

โครงการเลขที่ วศ.คพ. S002-2/2567

เรื่อง

เว็บไซต์ระบบสังคมสงเคราะห์

โดย

นายกิตปกรณ์ ทองโคตร	รหัส 650610749
นายนรภัทร จินดาสุน	รหัส 650610774
นายปณณวิชญ์ เศรษฐสิริวานิช	รหัส 650610784

โครงการนี้

เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเชียงใหม่

ปีการศึกษา 2567

PROJECT No. CPE S002-2/2567

Welfare Web System

Kitpakorn Thongkot 650610749

Norapat Chindasoon 650610774

Pannawit Setsiriwanit 650610784

**A Project Submitted in Partial Fulfillment of Requirements
for the Degree of Bachelor of Engineering
Department of Computer Engineering
Faculty of Engineering
Chiang Mai University
2024**

หัวข้อโครงการ : เว็บไซต์ระบบสังคมสงเคราะห์
: Welfare Web System
โดย : นายกิตติกรณ์ ทองโคตร รหัส 650610749
: นายนรภัทร จินดาสุน รหัส 650610774
: นายปณณวิชญ์ เศรษฐสิริวานิช รหัส 650610784
ภาควิชา : วิศวกรรมคอมพิวเตอร์
อาจารย์ที่ปรึกษา : ผศ. โดม โพธิ์กานนท์
ปริญญา : วิศวกรรมศาสตรบัณฑิต
สาขา : วิศวกรรมคอมพิวเตอร์
ปีการศึกษา : 2567

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเชียงใหม่ ได้อนุมัติให้โครงการนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต (สาขาวิศวกรรมคอมพิวเตอร์)

..... หัวหน้าภาควิชาวิศวกรรมคอมพิวเตอร์
(รศ.ดร. สันติ พิทักษ์กัญญ์)

คณะกรรมการสอบโครงการ

..... ประธานกรรมการ
(ผศ. โดม โพธิ์กานนท์)

..... กรรมการ
(ผศ.ดร. ลัชนา ระมิงค์วงศ์)

..... กรรมการ
(อ.ดร. นชิ ตันติธารานุกุล)

หัวข้อโครงการ : เว็บไซต์ระบบสังคมสงเคราะห์
: Welfare Web System
โดย : นายกิตติกรณ์ ทองโคตร รหัส 650610749
: นายณรภัทร จินดาสุน รหัส 650610774
: นายปณณวิชญ์ เศรษฐสิริวานิช รหัส 650610784
ภาควิชา : วิศวกรรมคอมพิวเตอร์
อาจารย์ที่ปรึกษา : ผศ. โดม โพธิ์กานนท์
ปริญญา : วิศวกรรมศาสตรบัณฑิต
สาขา : วิศวกรรมคอมพิวเตอร์
ปีการศึกษา : 2567

บทคัดย่อ

โครงการนี้มีวัตถุประสงค์เพื่อพัฒนา ระบบจัดเก็บข้อมูลสำหรับนักสังคมสงเคราะห์ ประจำโรงพยาบาลมหาราชนครเชียงใหม่ เนื่องจากปัจจุบันยังไม่มีระบบที่มีประสิทธิภาพในการจัดการข้อมูล ส่งผลให้กระบวนการทำงานอาจเกิดความล่าช้าและส่งผลต่อการให้บริการแก่ผู้รับสวัสดิการ

ระบบที่พัฒนาขึ้นเป็น เว็บแอปพลิเคชัน ที่ช่วยให้นักสังคมสงเคราะห์สามารถจัดเก็บและบริหารจัดการข้อมูลได้อย่างเป็นระบบ ทั้งยังสามารถนำข้อมูลที่บันทึกไว้มาวิเคราะห์และต่อยอดการทำงานได้อย่างสะดวกและมีประสิทธิภาพยิ่งขึ้น

เว็บแอปพลิเคชันดังกล่าวพัฒนาด้วย Next.js สำหรับฝั่ง Front-end และใช้ NestJS สำหรับ Back-end ในส่วนของฐานข้อมูล ใช้ PostgreSQL เพื่อจัดเก็บและบริหารจัดการข้อมูลอย่างเป็นระบบ นอกจากนี้ การทดสอบและการปรับใช้ (Deployment) ระบบจะดำเนินการผ่าน Docker เพื่อความยืดหยุ่นและประสิทธิภาพสูงสุดในการใช้งาน

Project Title : Welfare Web System
Name : Kitpakorn Thongkot 650610749
Norapat Chindasoon 650610774
Pannawit Setsiriwanit 650610784
Department : Computer Engineering
Project Advisor : Asst. Prof. Dome Potikanond
Degree : Bachelor of Engineering
Program : Computer Engineering
Academic Year : 2024

ABSTRACT

This project aims to develop a data management system for social workers at Maharaj Nakorn Chiang Mai Hospital. Currently, there is no efficient system for managing information, which may cause delays in work processes and affect the delivery of services to beneficiaries.

The developed system is a web application designed to help social workers systematically store and manage data. Additionally, it enables users to analyze and utilize stored information to enhance their work efficiency.

The web application is built using Next.js for the front-end and NestJS for the back-end. For database management, PostgreSQL is used to store and organize data effectively. Furthermore, the system's testing and deployment are carried out using Docker to ensure flexibility and optimal performance.

