายต่อการจัดการและเคลื่อนย้าย

เทคโนโลยีคอนเทนเนอร์และการจัดการ (Containerization & Orchestration)

2.1.3 Docker



รูปที่ 2.3 รูปโปรแกรม Docker

ใช้สำหรับทำ Containerization หรือการบรรจุแอปพลิเคชันและสิ่งที่จำเป็นต้องใช้ (Dependencies) ทั้งหมดลงใน "Image" เพื่อให้สามารถนำไปรันที่เครื่องใดก็ได้โดยทำงานได้เหมือนกัน โดยมีการเขียนไฟล์ Dockerfile เพื่อกำหนดขั้นตอนการสร้าง Image ทั้งส่วนของ Frontend (Nginx) และ Backend (Python)

2.1.4 Kubernetes (K8s)



รูปที่ 2.4 เครื่องมือ Kubernetes สำหรับบริหารจัดการคอนเทนเนอร์และกระบวนการนำส่งซอฟต์แวร์

ใช้สำหรับบริหารจัดการ Container จำนวนมาก (Orchestration) ช่วยในเรื่องการทำ Deployment และ Service เพื่อให้แอปพลิเคชันมีความเสถียร ในโครงงานนี้ใช้ Minikube ในการจำลอง Cluster ขนาดเล็กสำหรับการพัฒนาและทดสอบ

แนวคิด DevSecOps และระบบอัตโนมัติ (DevSecOps & Automation)

2.1.5 CI/CD Pipeline (GitLab CI/CD)



รูปที่ 2.5 เครื่องมือ GitLab สำหรับบริหารจัดการคอนเทนเนอร์และกระบวนการนำส่งซอฟต์แวร์

ใช้ GitLab ในการเก็บรวบรวมโค้ด (Version Control) และสร้างระบบอัตโนมัติ (Pipeline) ที่ประกอบด้วยขั้นตอนการ Build Scan และ Deploy ไปยัง Kubernetes ทันทีที่มีการอัปเดตโค้ด ช่วยลดความผิดพลาดจากการทำงานด้วยคน (Human Error)

2.1.6 Infrastructure as Code (IaC)

การจัดการโครงสร้างพื้นฐานด้วยการเขียนโค้ด เช่น การใช้ไฟล์ YAML สำหรับ Kubernetes Manifests เพื่อกำหนดสถานะของระบบ (Desired State)

ด้านความปลอดภัยและการตรวจสอบ (Security & Testing Tools)

2.1.7 Trivy



รูปที่ 2.6 เครื่องมือสำหรับการตรวจสอบช่องโหว่และการจัดการนโยบายความปลอดภัย

เครื่องมือสำหรับสแกนหาช่องโหว่ (Vulnerability Scanning) ใน Docker Image เพื่อตรวจสอบว่าไลบรารีหรือ Base Image ที่ใช้มีจุดอ่อนด้านความปลอดภัยหรือไม่

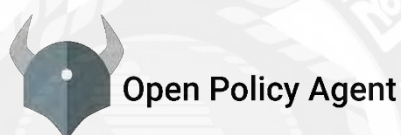
- Kube-bench



รูปที่ 2.7 เครื่องมือ Kube-bench สำหรับการตรวจสอบช่องโหว่และการจัดการนโยบายความปลอดภัย

ใช้สำหรับตรวจสอบความปลอดภัยของการตั้งค่า Kubernetes Cluster ว่าเป็นไปตามมาตรฐาน CIS Benchmark หรือไม่

2.1.8 OPA (Open Policy Agent)



รูปที่ 2.8 เครื่องมือ Open Policy Agent สำหรับการตรวจสอบช่องโหว่และการจัดการนโยบายความปลอดภัย

ใช้สำหรับการกำหนดนโยบายความปลอดภัยแบบโค้ด (Policy-as-Code) เพื่อตรวจสอบและป้องกันไม่ให้มีการ Deploy ทรัพยากรที่ไม่ปลอดภัยเข้าสู่ระบบ

2.1.9 OWASP ZAP



รูปที่ 2.9 เครื่องมือ OWASP ZAP สำหรับการตรวจสอบช่องโหว่และการจัดการนโยบายความปลอดภัย

ใช้สำหรับทดสอบความปลอดภัยของเว็บแอปพลิเคชันแบบ Dynamic (DAST) เพื่อจำลองการโจมตีและค้นหาช่องโหว่ในขณะที่ระบบกำลังทำงาน

ระบบติดตามและเฝ้าระวัง (Monitoring & Logging)

2.1.10 Prometheus & Grafana



รูปที่ 2.10 เครื่องมือ Prometheus & Grafana สำหรับการเฝ้าระวังระบบและการวิเคราะห์ข้อมูล
จรรยาบรรณคอมพิวเตอร์

ใช้ Prometheus ในการเก็บข้อมูลตัวชี้วัด (Metrics) เช่น การใช้ CPU Memory ของ Pods
และใช้ Grafana ในการนำข้อมูลเหล่านั้นมาแสดงผลเป็น Dashboard ที่เข้าใจง่าย

2.1.11 ELK Stack



รูปที่ 2.11 เครื่องมือ ELK Stack สำหรับการเฝ้าระวังระบบและการวิเคราะห์ข้อมูลจรรยา
บรรณคอมพิวเตอร์

ประกอบด้วย Elasticsearch และ Kibana ใช้สำหรับการรวบรวมและค้นหา Logs ของ
ระบบ เพื่อใช้ในการตรวจสอบข้อผิดพลาด (Troubleshooting) และวิเคราะห์การใช้งาน

2.2 เว็บแอปพลิเคชันบันทึกเวลาทำงานของนักศึกษาฝึกสหกิจศึกษา

PresentTrack (Internship Attendance Calendar)

เว็บแอปพลิเคชันบันทึกเวลาทำงานของนักศึกษาฝึกสหกิจศึกษา (PresentTrack) เป็นระบบที่พัฒนาขึ้นเพื่อเพิ่มประสิทธิภาพในการบริหารจัดการเวลาการเข้างานของนักศึกษาฝึกงาน โดยมีวัตถุประสงค์เพื่อลดความผิดพลาดจากการบันทึกข้อมูลด้วยมือและอำนวยความสะดวกในการตรวจสอบข้อมูลย้อนหลัง ระบบถูกออกแบบภายใต้สถาปัตยกรรมแบบ Client-Server ซึ่งแยกส่วนการทำงานระหว่างส่วนหน้าบ้าน (Frontend) และส่วนหลังบ้าน (Backend) อย่างชัดเจน เพื่อให้ระบบมีความยืดหยุ่นและง่ายต่อการบำรุงรักษา



รูปที่ 2.12 หน้าเว็บแอปพลิเคชันปฏิทินเช็คชื่อนักศึกษา

2.2.1 ส่วนประกอบของระบบ (System Components)

โครงสร้างของ PresentTrack ประกอบด้วยเทคโนโลยีและการทำงาน ดังนี้

- ส่วนหน้าบ้าน (Frontend): ทำหน้าที่เป็นส่วนติดต่อกับผู้ใช้งาน (User Interface) เพื่อรับค่าและแสดงผลข้อมูล โดยพัฒนาด้วยภาษามาตรฐานได้แก่ HTML สำหรับโครงสร้างหน้าเว็บ CSS สำหรับการตกแต่ง และ JavaScript สำหรับจัดการการทำงานโต้ตอบกับผู้ใช้งาน
- ส่วนหลังบ้าน (Backend): ทำหน้าที่ประมวลผลคำสั่งจากผู้ใช้งานและจัดการข้อมูล พัฒนาโดยใช้ภาษา Python ร่วมกับ Flask Framework ซึ่งเป็นเว็บเฟรมเวิร์กขนาดเล็กที่มีความคล่องตัวสูง ทำหน้าที่สร้าง API (Application Programming Interface) เพื่อรับส่งข้อมูลกับส่วนหน้าบ้าน

- ฐานข้อมูล (Database): ระบบใช้การจัดเก็บข้อมูลในรูปแบบไฟล์ JSON (db.json) ซึ่งเป็นรูปแบบการเก็บข้อมูลแบบ Text-based ที่มีโครงสร้างอ่านง่ายและประมวลผลได้รวดเร็ว เหมาะสำหรับการจัดเก็บข้อมูลผู้ใช้งานและประวัติการลงเวลาในโครงการนี้

2.2.2 โครงสร้างพื้นฐานและการจัดการ (Infrastructure & Orchestration)

เพื่อให้ระบบมีความทันสมัยและรองรับการขยายตัวในอนาคต PresentTrack ได้นำเทคโนโลยีคอนเทนเนอร์มาประยุกต์ใช้ ดังนี้

- Docker: ระบบถูกบรรจุลงใน Docker Container โดยแยก Container สำหรับ Frontend (ใช้ Nginx Web Server) และ Backend (ใช้ Python) ออกจากกัน เพื่อให้สามารถนำไปติดตั้งและใช้งานบนสภาพแวดล้อมต่างๆ ได้โดยไม่เกิดปัญหาความเข้ากันไม่ได้ของซอฟต์แวร์
- Kubernetes: มีการนำ Kubernetes (โดยจำลองผ่าน Minikube) มาใช้ในการบริหารจัดการ Container (Orchestration) ช่วยในการทำ Deployment และ Service เพื่อให้แอปพลิเคชันมีความเสถียรและสามารถเข้าถึงได้ผ่านเครือข่าย

