

แอปพลิเคชันสแกนฉลากยาและแจ้งเตือนการทานยา

โดย

นางสาว	ปรารธนา	สุภาวงศ์	6587057
นาย	ฤทธิช	พลราช	6587062
นางสาว	จิรัชญา	ราชพลแสน	6587074

อาจารย์ที่ปรึกษา
ดร.สนธิ์ แสงเหลา

โครงการนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

วิทยาศาสตร์บัณฑิต
(สาขาวิทยาการและเทคโนโลยีดิจิทัล)

คณะเทคโนโลยีสารสนเทศและการสื่อสาร
มหาวิทยาลัยมหิดล
2025

ลิขสิทธิ์ของมหาวิทยาลัยมหิดล

กิตติกรรมประกาศ

โครงการนี้สำเร็จลุล่วงได้ด้วยความร่วมมือและความกรุณาจากอาจารย์ที่ปรึกษา ดร.สนธิ แสงเหลา ซึ่งได้ให้คำแนะนำและแนวทางในการพัฒนาโครงการอย่างใกล้ชิด ตลอดจนให้คำปรึกษาและแก้ไขข้อบกพร่องต่าง ๆ มาโดยตลอด จนทำให้โครงการเล่มนี้เสร็จสมบูรณ์ ผู้จัดทำจึงขอกราบขอบพระคุณเป็นอย่างสูง

ขอขอบคุณผู้ร่วมทดสอบระบบทุกท่านที่ได้สละเวลาอันมีค่าในการทดสอบและให้ข้อเสนอแนะที่เป็นประโยชน์ ซึ่งช่วยให้ระบบมีความสมบูรณ์และตอบสนองความต้องการของผู้ใช้งานได้ดียิ่งขึ้น

ขอขอบคุณสำนักพัฒนามาตรฐานระบบข้อมูลสุขภาพไทยที่ได้เอื้อเพื่อข้อมูลจากระบบ Thai Medicine Terminology (TMT) ซึ่งเป็นแหล่งข้อมูลด้านยาที่มีความน่าเชื่อถือ และสำนักงานคณะกรรมการอาหารและยา (อย.) ที่ได้ให้บริการเผยแพร่ข้อมูลยาผ่านเว็บไซต์ เพื่อให้สามารถนำมาศึกษาและประยุกต์ใช้ในการพัฒนาและทดสอบระบบของโครงการนี้ได้อย่างมีประสิทธิภาพ

สุดท้ายนี้ ขอขอบคุณครอบครัวและผู้ปกครองที่ให้การสนับสนุน ให้กำลังใจ และเป็นแรงผลักดันให้ผู้จัดทำสามารถดำเนินโครงการนี้จนสำเร็จลุล่วงได้ด้วยดี

นางสาว ประรณนา สุภาวงศ์

นาย ฤทธิช พลราช

นางสาว จิรัชญา ราชพลแสน

แอปพลิเคชันสแกนฉลากยาและแจ้งเตือนการทานยา

นางสาว ปราวณา สุภาวงศ์ 6587057 ITDS/B

นาย ฤทธิช พลราช 6587062 ITDS/B

นางสาว จิรัชญา ราชพลแสน 6587074 ITDS/B

วท.บ. (วิทยาการและเทคโนโลยีดิจิทัล)

อาจารย์ที่ปรึกษาโครงการ: ดร.สนิท แสงเหลา

บทคัดย่อ

โครงการนี้จัดทำแอปพลิเคชันสำหรับสแกนฉลากยาและการแจ้งเตือนการทานยา โดยมีชื่อแอปพลิเคชันว่า CapYaDoo ซึ่งเป็นแอปพลิเคชันบนโทรศัพท์มือถือระบบปฏิบัติการ Android ที่ออกแบบมาเพื่อช่วยเหลือผู้สูงอายุในการจัดการยา โดยระบบสามารถ สแกนฉลากยา หรือ บาร์โค้ด เพื่อค้นหาข้อมูลยาได้อย่างสะดวกและรวดเร็ว ผู้ใช้งานสามารถ บันทึกยาที่กำลังใช้อยู่, จัดบันทึกอาการหลังการรับประทานยา และ ตั้งการแจ้งเตือนเพื่อเตือนเวลาทานยา ได้ภายในแอปพลิเคชันเดียว

