โครงงานเลขที่ วศ.คพ. **261492/3111**

เรื่อง

eDENT-ELDER: แพลตฟอร์มดิจิทัลสำหรับการค้นหาและวิเคราะห์รอยโรคก่อนมะเร็งและมะเร็งช่องปาก

โดย

Kawin Choosin	รหัส **650610745**
Natthapon Chanawerot	รหัส **650612082**
Thanawat Jaisert	รหัส **650610768**
Pariwat Wonginchan	รหัส **650612089**

โครงงานนี้
เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเชียงใหม่
ปีการศึกษา **3111**

PROJECT No. CPE 261492/3111

eDENT-ELDER: A Digital Platform for Detecting and Analyzing Potentially Malignant Disorders and Oral Cancer

Kawin Choosin	**650610745**
Natthapon Chanawerot	**650612082**
Thanawat Jaisert	**650610768**
Pariwat Wonginchan	**650612089**

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

2568

หัวข้อโครงงาน	: eDENT-ELDER: แพลตฟอร์มดิจิทัลสำหรับการค้นหาและวิเคราะห์รอยโรคก่อนมะเร็งและมะเร็งช่องปาก	
	: eDENT-ELDER: A Digital Platform for Detecting and Analyzing Potentially Malignant Disorders and Oral Cancer	
โดย	: Kawin Choosin	รหัส 650610745
	Natthapon Chanawerot	รหัส 650612082
	Thanawat Jaisert	รหัส 650610768
	Pariwat Wonginchan	รหัส 650612089
ภาควิชา	: วิศวกรรมคอมพิวเตอร์	
อาจารย์ที่ปรึกษา	: อ.ดร. ชินวัตร อิศราดิสัยกุล	
ปริญญา	: วิศวกรรมศาสตรบัณฑิต	
สาขา	: วิศวกรรมคอมพิวเตอร์	
ปีการศึกษา	: 3111	

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเชียงใหม่ ได้อนุมัติให้โครงงานนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต (สาขาวิศวกรรมคอมพิวเตอร์)

..... หัวหน้าภาควิชาวิศวกรรมคอมพิวเตอร์

(ผศ.ดร. นวดนย์ คุณเลิศกิจ)

คณะกรรมการสอบโครงงาน

..... ประธานกรรมการ
(อ.ดร. ชินวัตร อิศราดิสัยกุล)

..... กรรมการ
(ผศ.ดร. ภาสกร แช่มประเสริฐ)

..... กรรมการ
(รศ.ดร. ปฏิเวธ วุฒิสารวัฒนา)

..... กรรมการ
(ผศ.ดร. โดม โพธิ์กาญจนดล)

..... กรรมการ
(ผศ.ดร. นวดล คูณเลิศกิจ)

หัวข้อโครงงาน	: eDENT-ELDER: แพลตฟอร์มดิจิทัลสำหรับการค้นหาและวิเคราะห์รอยโรคก่อนมะเร็งและมะเร็งช่องปาก	
	: eDENT-ELDER: A Digital Platform for Detecting and Analyzing Potentially Malignant Disorders and Oral Cancer	
โดย	: Kawin Choosin	รหัส 650610745
	Natthapon Chanawerot	รหัส 650612082
	Thanawat Jaisert	รหัส 650610768
	Pariwat Wonginchan	รหัส 650612089
ภาควิชา	: วิศวกรรมคอมพิวเตอร์	
อาจารย์ที่ปรึกษา	: อ.ดร. ชินวัตร อิศราดิสัยกุล	
ปริญญา	: วิศวกรรมศาสตรบัณฑิต	
สาขา	: วิศวกรรมคอมพิวเตอร์	
ปีการศึกษา	: 3111	

บทคัดย่อ

เขียนบทคัดย่อของโครงงานที่นี่

การเขียนรายงานเป็นส่วนหนึ่งของการทำโครงงานวิศวกรรมคอมพิวเตอร์ เพื่อทบทวนทฤษฎีที่เกี่ยวข้อง อธิบายขั้นตอนวิธีแก้ปัญหาเชิงวิศวกรรม และวิเคราะห์และสรุปผลการทดลองอุปกรณ์และระบบต่างๆ อย่างไรก็ดี การสร้างรูปเล่มรายงานให้ถูกรูปแบบนั้นเป็นขั้นตอนที่ยุ่งยาก แม้ว่าจะมีต้นแบบสำหรับใช้ในโปรแกรม Microsoft Word แล้วก็ตาม แต่นักศึกษาส่วนใหญ่ยังคงค้นพบว่าการใช้งานมีความซับซ้อน และเกิดความผิดพลาดในการจัดรูปแบบ กำหนดเลขหัวข้อ และสร้างสารบัญอยู่ ภาควิชาวิศวกรรมคอมพิวเตอร์จึงได้จัดทำต้นแบบรูปเล่มรายงานโดยใช้ระบบจัดเตรียมเอกสาร LaTeX เพื่อช่วยให้นักศึกษาเขียนรายงานได้อย่างสะดวกและรวดเร็วมากยิ่งขึ้น

Project Title	: eDENT-ELDER: A Digital Platform for Detecting and Analyzing Potentially Malignant Disorders and Oral Cancer	
Name	: Kawin Choosin	650610745
	Natthapon Chanawerot	650612082
	Thanawat Jaisert	650610768
	Pariwat Wonginchan	650612089
Department	: Computer Engineering	
Project Advisor	: Chinawat Isradisaikul, Ph.D.	
Degree	: Bachelor of Engineering	
Program	: Computer Engineering	
Academic Year	: 2568	

ABSTRACT

The abstract would be placed here. It usually does not exceed 350 words long (not counting the heading), and must not take up more than one (1) page (even if fewer than 350 words long).

Make sure your abstract sits inside the `abstract` environment.

กิตติกรรมประกาศ

Your acknowledgments go here. Make sure it sits inside the `acknowledgment` environment.

Kawin Choosin
Natthapon Chanawerot
Thanawat Jaisert
Pariwat Wonginchan
25 พฤษภาคม 2563

สารบัญ

บทคัดย่อ . . . . . ค
Abstract . . . . . ง
กิตติกรรมประกาศ . . . . . จ
สารบัญ . . . . . ฉ
สารบัญรูป . . . . . ซ
สารบัญตาราง . . . . . ฌ

1 บทนำ **1**
1.1 ที่มาและความสำคัญของปัญหา 1
1.1.1 ข้อจำกัดทางเทคนิคของระบบเดิม 2
1.2 ระบบเดิม เฟส 1: AIDOC 3
1.3 ข้อจำกัดของระบบเฟส 1 3
1.4 วิธีดำเนินงาน: การ Re-engineer สถาปัตยกรรมระบบ 4
1.4.1 เสาหลักที่ 1: Architecture Re-engineering 4
1.4.2 เสาหลักที่ 2: High-Availability AI (Dual-Path Inference) 5
1.4.3 เสาหลักที่ 3: Data and Processing Layer 5
1.4.4 เสาหลักที่ 4: Enterprise Security and Compliance 5
1.5 แนะนำ eDENT-ELDER เฟส 2 5
1.5.1 วิสัยทัศน์และคำขวัญ 5
1.5.2 กลุ่มเป้าหมาย 5
1.5.3 4 เสาหลักของแพลตฟอร์ม 6
1.6 วัตถุประสงค์ของโครงงาน 6
1.6.1 ประโยชน์ที่คาดว่าจะได้รับ 6
1.7 ภาคีและผู้ร่วมดำเนินงาน 7