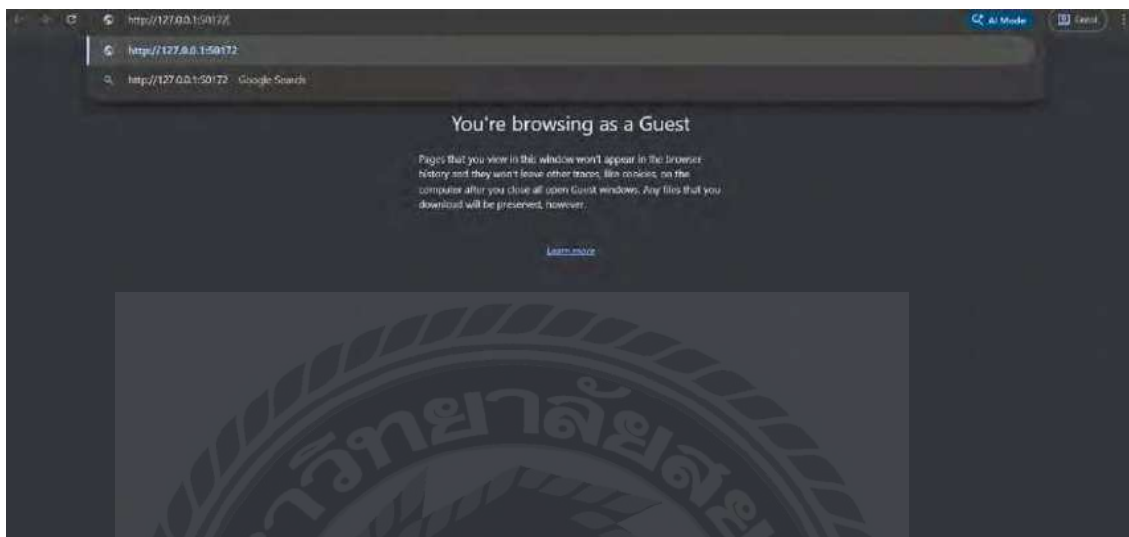
2.2.3 ฟังก์ชันการทำงานหลัก (Key Features)

ระบบ PresentTrack มีฟังก์ชันสำคัญที่ตอบโจทย์การใช้งาน ได้แก่

- ระบบปฏิทิน (Calendar): แสดงสถานะการเข้างานในรูปแบบปฏิทินรายเดือน โดยมีสถานะสีแจ้งเตือน (เช่น สีเขียวเมื่อบันทึกแล้ว) และปุ่มกดบันทึกการเข้าทำงานสำหรับวันที่ปัจจุบัน
- ระบบแดชบอร์ด (Dashboard): สรุปผลการเข้างานรายเดือนและรายสัปดาห์
- ระบบจัดการผู้ใช้ (User Management): รองรับการสมัครสมาชิก เข้าสู่ระบบ และแก้ไขข้อมูลส่วนตัว

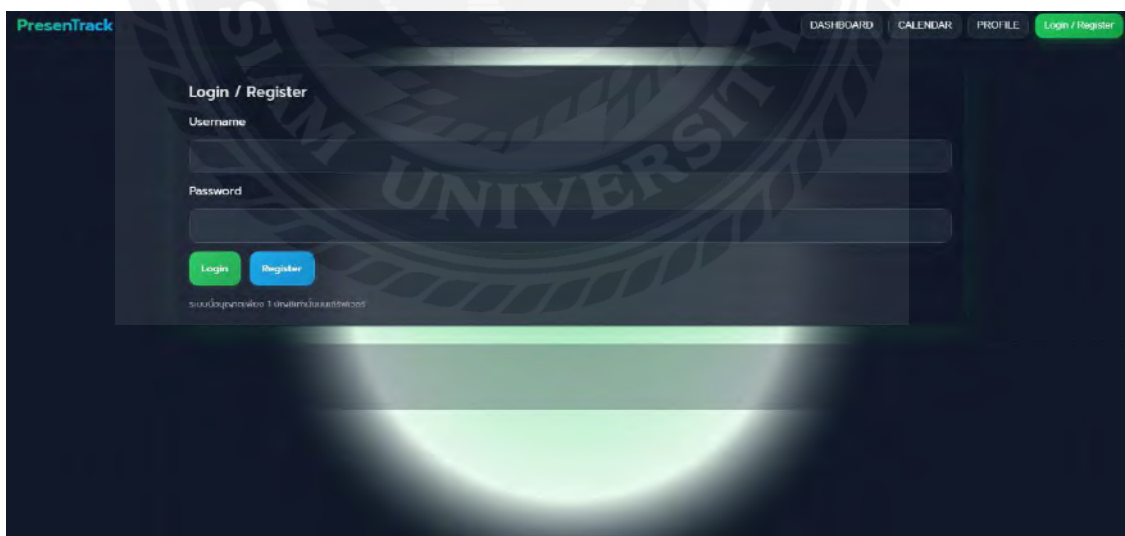
การสมัครและการทดลองใช้งานของเว็บแอปพลิเคชัน PresentTrack

1. เปิดเว็บเบราว์เซอร์เพื่อเข้าสู่เว็บไซต์ : <http://127.0.0.1:50172> เพื่อเริ่มต้นลงทะเบียนการใช้งาน



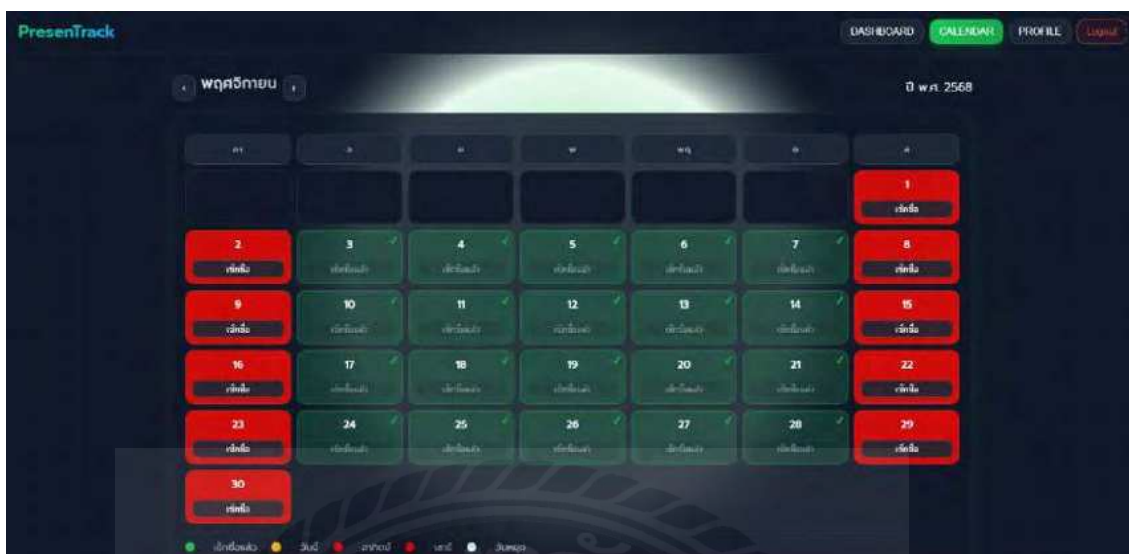
รูปที่ 2.13 หน้าเว็บเบราว์เซอร์

2. การสมัครสมาชิก (Register) : ในหน้าแรก ให้กรอก Username และ Password ที่ต้องการตั้งค่า คลิกที่ปุ่มสีฟ้า "Register"



รูปที่ 2.14 หน้า Login/Register

3. การเข้าสู่ระบบ (Login) : กรอก Username และ Password ที่สมัครไว้ คลิกที่ปุ่มสีเขียว "Login" เมื่อเข้าสู่ระบบสำเร็จ ระบบจะพาไปยังหน้า Dashboard โดยอัตโนมัติ



รูปที่ 2.15 หน้า calendar แสดงปฏิทินของเดือน

4. การเช็คชื่อ : คลิกที่เมนู “CALENDAR” ด้านบนขวา หน้าจอจะแสดงปฏิทินของเดือนปัจจุบัน

- จุดสีเหลือง / กรอบสีเหลือง: แสดงว่าเป็น “วันนี้” (Today)
- จุดสีเขียว: วันที่ “เช็คชื่อแล้ว” (Checked)
- สีแดง: วันหยุด (เสาร์-อาทิตย์)

5. การดูสรุปสถิติ (Dashboard) : คลิกที่เมนู "DASHBOARD"

การเลือกเดือนที่หัวข้อ "ข้อมูลย้อนหลัง" สามารถคลิกเลือกเดือนที่ต้องการดูข้อมูลได้จาก Dropdown ระบบจะแสดงกราฟ 2 รูปแบบ:

- กราฟวงกลม (สรุปรายเดือน): แสดงสัดส่วนระหว่าง "เช็คแล้ว" (สีฟ้า) กับ "ขาด/ยังไม่เช็ค" (สีแดง)
- กราฟแท่ง (สรุปรายสัปดาห์): แสดงจำนวนวันที่มาทำงานในแต่ละสัปดาห์



รูปที่ 2.16 หน้า dashboard แสดงกราฟวงกลมและกราฟแท่ง

6. การจัดการข้อมูลส่วนตัว (Profile) : คลิกที่เมนู "PROFILE" กรอกข้อมูลในแบบฟอร์มให้ครบถ้วน

- ชื่อ (First Name)
- นามสกุล (Last Name)
- วันที่เริ่มฝึกงาน (Start Date)
- วันที่สิ้นสุดการฝึกงาน (End Date)
- เบอร์โทรศัพท์ (Phone Number)

คลิกปุ่ม "บันทึกโปรไฟล์" (สีเขียว) เพื่อจัดเก็บข้อมูลลงระบบ

The screenshot shows the PresenTrack Profile page. At the top, there are navigation links: DASHBOARD, CALENDAR, PROFILE, and Login / Register. The main content area is titled 'โปรไฟล์' (Profile). It contains a form with the following fields: 'ชื่อ' (First Name) with a text input field, 'นามสกุล' (Last Name) with a text input field, 'วันที่เริ่มฝึกงาน' (Start Date) with a date picker showing 'mm/dd/yyyy', 'วันที่ฝึกงานวันสุดท้าย' (End Date) with a date picker showing 'mm/dd/yyyy', and 'เบอร์โทรศัพท์' (Phone Number) with a text input field. Below the form is a large green button labeled 'บันทึกโปรไฟล์' (Save Profile). At the bottom left, there is a small text link: 'คลิกเพื่อดูข้อมูลส่วนตัวของคุณ' (Click to view your personal information).