กิตติกรรมประกาศ

โครงการนี้จะไม่สำเร็จล่วงไปได้ หากไม่ได้รับความอนุเคราะห์และการสนับสนุนจากหลายฝ่าย ข้าพเจ้าขอแสดงความขอบคุณเป็นอย่างยิ่งต่อ ผศ.โตม โปธิกานนท์ อาจารย์ที่ปรึกษาโครงการ ที่สละเวลาอันมีค่าให้คำแนะนำ ให้แนวคิด และชี้แนะแหล่งความรู้ที่จำเป็นสำหรับการพัฒนาเว็บแอปพลิเคชัน ตลอดจนช่วยตรวจสอบและแก้ไขข้อบกพร่องต่าง ๆ ด้วยความเอาใจใส่มาโดยตลอด

นอกจากนี้ ข้าพเจ้าขอขอบพระคุณ ผศ.ดร.ลิขณา ระมิงค์วงศ์ และ อ.ดร.นชิ ตันติธารานุกุล ที่ให้คำปรึกษา และข้อเสนอแนะอันเป็นประโยชน์ จนทำให้โครงการนี้มีความสมบูรณ์ยิ่งขึ้น

ขอขอบคุณเพื่อนๆ และรุ่นพี่ทุกท่าน ที่คอยให้คำแนะนำ ให้กำลังใจ และเป็นแรงสนับสนุนตลอดมา ซึ่งเป็นพลังผลักดันสำคัญที่ทำให้ข้าพเจ้ามุ่งมั่นและตั้งใจทำงานจนสามารถพัฒนาโครงการนี้จนสำเร็จ

สุดท้ายนี้ ข้าพเจ้าขอขอบพระคุณทุกท่านที่ไม่ได้เอ่ยนาม ณ ที่นี้ ซึ่งได้ให้ความช่วยเหลือและสนับสนุนโครงการในรูปแบบต่างๆ ตลอดระยะเวลาการดำเนินงาน ขอบพระคุณเป็นอย่างสูง

นายกิตติกรณ์ ทองโคตร

นายณรภัทร จินดาสุน

นายปณณวิชญ์ เศรษฐสิริวานิช

5 มีนาคม 2568

สารบัญ

บทคัดย่อ	ข
Abstract	ค
กิตติกรรมประกาศ	ง
สารบัญ	จ
สารบัญรูป	ช
สารบัญตาราง	ซ
1 บทนำ	1
1.1 ที่มาของโครงการ	1
1.2 วัตถุประสงค์ของโครงการ	1
1.3 ขอบเขตของโครงการ	1
1.3.1 ขอบเขตด้านฮาร์ดแวร์	1
1.3.2 ขอบเขตด้านซอฟต์แวร์	1
1.4 ประโยชน์ที่ได้รับ	1
1.5 เทคโนโลยีและเครื่องมือที่ใช้	2
1.6 แผนการดำเนินงาน	2
1.7 บทบาทและความรับผิดชอบ	3
1.8 ผลกระทบด้านสังคม สุขภาพ ความปลอดภัย กฎหมาย และวัฒนธรรม	3
2 ทฤษฎีที่เกี่ยวข้อง	4
2.1 หน้าที่ของนักสังคมเคราะห์	4
2.1.1 ให้คำปรึกษาและแนะแนวทางช่วยเหลือ	4
2.1.2 ประสานงานและส่งต่อหน่วยงานที่เกี่ยวข้อง	4
2.1.3 ช่วยเหลือกลุ่มเปราะบางในสังคม	4
2.1.4 ติดตามผลและประเมินสถานการณ์	4
2.2 ด้านเทคโนโลยีที่ใช้พัฒนาเว็บแอปพลิเคชัน	5
2.2.1 Typescript	5
2.2.2 React	5
2.2.3 Next.js	6
2.2.4 Nest JS	6
2.2.5 PostgreSQL	7
2.2.6 MinIO	7
2.2.7 Docker	7
2.2.8 Tailwind CSS	8
2.2.9 Prisma	8
2.3 ความรู้ตามหลักสูตรซึ่งถูกนำมาใช้หรือบูรณาการในโครงการ	9
3 โครงสร้างและขั้นตอนการทำงาน	10
3.1 โครงสร้างของระบบ	10
3.1.1 Front-end	10
3.1.2 Back-end	10
3.1.3 Database และ Object Storage	11
3.1.4 Deploy	11
3.2 โครงสร้างฐานข้อมูล (Database Schema)	12
3.2.1 ข้อมูลผู้ป่วย	13

3.2.2	ข้อมูลเคส	13
3.2.3	ข้อมูลนักสังคมสงเคราะห์	13
3.2.4	ข้อมูลผู้กรอก	13
3.2.5	ข้อมูลที่ใช้ทำ Drop-down Box	13
4	การประเมินระบบ	14
4.1	การทดลองขึ้นหน้าเว็บแอปพลิเคชัน	14
4.2	ความช่วยเหลือในเบื้องต้น	15
	บรรณานุกรม	16

สารบัญรูป

2.1	Typescript	5
2.2	React	5
2.3	Next.js	6
2.4	Nest JS	6
2.5	PostgreSQL	7
2.6	MinIO	7
2.7	Docker	7
2.8	Tailwind CSS	8
2.9	Prisma	8
3.1	โครงสร้างของระบบ	10
3.2	โครงสร้างของฐานข้อมูล	12
4.1	หน้ากรอกข้อมูลส่วนตัวผู้ป่วย	14
4.2	หน้ากรอกที่อยู่ผู้ป่วย	15

สารบัญตาราง

บทที่ 1

บทนำ

1.1 ที่มาของโครงการ

ปัจจุบัน นักสังคมสงเคราะห์ที่ปฏิบัติงาน ณ โรงพยาบาลมหาสารคามเชียงใหม่ ยังคงใช้วิธีการจัดเก็บข้อมูลในรูปแบบกระดาษ ก่อนจะทำการบันทึกข้อมูลลงในคอมพิวเตอร์ของตนเองในรูปแบบไฟล์ .docx (Microsoft Word) อย่างไรก็ตาม คอมพิวเตอร์แต่ละเครื่องไม่สามารถเข้าถึงไฟล์ของกันและกันได้โดยตรง หากนักสังคมสงเคราะห์ต้องการเข้าถึงข้อมูลของผู้ใช้บริการที่ไม่ได้จัดเก็บอยู่ในคอมพิวเตอร์ของตนเอง พวกเขาจำเป็นต้องสอบถามเพื่อนร่วมงานเพื่อระบุว่ามีไฟล์ดังกล่าวถูกจัดเก็บอยู่ในเครื่องใด นอกจากนี้ สำหรับข้อมูลสถิติของผู้รับบริการ นักสังคมสงเคราะห์จะบันทึกข้อมูลลงในไฟล์ .xlsx (Microsoft Excel) ซึ่งสามารถแชร์ร่วมกันระหว่างคอมพิวเตอร์หลายเครื่องได้ อย่างไรก็ตาม การจัดเก็บข้อมูลดังกล่าวดำเนินการในไฟล์เดียว โดยมีจำนวนข้อมูลสะสมหลายพันบรรทัด ส่งผลให้การสรุปผลข้อมูลในระดับรายวันหรือรายเดือนทำได้ยาก และอาจส่งผลต่อประสิทธิภาพในการบริหารจัดการข้อมูลโดยรวม