ระบบพัฒนาด้วย Flutter สำหรับฝั่ง Mobile Application และ Spring Boot ร่วมกับ PostgreSQL สำหรับฝั่งเซิร์ฟเวอร์ มีการประมวลผลข้อมูลจากสำนักพัฒนามาตรฐานระบบข้อมูลสุขภาพไทย ผ่านระบบ Thai Medicine Terminology (TMT) และ ข้อมูลจากสำนักงานคณะกรรมการอาหารและยา (อย.) ซึ่งเป็นแหล่งข้อมูลเปิดที่ให้บริการรายละเอียดเกี่ยวกับยา โดยใช้ PySpark สำหรับจัดการและเตรียมข้อมูลก่อนนำเข้าฐานข้อมูล และทำการ Deploy ระบบโดยใช้ Kind ซึ่งเป็นเครื่องมือจำลอง Kubernetes บนเครื่อง เพื่อจัดการสภาพแวดล้อม Container

คำสำคัญ: Application /Framework /API /RESTful API /Text-to-Speech /Notification /Optical Character Recognition - OCR /Frontend /Backend

สารบัญ

หน้า

กิตติกรรมประกาศ	ii
บทคัดย่อ	iii
LIST OF FIGURES	vii
LIST OF TABLES	viii
1 บทนำ	1
1.1 ที่มาและความสำคัญของปัญหา.....	1
1.2 วัตถุประสงค์	1
1.3 ขอบเขตงาน	2
1.3.1 ด้านฟังก์ชันการทำงาน.....	2
1.3.2 ด้านข้อมูล	2
1.3.3 ด้านเทคนิค	2
1.3.4 เครื่องมือเสริม	3
1.4 ประโยชน์ที่คาดว่าจะได้รับ.....	3
2 ทฤษฎีที่เกี่ยวข้อง	4
2.1 สังคมสูงวัย (AGED SOCIETY และ SUPER-AGED SOCIETY)	4
2.2 โรคไม่ติดต่อเรื้อรัง (NON-COMMUNICABLE DISEASES: NCDs).....	4
2.3 การออกแบบระบบที่เหมาะสมกับผู้สูงอายุ (HUMAN-COMPUTER INTERACTION: HCI)	4
2.3.1 การออกแบบเพื่อการเข้าถึงผู้สูงอายุ (ACCESSIBILITY).....	4
2.3.2 สุขทรียภาพและประสบการณ์ผู้ใช้ (USABILITY & AESTHETICS)	4
2.3.3 การออกแบบเพื่อความเข้าใจและประสบการณ์ผู้ใช้ (COGNITIVE & USER EXPERIENCE)	5
2.4 พื้นฐานและการออกแบบฐานข้อมูล	5
2.5 OPTICAL CHARACTER RECOGNITION (OCR).....	5
2.6 POSTGRESQL	6
2.6.1 ประโยชน์ของ POSTGRESQL	6
2.7 สถาปัตยกรรมแบบ CLIENT-SERVER (CLIENT-SERVER ARCHITECTURE)	6
2.7.1 องค์ประกอบหลัก	6
2.7.2 ข้อดีของสถาปัตยกรรม CLIENT-SERVER	7