รูปที่ 2.17 หน้า Profile กรอกข้อมูลในแบบฟอร์ม

7. การออกจากระบบ (Logout) : คลิกที่ปุ่ม "Logout" (สีแดง) ที่มุมขวาบนของเมนู
ระบบจะพากลับมายังหน้าจอ Login อีกครั้ง



บทที่ 3

รายละเอียดและการปฏิบัติงาน

3.1 ชื่อและที่ตั้งของสถานประกอบการ

บริษัทรีเลิร์น โซลูชัน จำกัด

ที่อยู่ : 35 อาคารวรรณสรณ์ ชั้นที่ 12A ห้องเลขที่ 1303 ถนนพญาไท แขวงถนนพญาไท เขตราชเทวี กรุงเทพมหานคร 10400

โทรศัพท์ : 062 - 094 - 5538

อีเมล : Admin@relearn-solution.com



รูปที่ 3.1 แผนที่อาคารวรรณสรณ์

3.2 ลักษณะการประกอบการ

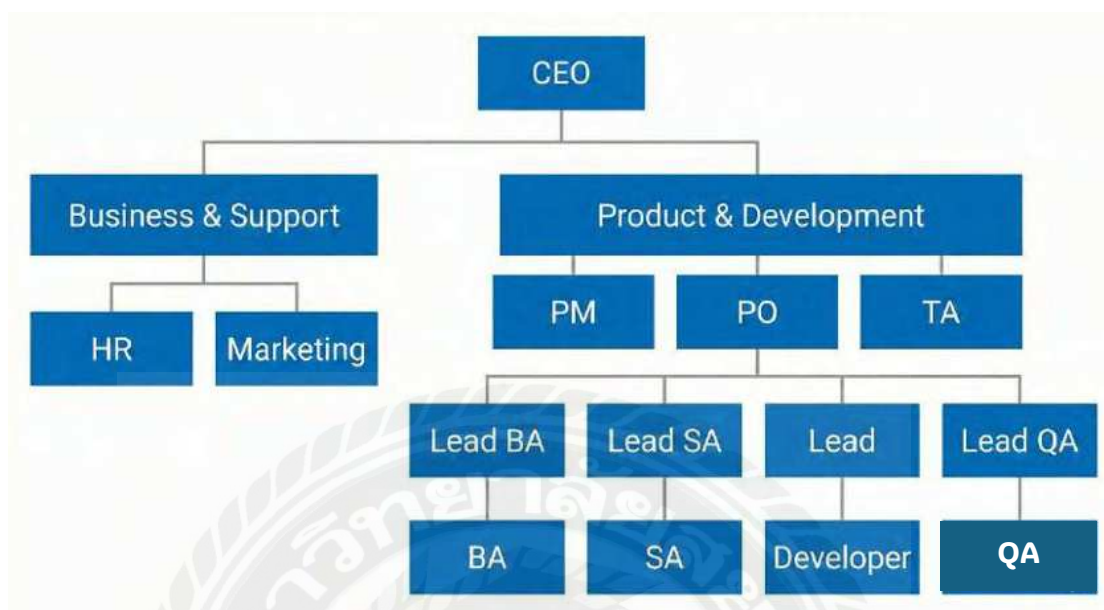
บริษัท รีเลิร์น โซลูชัน จำกัด เป็นบริษัทที่ประกอบธุรกิจด้านเทคโนโลยีสารสนเทศและการศึกษา โดยมุ่งเน้นการพัฒนาศักยภาพบุคลากรด้านดิจิทัลและการให้บริการทางด้านซอฟต์แวร์และคลาวด์ ซึ่งมีลักษณะการให้บริการหลักแบ่งออกเป็น 3 ด้าน ได้แก่

3.2.1 โครงการ Code Camp Thailand (บริการด้านการศึกษาและฝึกอบรม) บริษัทเป็นผู้ดำเนินการโครงการ Code Camp Thailand และแพลตฟอร์ม GetCode ซึ่งให้บริการจัดฝึกอบรมและสอนหลักสูตรการเขียนโปรแกรม (Coding) และทักษะด้านเทคโนโลยี ทั้งในรูปแบบออนไลน์ (Online) และออฟไลน์ (Offline) โดยมีกลุ่มเป้าหมายทั้งบุคคลทั่วไปที่ต้องการเปลี่ยนสายงานมาเป็นโปรแกรมเมอร์ และองค์กรที่ต้องการพัฒนาทักษะบุคลากร โดยเน้นการเรียนการสอนที่นำไปใช้งานได้จริงเพื่อผลิตนักพัฒนาระบบเข้าสู่ตลาดแรงงาน

3.2.2 บริการพัฒนาซอฟต์แวร์ (Software Development) ให้บริการจัดทำโปรแกรมคอมพิวเตอร์ เว็บไซต์ และระบบเครือข่ายตามความต้องการของลูกค้า (Software House) โดยรองรับกลุ่มเจ้าของธุรกิจที่ต้องการโซลูชันด้านโค้ดหรือซอฟต์แวร์เพื่อนำไปยกระดับและปรับปรุงประสิทธิภาพการทำงานขององค์กรให้มีความทันสมัย

3.2.3 บริการด้านเทคโนโลยีสารสนเทศและการลงทุน นอกเหนือจากการศึกษาและพัฒนาซอฟต์แวร์ บริษัทยังประกอบกิจกรรมการจัดทำโปรแกรมเว็บไซต์และเครือข่าย และมีการลงทุนในบริษัทอื่น รวมถึงการรับงานโครงการภาครัฐที่เกี่ยวข้องกับการพัฒนาเทคโนโลยีและการศึกษาด้านดิจิทัล

3.3 รูปแบบการจัดการองค์กรและการบริหารขององค์กร



รูปที่ 3.2 รูปแบบการจัดการองค์กร

3.4 ตำแหน่งและลักษณะงานที่ได้รับมอบหมาย

3.4.1 ตำแหน่ง Tester

3.4.2 ลักษณะงานที่ได้รับมอบหมาย

- ปฏิบัติการทดสอบการทำงานของระบบเว็บไซต์สำนักงานประกันสังคม (Social Security System) เพื่อตรวจสอบความถูกต้องและความเสถียรของระบบ
- จัดทำเอกสารกรณีทดสอบ (Test Cases) อย่างละเอียด พร้อมทั้งระบุข้อบกพร่อง (Defects) และวิเคราะห์ปัญหาที่เกี่ยวข้องกับประสบการณ์ผู้ใช้งาน (UX) และส่วนติดต่อผู้ใช้งาน (UI)
- ดำเนินการทดสอบทั้งในรูปแบบการทดสอบด้วยตนเอง (Manual Testing) และการทดสอบแบบอัตโนมัติ (Automated Testing) โดยประยุกต์ใช้เครื่องมือ Visual Studio Code (VS Code) ร่วมกับเฟรมเวิร์ก Playwright ในการเขียนชุดคำสั่งทดสอบ

3.5 ชื่อและตำแหน่งของพนักงานที่ปรึกษา

นายชาญชัย ศรีจันทร์ ตำแหน่ง Lead BA tester

3.6 ระยะเวลาในการปฏิบัติสหกิจศึกษา

ระยะเวลาในการปฏิบัติสหกิจศึกษา : วันที่ 26 พฤษภาคม พ.ศ. 2568 ถึง วันที่ 5 กันยายน พ.ศ. 2568

ขั้นตอนการดำเนินงาน	พ.ค. 2568	มิ.ย. 2568	ก.ค. 2568	ส.ค. 2568	ก.ย. 2568
1. ศึกษาระเบียบวิธีปฏิบัติและกระบวนการทำงานมาตรฐานของบริษัท	←→				
2. ศึกษาหลักการออกแบบกรณีทดสอบ (Test Case)		←→			
3. ปฏิบัติหน้าที่ตรวจสอบและทดสอบฟังก์ชันการทำงานของระบบตามขอบเขตงานที่ได้รับมอบหมาย		←→			
4. ศึกษาเครื่องมือสำหรับการทดสอบอัตโนมัติ ใช้งานเฟรมเวิร์ก Playwright ในการทำ Automate Testing			←→		
5. ศึกษาและวิเคราะห์กระบวนการ DevSecOps นำองค์ความรู้มาประยุกต์ใช้ในการพัฒนาโครงการที่ได้รับมอบหมาย			←→		

ตารางที่ 3.1 ระยะเวลาในการฝึกสหกิจศึกษา

3.7 ขั้นตอนและการดำเนินงาน

ขั้นตอนและการดำเนินงานแบ่งออกเป็น 2 ส่วน

3.7.1 ในการปฏิบัติงานสหกิจศึกษา ณ บริษัทรีลิร์น โซลูชั่น จำกัด ผู้จัดทำได้รับมอบหมายให้ปฏิบัติหน้าที่ในตำแหน่ง Software Tester (หรือ QA) โดยมีหน้าที่หลักในการตรวจสอบคุณภาพระบบเว็บไซต์สำนักงานประกันสังคม (Social Security System) เพื่อให้มั่นใจในความถูกต้องและความเสถียรของระบบ

ขั้นตอนการปฏิบัติงานเริ่มจากการวิเคราะห์ความต้องการของระบบเพื่อจัดทำเอกสารกรณีทดสอบ (Test Cases) อย่างละเอียด ครอบคลุมทั้งฟังก์ชันการทำงานหลักและกรณีการใช้งานรูปแบบต่างๆ พร้อมทั้งทำการระบุข้อบกพร่อง (Defects) และวิเคราะห์ปัญหาที่ส่งผลกระทบต่อประสบการณ์ผู้ใช้งาน (UX) และส่วนติดต่อผู้ใช้งาน (UI)

ด้านกระบวนการทดสอบ ได้ดำเนินการทั้งในรูปแบบการทดสอบด้วยตนเอง (Manual Testing) เพื่อตรวจสอบความถูกต้องเบื้องต้น และได้นำเทคโนโลยีเข้ามาช่วยเพิ่มประสิทธิภาพด้วยการทำ Automated Testing โดยประยุกต์ใช้เครื่องมือ Visual Studio Code (VS Code) ร่วมกับเฟรมเวิร์ก Playwright ในการพัฒนาชุดคำสั่งทดสอบอัตโนมัติ เพื่อให้การตรวจสอบระบบมีความรวดเร็วและแม่นยำยิ่งขึ้น