จากปัญหาที่กล่าวมาข้างต้น คณะผู้จัดทำโครงการได้ตระหนักถึงข้อจำกัดในการจัดเก็บและบริหารจัดการข้อมูลของนักสังคมสงเคราะห์ จึงมีแนวคิดในการพัฒนา เว็บแอปพลิเคชัน เพื่อช่วยให้กระบวนการจัดการข้อมูลมีประสิทธิภาพมากยิ่งขึ้น ระบบดังกล่าวได้รับการออกแบบมาเพื่อแก้ไขปัญหาด้านการค้นหาข้อมูลให้สะดวกและรวดเร็วขึ้น รวมถึงปรับปรุงกระบวนการสรุปผลข้อมูลให้สามารถดำเนินการได้อย่างเป็นระบบและมีประสิทธิภาพยิ่งขึ้น

1.2 วัตถุประสงค์ของโครงการ

1. เพื่อพัฒนาเว็บแอปพลิเคชันที่สามารถจัดเก็บข้อมูล และเข้าถึงข้อมูลได้อย่างสะดวก รวดเร็ว และมีประสิทธิภาพ
2. เพื่อพัฒนาเว็บแอปพลิเคชันที่สามารถสรุปผลข้อมูลที่ใช้ต้องการได้ โดยไม่ต้องใช้โปรแกรมอื่นเพิ่มเติม

1.3 ขอบเขตของโครงการ

1.3.1 ขอบเขตด้านฮาร์ดแวร์

โครงการนี้เป็นการพัฒนาเว็บแอปพลิเคชันที่สามารถใช้งานได้บนคอมพิวเตอร์ที่ใช้ระบบปฏิบัติการ macOS หรือ Windows และบนโทรศัพท์มือถือที่ใช้ระบบปฏิบัติการ iOS หรือ Android

1.3.2 ขอบเขตด้านซอฟต์แวร์

1. เว็บแอปพลิเคชันที่สามารถใช้งานได้เฉพาะ CMU Account ที่กำหนดไว้เท่านั้น
2. เว็บแอปพลิเคชันที่สามารถเก็บข้อมูลไฟล์ต่างๆ ได้

1.4 ประโยชน์ที่ได้รับ

1. ผู้ใช้สามารถจัดเก็บข้อมูลได้อย่างมีประสิทธิภาพ

2. ผู้ใช้สามารถเข้าถึงข้อมูลได้อย่างสะดวก รวดเร็ว
3. ผู้ใช้สามารถนำข้อมูลมาสรุปผลตามเวลาที่เลือกได้

1.5 เทคโนโลยีและเครื่องมือที่ใช้

- ภาษาโปรแกรมมิ่ง: Typescript
- Front-end: Next.js, React, Tailwind CSS
- Back-end: NestJs, Prisma
- ฐานข้อมูล: PostgreSQL
- Object Storage: MinIO
- เทคโนโลยีและเครื่องมืออื่นๆ: Docker, Figma, Github

1.6 แผนการดำเนินงาน

ขั้นตอนการดำเนินงาน	ม.ค. 2568	ก.พ. 2568	มี.ค. 2568	เม.ย. 2568	พ.ค. 2568	มิ.ย. 2568	ก.ค. 2568	ส.ค. 2568	ก.ย. 2568	ต.ค. 2568	พ.ย. 2568	ธ.ค. 2568	ม.ค. 2569	ก.พ. 2569	มี.ค. 2569	เม.ย. 2569
Tools Research																
Requirements																
Design																
Set up Environments																
Sprint 1: ทำระบบจัดเก็บข้อมูล																
Sprint 2: ทำระบบ login																
Sprint 3: ทำระบบสรุปข้อมูล																
Report																

1.7 บทบาทและความรับผิดชอบ

1. นายกิตติกรณ์ ทองโคตร รับหน้าที่ในการทำ Components ของ Front-end, Initialize Project โดยใช้ Github, วางโครงสร้างของระบบ ทั้ง Front-end และ Back-end
2. นายนรภัทร จินดาสุน์ รับหน้าที่ในการนัดหมายเพื่อพูดคุยกับ Stakeholder, ออกแบบระบบโดยใช้ Figma, จัดวางระบบ Front-end, ติดตามผลการทำงานของเพื่อนในกลุ่ม
3. นายปณณวิชญ์ เศรษฐสิริวานิช รับหน้าที่ในการทำระบบฐานข้อมูลทั้งหมด, ทำระบบ Back-end

1.8 ผลกระทบด้านสังคม สุขภาพ ความปลอดภัย กฎหมาย และวัฒนธรรม

เมื่อนักสังคมสงเคราะห์สามารถทำงานได้สะดวก รวดเร็วขึ้น ทางผู้รับบริการก็สามารถได้รับการได้ดีขึ้นทั้งทางตรงและทางอ้อม เช่นการบริการที่เร็วขึ้น ทำให้รับผู้รับบริการได้มากขึ้น และการสรุปผลข้อมูลก็สามารถลดภาระของนักสังคมสงเคราะห์ที่ต้องไปสรุปข้อมูลเองด้วย

บทที่ 2

ทฤษฎีที่เกี่ยวข้อง

2.1 หน้าที่ของนักสังคมเคราะห์

นักสังคมสงเคราะห์มีหน้าที่สำคัญในการช่วยเหลือและสนับสนุนบุคคล ครอบครัว และชุมชนที่เผชิญปัญหาทางสังคม อารมณ์ หรือเศรษฐกิจ เพื่อให้สามารถปรับตัวและดำเนินชีวิตในสังคมได้อย่างเหมาะสม โดยมีบทบาทที่ครอบคลุมหลายด้าน ซึ่งแต่ละบทบาทล้วนมีความสำคัญต่อคุณภาพชีวิตของผู้ที่ต้องการความช่วยเหลือ