2 การทำงานของระบบ **8**
2.1 เครื่องมือคัดกรองสุขภาพช่องปาก 8
2.2 กระบวนการยืนยันตัวตนและ Flow การทำงาน 8
2.2.1 ขั้นตอนการยืนยันตัวตน 8
2.2.2 Flow การทำงานระบบแบบ 5 เลน 9
2.2.3 อินเทอร์เฟซ Mobile สำหรับผู้ป่วยและ อสม. 9
2.2.4 Screener Dashboard 10
2.2.5 Specialist Dashboard 10
2.2.6 ภาพหน้าจอจากโปสเตอร์โครงงาน 10

3 สถาปัตยกรรม ระบบ AI และความปลอดภัย **12**
3.1 สถาปัตยกรรมระบบ 12
3.1.1 เปรียบเทียบระบบเดิมและระบบใหม่ 12
3.2 ระบบ AI แบบ Dual-Path ที่มีความพร้อมใช้งานสูง 14
3.2.1 Primary Path 14
3.2.2 Fallback Path 14
3.2.3 ผลการทดสอบประสิทธิภาพจริง 14
3.2.4 ผลลัพธ์เชิงสถาปัตยกรรม 15
3.3 โครงสร้างพื้นฐานสมัยใหม่และ DevOps 16
3.4 กรอบความปลอดภัยและการปฏิบัติตามมาตรฐาน 16

3.4.1 การคุ้มครองข้อมูลและความเป็นส่วนตัว 17
3.4.2 การควบคุมการเข้าถึงและการยืนยันตัวตน 18
3.4.3 Audit Trail และการตรวจสอบย้อนหลัง 19
3.4.4 การสอดคล้องกับมาตรฐานสาธารณสุข 20

4 การทดสอบและการรับรองคุณภาพ 21
4.1 การทดสอบและการรับรองคุณภาพทางวิศวกรรม 21
4.1.1 กลยุทธ์และเครื่องมือทดสอบ 21
4.1.2 ผลการทดสอบ 21
4.1.3 Module หลักที่ผ่านการตรวจสอบ 22
4.2 Feedback จากการพัฒนา 22

5 บทสรุป 23
5.1 สรุปและบทสรุป 23

สารบัญรูป

1.1 สื่อมวลชนรายงานปัญหาการวินิจฉัยมะเร็งช่องปากล่าช้า ผู้ป่วยในชนบทมักซื้อยามารับประทานเองจนอาการรุนแรง ที่มา: Amarin News / กรมอนามัย 1
1.2 รอยโรคช่องปาก ตัวอย่างเคสที่ 1 1
1.3 รอยโรคช่องปาก ตัวอย่างเคสที่ 2 1
1.4 การ Annotation รอยโรคก่อนมะเร็งด้วย AI เส้นกรอบสีเหลืองแสดงบริเวณที่ส่งให้ผู้เชี่ยวชาญตรวจสอบ 2
1.5 ข้อมูลรอยโรคช่องปากทั่วประเทศ ปีงบประมาณ 2559 (ยกเว้น กทม.) จำแนกเป็น 3 กลุ่มตามช่วงอายุ ที่มา: กรมอนามัย 2
1.6 จำนวนทันตแพทย์เฉพาะทาง ณ วันที่ 21 พฤศจิกายน 2567 แถวที่ 9 (ไฮไลต์) สาขาวิทยาการวินิจัยโรคช่องปาก มีทันตแพทย์เพียง 96 คน 2
1.7 ระบบ AIDOC เฟส 1: (ซ้าย) หน้า Consent และ Login; (ขวา) Workflow อัปโหลดภาพและคัดกรองด้วย AI สร้างด้วย Flask + Bootstrap + MySQL 3
1.8 สถาปัตยกรรมระบบเดิม: Flask ทำหน้าที่ทั้ง Frontend, Backend, AI Inference และ Image Storage ในหน่วยเดียว ความล้มเหลวของส่วนใดทำให้ระบบทั้งหมดหยุด 4

2.1 ขั้นตอนการยืนยันตัวตนของ eDENT-ELDER: กรอกเบอร์โทร ยืนยัน OTP ลง Consent PDPA และได้รับสิทธิ์ตามบทบาท 8
2.2 Flow การทำงานระบบ eDENT-ELDER แบบ 5 เลน 9
2.3 อินเทอร์เฟซ Mobile: รายชื่อผู้ป่วย ผล OHAT แบบ OHIP การถ่ายภาพ AI-Guided และหน้าแสดงผล AI (จากซ้ายไปขวา) 9
2.4 Screener Dashboard: คิวภาพเคสพร้อม AI Tag (ซ้าย) รายละเอียดเคสพร้อม Annotation (กลาง) และตาราง OHAT/OHIP (ขวา) 10
2.5 Specialist Dashboard: คิว Referral (ซ้าย) รายละเอียดเคส (กลาง) แผนที่ความเสี่ยงระดับจังหวัด (ขวา) 10
2.6 eDENT-ELDER Platform Interface (โปสเตอร์ v1): ระบบ AI ช่วยตรวจจับรอยโรค แสดง Workflow การคัดกรองแบบ Mobile-first 10
2.7 อินเทอร์เฟซวินิจฉัยสำหรับผู้เชี่ยวชาญ (บน) และผู้คัดกรอง (ล่าง) พร้อมการเชื่อมต่อผ่าน QR Code 11

3.1 สถาปัตยกรรมระบบเฟส 2 (High-Level): Next.js + FastAPI (DDD) + PostgreSQL/MinIO/Redis + Celery + Dual-Path AI + Security Layer + Integrations . . . 12
3.2 สถาปัตยกรรม Low-Level: JWT, gRPC/HTTP, Async Queue (Redis $\rightarrow$ Celery), และประเภทงาน Worker ทั้งหมด 13
3.3 ระบบ AI แบบ Dual-Path: Primary Path ผ่าน NVIDIA Triton (gRPC + TensorRT, $<$200 ms) พร้อม Auto Failover ไปยัง TFLite/FastAPI ภายใน $<$2 วินาที . 14
3.4 ตัวชี้วัดประสิทธิภาพ AI Inference Server: Request Duration ($<$200 ms), Failure Rate ($\approx$0%), DALI vs. TensorRT Pipeline ($\approx$19.7 vs. 20.5 ms) รวม 694 การคัดกรองที่สำเร็จ 16
3.5 โครงสร้างพื้นฐานสมัยใหม่: Docker Containerization ครบทุก Service, CI/CD Pipeline อัตโนมัติ Zero-Downtime Deployment และการแยก Staging/Production อย่างเข้มงวด 16