3.7.2 การพัฒนาเว็บแอปพลิเคชันปฏิทินเชิงซ้อนนักศึกษา (PresenTrack) เป็นระบบที่พัฒนาขึ้นเพื่อเพิ่มประสิทธิภาพในการบริหารจัดการเวลาการเข้างานของนักศึกษาฝึกงาน โดยมีวัตถุประสงค์เพื่อลดความผิดพลาดจากการบันทึกข้อมูลด้วยมือและอำนวยความสะดวกในการตรวจสอบข้อมูลย้อนหลัง ระบบถูกออกแบบภายใต้สถาปัตยกรรมแบบ Client-Server ซึ่งแยกส่วนการทำงานระหว่างส่วนหน้าบ้าน (Frontend) และส่วนหลังบ้าน (Backend) อย่างชัดเจน เพื่อให้ระบบมีความยืดหยุ่นและง่ายต่อการบำรุงรักษา

3.8 อุปกรณ์และเครื่องมือที่ใช้

3.8.1 ฮาร์ดแวร์

- Laptop 1 เครื่อง

3.8.2 ซอฟต์แวร์

- โปรแกรม Google Sheet
- โปรแกรม Visual Studio Code
- โปรแกรม Playwright
- โปรแกรม Jira
- โปรแกรม Taiga
- โปรแกรม NotebookLM

บทที่ 4

ผลการปฏิบัติงาน

เนื่องจากระบบซอฟต์แวร์สำหรับการจัดเก็บและการจัดการข้อมูลลูกค้าที่บริษัทพัฒนาขึ้นมีความเกี่ยวข้องกับข้อมูลบริษัท ซึ่งต้องดำเนินการภายใต้เกณฑ์การรักษาความปลอดภัยของข้อมูล ดังนั้นผลการปฏิบัติงานในบทนี้จะนำเสนอในรูปแบบเชิงสมมติหรือใช้ข้อมูลบางส่วนเท่านั้น

4.1 ขั้นตอนการทดสอบระบบมี 2 รูปแบบ

ขั้นตอนการทดสอบระบบแบบ Manual Testing

1. การศึกษาและวิเคราะห์ข้อกำหนดของระบบ (System Requirements Analysis) ศึกษาคู่มือการทดสอบระบบและทำความเข้าใจรายละเอียดความต้องการของระบบ (Requirements) เพื่อให้ทราบถึงขอบเขตและฟังก์ชันการทำงานที่ต้องทำการตรวจสอบ



Case No.	Name	Test Case	Input	Output	Owner	API	Status (Current Observation)	Test Result
1	TC-001	Test Case 1	Input 1	Output 1	Owner 1	API 1	Pass	Pass
2	TC-002	Test Case 2	Input 2	Output 2	Owner 2	API 2	Pass	Pass
3	TC-003	Test Case 3	Input 3	Output 3	Owner 3	API 3	Pass	Pass
4	TC-004	Test Case 4	Input 4	Output 4	Owner 4	API 4	Pass	Pass
5	TC-005	Test Case 5	Input 5	Output 5	Owner 5	API 5	Pass	Pass
6	TC-006	Test Case 6	Input 6	Output 6	Owner 6	API 6	Pass	Pass
7	TC-007	Test Case 7	Input 7	Output 7	Owner 7	API 7	Pass	Pass
8	TC-008	Test Case 8	Input 8	Output 8	Owner 8	API 8	Pass	Pass
9	TC-009	Test Case 9	Input 9	Output 9	Owner 9	API 9	Pass	Pass
10	TC-010	Test Case 10	Input 10	Output 10	Owner 10	API 10	Pass	Pass
11	TC-011	Test Case 11	Input 11	Output 11	Owner 11	API 11	Pass	Pass
12	TC-012	Test Case 12	Input 12	Output 12	Owner 12	API 12	Pass	Pass
13	TC-013	Test Case 13	Input 13	Output 13	Owner 13	API 13	Pass	Pass
14	TC-014	Test Case 14	Input 14	Output 14	Owner 14	API 14	Pass	Pass
15	TC-015	Test Case 15	Input 15	Output 15	Owner 15	API 15	Pass	Pass
16	TC-016	Test Case 16	Input 16	Output 16	Owner 16	API 16	Pass	Pass
17	TC-017	Test Case 17	Input 17	Output 17	Owner 17	API 17	Pass	Pass
18	TC-018	Test Case 18	Input 18	Output 18	Owner 18	API 18	Pass	Pass
19	TC-019	Test Case 19	Input 19	Output 19	Owner 19	API 19	Pass	Pass
20	TC-020	Test Case 20	Input 20	Output 20	Owner 20	API 20	Pass	Pass

รูปที่ 4.1 ภาพการทดสอบตาม Requirements

2. การออกแบบกรณีการใช้งาน (Use Case Design) จัดทำและระบุรายละเอียดของกรณีการใช้งาน (Use Case) โดยกำหนดขั้นตอนการทดสอบในแต่ละสถานการณ์อย่างเป็นลำดับขั้น เพื่อใช้เป็นแนวทางในการตรวจสอบระบบ

Test Scenario ID	Test Scenario Description	Test Case ID	Test Case Description	Test Steps	Pre Conditions	Test Data	Post Conditions	Expected Result	Test Result
3001_0001	Test Scenario 1	3001_0001	Test Case 1	Test Step 1	Pre Condition 1	Test Data 1	Post Condition 1	Expected Result 1	pass
3002_0001	Test Scenario 2	3002_0001	Test Case 2	Test Step 2	Pre Condition 2	Test Data 2	Post Condition 2	Expected Result 2	pass
3003_0001	Test Scenario 3	3003_0001	Test Case 3	Test Step 3	Pre Condition 3	Test Data 3	Post Condition 3	Expected Result 3	pass
3004_0001	Test Scenario 4	3004_0001	Test Case 4	Test Step 4	Pre Condition 4	Test Data 4	Post Condition 4	Expected Result 4	pass
3005_0001	Test Scenario 5	3005_0001	Test Case 5	Test Step 5	Pre Condition 5	Test Data 5	Post Condition 5	Expected Result 5	pass
3006_0001	Test Scenario 6	3006_0001	Test Case 6	Test Step 6	Pre Condition 6	Test Data 6	Post Condition 6	Expected Result 6	pass
3007_0001	Test Scenario 7	3007_0001	Test Case 7	Test Step 7	Pre Condition 7	Test Data 7	Post Condition 7	Expected Result 7	pass

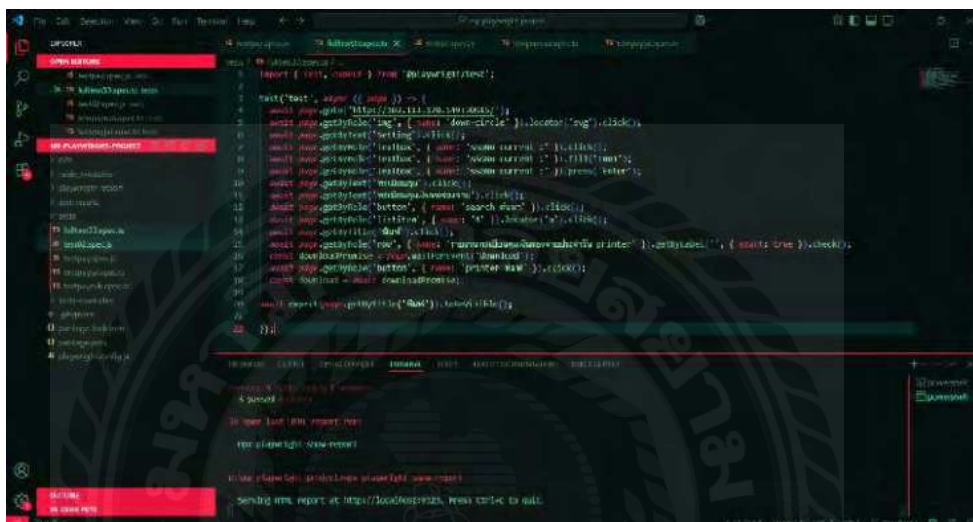
รูปที่ 4.2 ภาพการเขียน Use Case ตาม Requirements

3. การดำเนินการทดสอบระบบ (System Execution) ดำเนินการทดสอบระบบด้วยวิธีการป้อนข้อมูลนำเข้า (Input Data) ตามชุดข้อมูลที่กำหนดไว้ในเอกสารประกอบการทดสอบ เพื่อตรวจสอบความถูกต้องของการทำงานและการตอบสนองของระบบ

รูปที่ 4.3 ภาพการบันทึกข้อผิดพลาดของระบบ (Bug Reporting) ด้วยเครื่องมือ Jam

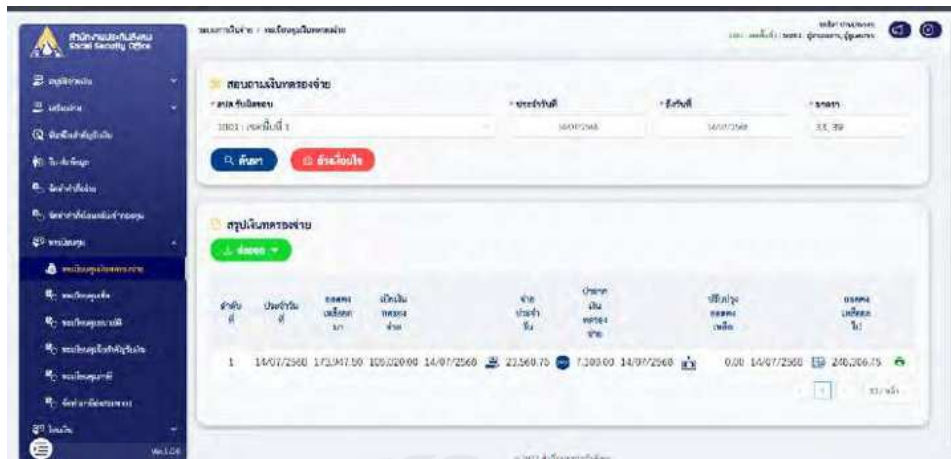
ขั้นตอนการดำเนินการทดสอบแบบอัตโนมัติ (Automated Testing Process)