2.1.1 ให้คำปรึกษาและแนะแนวทางช่วยเหลือ

นักสังคมสงเคราะห์ให้คำปรึกษาโดยรับฟัง ประเมินปัญหา และให้แนวทางแก้ไข พร้อมแนะนำสิทธิและสวัสดิการที่เหมาะสม เพื่อให้ผู้ที่ต้องการความช่วยเหลือได้รับการดูแลอย่างเหมาะสม

2.1.2 ประสานงานและส่งต่อหน่วยงานที่เกี่ยวข้อง

นักสังคมสงเคราะห์ทำงานร่วมกับบุคลากรด้านสุขภาพและสวัสดิการเพื่อให้ผู้ป่วยได้รับการดูแลครบถ้วน พวกเขาประสานงานกับหน่วยงานต่าง ๆ เพื่อจัดหาความช่วยเหลือเพิ่มเติม เช่น ที่พักพิง การสนับสนุนทางกฎหมาย และการเข้าถึงบริการทางสังคม

2.1.3 ช่วยเหลือกลุ่มเปราะบางในสังคม

นักสังคมสงเคราะห์ช่วยเหลือกลุ่มเปราะบาง เช่น เด็กถูกทอดทิ้ง ผู้สูงอายุไร้ผู้ดูแล ผู้พิการ และคนไร้บ้าน โดยจัดหาทรัพยากรที่จำเป็น เช่น อาหาร ที่พัก และเงินสนับสนุน พร้อมให้ความรู้เพื่อพัฒนาคุณภาพชีวิตและส่งเสริมการพึ่งพาตนเอง

2.1.4 ติดตามผลและประเมินสถานการณ์

นักสังคมสงเคราะห์ติดตามผลการช่วยเหลืออย่างต่อเนื่อง เพื่อให้แน่ใจว่าตรงกับความต้องการของแต่ละบุคคล หากพบปัญหาเพิ่มเติม พวกเขาจะปรับแผนให้เหมาะสม พร้อมจัดทำรายงานเพื่อใช้พัฒนาระบบสวัสดิการในอนาคต

2.2 ด้านเทคโนโลยีที่ใช้พัฒนาเว็บแอปพลิเคชัน

2.2.1 Typescript

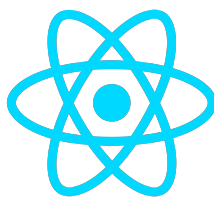
TypeScript เป็นภาษาโปรแกรมที่พัฒนาโดย Microsoft ซึ่งเพิ่มระบบตรวจสอบ Type ให้กับ JavaScript ทำให้สามารถควบคุมและจัดการ Type ในโปรแกรมได้อย่างมีประสิทธิภาพ ช่วยลดข้อผิดพลาดที่อาจเกิดขึ้นก่อนรันโปรแกรม ซึ่งเป็นข้อได้เปรียบสำคัญในการพัฒนาแอปพลิเคชันขนาดใหญ่ เนื่องจากใน JavaScript ข้อผิดพลาดเกี่ยวกับ Type มักจะปรากฏขึ้นในระหว่างการทำงานของโปรแกรม [8]



รูปที่ 2.1: Typescript

2.2.2 React

React เป็นไลบรารี JavaScript ที่พัฒนาโดย Facebook โดยถูกออกแบบมาเพื่อช่วยในการสร้าง User Interface (UI) ที่มีประสิทธิภาพและยืดหยุ่นสูง โครงสร้างหลักของ React คือแนวคิดเกี่ยวกับ Component-Based Architecture ซึ่งช่วยให้การพัฒนา UI เป็นไปอย่างเป็นระบบ แต่ละ Component สามารถนำกลับมาใช้ซ้ำได้ React ใช้แนวคิด Virtual DOM เพื่อเพิ่มประสิทธิภาพในการอัปเดต UI โดยการเปรียบเทียบความเปลี่ยนแปลงก่อนที่จะเรนเดอร์จริง ทำให้การทำงานรวดเร็วและลดภาระของเบราว์เซอร์ React ได้รับความนิยมอย่างแพร่หลายในวงการพัฒนาเว็บแอปพลิเคชัน เนื่องจากช่วยให้การจัดการ UI ซับซ้อนทำได้ง่ายขึ้น และสามารถรองรับการพัฒนาแบบไดนามิกได้เป็นอย่างดี [6]



รูปที่ 2.2: React

2.2.3 Next.js

Next.js เป็น React Web Framework ที่ออกแบบมาเพื่อให้การพัฒนาเว็บแอปพลิเคชันสะดวกขึ้น โดยมีการตั้งค่าและการคอนฟิกหลายๆ อย่างให้พร้อมใช้งาน เช่น การไม่ต้องตั้งค่าใดๆ เอง (Zero Config) และสามารถใช้งานได้ทันทีในสภาพแวดล้อมการผลิต (Ready for Production) นอกจากนี้ Next.js รองรับการทำ SEO ได้ดีกว่า React ปกติ เนื่องจากการเรนเดอร์แบบ Server Side Rendering (SSR) ซึ่งแตกต่างจาก React ที่ส่วนใหญ่ใช้ Client Side Rendering ในการทำงาน นอกจากนี้ยังรองรับฟีเจอร์เช่น Code Splitting, Static Site Generation (SSG), การสร้าง Dynamic Page, การสร้าง APIs และการปรับแต่งเซิร์ฟเวอร์ได้ รวมถึงมีเอกสารประกอบที่อ่านเข้าใจง่ายและมีตัวอย่างให้ศึกษา ทำให้ผู้ที่เคยใช้งาน React สามารถเริ่มใช้งาน Next.js ได้อย่างรวดเร็วและไม่ยาก [7]