4.1 ผลการรับรองคุณภาพ: Coverage $\approx$75% โดยรวม พร้อม 100% บน Critical Module; Vitest UI ยืนยัน 900 Test ผ่านภายใน $<$13 s 21

สารบัญตาราง

บทที่ 1
บทนำ

1.1 ที่มาและความสำคัญของปัญหา

มะเร็งช่องปาก (Oral Cancer) ติดอันดับ 1 ใน 10 โรคที่พบบ่อยที่สุดในประเทศไทย ความท้าทายสำคัญในชุมชนชนบทคือพฤติกรรมการซื้อยาปฏิชีวนะมารับประทานเองเมื่อมีแผลในช่องปาก โดยไม่ปรึกษาแพทย์หรือทันตแพทย์ ส่งผลให้การวินิจฉัยล่าช้าและพบโรคในระยะลุกลาม ทำให้ผลการรักษาแย่ลงอย่างมีนัยสำคัญ

[ช่องว่างสำหรับรูปภาพ]
fig_bg_news.png
ภาพข่าวแสดงกรณีมะเร็งช่องปากและการวินิจฉัยล่าช้าในชนบทไทย (สไลด์ 3)
แทนที่ด้วยไฟล์ภาพ: fig_bg_news.png

รูปที่ 1.1: สื่อมวลชนรายงานปัญหาการวินิจฉัยมะเร็งช่องปากล่าช้า ผู้ป่วยในชนบทมักซื้อยามารับประทานเองจนอาการรุนแรง ที่มา: Amarin News / กรมอนามัย

ข้อมูลจากกรมอนามัย กระทรวงสาธารณสุข แบ่งรอยโรคในช่องปากออกเป็น 3 กลุ่มหลัก:

- **รอยโรคช่องปากทั่วไป (Oral Lesion)** — พบมากที่สุด ครอบคลุมทุกช่วงอายุ
- **รอยโรคก่อนมะเร็ง (PMD — Potentially Malignant Disorders)** — รอยขาว (Leukoplakia) รอยแดง (Erythroplakia) ก้อน และแผลเล็กในช่องปาก
- **มะเร็งช่องปาก (OSCC — Oral Squamous Cell Carcinoma)** — พบมากในกลุ่มอายุ 60 ปีขึ้นไป

[ช่องว่างสำหรับรูปภาพ]
fig_bg_lesion1.png
ภาพทางคลินิก ภาพซ้าย (สไลด์ 4)
แทนที่ด้วยไฟล์ภาพ: fig_bg_lesion1.png

รูปที่ 1.2: รอยโรคช่องปาก ตัวอย่างเคสที่ 1

[ช่องว่างสำหรับรูปภาพ]
fig_bg_lesion2.png
ภาพทางคลินิก ภาพขวา (สไลด์ 4)
แทนที่ด้วยไฟล์ภาพ: fig_bg_lesion2.png

รูปที่ 1.3: รอยโรคช่องปาก ตัวอย่างเคสที่ 2

[ช่องว่างสำหรับรูปภาพ]

fig_bg_lesion_annotated.png

ภาพทางคลินิกพร้อม Annotation สีเหลือง (สไลด์ 5)

แทนที่ด้วยไฟล์ภาพ: fig_bg_lesion_annotated.png

รูปที่ 1.4: การ Annotation รอยโรคก่อนมะเร็งด้วย AI เส้นกรอบสีเหลืองแสดงบริเวณที่ส่งให้ผู้เชี่ยวชาญตรวจสอบ

[ช่องว่างสำหรับรูปภาพ]

fig_bg_stats_chart.png

กราฟสถิติรอยโรคช่องปากระดับประเทศ ปีงบประมาณ 2559 (สไลด์ 6)

แทนที่ด้วยไฟล์ภาพ: fig_bg_stats_chart.png

รูปที่ 1.5: ข้อมูลรอยโรคช่องปากทั่วประเทศ ปีงบประมาณ 2559 (ยกเว้น กทม.) จำแนกเป็น 3 กลุ่ม ตามช่วงอายุ ที่มา: กรมอนามัย

[ช่องว่างสำหรับรูปภาพ]

fig_bg_registry_table.png

ตารางจำนวนทันตแพทย์เฉพาะทาง ราชวิทยาลัยทันตแพทย์ (สไลด์ 7)

แทนที่ด้วยไฟล์ภาพ: fig_bg_registry_table.png

รูปที่ 1.6: จำนวนทันตแพทย์เฉพาะทาง ณ วันที่ 21 พฤศจิกายน 2567 แถวที่ 9 (ไฮไลต์) สาขาวิทยาการวินิจัยโรคช่องปาก มีทันตแพทย์เพียง 96 คน

ปัญหาหลัก

ผู้สูงอายุในชนบทอายุ 50 ปีขึ้นไป โดยเฉพาะกลุ่มเปราะบางที่พึ่งพาผู้ดูแล เผชิญอุปสรรคหนักในการเข้าถึงทันตแพทย์เฉพาะทาง เมื่อได้รับการวินิจฉัย มักพบว่าโรคลุกลามสู่ระยะท้ายแล้ว ส่งผลให้ผลการรักษาแย่ลงและค่าใช้จ่ายสูงขึ้นอย่างมาก

1.1.1 ข้อจำกัดทางเทคนิคของระบบเดิม

แม้ระบบต้นแบบในเฟส 1 จะมีความแม่นยำของ AI ที่น่าพอใจ แต่ยังไม่พร้อมสำหรับการใช้งานจริงในฐานะซอฟต์แวร์ทางการแพทย์ *ในบริบทซอฟต์แวร์ทางการแพทย์ ความแม่นยำของ AI เพียงอย่างเดียวไม่เพียงพอ*

— *ระบบต้องปลอดภัย โปร่งใส ตรวจสอบได้ และพร้อมใช้งานตลอดเวลา* ระบบเดิมมีช่องโหว่สำคัญ 5 ประการ:

- **ไม่มีกรอบการปฏิบัติตาม PDPA / HIPAA** — ข้อมูลสุขภาพผู้ป่วยถูกจัดเก็บโดยไม่ผ่านมาตรฐานคุ้มครองข้อมูลส่วนบุคคล
- **AI แบบ Monolith เป็น Single Point of Failure** — หาก AI Service ล่ม ระบบทั้งหมดหยุดทำงานทันที ไม่มี Fallback
- **ไม่มีระบบ Audit Trail ป้องกันการแก้ไขย้อนหลัง** — ไม่มีกลไกตรวจจับหรือป้องกันการแก้ไขเวชระเบียนย้อนหลัง
- **โค้ดแบบ Monolith ขยายระบบไม่ได้** — การเปลี่ยนแปลงใดก็ตามเสี่ยงกระทบส่วนอื่น ไม่รองรับการ Scale Out แต่ละ Service
- **ไม่มี Role-Based Access Control (RBAC)** — ผู้ใช้ทุกคนที่ผ่านการยืนยันตัวตนเข้าถึงข้อมูลและฟังก์ชันได้เหมือนกัน