2.8	SPRING BOOT FRAMEWORK	7
2.8.1	คุณสมบัติหลัก	7
2.9	FLUTTER FRAMEWORK	7
2.9.1	ข้อดีของ FLUTTER	8
2.9.2	FLUTTER TEXT-TO-SPEECH (TTS) PLUGIN.....	8
2.9.3	FLUTTER_LOCAL_NOTIFICATIONS.....	9
2.9.4	GOOGLE_MLKIT_TEXT_RECOGNITION	9
2.9.5	MOBILE_SCANNER PLUGIN	9
2.10	JAVA.....	9
2.10.1	คุณสมบัติเด่นของ JAVA	10
2.11	DART.....	10
2.11.1	คุณสมบัติเด่นของ DART	10
2.12	AGILE METHODOLOGY.....	11
2.12.1	หลักการของ AGILE	11
2.12.2	ลักษณะการทำงานของ AGILE.....	11
2.12.3	ประโยชน์ของการใช้ AGILE	11
2.13	คำศัพท์.....	12
3	ผลงานที่เกี่ยวข้อง	13
3.1	แอปพลิเคชันช่วยอ่านฉลากยาโดยเฉพาะฟังก์ชันที่ช่วยให้อ่านข้อมูลยาได้ง่ายขึ้นอย่างชัดเจน	13
3.2	แอปพลิเคชันบน LINE LIFF ร่วมกับ GOOGLE APPS SCRIPT สำหรับติดตามและประเมินผลความสำเร็จในการรับประทานยาของผู้ป่วยวัณโรค โรงพยาบาลชัชวาลนาทนคร.....	13
3.3	โครงการวิจัยเรื่อง เทคโนโลยีเภสัชสารสนเทศแสดงข้อมูลฉลากยาเอกสารกำกับยาแบบอัตโนมัติสำหรับบริหารจัดการคลังยาปฏิชีวนะเพื่อความปลอดภัย	14
3.4	การเปรียบเทียบแอปพลิเคชันที่มีอยู่ในตลาด	15
3.4.1	ตารางเปรียบเทียบแอปพลิเคชันในตลาด	15
3.4.2	สรุปการเปรียบเทียบ	17
3.5	ประสบการณ์การพัฒนาที่ผ่านมา	17
3.5.1	PROJECT 1: APPLICATION CHAT MUSIC PLAYER.....	17
3.5.2	PROJECT 2: APPLICATION LOCATION DISCOVERY.....	17
3.5.3	บทเรียนที่ได้จากประสบการณ์การพัฒนา.....	18
4	ระเบียบวิธีดำเนินงาน	19
4.1	การวิเคราะห์และออกแบบระบบ	19

4.1.1	PERSONA.....	19
4.1.2	USER JOURNEY MAP.....	21
4.1.3	ปัญหาและแนวทางการแก้ไข	22
4.1.4	การวิเคราะห์ความต้องการ (REQUIREMENT ANALYSIS)	22
4.1.5	การออกแบบสถาปัตยกรรม (SYSTEM ARCHITECTURE DESIGN).....	23
4.1.6	ขั้นตอนการออกแบบระบบ	24
4.1.7	การพัฒนา UX/UI:	30
4.2	ขั้นตอนการพัฒนา	33
4.2.1	ขั้นตอนที่ 1: การรวบรวมปัญหา วิเคราะห์ความต้องการของผู้ใช้งานและการออกแบบ	33
4.2.2	ขั้นตอนที่ 2: การศึกษาและพัฒนา DATA PIPELINE ด้วย PYSPARK.....	33
4.2.3	ขั้นตอนที่ 3: การจัดการ CONTAINER ด้วย KUBERNETES IN DOCKER (KIND)	33
4.2.4	ขั้นตอนที่ 4: พัฒนาระบบในส่วนของหน้าบ้าน (FRONTEND).....	34
4.2.5	ขั้นตอนที่ 5: การพัฒนาระบบในส่วนของหลังบ้าน (BACKEND)	34
4.2.6	ขั้นตอนที่ 6: การทดสอบและปรับปรุงระบบ	34
4.2.7	เครื่องมือและสภาพแวดล้อมในการพัฒนา	35
4.3	การทดสอบและการประเมินผล.....	35
4.3.1	การทดสอบระบบ (SYSTEM TESTING).....	35
4.3.2	การทดสอบประสิทธิภาพ (PERFORMANCE TESTING)	35
4.3.3	การทดสอบการยอมรับจากผู้ใช้งาน (USER ACCEPTANCE TESTING - UAT)	35
4.4	ผลลัพธ์เบื้องต้นจากการพัฒนา.....	35
4.4.1	การทดสอบระบบเบื้องต้นในสภาพแวดล้อม CONTAINER ด้วย KIND	35
4.4.2	กระบวนการ DATA PIPELINE	36
4.4.3	BACKEND	36
	เอกสารอ้างอิง	37
	ข้อความเปิดเผยการใช้เทคโนโลยีปัญญาประดิษฐ์	39