1. การจัดทำชุดคำสั่งทดสอบ (Test Script Development) ดำเนินการเขียนและพัฒนาชุดคำสั่ง (Source Code) ผ่านโปรแกรม Visual Studio Code (VS Code) โดยอ้างอิงขั้นตอนจากกรณีการทดสอบ (Test Cases) ที่ออกแบบไว้ เพื่อกำหนดเงื่อนไขและลำดับการทำงานที่ต้องการตรวจสอบ

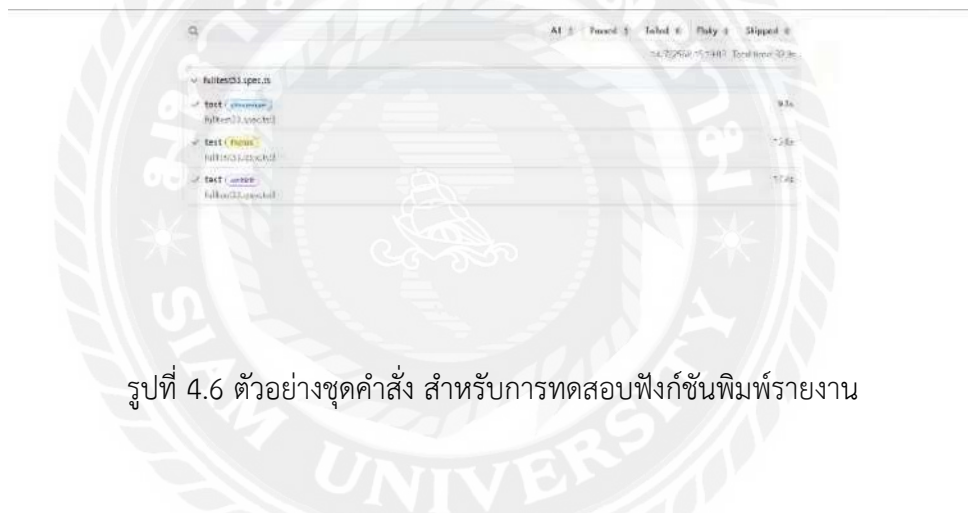


รูปที่ 4.4 ภาพ source code ผ่านโปรแกรม Visual Studio Code

2. การประมวลผลและตรวจสอบระบบ (Execution and Verification) ดำเนินการทดสอบโดยสั่งการให้ Playwright Framework ทำงานตามชุดคำสั่งที่จัดทำไว้ เพื่อจำลองพฤติกรรมการใช้งานจริงของผู้ใช้ (User Simulation) บนเว็บแอปพลิเคชัน พร้อมทั้งตรวจสอบความถูกต้องของผลลัพธ์และการตอบสนองของระบบ (System Assertion) แบบอัตโนมัติ



รูปที่ 4.5 หน้าจอแสดงผลการทำงานของสคริปต์ทดสอบอัตโนมัติ



รูปที่ 4.6 ตัวอย่างชุดคำสั่ง สำหรับการทดสอบฟังก์ชันพิมพ์รายงาน

```
import { test, expect } from '@playwright/test';

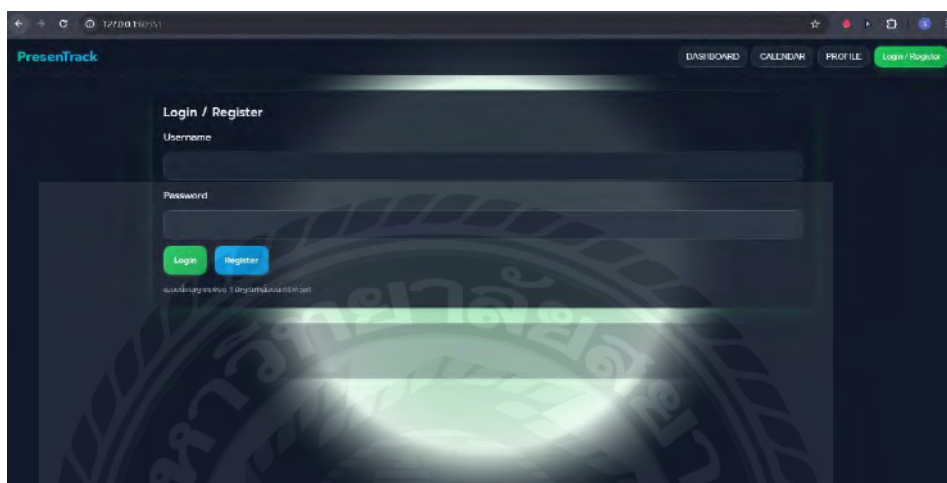
test('test', async ({ page }) => {
  await page.goto('http://103.113.178.149:30615/');
  await page.getByRole('img', { name: 'down-circle' }).locator('svg').click();
  await page.getByText('Setting').click();
  await page.getByRole('textbox', { name: 'ssoNo current : ' }).click();
  await page.getByRole('textbox', { name: 'ssoNo current : ' }).fill('1001');
  await page.getByRole('textbox', { name: 'ssoNo current : ' }).press('Enter');
  await page.getByText('ทะเบียนคนงาน').click();
  await page.getByText('ทะเบียนคนงานตรวจสอบราย').click();
  await page.getByRole('button', { name: 'search ค้นหา' }).click();
  await page.getByRole('listitem', { name: '4' }).locator('a').click();
  await page.getByText('พิมพ์').click();
  await page.getByRole('row', { name: 'รายงานทะเบียนคนงานรายวัน printer' }).getByLabel('a', { exact: true }).click();
  const downloadPromise = page.waitForEvent('download');
  await page.getByRole('button', { name: 'printer พิมพ์' }).click();
  const download = await downloadPromise;

  await expect(page.getByText('พิมพ์')).toBeVisible();
});
```

รูปที่ 4.7 ผลลัพธ์การรันทดสอบผ่าน Playwright (Test Passed)

4.2 ขั้นตอนการปฏิบัติตามโครงการ Devsecop

1. สร้างส่วน Frontend สำหรับการสร้างเว็บแอปพลิเคชัน โดยมีองค์ประกอบพื้นฐานในการพัฒนาระบบส่วนติดต่อผู้ใช้งาน ทำหน้าที่เป็นส่วนติดต่อกับผู้ใช้งาน เพื่อรับข้อมูลและแสดงผล โดยใช้เทคโนโลยีหลัก คือ HTML, CSS, Javascript



รูปที่ 4.8 หน้าระบบ Login/Register



รูปที่ 4.9 หน้าระบบบันทึกเวลาเข้าทำงาน

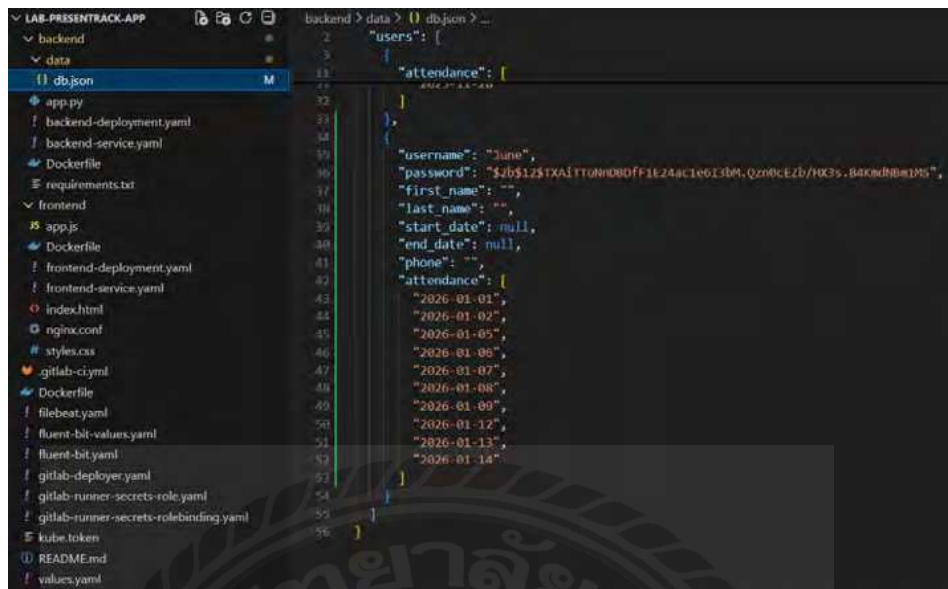


รูปที่ 4.10 หน้า Dashboard สรุปผลการบันทึกเวลาเข้าทำงาน

The screenshot shows the PresenTrack user profile page. At the top, there are tabs for 'DASHBOARD', 'CALENDAR', 'PROFILE', and 'Logout'. The main content area is titled 'โปรไฟล์' (Profile) and contains a list of personal information fields: 'ชื่อ' (Name), 'นามสกุล' (Surname), 'วันเกิด' (Date of Birth), 'วันเริ่มใช้งาน' (Start Date), 'วันเลิกใช้งาน' (End Date), and 'อีเมล' (Email). Below these fields is a progress bar showing '0.00/0.00' and a green bar with the text 'บันทึกเวลา' (Record Time). The background is dark blue with a subtle grid pattern.