1.2 ระบบเดิม เฟส 1: AIDOC

ระบบต้นแบบในเฟส 1 คือ **AIDOC** (*Artificial Intelligent System for Detecting and Analyzing PMDs and Oral Cancer*) พัฒนาและนำไปใช้งานร่วมกับกรมอนามัย สร้างบน Flask Monolith Architecture ใช้ Bootstrap เป็น UI จัดเก็บรูปภาพบน File System และใช้ MySQL เป็นฐานข้อมูล

[ช่องว่างสำหรับรูปภาพ]

`fig_aidoc_screenshot.png`

หน้าจอระบบ AIDOC เฟส 1 (สไลด์ 8)

แทนที่ด้วยไฟล์ภาพ: `fig_aidoc_screenshot.png`

รูปที่ 1.7: ระบบ AIDOC เฟส 1: (ซ้าย) หน้า Consent และ Login; (ขวา) Workflow อัปโหลดภาพและคัดกรองด้วย AI สร้างด้วย Flask + Bootstrap + MySQL

1.3 ข้อจำกัดของระบบเฟส 1

"ซอฟต์แวร์ทางการแพทย์ต้องมีความปลอดภัย โปร่งใส ตรวจสอบได้ และพร้อมใช้งานตลอดเวลา"

[ช่องว่างสำหรับรูปภาพ]

fig_legacy_arch.png

Diagram สถาปัตยกรรม Flask Monolith (สไลด์ 9/18)

แทนที่ด้วยไฟล์ภาพ: fig_legacy_arch.png

รูปที่ 1.8: สถาปัตยกรรมระบบเดิม: Flask ทำหน้าที่ทั้ง Frontend, Backend, AI Inference และ Image Storage ในหน่วยเดียว ความล้มเหลวของส่วนใดทำให้ระบบทั้งหมดหยุด

ข้อจำกัด	ผลกระทบ
ไม่มีการปฏิบัติตาม PDPA / HIPAA	ข้อมูลผู้ป่วยขาดการคุ้มครองตามกฎหมายและข้อกำหนดที่เกี่ยวข้อง
Single Point of Failure	หาก Flask ล่ม ระบบทั้งหมดใช้งานไม่ได้ทันที
ไม่มี Audit Trail ป้องกันการแก้ไขย้อนหลัง	เวชระเบียนสามารถถูกแก้ไขโดยไม่มีหลักฐาน
โค้ดแบบ Monolith ขยายระบบยาก	การแก้ไข Bug หรืออัปเดต AI เสี่ยงทำให้ระบบอื่นพัง

1.4 วิธีดำเนินงาน: การ Re-engineer สถาปัตยกรรมระบบ

วิธีดำเนินงานหลักของเฟส 2 คือการ Re-engineer สถาปัตยกรรมระบบใหม่ทั้งหมด โดยแบ่งออกเป็น 4 เสาหลักทางวิศวกรรม:

1.4.1 เสาหลักที่ 1: Architecture Re-engineering

- **Next.js** (Layered Design, Mobile-first) สำหรับ Frontend พร้อม Auth Guard และการแยก State Management
- **FastAPI + Clean Architecture + Domain-Driven Design (DDD)** สำหรับ Backend — Business Logic แยกจาก AI Layer อย่างสมบูรณ์
- **Async Queue Processing** (Redis + Celery Workers) ขจัดปัญหาเว็บค้างและ API Timeout จาก Synchronous AI Inference

1.4.2 เสาหลักที่ 2: High-Availability AI (Dual-Path Inference)

- **Primary Path:** NVIDIA Triton Inference Server (GPU, gRPC) พร้อม TensorRT Optimization
- **Fallback Path:** TensorFlow Lite (CPU, HTTP) พร้อม Auto Failover อัตโนมัติ
- **Health Monitoring:** ตรวจสอบสถานะต่อเนื่อง เปิด Auto-Switch ภายใน $< 2\,\mathrm{s}$

1.4.3 เสาหลักที่ 3: Data and Processing Layer

- **PostgreSQL** (OLTP + Audit) — ฐานข้อมูลหลักพร้อม Immutable Audit Schema
- **Redis** — Task Queue สำหรับงาน AI แบบ Async และ Cache
- **MinIO** — Object Storage สำหรับรูปภาพทางคลินิก
- **Celery Workers** — ประมวลผลเบื้องหลัง: AI Inference Queue, 2-Tier Audit Chain, Archival และงาน Compliance

1.4.4 เสาหลักที่ 4: Enterprise Security and Compliance

- **TLS 1.3** เข้ารหัสการสื่อสารทุกส่วนแบบ End-to-End
- **RBAC** ครอบคลุม 6 บทบาท: ผู้ป่วย, อสม., ทันตแพทย์, ผู้คัดกรอง, ผู้เชี่ยวชาญ, ผู้ดูแลระบบ
- **2-Tier Audit Logging + Hash-Chain** ป้องกันการแก้ไขย้อนหลัง ตรวจสอบได้ 100%
- **PDPA / HIPAA-ready** สอดคล้องกับกฎหมายคุ้มครองข้อมูลส่วนบุคคลของไทย

1.5 แนะนำ eDENT-ELDER เฟส 2

1.5.1 วิสัยทัศน์และคำขวัญ

“พบเร็ว – ส่งต่อเร็ว – ติดตามครบ”

“Detect Early — Refer Fast — Follow Up Completely”

1.5.2 กลุ่มเป้าหมาย

1. **ผู้สูงอายุที่มีอายุ 50 ปีขึ้นไป** — กลุ่มเปราะบาง อาศัยในชนบท และต้องพึ่งพาผู้ดูแล
2. **บุคลากรสาธารณสุขแนวหน้า** — อาสาสมัครสาธารณสุขประจำหมู่บ้าน (อสม.) ทันตาภิบาล และเจ้าหน้าที่ รพ.สต. ที่ผ่าน Train-the-Trainer 6–8 รุ่น

1.5.3 4 เสาหลักของแพลตฟอร์ม

Hybrid Mobile Platform	อินเทอร์เฟซที่ใช้งานง่ายผ่าน LINE Chatbot และ Web Dashboard สำหรับผู้ดูแลและเจ้าหน้าที่ภาคสนาม
AI-Guided Screening	AI ตรวจจับรอยโรค OPMD/OSCC พร้อม Assisted-Capture และ Quality-Gate รับรองคุณภาพภาพถ่าย
Elderly-Centric Empowerment	เสริมพลัง อสม. (อายุ 50+) ในการคัดกรอง สื่อสารความเสี่ยง และส่งต่ออย่างเป็นระบบ
Async. Tele-Consult	ระบบ Store-and-Forward เชื่อมต่อ อสม. กับผู้เชี่ยวชาญ โดยไม่ต้องนัดหมายพร้อมกัน