รูปที่ 2.3: Next.js

2.2.4 Nest JS

Nest JS เป็น Framework สำหรับพัฒนาแอปพลิเคชันฝั่งเซิร์ฟเวอร์ที่สร้างขึ้นบน Node.js โดยออกแบบมาให้มีประสิทธิภาพและยืดหยุ่นสูง โครงสร้างของ Nest JS ถูกออกแบบให้ใช้ TypeScript เป็นหลัก (แม้ว่าจะรองรับ JavaScript ด้วย) และยึดแนวคิดเชิง Object-Oriented Programming (OOP) อย่างสมบูรณ์ ซึ่งช่วยให้การจัดระเบียบโค้ดเป็นไปอย่างเป็นระบบและมีโครงสร้างที่ชัดเจน นอกจากนี้ NestJS ยังสามารถทำงานร่วมกับไลบรารีอื่นๆ ได้อย่างง่ายดาย เช่น Express.js และ Socket.io ทำให้สามารถขยายขีดความสามารถของแอปพลิเคชันได้อย่างยืดหยุ่นและมีประสิทธิภาพ [4]



รูปที่ 2.4: Nest JS

2.2.5 PostgreSQL

PostgreSQL เป็นระบบจัดการฐานข้อมูลเชิงสัมพันธ์แบบโอเพ่นซอร์สระดับองค์กรที่ทรงพลัง รองรับทั้งการจัดเก็บข้อมูลแบบเชิงสัมพันธ์ผ่าน SQL และข้อมูลแบบไม่เชิงสัมพันธ์ผ่าน JSON โดดเด่นด้วยความน่าเชื่อถือ ความปลอดภัย และความแม่นยำในระดับสูง อีกทั้งยังได้รับการสนับสนุนจากชุมชนผู้พัฒนาที่แข็งแกร่ง ด้วยประสิทธิภาพและความยืดหยุ่นของ PostgreSQL ทำให้เป็นตัวเลือกหลักสำหรับเว็บแอปพลิเคชัน อุปกรณ์พกพา และโซลูชันด้านการวิเคราะห์ข้อมูลเชิงพื้นที่ที่ต้องการระบบฐานข้อมูลที่มีเสถียรภาพสูง [2]



รูปที่ 2.5: PostgreSQL

2.2.6 MinIO

MinIO เป็นระบบจัดเก็บข้อมูลแบบ Object Storage ที่รองรับการจัดเก็บข้อมูลประเภทต่างๆ เช่น ไฟล์ ไบนารี, ไฟล์เสียง, รูปภาพ, วิดีโอ และสเปรดชีต โดยผู้ใช้งานสามารถเข้าถึงและจัดการข้อมูลที่เก็บอยู่ใน MinIO ผ่าน REST API ทำให้สามารถเชื่อมต่อกับแอปพลิเคชันหรือระบบต่างๆ ได้อย่างสะดวกและมีประสิทธิภาพ [3]



รูปที่ 2.6: MinIO

2.2.7 Docker

Docker เป็นแพลตฟอร์มซอฟต์แวร์ที่ช่วยให้การสร้าง ทดสอบ และปรับใช้แอปพลิเคชันเป็นไปอย่างรวดเร็วและมีประสิทธิภาพ โดยใช้ คอนเทนเนอร์ (Containers) ซึ่งเป็นหน่วยบรรจุซอฟต์แวร์ที่รวมทุกองค์ประกอบที่จำเป็นสำหรับการทำงาน เช่น ไลบรารี เครื่องมือระบบ โค้ด และรันไทม์ ทำให้สามารถรันแอปพลิเคชันได้อย่างเสถียรในทุกสภาพแวดล้อม ด้วยความสามารถนี้ Docker ช่วยให้การปรับใช้และขยายขนาดแอปพลิเคชันเป็นไปอย่างราบรื่นและมั่นใจได้ว่าโค้ดจะทำงานได้อย่างสม่ำเสมอในทุกระบบ [1]



รูปที่ 2.7: Docker

2.2.8 Tailwind CSS

Tailwind CSS เป็นหนึ่งใน CSS Framework ที่ออกแบบมาในแนวทาง Utility-first ซึ่งหมายความว่าแต่ละคลาสใน Tailwind จะกำหนดคุณสมบัติของ CSS เพียงหนึ่งอย่างโดยเฉพาะ เช่น สีพื้นหลัง (background-color), ขอบ (border), ขนาดตัวอักษร (font-size) และอื่น ๆ แทนที่จะเป็นคลาสที่รวมชุดคำสั่งสำเร็จรูปไว้

Tailwind จึงเหมาะสำหรับนักพัฒนาที่ต้องการความยืดหยุ่นสูงในการออกแบบหน้าเว็บ เนื่องจากสามารถกำหนดสไตล์ขององค์ประกอบต่าง ๆ ได้อย่างแม่นยำโดยไม่ต้องพึ่งพาคลาสคอมโพเนนต์สำเร็จรูป ซึ่งมักมีข้อจำกัดในการปรับแต่ง นอกจากนี้ การใช้ Tailwind ยังช่วยให้การจัดการโค้ด CSS เป็นระบบมากขึ้น ลดความซับซ้อนของไฟล์สไตล์ชีต และส่งเสริมแนวทางการพัฒนาเว็บที่มีประสิทธิภาพ [9]

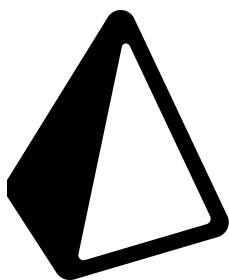


รูปที่ 2.8: Tailwind CSS

2.2.9 Prisma

Prisma ORM เป็นเครื่องมือสำหรับจัดการฐานข้อมูลในแอปพลิเคชัน TypeScript และ JavaScript ที่ช่วยให้การทำงานกับฐานข้อมูลมีประสิทธิภาพและใช้งานง่ายขึ้น