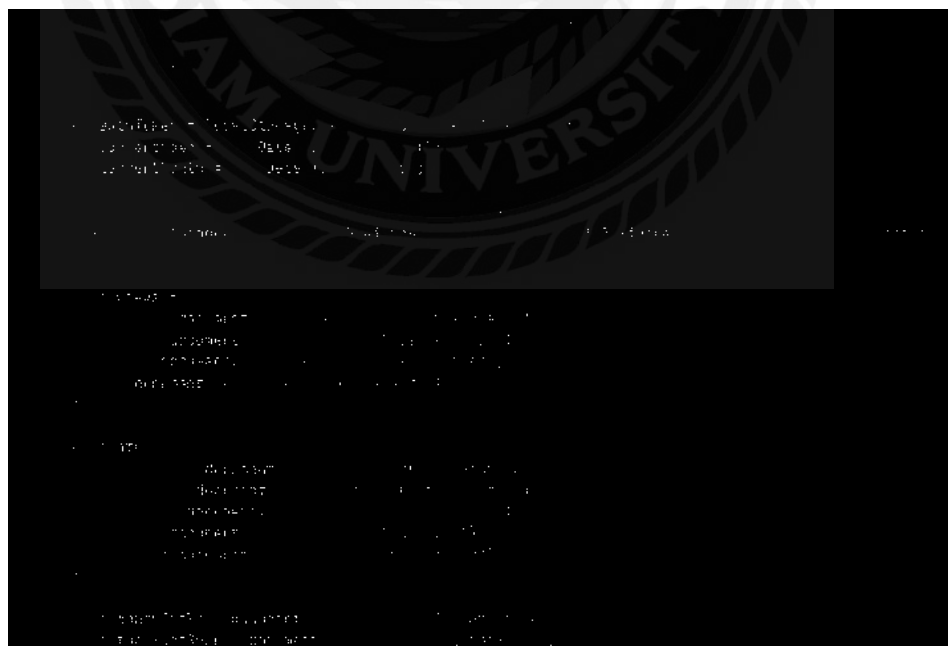
รูปที่ 4.11 หน้าบันทึกข้อมูล Profile ของผู้ใช้งาน

2. สร้างส่วน Backend สำหรับทำหน้าที่ประมวลผลข้อมูลและเชื่อมต่อกับฐานข้อมูล โครงสร้างนี้เลือกใช้ภาษา Python ร่วมกับเฟรมเวิร์ก Flask ซึ่งเป็น Micro-framework ที่มีความยืดหยุ่นสูง โดยมีการสื่อสารระหว่าง Frontend และ Backend ผ่านรูปแบบ REST API



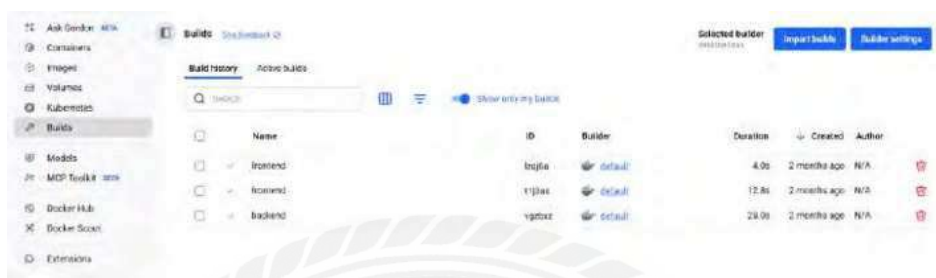
รูปที่ 4.12 ภาพ source code database การทำงานของส่วนหลังบ้าน

3. สร้างส่วนการเชื่อมต่อ Frontend และ Backend เพื่อเชื่อมต่อข้อมูลการจัดการ การทำงาน โต้ตอบของผู้ใช้ เช่น การกดปุ่มบันทึกเวลาเข้าทำงาน หรือการแสดงผลสถิติในรูปแบบกราฟ ใช้ ภาษา Javascript



รูปที่ 4.13 ภาพ source code การเชื่อมต่อ Frontend และ Backend

4. สร้างส่วนเทคโนโลยีคอนเทนเนอร์และการจัดการ ด้วยโปรแกรม Docker สำหรับการบรรจุแอปพลิเคชันและสิ่งจำเป็นจ้องใช้ทั้งหมดลงใน “Image” เพื่อให้สามารถนำไปรันที่เครื่องใดก็ได้ใช้เครื่องมือ kubernetes สำหรับการบริหารจัดการจำนวนมาก และโครงสร้างนี้ใช้ Minikube ในการจำลอง Cluster ขนาดเล็กเพื่อพัฒนาและทดสอบ

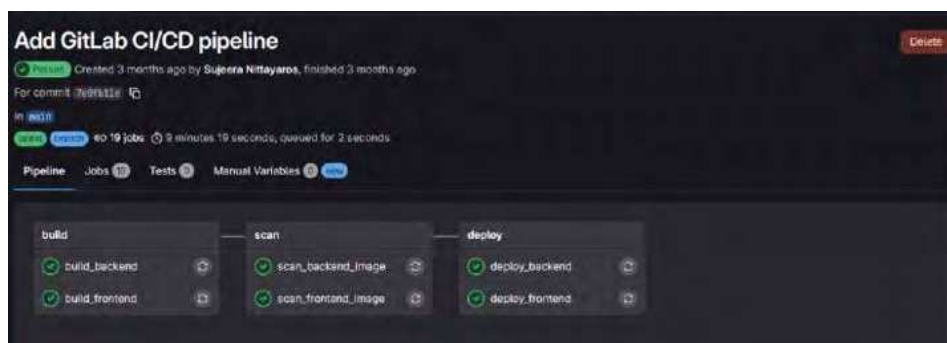


รูปที่ 4.14 ภาพระบบโปรแกรม Docker



รูปที่ 4.15 ภาพระบบโปรแกรม Docker ที่ใช้เครื่องมือ kubernetes

5. สร้างเครื่องมือ GitLab สำหรับบริหารจัดการเก็บรวบรวมโค้ดและสร้างระบบอัตโนมัติที่ประกอบด้วย การ Build Scan และ deploy ไปยัง Kubernetes พื้นที่ที่มีการอัปเดตโค้ด ช่วยลดข้อผิดพลาดจากการทำงาน



รูปที่ 4.16 ภาพเครื่องมือ GitLab ที่ทำการ Build Scan และ deploy

6. เครื่องมือ Trivy สำหรับสแกนหาช่องโหว่ใน Docker Image เพื่อตรวจสอบว่าไลบรารีมีจุดอ่อนด้านความปลอดภัยหรือไม่

Package	ID	Title	Severity	CVSS	Remediation
libssl	GHSA-2335-4333-4333	libssl: A vulnerability in the libssl library allows an attacker to bypass the certificate verification process and impersonate a trusted entity.	CRITICAL	9.8	Upgrade to libssl 1.1.1f or later.
openssl	GHSA-2335-4333-4333	openssl: A vulnerability in the openssl library allows an attacker to bypass the certificate verification process and impersonate a trusted entity.	CRITICAL	9.8	Upgrade to openssl 1.1.1f or later.

รูปที่ 4.17 ภาพสแกนหาช่องโหว่ด้วยเครื่องมือ Trivy

7. เครื่องมือ Kube-bench สำหรับการตรวจสอบช่องโหว่และการจัดการนโยบายความปลอดภัยของการตั้งค่า Kubernetes Cluster ว่าเป็นไปตามมาตรฐานหรือไม่

Kube-bench จะแสดงรายงานที่ละเอียดว่าการตั้งค่าของ Kubernetes Node และ Master ของคุณเป็นไปตามข้อกำหนดของ CIS Benchmark หรือไม่ โดยจะบอกว่าการทดสอบไหน PASSED (ผ่าน) FAILED (ไม่ผ่าน) หรือ INFO (ข้อมูลเพิ่มเติม)

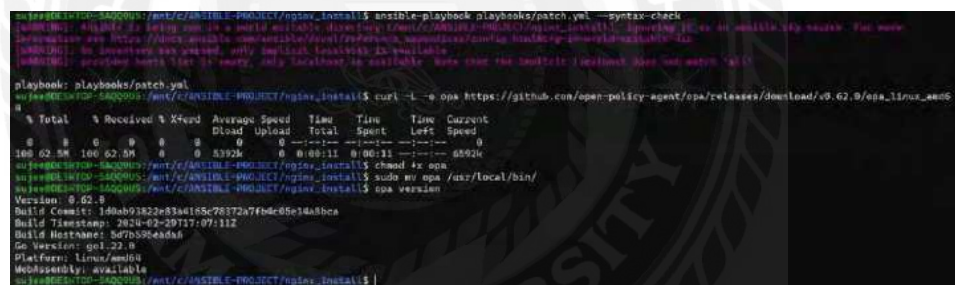
```
== Summary policies ==
0 checks PASS
0 checks FAIL
35 checks WARN
0 checks INFO

== Summary total ==
57 checks PASS
14 checks FAIL
62 checks WARN
0 checks INFO
```

รูปที่ 4.18 ภาพผลการสแกนทดสอบช่องโหว่

8. เครื่องมือ OPA (Open Policy Agent)

- **Security Automation:** คือการใช้ซอฟต์แวร์และเครื่องมือต่างๆ ในการจัดการงานด้านความปลอดภัยโดยอัตโนมัติ เช่น การสแกนช่องโหว่ การแก้ไขค่าคอนฟิกที่ไม่ปลอดภัย การจัดการ Patch การตอบสนองต่อเหตุการณ์ (Incident Response)
- **Benefits of Security Automation:**
 - **Speed:** ตอบสนองต่อภัยคุกคามได้เร็วกว่ามนุษย์
 - **Consistency:** ทำงานได้เหมือนกันทุกครั้ง ลด Human Error
 - **Scalability:** จัดการระบบจำนวนมากได้โดยไม่เพิ่มภาระงาน
 - **Efficiency:** ช่วยให้ทีม Security มีเวลาไปโฟกัสกับงานที่ซับซ้อนมากขึ้น
- **Integration with CI/CD:** Security Automation มักจะถูกรวมเข้ากับ CI/CD Pipeline เพื่อให้การตรวจสอบและแก้ไขความปลอดภัยเป็นส่วนหนึ่งของกระบวนการพัฒนาซอฟต์แวร์