LIST OF FIGURES

	Page
Figure 4.1: ตัวตนสมมติที่เป็นตัวแทนของกลุ่มผู้ใช้เป้าหมายคนที่ 1	19
Figure 4.2: ตัวตนสมมติที่เป็นตัวแทนของกลุ่มผู้ใช้เป้าหมายคนที่ 2	20
Figure 4.3: ตัวตนสมมติที่เป็นตัวแทนของกลุ่มผู้ใช้เป้าหมายคนที่ 3	20
Figure 4.4: ประสบการณ์ของผู้ใช้เมื่อมีปฏิสัมพันธ์กับแอปพลิเคชันของเราคนที่ 1	21
Figure 4.5: ประสบการณ์ของผู้ใช้เมื่อมีปฏิสัมพันธ์กับแอปพลิเคชันของเราคนที่ 2	21
Figure 4.6: ประสบการณ์ของผู้ใช้เมื่อมีปฏิสัมพันธ์กับแอปพลิเคชันของเราคนที่ 3	22
Figure 4.7: System Architecture Design	24
Figure 4.8: Entity Relationship Diagram (ERD)	25
Figure 4.9: Functional Decomposition	26
Figure 4.10: Data Flow Diagram Level 0	26
Figure 4.11: Data Flow Diagram Level 1	27
Figure 4.12: Data Flow Diagram Level 2.1 - Manage User Information	28
Figure 4.13: Data Flow Diagram Level 2.2 - OCR/ Barcode/ Text-To-Speech	28
Figure 4.14: Data Flow Diagram Level 2.3 - Medication Management	29
Figure 4.15: Data Flow Diagram Level 2.4 - Notification Management	29
Figure 4.16: Data Flow Diagram Level 2.5 - History Logging	30
Figure 4.17: หน้าการจัดการข้อมูลผู้ใช้งาน	30
Figure 4.18: หน้าสำหรับการจัดการข้อมูลยา	31
Figure 4.19: หน้าสำหรับการจัดการข้อมูลอาการหลังการใช้ยา	31
Figure 4.20: หน้าสำหรับการจัดการตั้งค่าการแจ้งเตือน	31
Figure 4.21: หน้าสำหรับการค้นหาข้อมูลยา	32
Figure 4.22: หน้าสำหรับการถ่ายภาพยาเพื่อยืนยันการทานยา การสแกนบาร์โค้ด และการแจ้งเตือนการทานยา	32
Figure 4.23: หน้าสำหรับการเข้าสู่ระบบ	33
Figure 4.24: แผนการดำเนินโครงการ	35
Figure 4.25: รูปผลลัพธ์ container ที่สร้างขึ้นจาก Kind	36
Figure 4.26: รูปผลลัพธ์จากฐานข้อมูล PostgreSQL ที่ได้จากการประมวลผลข้อมูลด้วย PySpark	36
Figure 4.27: รูปผลลัพธ์ข้อมูลจาก API ที่ได้พัฒนาด้วย Spring Boot	36

LIST OF TABLES

Page

Table 3.1:	เปรียบเทียบแอปพลิเคชันในตลาด	16
------------	------------------------------------	----

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญของปัญหา

ประเทศไทยกำลังเผชิญหน้ากับการเปลี่ยนแปลงทางประชากรครั้งใหญ่ โดยกำลังก้าวเข้าสู่ “สังคมสูงวัยระดับสุดยอด” (Super-Aged Society) ซึ่งเป็นผลมาจากสัดส่วนประชากรผู้สูงอายุที่เพิ่มขึ้นอย่างรวดเร็ว ข้อมูลระบุว่าในปี พ.ศ. 2548 ไทยได้เข้าสู่ “สังคมสูงวัย” (Aged Society) ที่มีประชากรอายุเกิน 60 ปี มากกว่าร้อยละ 10 ของประชากรทั้งหมด และในปี พ.ศ. 2566 ได้ก้าวเข้าสู่ “สังคมสูงวัยอย่างสมบูรณ์” (Complete Aged Society) ด้วยสัดส่วนประชากรกลุ่มนี้ที่เพิ่มขึ้นมากกว่าร้อยละ 20 และมีแนวโน้มที่จะเข้าสู่ “สังคมสูงวัยระดับสุดยอด” (Super-Aged Society) ในปี พ.ศ. 2576 ซึ่งจะมีประชากรสูงอายุเกิน 60 ปี มากกว่าร้อยละ 28 หรือมีผู้สูงอายุเกิน 65 ปี มากกว่าร้อยละ 20 ของประชากรทั้งหมด [1]