Prisma มีเครื่องมือสำหรับ จัดการโครงสร้างฐานข้อมูล (schema management), ทำ migrations และเรียกใช้งานข้อมูลผ่าน Prisma Client ซึ่งช่วยลดความซับซ้อนของ SQL Query รองรับฐานข้อมูลยอดนิยม เช่น PostgreSQL, MySQL, SQLite และ MongoDB ทำให้เป็นตัวเลือกที่เหมาะสมสำหรับนักพัฒนาที่ต้องการความยืดหยุ่นในการจัดการข้อมูล [5]



รูปที่ 2.9: Prisma

2.3 ความรู้ตามหลักสูตรซึ่งถูกนำมาใช้หรือบูรณาการในโครงการ

- **Fundamentals of Database Systems (261342)** ศึกษาหลักการออกแบบและการใช้งานฐานข้อมูล โดยใช้ MySQL ในการสร้าง เชื่อมต่อ และจัดการฐานข้อมูล รวมถึงการเขียนคำสั่ง SQL เพื่อจัดการข้อมูลภายในฐานข้อมูล
- **Software Engineering (261361)** ศึกษากระบวนการทางวิศวกรรมซอฟต์แวร์ ตั้งแต่การเก็บรวบรวมความต้องการของผู้ใช้ การกำหนดเป้าหมายของระบบ การวางแผนการทำงาน การประเมินและทดสอบระบบ ไปจนถึงการปรับปรุงและแก้ไขระบบให้มีประสิทธิภาพ
- **Basic CPE Lab (261207)** ฝึกปฏิบัติการพัฒนาเว็บแอปพลิเคชัน โดยประยุกต์ใช้ความรู้ด้าน HTML สำหรับการพัฒนา Front-End และ Python สำหรับการพัฒนา Back-End รวมถึงการออกแบบ UX/UI เพื่อให้ระบบใช้งานได้อย่างมีประสิทธิภาพ
- **Selected Topics in Computer Engineering (261494)** ศึกษาและประยุกต์ใช้เทคโนโลยี Docker และ Cloud ในการสร้างระบบจำลองสำหรับพัฒนาเว็บแอปพลิเคชัน
- **Object-Oriented Programming (261200)** ศึกษาหลักการเขียนโปรแกรมเชิงวัตถุ (OOP) และ การใช้ Model-View-Controller (MVC) รวมถึงการใช้ Web socket ในการออกแบบและพัฒนาเว็บแอปพลิเคชัน

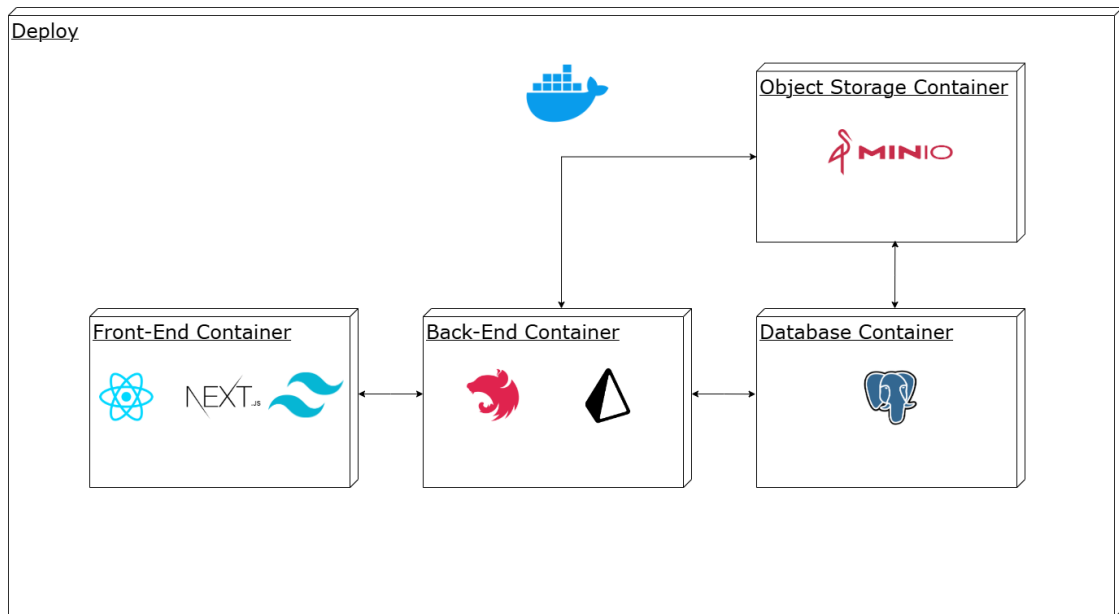
บทที่ 3

โครงสร้างและขั้นตอนการทำงาน

ในบทนี้จะกล่าวถึงหลักการ และการออกแบบระบบ

3.1 โครงสร้างของระบบ

ภาพรวมโครงสร้างของระบบ จะเป็นดังรูปนี้



รูปที่ 3.1: โครงสร้างของระบบ

3.1.1 Front-end

- เลือกใช้ Next.js และ React เพราะมีความสำเร็จรูปในตัวเอง มีเครื่องมือที่มีให้ใช้พร้อม และยังเป็นที่ใช้แพร่หลาย ทำให้สามารถค้นหาข้อมูลเกี่ยวกับ Framework ได้ง่าย
- เลือกใช้ Tailwind CSS เพราะเป็น component สำเร็จรูปที่สามารถ custom ได้ง่ายและเป็นที่ยอมรับ

3.1.2 Back-end

- เลือกใช้ Nest JS เพราะมีความสำเร็จรูปในตัวเอง มีเครื่องมือที่พร้อมใช้งานได้เลยโดยไม่ต้องสร้างเพิ่ม
- เลือกใช้ MinIO เพราะเหมาะสำหรับกรณีที่ต้องการตัวเลือกการจัดเก็บข้อมูลที่มีความเร็วสูงและมีต้นทุนต่ำ สามารถใช้งานในระบบคลาวด์หรือเซิร์ฟเวอร์ขององค์กรได้ โดยไม่ต้องพึ่งพาผู้ให้บริการคลาวด์