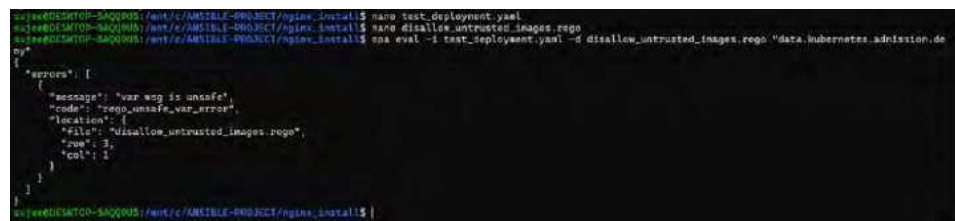


```

aws@DESKTOP-SAQQUUS:/mnt/c/ANSIBLE-PWD/ECT/engine_install$ ansible-playbook playbooks/patch.yml --syntax-check
[WARNING]: No inventory was parsed, only implicit localhost is available
[WARNING]: No inventory was parsed, only implicit localhost is available
[WARNING]: provided hosts list is empty, only localhost is available. Note that the implicit localhost does not match 'all'

playbook: playbooks/patch.yml
aws@DESKTOP-SAQQUUS:/mnt/c/ANSIBLE-PWD/ECT/engine_install$ curl -L -o opa https://github.com/open-policy-agent/opa/releases/download/v0.62.0/opa_linux_amd64
% Total % Received % Xferd Average Speed Time Time Time Current
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
100 62.5M 100 62.5M 0 0 8392k 0 0:00:11 0:00:11 0:00:11 8392k
aws@DESKTOP-SAQQUUS:/mnt/c/ANSIBLE-PWD/ECT/engine_install$ sha256sum opa
opa_linux_amd64 1d0eb91822e83a6185e78172a7f0dc85e3da8bca
aws@DESKTOP-SAQQUUS:/mnt/c/ANSIBLE-PWD/ECT/engine_install$ chmod +x opa
aws@DESKTOP-SAQQUUS:/mnt/c/ANSIBLE-PWD/ECT/engine_install$ sudo mv opa /usr/local/bin/
aws@DESKTOP-SAQQUUS:/mnt/c/ANSIBLE-PWD/ECT/engine_install$ opa version
Version: 0.62.0
Build Commit: 1d0eb91822e83a6185e78172a7f0dc85e3da8bca
Build Timestamp: 2024-02-29T17:07:11Z
Build Hostname: 3d7d39c6adab
Go Version: go1.22.0
Platform: linux/amd64
Modversion: available
aws@DESKTOP-SAQQUUS:/mnt/c/ANSIBLE-PWD/ECT/engine_install$
  
```

รูปที่ 4.19 ภาพการติดตั้งเครื่องมือ



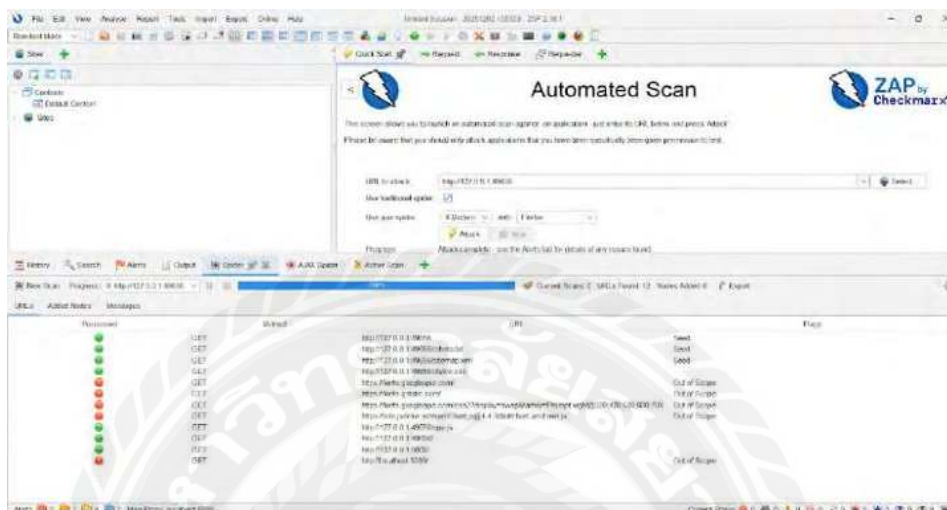
```

aws@DESKTOP-SAQQUUS:/mnt/c/ANSIBLE-PWD/ECT/engine_install$ nano test_deployment.yml
aws@DESKTOP-SAQQUUS:/mnt/c/ANSIBLE-PWD/ECT/engine_install$ nano disallow_untrusted_images.rego
aws@DESKTOP-SAQQUUS:/mnt/c/ANSIBLE-PWD/ECT/engine_install$ opa eval -i test_deployment.yml -d disallow_untrusted_images.rego "data.hubnetes.admission.de
py"
{
  "errors": [
    {
      "message": "var img is unsafe",
      "code": "rego_unsafe_var_error",
      "location": {
        "file": "disallow_untrusted_images.rego",
        "row": 3,
        "col": 1
      }
    }
  ]
}
aws@DESKTOP-SAQQUUS:/mnt/c/ANSIBLE-PWD/ECT/engine_install$
  
```

รูปที่ 4.20 ภาพการทดสอบการใช้เครื่องมือ OPA

9. โปรแกรม OWAPS ZAP

เป็นเครื่องมือสำหรับการตรวจสอบความปลอดภัยของเว็บแอปพลิเคชันแบบ Dynamic (DAST) เพื่อจำลองการโจมตีและค้นหาช่องโหว่ในขณะที่ระบบกำลังทำงาน



รูปที่ 4.21 ภาพเครื่องมือ OWAPS ZAP สแกนหาช่องโหว่ของเว็บแอปพลิเคชัน

เมื่อสแกนเสร็จ ให้อูที่แท็บ "Alerts" ด้านล่างซ้าย จะเห็นธงสีต่างๆ

-  Red (High): อันตรายมาก ต้องแก้ทันที
-  Orange (Medium): ควรแก้ไข
-  Yellow (Low): ความเสี่ยงต่ำ
-  Blue (Informational): ข้อมูลทั่วไป

บทที่ 5

สรุปผลและข้อเสนอแนะ

5.1 สรุปผลโครงการ

จากการนำเว็บแอปพลิเคชันที่พัฒนาไปทดสอบการใช้งาน พบว่ามีผลสรุป ข้อจำกัด และ ข้อเสนอแนะ ดังนี้

5.1.1 ผลสรุปการทำโครงการ

จากการทดสอบเว็บแอปพลิเคชัน PresentTrack พบว่าระบบช่วยให้การบริหารจัดการเวลาในการบันทึกเวลาทำงานของนักศึกษาฝึกสหกิจศึกษา มีความสะดวกและรวดเร็วยิ่งขึ้น โดยระบบมีฟังก์ชันรองรับการบันทึกข้อมูลย้อนหลัง พร้อมทั้งมีการแสดงผลสถิติในรูปแบบกราฟรายสัปดาห์และรายเดือน ซึ่งช่วยให้ผู้ใช้งานสามารถติดตามข้อมูลการเข้าเรียน/เข้าทำงานได้อย่างมีประสิทธิภาพ

5.1.2 ข้อจำกัดของโครงการ

เว็บแอปพลิเคชันสามารถทำงานได้เพียงขั้นพื้นฐาน ทำงานผ่านเครือข่ายอินเทอร์เน็ตเท่านั้น ไม่สามารถใช้งานในรูปแบบออฟไลน์ได้ และยังไม่ครอบคลุมถึงการเชื่อมต่อกับระบบฐานข้อมูลทะเบียนกลางของบริษัท ทำให้ไม่สามารถทำงานที่ซับซ้อนได้

5.1.3 ข้อเสนอแนะ

1. เพิ่มระบบวิเคราะห์ KPI: นำสถิติการเข้างานมาประมวลผลเพื่อช่วยประเมินผลนักศึกษา
2. เช็กชื่อ Real-time: บันทึกเวลาตามจริง และระบุสถานะ "มาสาย" อัตโนมัติทันทีเมื่อเกินเวลา

5.2 สรุปผลการปฏิบัติสหกิจศึกษา

จากการปฏิบัติสหกิจศึกษา บริษัทรีเลิร์น โซลูชัน จำกัด สามารถสรุปผลได้ต่อไปนี้

5.2.1 ผลที่ได้รับจากการปฏิบัติงานสหกิจศึกษา

ผู้จัดทำได้เรียนรู้รายละเอียดและหลักการทำงานในการพัฒนาเว็บแอปพลิเคชันของบริษัทอย่างเป็นระบบ ทำให้มีความเข้าใจในโครงสร้างและขั้นตอนการทำงานของซอฟต์แวร์ในระดับที่ใช้งานจริงอย่างชัดเจน อีกทั้งยังได้ฝึกทักษะการทดสอบระบบ (Software Testing) ในรูปแบบต่าง ๆ ซึ่งเป็นกระบวนการสำคัญในการตรวจสอบคุณภาพของงานก่อนส่งมอบ

นอกจากนี้ การได้มีโอกาสใช้งานระบบจริงยังช่วยให้ผู้จัดทำสามารถวิเคราะห์ปัญหาที่เกิดขึ้นจากการบันทึกและจัดเก็บข้อมูล จนนำไปสู่การเรียนรู้แนวทางการแก้ไขปัญหาและการจัดการฐานข้อมูลที่มีประสิทธิภาพ ตลอดจนได้เรียนรู้กระบวนการทำงานร่วมกับผู้อื่นในสภาพแวดล้อมการทำงานจริง ซึ่งถือเป็นประสบการณ์ที่มีค่าและช่วยเสริมสร้างทักษะการปรับตัวที่แตกต่างจากการเรียนรู้ในภาคทฤษฎีเพียงอย่างเดียว

5.2.2 ปัญหาหรืออุปสรรคที่พบในการปฏิบัติงานสหกิจศึกษา

เนื่องจากการปฏิบัติงานในองค์กรขนาดเล็กที่เน้นด้านการพัฒนาซอฟต์แวร์เป็นหลัก แม้จะส่งผลดีให้ผู้จัดทำได้เรียนรู้กระบวนการทำงานที่ครอบคลุมและหลากหลาย แต่ในขณะเดียวกันก็มีข้อจำกัดด้านบุคลากร โดยเฉพาะการมีที่ปรึกษาเพียงท่านเดียว ทำให้ในบางครั้งเกิดความกังวลในการสอบถามรายละเอียดเชิงลึกของงานที่ได้รับมอบหมาย ซึ่งส่งผลให้เกิดความคลาดเคลื่อนในการตีความเนื้อหาและข้อกำหนดต่าง ๆ จนนำไปสู่การใช้ระยะเวลาในการดำเนินงานที่นานขึ้นกว่าที่คาดการณ์ไว้ อย่างไรก็ตาม อุปสรรคดังกล่าวกระตุ้นให้ผู้จัดทำต้องเพิ่มความระมัดระวังในการทบทวนความเข้าใจ และฝึกฝนทักษะการสื่อสารเพื่อประสานงานให้เกิดความชัดเจนและมีประสิทธิภาพมากยิ่งขึ้นในอนาคต