การเข้าสู่สังคมสูงอายุอย่างรวดเร็วของประเทศไทยส่งผลให้เกิดความท้าทายด้านสุขภาพอย่างหลีกเลี่ยงไม่ได้ โดยเฉพาะอย่างยิ่งในเรื่องของ กลุ่มโรคไม่ติดต่อเรื้อรัง (Non-Communicable Diseases หรือ NCDs) ซึ่งเป็นสาเหตุการเสียชีวิตอันดับต้น ๆ ของประชากรทั่วโลก ปัจจัยเสี่ยงสำคัญ ได้แก่ การสูบบุหรี่ การไม่ออกกำลังกาย การรับประทานอาหารไม่เหมาะสม และการดื่มแอลกอฮอล์มากเกินไป ซึ่งนำไปสู่โรคหัวใจ มะเร็ง โรคระบบหายใจเรื้อรัง เบาหวาน และความดันโลหิตสูง [2]

ผู้สูงอายุที่ป่วยด้วยโรคไม่ติดต่อเรื้อรัง (NCDs) มักต้องจัดการการใช้ยาที่ซับซ้อน เช่น การจดจำชนิดของยา เวลารับประทาน และผลข้างเคียง การลืมทานยาหรือทานผิดเวลาพบได้บ่อย และส่งผลต่อประสิทธิภาพการรักษา ขณะเดียวกัน แอปพลิเคชันสุขภาพทั่วไปยังไม่เหมาะสมกับการใช้งานของผู้สูงอายุ ในเรื่องของแอปพลิเคชันที่มีฟังก์ชันที่เกี่ยวข้องกับการดูแลสุขภาพ บันทึกอาการ และการแจ้งเตือนการรับประทานยามักถูกแยกออกจากกันในคนละแอปพลิเคชัน ทำให้ผู้ใช้ต้องสลับการใช้งานหลายระบบ ส่งผลให้เกิดความยุ่งยากและเพิ่มโอกาสในการลืมหรือใช้งานผิดพลาด

จากความท้าทายเหล่านี้ การพัฒนาแอปพลิเคชันที่สามารถแก้ไขปัญหาดังกล่าวได้จึงมีความสำคัญอย่างยิ่ง แอปพลิเคชันนี้จะทำหน้าที่เป็นเครื่องมือที่ใช้งานง่าย มีข้อมูลที่น่าเชื่อถือ และมีฟังก์ชันที่ช่วยลดความผิดพลาดในการใช้ยา เพื่อช่วยให้ผู้สูงอายุสามารถจัดการสุขภาพได้อย่างมีประสิทธิภาพและปลอดภัยสูงสุดในยุคที่ประเทศไทยกำลังก้าวเข้าสู่สังคมสูงวัยอย่างเต็มตัว

1.2 วัตถุประสงค์

- เพื่อให้ผู้ใช้สามารถค้นหาข้อมูลยาได้อย่างรวดเร็วและถูกต้อง ผ่านการค้นหาด้วยข้อความ การสแกนบาร์โค้ด หรือการอ่านข้อความบนฉลากยา (OCR)
- เพื่อเชื่อมโยงข้อมูลยาจากฐานข้อมูลที่เชื่อถือได้ ได้แก่ ระบบ Thai Medicine Terminology (TMT) และข้อมูลจาก สำนักงานคณะกรรมการอาหารและยา (อย.) เพื่อให้ผู้ใช้งานได้รับข้อมูลที่ถูกต้องและครบถ้วน

- เพื่อให้ผู้ใช้สามารถติดตามอาการหลังการใช้จ่ายได้ ผ่านฟังก์ชันการบันทึกอาการหลังการใช้จ่าย
- เพื่อส่งเสริมความเข้าใจด้านข้อมูลยาแก่ผู้ใช้ จากระบบเสียงอ่านข้อมูลยา (Text-to-Speech)
- เพื่อพัฒนาแอปพลิเคชันที่ช่วยให้ผู้ใช้สามารถจัดการข้อมูลยาที่รับประทานได้อย่างมีประสิทธิภาพ