1.6 วัตถุประสงค์ของโครงงาน

1. **พัฒนาและทดสอบระบบ AI คัดกรอง** OPMD/OSCC ให้มีความแม่นยำสูง ใช้งานได้ในภาคสนาม และเป็นที่ยอมรับของทันตแพทย์เฉพาะทาง
2. **พัฒนาแพลตฟอร์ม eDENT-ELDER** รองรับการคัดกรองด้วยตัวเองและการคัดกรองชุมชน
3. **ประเมินผลลัพธ์ด้านบริการและสังคม** วัดการเข้าถึง เวลาพบผู้เชี่ยวชาญ อัตราตรวจพบระยะแรก และความยั่งยืน

1.6.1 ประโยชน์ที่คาดว่าจะได้รับ

- **ระดับระบบ:** แพลตฟอร์มที่ผสาน AI ปลอดภัย ตรวจสอบได้ ลดภาระแนวหน้าและเพิ่มโอกาสตรวจพบระยะแรก (Stage-Shift)
- **ระดับชุมชน:** ผู้สูงอายุและครอบครัวในพื้นที่นำร่องได้รับการคัดกรองเร็วขึ้น หัวหน้า อสม. (อายุเฉลี่ย 50+) เป็นทั้งผู้ให้บริการและผู้รับประโยชน์

พื้นที่นำร่องหลัก: เชียงใหม่ • พะเยา • ภูเก็ต • หนองบัวลำภู
เครือข่ายเสริม: สุรินทร์ สมุทรปราการ และจังหวัดอื่นตามความพร้อม

1.7 ภาคีและผู้ร่วมดำเนินงาน

หน่วยงาน	บทบาท
คณะทันตแพทยศาสตร์ มหาวิทยาลัยธรรมศาสตร์	คำปรึกษาทางคลินิกและเครือข่ายผู้เชี่ยว-ชาญ
คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเชียงใหม่	ทีมพัฒนาระบบ
คณะทันตแพทยศาสตร์ มหาวิทยาลัยเชียงใหม่	ตรวจสอบและให้ Feedback ทางคลินิก
กรมอนามัย ศูนย์ทันตสาธารณสุขระหว่างประเทศ เชียงใหม่	ความร่วมมือด้านข้อมูลและการ Deploy
NurseCMU	บูรณาการงานสาธารณสุขชุมชน

บทที่ 2
การทำงานของระบบ

2.1 เครื่องมือคัดกรองสุขภาพช่องปาก

Oral Health Assessment Tool (OHAT) — ประเมินสภาพช่องปากเพื่อคัดกรองความเสี่ยงและตัดสินใจว่าควรส่งภาพเข้า AI วิเคราะห์หรือไม่

Oral Health Impact Profile (OHIP) — ประเมินผลกระทบของสุขภาพช่องปากต่อคุณภาพชีวิต ทั้งความเจ็บปวด การใช้ชีวิตประจำวัน และการเข้าสังคม

2.2 กระบวนการยืนยันตัวตนและ Flow การทำงาน

2.2.1 ขั้นตอนการยืนยันตัวตน

1. ผู้ใช้งานกรอกหมายเลขโทรศัพท์
2. ระบบตรวจสอบสถานะการลงทะเบียนในทะเบียนราษฎร์
3. *กรณีผู้ใช้ใหม่:* ลงทะเบียน Consent ตาม PDPA และยืนยันบทบาท
4. ส่งและยืนยัน OTP ผ่าน SMS
5. ให้สิทธิ์ตามบทบาท (ผู้ป่วย หรือ อสม./ผู้คัดกรอง)

[ช่องว่างสำหรับรูปภาพ]

`fig_auth_flow.png`

Diagram Flow การยืนยันตัวตน พร้อมภาพหน้าจอ Mobile (สไลด์ 12)

แทนที่ด้วยไฟล์ภาพ: `fig_auth_flow.png`

รูปที่ 2.1: ขั้นตอนการยืนยันตัวตนของ eDENT-ELDER: กรอกเบอร์โทร ยืนยัน OTP ลง Consent PDPA และได้รับสิทธิ์ตามบทบาท

2.2.2 Flow การทำงานระบบแบบ 5 เลน

เลน	หน้าที่
เลน 1 — ผู้ป่วย / อสม.	รับข้อมูลและ Consent; แบบสอบถาม OHAT/OHIP; ถ่ายภาพผ่าน Quality Gate; ส่งเข้า AI
เลน 2 — ระบบ AI	ตรวจสอบคุณภาพภาพ; ตรวจจับรอยโรคและประเมินความเสี่ยง; จัดการ Fallback
เลน 3 — ผู้คัดกรอง	Screener Dashboard; Triage ผล AI; Flag Retrain หรือ Escalate
เลน 4 — ผู้เชี่ยวชาญ	ยืนยันวินิจฉัย; แจ้งเตือน LINE; ติดตาม 14 วัน; ปิดเคส
เลน 5 — ผู้ดูแลระบบ / วิชาการ	Dashboard ระดับประชากร; แผนที่ความเสี่ยงจังหวัด; กำกับระบบ

[ช่องว่างสำหรับรูปภาพ]

`fig_system_flow.png`

Diagram Flow 5 เลนพร้อมภาพหน้าจอทุกกลุ่มผู้ใช้ (สไลด์ 13)

แทนที่ด้วยไฟล์ภาพ: `fig_system_flow.png`

รูปที่ 2.2: Flow การทำงานระบบ eDENT-ELDER แบบ 5 เลน

2.2.3 อินเทอร์เฟซ Mobile สำหรับผู้ป่วยและ อสม.

[ช่องว่างสำหรับรูปภาพ]

`fig_patient_vhv_ui.png`

หน้าจอ Mobile ผู้ป่วยและ อสม. (สไลด์ 14)

แทนที่ด้วยไฟล์ภาพ: `fig_patient_vhv_ui.png`

รูปที่ 2.3: อินเทอร์เฟซ Mobile: รายชื่อผู้ป่วย ผล OHAT แบบ OHIP การถ่ายภาพ AI-Guided และหน้าแสดงผล AI (จากซ้ายไปขวา)

2.2.4 Screener Dashboard

[ช่องว่างสำหรับรูปภาพ]

fig_screener_dashboard.png

Screener Dashboard บน Tablet (สไลด์ 15)

แทนที่ด้วยไฟล์ภาพ: fig_screener_dashboard.png

รูปที่ 2.4: Screener Dashboard: คิวภาพเคสพร้อม AI Tag (ซ้าย) รายละเอียดเคสพร้อม Annotation (กลาง) และตาราง OHAT/OHIP (ขวา)

2.2.5 Specialist Dashboard

[ช่องว่างสำหรับรูปภาพ]

fig_specialist_dashboard.png

Specialist Dashboard บน Tablet (สไลด์ 16)

แทนที่ด้วยไฟล์ภาพ: fig_specialist_dashboard.png

รูปที่ 2.5: Specialist Dashboard: คิว Referral (ซ้าย) รายละเอียดเคส (กลาง) แผนที่ความเสี่ยงระดับจังหวัด (ขวา)