5.2.3 ข้อเสนอแนะในการปฏิบัติงานสหกิจศึกษา

นักศึกษาควรมีการสื่อสารระหว่างเพื่อนร่วมงานเกี่ยวกับงานที่ได้รับมอบหมายให้มากขึ้น เพื่อที่จะสามารถทำความเข้าใจงานได้อย่างถูกต้อง และเพิ่มประสิทธิภาพการทำงาน

บรรณานุกรม

- Atlassian. (n.d.). Atlassian | Software development and collaboration tools.
<https://www.atlassian.com/>
- Blackboy96. (2024, September 18). ปูพื้นฐานและฝึกทำ Automated testing ผ่าน Playwright | Part01 [Video]. YouTube.
<http://www.youtube.com/watch?v=wBf4vx03Tw4>
- Creden Data. (n.d.). ข้อมูลบริษัท เก็ท โค้ด แคมป์ จำกัด.
<https://data.creden.co/company/general/0105563003735>
- Docker. (n.d.). Docker icons. Icons8. <https://icons8.com/icons/set/docker>
- Ebiarede, P. (2023). The frontend, backend, and database of a web-based application [Image]. ResearchGate.
<https://www.researchgate.net/profile/Perekebina-Ebiarede-2/publication/371131312/figure/fig1/AS:11431281162578042@1685370097903/The-frontend-backend-and-database-of-a-web-based-application.jpg>
- GetCodeCamp. (n.d.). เกี่ยวกับเรา. <https://getcodecamp.com/about-us/>
- GitHub. (n.d.). GitHub Documentation. <https://docs.github.com/en>
- GitLab. (n.d.). The DevSecOps Platform. <https://about.gitlab.com/>
- Google. (n.d.). NotebookLM. <https://notebooklm.google.com/>
- Microsoft. (n.d.). Playwright: Fast and reliable end-to-end testing for modern web apps. <https://playwright.dev/>
- PatternCraft. (n.d.). PatternCraft - Learning through play. <https://patterncraft.fun/>
- Prometheus. (2020, April 24). Prometheus and Grafana: The perfect combo. CNCF Blog.
<https://www.cncf.io/blog/2020/04/24/prometheus-and-grafana-the-perfect-combo/>



ภาคผนวก



ภาคผนวก ก หนังสือยินยอมให้เผยแพร่โครงงานสหกิจ

บริษัทรีเลชั่น โซลูชัน จำกัด

35 อาคารวรรณสรณ์ ชั้นที่ 12A ถนนพญาไท

แขวงพญาไท เขตราชเทวี กทม. 10400

วันที่ 28 มกราคม พ.ศ.2568

เรื่อง หนังสือยินยอมเผยแพร่รายงานการปฏิบัติงานโครงการสหกิจศึกษา

เรียน อธิการบดี มหาวิทยาลัยสยาม

ข้าพเจ้า นายชาญชัย ศรีจันทร์ พนักงานตำแหน่ง Lead BA Tester ของบริษัทรีเลชั่น โซลูชัน จำกัด

ได้ตรวจสอบข้อมูลทั้งหมดในรายงานการปฏิบัติโครงการสหกิจศึกษาและการศึกษาเชิงบูรณาการกับการทำงาน โครงการเรื่องบันทึกเวลาเข้าทำงานนักศึกษาฝึกสหกิจศึกษา สำหรับบริษัท รีเลชั่น โซลูชัน จำกัด ของนางสาวสุจิตา นิตยารส หลัสนักศึกษา 6504000011 สาขาวิศวกรรม คอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยสยาม

ยินยอมให้นักศึกษาและมหาวิทยาลัยสยาม เผยแพร่รายงานปฏิบัติงานโครงการสหกิจศึกษา ดังกล่าวต่อสาธารณะ เพื่อประโยชน์ต่อการศึกษาต่อไป

ลงชื่อ.....

(นายชาญชัย ศรีจันทร์)

รูปที่ ก.1 หนังสือยินยอมให้เผยแพร่โครงการสหกิจศึกษา



ภาคผนวก ข ภาพการนิเทศงานของอาจารย์



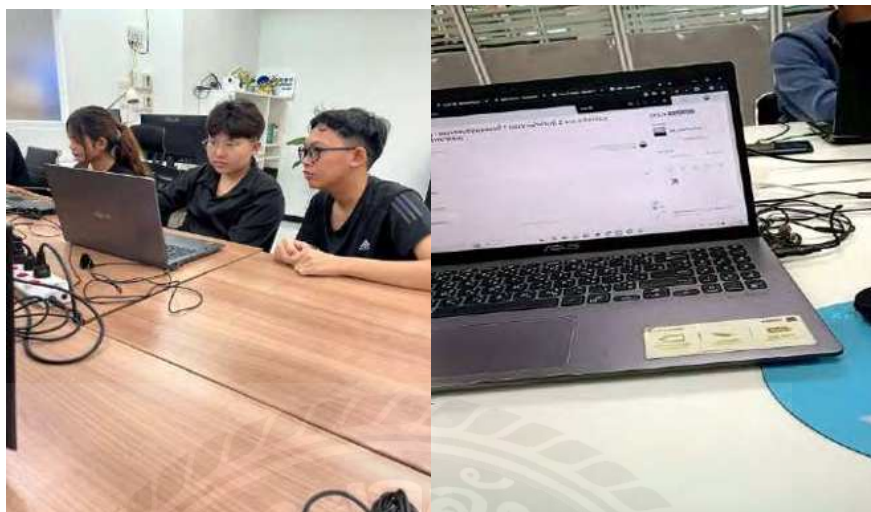
รูปที่ ข.1 อาจารย์นิเทศงานครั้งที่ 1 ณ วันที่ 24 มิถุนายน พ.ศ.2568



รูปที่ ข.2 อาจารย์นิเทศงานครั้งที่ 2 ณ วันที่ 8 สิงหาคม พ.ศ.2568



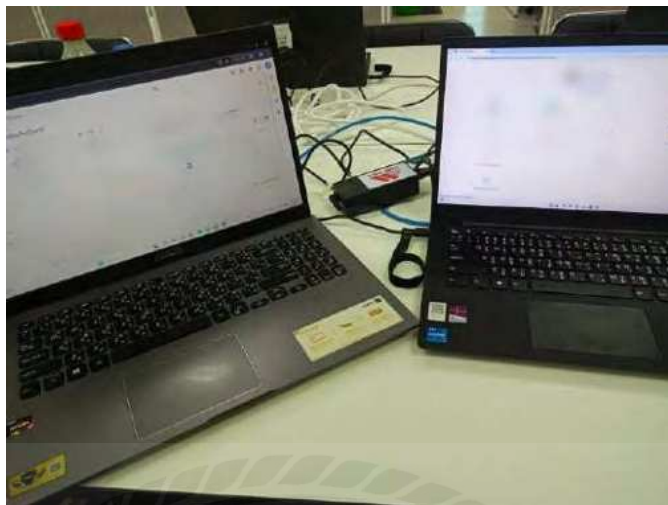
ภาคผนวก ค ภาพขณะปฏิบัติงาน



รูปที่ ค.1 ภาพการศึกษาการทำงานระบบประกันสังคมจากคู่มือมาตรฐานการปฏิบัติงาน



รูปที่ ค.2 บรรยากาศการสอนงานเบื้องต้นและการถ่ายทอดความรู้ภายในทีมพัฒนาซอฟต์แวร์ผ่าน
Discord



รูปที่ ค.3 ภาพระหว่างการทดสอบระบบเว็บไซต์ที่ได้รับมอบหมาย





ภาคผนวก ง ภาพการนำเสนอโครงการ



รูปที่ ง.1 การนำเสนอโครงการ ณ สถานประกอบการ



รูปที่ ง.2 การนำเสนอโครงการ ณ มหาวิทยาลัยสยาม



ภาคผนวก จ

การตรวจสอบการลอกเลียนวรรณกรรมทางวิชาการ

โดยใช้โปรแกรมอักขราวิสุทธิ์

Plagiarism Checking Report

Created on 2025-01-25 11:54:07 at 11:54 AM

[Print Report](#)[View Full Document](#)

Submission information

ID	SUBMISSION DATE	SUBMITTED BY	ORGANIZATION	FILENAME	STATUS	SIMILARITY INDEX
4606207	Jan 28, 2025 at 11:49 AM	สฤทธธนาภิวัฒน์	มหาวิทยาลัยสยาม	หนังสือ-ทุนธนาภิวัฒน์.pdf	Complete	View %

รูปที่ จ.1 ผลการตรวจสอบที่ได้จากโปรแกรมอักขราวิสุทธิ์



ประวัติผู้จัดทำ



ชื่อ – นามสกุล : นางสาวสุจีรา นิตยารส

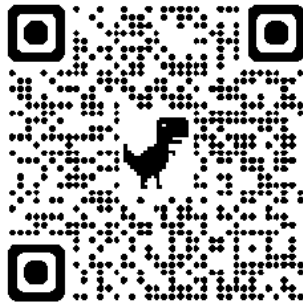
รหัสนักศึกษา : 6504000011

คณะ : วิศวกรรมศาสตร์

ภาควิชา : วิศวกรรมคอมพิวเตอร์

ที่อยู่ : 75/169 หอพักคู่ขวัญ ถนนกำนันมน ชอยกำนันมน 3
แขวงคลองบางพราน เขตบางบอน จังหวัดกรุงเทพมหานคร 10150

เบอร์โทรศัพท์ : 065-881-9155



https://drive.google.com/drive/folders/1dT3-Mx0QRttYViKDPJkgIfGfJLpvTR5?usp=drive_link

รายงานการปฏิบัติงานสหกิจศึกษา

โครงการบันทึกเวลาทำงานของนักศึกษาฝึกสหกิจศึกษา

สำหรับบริษัท รีเลิร์น โซลูชั่น จำกัด

Project PresentTrack (Intern Attendance Calendar)

For Relearn Solution Co., Ltd

โดย

นางสาวสุจีรา นิตยารส 6504000011

รายงานนี้เป็นส่วนหนึ่งของรายวิชา 155-393 สหกิจศึกษา

หลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยสยาม

ภาคการศึกษา 3 ปีการศึกษา 2567