1.3 ขอบเขตงาน

1.3.1 ด้านฟังก์ชันการทำงาน

- ผู้ใช้งานสามารถเพิ่ม, แก้ไข และลบข้อมูลยาที่ทานได้
- ผู้ใช้งานสามารถเพิ่ม, แก้ไข และลบข้อมูลอาการหลังทานยาได้
- ผู้ใช้งานสามารถค้นหาข้อมูลยาผ่านการสแกนบาร์โค้ด, สแกนฉลากยา และกรอกข้อมูลยา เพื่อดูรายละเอียดเพิ่มเติมได้
- ผู้ใช้งานสามารถใช้ฟังก์ชันการแปลงข้อความเป็นเสียงเพื่อเข้าถึงข้อมูลยาที่ผ่านการค้นหาได้
- ผู้ใช้งานสามารถตั้งแจ้งเตือนการทานยาตามเวลาที่ต้องการได้
- ผู้ใช้งานสามารถเพิ่มรูปหลังทานเพื่อยืนยันการทานยาได้

1.3.2 ด้านข้อมูล

- รวบรวมข้อมูลยาจากระบบ Thai Medicine Terminology (TMT) และสำนักงานคณะกรรมการอาหารและยา
- จัดเก็บข้อมูลการทานยาของผู้ใช้งานอย่างปลอดภัย
- จัดทำฐานข้อมูลที่มีความสอดคล้องกันของข้อมูลและมีความปลอดภัย

1.3.3 ด้านเทคนิค

- พัฒนาด้วยภาษา Dart และเฟรมเวิร์ก Flutter
- พัฒนาด้วยภาษา Java และเฟรมเวิร์ก Spring Boot
- ใช้ฐานข้อมูล PostgreSQL สำหรับจัดเก็บข้อมูล
- ใช้ PySpark ในการจัดการข้อมูลจากระบบ Thai Medicine Terminology (TMT) และจากเว็บไซต์ของทางสำนักงานคณะกรรมการอาหารและยาก่อนเก็บลงฐานข้อมูล
- ใช้ Kind เป็นเครื่องมือจำลอง Kubernetes บนเครื่องพัฒนา (Local Environment) เพื่อให้สามารถ deploy และจัดการหลาย container ได้ โดยมี container ดังนี้ Spring Boot Application, PySpark และ ฐานข้อมูล PostgreSQL

1.3.4 เครื่องมือเสริม

- google_mlkit_text_recognition สำหรับอ่านข้อความบนฉลากยา
- mobile_scanner สำหรับสแกนบาร์โค้ด
- flutter_tts สำหรับทำ Text-to-Speech
- flutter_local_notifications สำหรับแจ้งเตือน

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1. ทำให้เป็นแอปพลิเคชันที่มีฟังก์ชันที่จำเป็นต่อการจัดการปัญหาด้านการทานยาอย่างครบถ้วน โดยรวมฟังก์ชันการค้นหายา แจ้งเตือนการทานยา และบันทึกอาการหลังการใช้อย่างมาไว้ในแอปพลิเคชันเดียว ทำให้ผู้ใช้ไม่ต้องใช้งานหลายแอปพลิเคชันเพื่อจัดการปัญหาด้านการทานยา
2. ผู้ใช้สามารถค้นหาข้อมูลยาได้อย่างรวดเร็วผ่านการค้นหาด้วยข้อความ การสแกนบาร์โค้ด หรือการอ่านข้อความบนฉลากยา (OCR)
3. ผู้ใช้สามารถรับฟังข้อมูลยาในรูปแบบเสียงได้ ช่วยให้ผู้ใช้สามารถเข้าใจข้อมูลยาได้อย่างง่ายขึ้น
4. ระบบแจ้งเตือนการทานยาตามเวลาที่กำหนดอย่างแม่นยำ ช่วยลดปัญหาการลืมทานยา การทานยาซ้ำ หรือทานยาไม่ตรงเวลา
5. แอปพลิเคชันสามารถเก็บประวัติการใช้จ่ายและการแจ้งเตือนต่าง ๆ ในเครื่อง ทำให้ผู้ใช้สามารถเรียกดูย้อนหลังได้