3.1.3 Database และ Object Storage

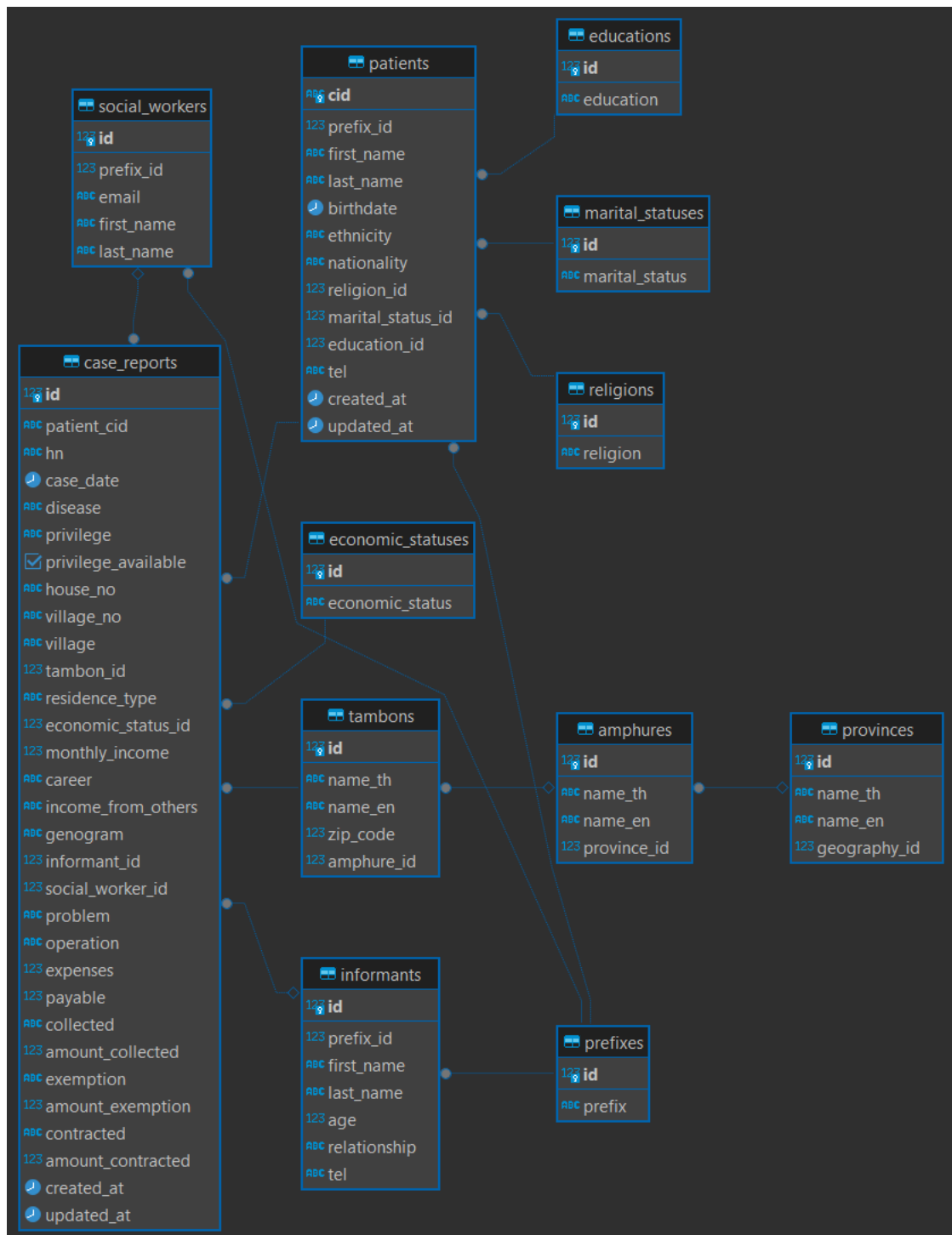
- เลือกใช้ PostgreSQL เพราะเหมาะกับข้อมูลขนาดใหญ่ ทำงานได้ดีในการอ่าน หรือ เขียน และ การวิเคราะห์ข้อมูลอย่างละเอียด
- เลือกใช้ Prisma เพราะมี Learning Curve ที่สูง ใช้งานง่าย และมีประสิทธิภาพดีเท่ากับตัวอื่น

3.1.4 Deploy

- เลือกใช้ Docker เพราะเหมาะสำหรับการรันแอปพลิเคชันที่ต้องการความยืดหยุ่นในการปรับใช้และทรัพยากรที่มีข้อจำกัด

3.2 โครงสร้างฐานข้อมูล (Database Schema)

ภาพรวมโครงสร้างของฐานข้อมูล จะเป็นดังรูปนี้



รูปที่ 3.2: โครงสร้างของฐานข้อมูล

3.2.1 ข้อมูลผู้ป่วย

โดยผู้ป่วย 1 คนจะมีข้อมูลได้แค่ 1 ข้อมูล

3.2.2 ข้อมูลเคส

โดยผู้ป่วย 1 คนสามารถมีได้หลายเคส (one to many)

3.2.3 ข้อมูลนักสังคมสงเคราะห์

โดยนักสังคมสงเคราะห์ 1 คนจะมีข้อมูลได้แค่ 1 ข้อมูล

3.2.4 ข้อมูลผู้กรอก

โดยเคส 1 เคสจะมีข้อมูลผู้กรอกได้แค่ 1 คนโดยอาจจะเป็นผู้ป่วยกรอกเองหรือญาติกรอกก็ได้

3.2.5 ข้อมูลที่ใช้ทำ Drop-down Box

ข้อมูลบางชนิดจะถูกเก็บไว้ในฐานข้อมูล แล้วจะถูกเรียกใช้ใน Drop-down Box เพื่อลดข้อผิดพลาดจากการพิมพ์ของนักสังคมสงเคราะห์

- educations
- marital_statuses
- economic_statuses
- religions
- prefixes
- provinces
- amphures
- tambons