2.2.6 ภาพหน้าจอจากโปสเตอร์โครงงาน

[ช่องว่างสำหรับรูปภาพ]

fig_poster_interface.png

อินเทอร์เฟซแพลตฟอร์มจากโปสเตอร์ v1 (บน)

แทนที่ด้วยไฟล์ภาพ: fig_poster_interface.png

รูปที่ 2.6: eDENT-ELDER Platform Interface (โปสเตอร์ v1): ระบบ AI ช่วยตรวจจับรอยโรค แสดง Workflow การคัดกรองแบบ Mobile-first

[ช่องว่างสำหรับรูปภาพ]

fig_poster_screener.png

หน้าจอผู้เชี่ยวชาญและผู้คัดกรองจากโปสเตอร์ v1 (ล่าง)

แทนที่ด้วยไฟล์ภาพ: fig_poster_screener.png

รูปที่ 2.7: อินเทอร์เฟซวินิจฉัยสำหรับผู้เชี่ยวชาญ (บน) และผู้คัดกรอง (ล่าง) พร้อมการเชื่อมต่อผ่าน QR Code

บทที่ 3
สถาปัตยกรรม ระบบ AI และความปลอดภัย

3.1 สถาปัตยกรรมระบบ

3.1.1 เปรียบเทียบระบบเดิมและระบบใหม่

เฟส 1 รวมทุกอย่างไว้ใน Flask Deployment เดียว เฟส 2 ใช้ **Service-Based Hybrid Architecture ที่มี Modular Monolith Core**

[ช่องว่างสำหรับรูปภาพ]

fig_new_arch.png

ภาพรวมสถาปัตยกรรมระบบเฟส 2 (สไลด์ 17/19)

แทนที่ด้วยไฟล์ภาพ: fig_new_arch.png

รูปที่ 3.1: สถาปัตยกรรมระบบเฟส 2 (High-Level): Next.js + FastAPI (DDD) + PostgreSQL/MinIO/Redis + Celery + Dual-Path AI + Security Layer + Integrations

Scalable · Maintainable · Future-proof

Technology Stack

Layer	เทคโนโลยี
Frontend	Next.js (Layered Design)
Backend	FastAPI (Domain-Driven Design)
ฐานข้อมูลหลัก	PostgreSQL (OLTP + Audit)
Object Storage	MinIO
Queue และ Cache	Redis
Background Workers	Celery
AI Primary	NVIDIA Triton Inference Server (TensorRT, gRPC)
AI Fallback	TensorFlow Lite via FastAPI (HTTP)
Integrations	SMS/OTP (Thaibureau), ทะเบียนราษฎร์, Maps

ข้อได้เปรียบหลัก

- Business Logic แยกออกจาก AI Inference Layer อย่างสมบูรณ์
- Schema Validation End-to-End: Zod (Frontend) + Pydantic (Backend)
- พร้อมสำหรับ Microservices ในการ Scale AI หรือข้อมูลในอนาคต

รายละเอียดสถาปัตยกรรม Low-Level

Component	รายละเอียด
JWT Auth Guard	Access/Refresh Token แบบ Stateless บังคับใช้ที่ทุก API Boundary
gRPC / HTTP	Primary Path ใช้ gRPC ไปยัง Triton; Fallback ใช้ HTTP REST ไปยัง TFLite
AI Inference Queue	Celery Queue บน Redis แยก Upload จาก Inference
2-Tier Audit Chain	ทุก Write เพิ่ม Record ที่มี Hash ของ Record ก่อนหน้า
Archival & Cleanup	Worker เก็บถาวร Record เก่าและบังคับนโยบาย Retention
Compliance Tasks	ตรวจสอบ Consent หมดอายุ, Anonymise ข้อมูล, รายงาน PDPA

[ช่องว่างสำหรับรูปภาพ]

`fig_lowlevel_arch.png`

Diagram สถาปัตยกรรม Low-Level (โปสเตอร์ v2)

แทนที่ด้วยไฟล์ภาพ: `fig_lowlevel_arch.png`

รูปที่ 3.2: สถาปัตยกรรม Low-Level: JWT, gRPC/HTTP, Async Queue (Redis → Celery), และประเภทงาน Worker ทั้งหมด

3.2 ระบบ AI แบบ Dual-Path ที่มีความพร้อมใช้งานสูง

[ช่องว่างสำหรับรูปภาพ]

fig_dual_ai_arch.png

Diagram Dual-Path AI: Triton Primary และ TFLite Fallback (สไลด์ 20)

แทนที่ด้วยไฟล์ภาพ: fig_dual_ai_arch.png

รูปที่ 3.3: ระบบ AI แบบ Dual-Path: Primary Path ผ่าน NVIDIA Triton (gRPC + TensorRT, <200 ms) พร้อม Auto Failover ไปยัง TFLite/FastAPI ภายใน <2 วินาที

3.2.1 Primary Path

- **Engine:** NVIDIA Triton Inference Server
- **การประมวลผล:** เร่งด้วย GPU (gRPC + TensorRT Optimization)
- **Latency:** $<$ 200 ms ต่อการ Inference
- **Models:** รัน Ensemble Models พร้อมกัน (ตรวจรอยโรค + ตรวจคุณภาพภาพ)

3.2.2 Fallback Path

- **Engine:** TensorFlow Lite (CPU, HTTP via FastAPI)
- **Auto Failover:** <1 วินาที หาก Primary Path ล้มเหลว

3.2.3 ผลการทดสอบประสิทธิภาพจริง

สภาพแวดล้อมทดสอบ: CPU 4 Core, NVIDIA GTX 1060 GPU, RAM 16 GB

ตัวชี้วัด	ผลลัพธ์
จำนวนการคัดกรองที่สำเร็จทั้งหมด	**694**
อัตราความสำเร็จ Throughput	$\approx$99.9%
Average Inference Latency (End-to-End)	< 500 ms
ความเร็ว Quality Check	< 100 ms
ความเร็ว Lesion Detection	$\approx$400 ms
GPU Compute (TensorRT)	38.1 ms
DALI Preprocessing Stage	$\approx$19.7 ms
เวลา Auto-Switch Failover	< 2 s
อัตราการ Inference ล้มเหลว	$\approx$0%

3.2.4 ผลลัพธ์เชิงสถาปัตยกรรม

- **Zero-Downtime AI Capability** — สลับจาก Triton (GPU) ไปยัง TFLite (CPU) โดยไม่หยุดให้บริการ แม้ AI Server หลักจะ Crash
- **ขจัดปัญหา API Timeout** — AI Inference ประมวลผลแบบ Async (Batch/Queue) แก้ปัญหา Request บล็อกภายใต้โหลดสูง
- **Healthcare Compliance Ready** — RBAC และ 2-Tier Audit Logging ตรวจสอบข้อมูลย้อนหลังได้ 100% ตามมาตรฐาน PDPA/HIPAA
- **Modular Design และ Scalability** — Scale Frontend, Backend และ AI ได้อย่างอิสระ เพิ่มฟีเจอร์ใหม่โดยไม่กระทบ Component อื่น
- **High Maintainability** — Schema Validation ด้วย Zod + Pydantic ลด Technical Debt แบบ End-to-End

[ช่องว่างสำหรับรูปภาพ]

fig_performance_metrics.png

Dashboard ตัวชี้วัดประสิทธิภาพ (สไลด์ 21)

แทนที่ด้วยไฟล์ภาพ: fig_performance_metrics.png

รูปที่ 3.4: ตัวชี้วัดประสิทธิภาพ AI Inference Server: Request Duration ($<$200 ms), Failure Rate ($\approx$0%), DALI vs. TensorRT Pipeline ($\approx$19.7 vs. 20.5 ms) รวม 694 การคัดกรองที่สำเร็จ

3.3 โครงสร้างพื้นฐานสมัยใหม่และ DevOps

[ช่องว่างสำหรับรูปภาพ]

fig_devops.png

Docker, CI/CD Pipeline, Environment Segregation (สไลด์ 22)

แทนที่ด้วยไฟล์ภาพ: fig_devops.png

รูปที่ 3.5: โครงสร้างพื้นฐานสมัยใหม่: Docker Containerization ครบทุก Service, CI/CD Pipeline อัตโนมัติ Zero-Downtime Deployment และการแยก Staging/Production อย่างเข้มงวด

Full Containerization	ทุก Service อยู่ใน Docker ขจัด XAMPP Dependency ทำงานได้เหมือนกันทั้งบน Cloud และ On-Premise
CI/CD Pipeline	Pipeline อัตโนมัติแทนการ Clone/Install ด้วยมือ รองรับ Zero-Downtime Deployment
Environment Segregation	แยก Staging และ Production อย่างเด็ดขาด ป้องกัน Hardcoded Settings ในระบบจริง

3.4 กรอบความปลอดภัยและการปฏิบัติตามมาตรฐาน

ส่วนนี้ประเมินสถานะความปลอดภัยของเฟส 2 อย่างโปร่งใส โดยแยกแยะสิ่งที่ดำเนินการแล้ว สิ่งที่อยู่ระหว่างดำเนินการ และสิ่งที่อยู่นอกขอบเขตโครงงาน สัญลักษณ์:

3 ดำเนินการแล้ว สร้าง ทดสอบ และใช้งานอยู่จริง

~ บางส่วน มีกลไกหลักแล้ว แต่ยังไม่ได้รับการรับรองภายนอก

7 ยังไม่ได้ดำเนินการ วางแผนไว้แต่อยู่นอกขอบเขตเฟส 2

3.4.1 การคุ้มครองข้อมูลและความเป็นส่วนตัว

ข้อกำหนด	สถานะ	หมายเหตุ
เข้ารหัสข้อมูลระหว่างรับส่ง (TLS 1.3)	3	การสื่อสารทั้งหมดเข้ารหัสด้วย TLS 1.3
เข้ารหัสข้อมูลที่จัดเก็บ (ฐานข้อมูล)	~	เปิด PostgreSQL Encryption แล้ว แต่การเข้ารหัสระดับ Disk ขึ้นอยู่กับ Infrastructure
เข้ารหัสข้อมูลที่จัดเก็บ (Object Storage)	~	เปิด MinIO Server-Side Encryption แล้ว แต่ยังไม่ได้เชื่อมต่อ KMS
การรับ Consent และจัดเก็บ (PDPA)	3	รับ Consent ผ่าน OTP พร้อม Timestamp และ User ID
สิทธิ์การลบข้อมูล (Right to Erasure)	7	ยังไม่ได้สร้าง Workflow ลบข้อมูลตาม PDPA ม.33 วางแผนสำหรับเฟส 3
นโยบาย Data Minimisation	~	เก็บเฉพาะข้อมูลที่จำเป็นทางคลินิก แต่ยังไม่มีเอกสารนโยบายอย่างเป็นทางการ
ควบคุมการถ่ายโอนข้อมูลข้ามพรมแดน	7	ข้อมูลทั้งหมดอยู่ในไทย แต่ยังไม่ได้จัดทำ Transfer Impact Assessment

3.4.2 การควบคุมการเข้าถึงและการยืนยันตัวตน

ข้อกำหนด	สถานะ	หมายเหตุ
Role-Based Access Control (RBAC)	3	6 บทบาทบังคับใช้ที่ทุก API Route
JWT Authentication (Access + Refresh)	3	Access Token 15 นาที + Refresh Token อายุยาว บังคับผ่าน Auth Guard
OTP / SMS ยืนยันตัวตน	3	OTP ผ่าน Thaibureau SMS Gateway ใช้ทั้งการลงทะเบียนและยืนยันซ้ำ
Multi-Factor Authentication (MFA)	7	มี OTP เฉพาะตอนลงทะเบียน ยังไม่มี Step-Up MFA สำหรับบทบาทสิทธิ์สูง
Session Timeout และการยกเลิก	3	บังคับหมดอายุ; Redis Blacklist Refresh Token เมื่อ Logout
การจัดการ Credential (ไม่ Hardcode)	3	ใช้ Environment Variables ทั้งหมด ไม่มี Credential ในโค้ดหรือ Version Control
API Rate Limiting	3	Per-IP และ Per-User Rate Limiting ที่ API Gateway ป้องกัน Brute-Force
CORS Policy	3	CORS Allowlist เข้มงวด อนุญาตเฉพาะ Frontend Origins ที่รู้จัก

3.4.3 Audit Trail และการตรวจสอบย้อนหลัง

ข้อกำหนด	สถานะ	หมายเหตุ
2-Tier Audit Logging	3	บันทึกทุก Clinical Write และ Access Event ใน Immutable Audit Schema
Hash-Chain Integrity (ป้องกันการแก้ไข)	3	SHA-256 Hash ของ Record ก่อนหน้าในทุก Entry การแก้ไขย้อนหลังทำลาย Chain ตรวจพบได้ทันที
ตรวจสอบย้อนหลังได้ 100%	3	ทุก Record ตรวจสอบได้นับจากวันเริ่มระบบ
Export Audit Log สำหรับผู้กำกับดูแล	~	Admin Dashboard รองรับ Export CSV แต่รูปแบบรายงานสำหรับผู้กำกับดูแลยังไม่ได้มาตรฐาน
แจ้งเตือน Anomaly / Intrusion อัตโนมัติ	7	ยังไม่มีการแจ้งเตือนอัตโนมัติ เช่น การ Export ข้อมูลจำนวนมากหรือเข้าถึงนอกเวลา วางแผนสำหรับ Release ถัดไป