บทที่ 4

การประเมินระบบ

เนื่องจากนักสังคมสงเคราะห์ที่ปฏิบัติงานในโรงพยาบาลมหาราชนครเชียงใหม่ยังไม่มีระบบที่สามารถรองรับการทำงานได้อย่างมีประสิทธิภาพเพียงพอ หากต้องรอให้ระบบที่พัฒนาขึ้นเสร็จสมบูรณ์ อาจใช้ระยะเวลาหลายเดือน

เพื่อแก้ไขปัญหาในระยะเริ่มต้น ทางคณะผู้จัดทำโครงการได้วางแผนทางในการให้ความช่วยเหลือเบื้องต้นแก่เจ้าหน้าที่นักสังคมสงเคราะห์ ควบคู่ไปกับการสร้างต้นแบบของแบบฟอร์มบันทึกข้อมูลเบื้องต้น เพื่อศึกษาประเด็นปัญหาที่อาจเกิดขึ้นในการพัฒนาและนำไปสู่การปรับปรุงแนวทางการทำงานใน Sprint 1 อย่างมีประสิทธิภาพ

4.1 การทดลองขึ้นหน้าเว็บแอปพลิเคชัน

โดยจะเป็นการทดลองขึ้นหน้าแบบฟอร์มกรอกประวัติผู้ป่วย ในหน้าแรกจะเป็นการกรอกข้อมูลส่วนตัวของผู้ป่วย โดยจะมีข้อมูลบางชนิดที่ต้องทำเป็น Drop-down box เพื่อลดความผิดพลาดจากการพิมพ์ของนักสังคมสงเคราะห์ และต้องกรอกให้สมบูรณ์จึงจะสามารถกรอกหน้าต่อไปได้

ส่งแบบฟอร์ม

ข้อมูลส่วนตัวผู้ป่วย

ข้อมูลที่อยู่ผู้ป่วย

รายละเอียดเคส

การประเมินปัญหา

ส่วนอายุ * *	ชื่อ *	นามสกุล *	เลขที่บัตรประชาชน *
วันเดือนปีเกิด *	สัญชาติ *	เชื้อชาติ *	ศาสนา *
การศึกษา *	หมายเลขโทรศัพท์ *	HN *	สถานภาพเศรษฐกิจ *
รายได้นี้อยู่คนเดียว *			

[ต่อไป >](#)

รูปที่ 4.1: หน้ากรอกข้อมูลส่วนตัวผู้ป่วย

หน้าต่อมาคือหน้าที่ใช้กรอกที่อยู่ของผู้ป่วย ซึ่งจะสามารถกรอกได้ 2 รูปแบบ คือ

1. กรอกจังหวัด อำเภอ และตำบลจากนั้นรหัสไปรษณีย์จะ Auto-complete ให้เอง
2. กรอกแค่หมายเลขไปรษณีย์แล้ว จังหวัด อำเภอ และตำบลจะ Auto-complete ให้เอง

ส่งแบบฟอร์ม

ข้อมูลส่วนตัวผู้ป่วย

ข้อมูลโรคของผู้ป่วย

รายละเอียดเคส

การประเมินปัญหา

จังหวัด *

รหัสไปรษณีย์ *

< กลับ

ต่อไป >

รูปที่ 4.2: หน้ากรอกที่อยู่ผู้ป่วย

4.2 ความช่วยเหลือในเบื้องต้น

ทางคณะผู้จัดทำโครงการ ได้ทดลองใช้ Microsoft Form ร่วมกับ Microsoft Onedrive และ Microsoft Power Automate เพื่อให้สามารถเก็บข้อมูลและเข้าถึงข้อมูลผู้ป่วย โดยที่คอมพิวเตอร์ทุกเครื่องที่กำหนดสามารถที่จะเข้าถึงไฟล์ได้โดยตรง โดยเมื่อนักสังคมสงเคราะห์กรอกข้อมูลผู้ป่วยลงใน Microsoft Form ทาง Microsoft Power Automate จะทำการสร้างโฟลเดอร์ของผู้ป่วยแต่ละคนใน Onedrive หากผู้ป่วยคนนั้นยังไม่มีโฟลเดอร์ของตนเอง และนำข้อมูลที่กรอกไว้ ใส่ลงในโฟลเดอร์นั้น

แต่เนื่องจาก Onedrive มีพื้นที่ที่ใช้ได้อย่างจำกัด (15 GB) ทำให้ไม่สามารถจัดเก็บข้อมูลผู้ป่วยได้อย่างเพียงพอ จึงไม่นำวิธีการนี้ไปใช้จริง

บรรณานุกรม

- [1] Amazon. Docker คืออะไร. <https://aws.amazon.com/th/docker/>.
- [2] AppMaster. Postgresql คืออะไร. <https://appmaster.io/th/blog/postgresql-khuue-aair>, 2022.
- [3] Gram. บันทึกการทดสอบและการใช้งาน object storage (minio) แบบ active/active. <https://gramkittisak.medium.com/บันทึกการทดสอบการใช้งาน-object-storage-minio-แบบ-active-active-4fddb461fc91>, 2023.
- [4] Kritsada L. ทำความรู้จัก nest js และวิธีการใช้! <https://kritsadapk.medium.com/ทำความรู้จัก-nest-js-และวิธีการใช้-6a498bef8b01>, 2020.
- [5] mikelopster. รู้จักกับ prisma orm. <https://mikelopster.dev/posts/next-prisma/>, 2024.
- [6] Papimpat Nimprasert. ทำความรู้จักกับ react และการใช้งานเบื้องต้น. <https://www.borntodev.com/2024/05/13/ทำความรู้จักกับ-react/>, 2024.
- [7] Chai Phonbopit. Next.js คืออะไร? มาเริ่มเขียนเว็บด้วย next.js กันดีกว่า. <https://www.devahoy.com/blog/2020/03/getting-started-with-nextjs/>, 2020.
- [8] Nattapon Rakthong. แนะนำภาษา typescript. <https://marcuscode.com/lang/typescript/introduction>, 2021.
- [9] Kittikorn Voraprateep. เปรียบเทียบการใช้งาน css ธรรมดา กับ tailwind css. <https://www.borntodev.com/2023/11/28/เปรียบเทียบการใช้งาน-css/>, 2023.