3.4.4 การสอดคล้องกับมาตรฐานสาธารณสุข

ข้อกำหนด	สถานะ	หมายเหตุ
การสอดคล้องกับ PDPA	~	Consent, Access Control และ Audit ดำเนินการแล้ว การรับรองสมบูรณ์ต้องการตรวจสอบทางกฎหมายและ Erasure Workflow (เฟส 3)
HIPAA Technical Safeguards	~	Access Control, Encryption และ Audit ดำเนินการแล้ว แต่ยังไม่มี Business Associate Agreement (BAA) และ Risk Assessment
ISO 27001 Information Security	7	สถาปัตยกรรม ออกแบบ ตาม หลัก ISO 27001 แต่ยังไม่เริ่มกระบวนการรับรองอย่างเป็นทางการ
ซอฟต์แวร์ อุปกรณ์ ทางการ แพทย์ (IEC 62304)	7	ไม่อยู่ในขอบเขตเฟส 2 จำเป็นหากต้องการขึ้นทะเบียน อย. ไทย หรือ CE Class II
Penetration Testing / Security Audit	7	ยังไม่ได้ทำ Penetration Test โดยบุคคลที่สาม มีเพียง Code Review ภายในและ Static Analysis

สรุป: เฟส 2 ดำเนินการควบคุมความปลอดภัยทางเทคนิคหลักครบถ้วนแล้ว ได้แก่ TLS Encryption, RBAC, JWT, OTP, Tamper-Proof Audit Logging และการจัดการ Credential อย่างถูกต้อง ช่องว่างที่เหลืออยู่คือ *การรับรองอย่างเป็นทางการ* (PDPA ทางกฎหมาย, HIPAA BAA, ISO 27001) *การตอบสนองภัยคุกคามขั้นสูง* (MFA สำหรับ Admin, Anomaly Alerting) และ *Workflow สิทธิ์เจ้าของข้อมูล* (การลบ, Data Portability) ทั้งหมดวางแผนสำหรับเฟส 3 ขณะที่แพลตฟอร์มมุ่งสู่การ Deploy ระดับประเทศ

บทที่ 4
การทดสอบและการรับรองคุณภาพ

4.1 การทดสอบและการรับรองคุณภาพทางวิศวกรรม

4.1.1 กลยุทธ์และเครื่องมือทดสอบ

- **Unit Test** — ทดสอบ Function, Method และ Class แต่ละตัวแยกกัน
- **Integration Test** — ทดสอบการทำงานร่วมกันของหลาย Component

เครื่องมือ	วัตถุประสงค์
Vitest	Test Runner สำหรับ Unit และ Integration Test ที่รวดเร็ว
React Testing Library	ทดสอบ UI ระดับ Component
jsdom	จำลองสภาพแวดล้อม Browser

4.1.2 ผลการทดสอบ

[ช่องว่างสำหรับรูปภาพ]

`fig_test_results.png`

รายงาน Coverage และ Vitest UI: 900 ผ่าน / 0 ล้มเหลว (สไลด์ 27)

แทนที่ด้วยไฟล์ภาพ: `fig_test_results.png`

รูปที่ 4.1: ผลการรับรองคุณภาพ: Coverage ≈75% โดยรวม พร้อม 100% บน Critical Module; Vitest UI ยืนยัน 900 Test ผ่านภายใน <13 s

จำนวน Test ทั้งหมด	900 (Unit + Integration)
สถานะ	**ผ่าน 100% — ล้มเหลว 0 Test**
Code Coverage โดยรวม	≈75%
Coverage บน Critical Business Logic	**100%**
เวลาที่ใช้รัน	< 13 s

4.1.3 Module หลักที่ผ่านการตรวจสอบ

- **Auth** — การยืนยันตัวตนและการควบคุมสิทธิ์ตามบทบาท
- **Clinical** — Logic แบบสอบถาม OHIP, OHAT และการคำนวณคะแนนวินิจฉัย
- **Upload** — Pipeline อัปโหลดไฟล์ปลอดภัยพร้อม Quality-Gate Validation

4.2 Feedback จากการพัฒนา

กระบวนการพัฒนาได้รับ Feedback โดยตรงจากผู้เชี่ยวชาญที่มีประสบการณ์จริง ในการพัฒนาแพลตฟอร์ม Web ทางคลินิกทันตกรรม ข้อมูลนี้ชี้นำการออกแบบ Interface, Clinical Workflow Integration และ Logic ของแบบสอบถาม เพื่อให้แพลตฟอร์มสะท้อนความต้องการในการปฏิบัติงานจริง

บทที่ 5
บทสรุป

5.1 สรุปและบทสรุป

eDENT-ELDER เป็นการ Re-engineer ระบบ AIDOC เฟส 1 อย่างครบวงจร แปลงระบบที่เปราะบางให้กลายเป็นแพลตฟอร์มสุขภาพดิจิทัลที่ปลอดภัย ขยายได้ และผ่านการรับรองทางคลินิก

ผลสำเร็จหลัก

- **สถาปัตยกรรม:** Hybrid Modular System (Next.js + FastAPI + DDD) แทน Legacy Monolith
- **ระบบ AI:** Dual-Path Inference Latency < 200 ms อัตราสำเร็จ $\approx$99.9% จำนวน 694 การคัดกรองที่สำเร็จ
- **ความปลอดภัย:** กรอบ PDPA/HIPAA พร้อม Tamper-Proof Audit Logging และ RBAC ครอบคลุม 6 บทบาท
- **DevOps:** Docker Containerization ครบวงจร พร้อม CI/CD Pipeline และการแยก Environment เข้มงวด
- **คุณภาพ:** 900 Automated Test ผ่าน 100% Coverage 100% บน Critical Business Logic
- **ความร่วมมือ:** หลายสถาบัน ครอบคลุมทันตแพทยศาสตร์ วิศวกรรมศาสตร์ และสาธารณสุขชุมชน

แพลตฟอร์มพร้อม Deploy ระดับนำร่องในเชียงใหม่ พะเยา ภูเก็ต และหนองบัวลำภู เพื่อช่วยให้ตรวจพบมะเร็งช่องปากในผู้สูงอายุชนบทเร็วขึ้น ลดการวินิจฉัยระยะท้าย และพัฒนาผลลัพธ์ผู้ป่วยในวงกว้าง

โครงการนี้ประสบความสำเร็จในการแก้ปัญหาคอขวดและจุดล้มเหลวของระบบเดิม เปลี่ยนผ่านสู่ **แพลตฟอร์มทางการแพทย์พร้อมใช้งานจริง (Production-Ready)** ด้วยสถาปัตยกรรมที่เสถียร ปลอดภัย และสมบูรณ์ระดับ Enterprise

งานในอนาคต (Future Work):

- ขยาย AI Model ครอบคลุมรอยโรคช่องปากเพิ่มเติม และปรับปรุงความแม่นยำด้วย Dataset Annotation จากพื้นที่นำร่อง
- ขยายสู่จังหวัดเพิ่มเติมและเชื่อมต่อกับระบบสารสนเทศสุขภาพ (HIS) ของไทย
- ดำเนินการศึกษา Clinical Validation อย่างเป็นทางการ วัด Stage-Shift และเวลาพบผู้เชี่ยวชาญในกลุ่มนำร่อง

- สำรวจการเชื่อมต่อ LINE OA และ PWA Offline Mode สำหรับชุมชนที่มีอินเทอร์เน็ตจำกัด