บทที่ 2

ทฤษฎีที่เกี่ยวข้อง

2.1 สังคมสูงวัย (Aged Society และ Super-Aged Society)

คาดว่าประมาณปี พ.ศ. 2576 ประเทศไทยจะก้าวเข้าสู่ สังคมสูงวัยระดับสุดยอด (Super-Aged Society) ซึ่งมีประชากรอายุเกิน 60 ปี มากกว่าร้อยละ 28 ของประชากรทั้งหมด การเปลี่ยนแปลงนี้ทำให้เกิดปัญหาด้านสุขภาพและความต้องการเทคโนโลยีเพื่อช่วยเหลือการดูแลผู้สูงอายุ โดยเฉพาะในเรื่อง การจัดการการใช้ยา ซึ่งมีความซับซ้อนและมีความเสี่ยงต่อการใช้ยาผิดพลาด

2.2 โรคไม่ติดต่อเรื้อรัง (Non-Communicable Diseases: NCDs)

ผู้สูงอายุจำนวนมากป่วยด้วยโรคเรื้อรัง เช่น ความดันโลหิตสูง เบาหวาน โรคหัวใจ ซึ่งต้องใช้ยาหลายชนิดต่อเนื่องเป็นเวลานาน การลืมทานยาหรือทานผิดเวลาอาจส่งผลกระทบต่อประสิทธิภาพการรักษา และก่อให้เกิดภาวะแทรกซ้อน การพัฒนาแอปพลิเคชันที่ช่วยแจ้งเตือนและบันทึกประวัติการใช้ยา จึงตอบโจทย์การจัดการโรคเรื้อรังอย่างมีระบบ

2.3 การออกแบบระบบที่เหมาะสมกับผู้สูงอายุ (Human-Computer Interaction: HCI)

แนวทางการออกแบบส่วนประสานผู้ใช้งานกับผู้สูงอายุไทยในยุคดิจิทัล [3] ได้แก่

2.3.1 การออกแบบเพื่อการเข้าถึงผู้สูงอายุ (Accessibility)

- แนวคิด Universal Design / อารยสถาปัตย์: ออกแบบให้ทุกคนเข้าถึงและใช้งานได้เท่าเทียม รวมถึงผู้สูงอายุที่มีความเสื่อมสภาพทางร่างกายและประสาทสัมผัส
- การออกแบบเฉพาะผู้สูงอายุ: คำนึงถึงปัญหาสุขภาพและข้อจำกัดด้านการมองเห็น การได้ยิน การเคลื่อนไหว รวมถึงโรคประจำตัวที่ส่งผลกระทบต่อการใช้งาน

2.3.2 สุขทรีียภาพและประสบการณ์ผู้ใช้ (Usability & Aesthetics)

- ความสุขทางประสาทสัมผัส (Sensory Experience): การออกแบบส่วนประสานผู้ใช้งานต้องสอดคล้องกับการรับรู้ของผู้สูงอายุ เช่น การเลือกขนาดตัวอักษร สี เสียง และรูปทรงที่เหมาะสม
- แนวคิดหวนระลึกอดีต (Nostalgia): การเชื่อมโยงกับความทรงจำหรือประสบการณ์ในช่วงวัยหนุ่มสาวช่วยให้จดจำและใช้งานได้ง่ายขึ้น
- ความสวยงามและกราฟิก (GUI): เลือกสไตล์ที่ผู้สูงอายุตอบสนองดี เช่น หน้าที่เรียบง่าย สบายตา ใช้สีโทนอ่อนหรือคมชัดตาม

ความเหมาะสม รวมถึงการใช้ภาพเหมือนจริง รูปแบบสเกียมอร์ฟิซึม หรือภาพการ์ตูน 2D/3D ตามความพึงพอใจ

2.3.3 การออกแบบเพื่อความเข้าใจและประสบการณ์ผู้ใช้ (Cognitive & User Experience)

- สัญลักษณ์และไอคอน: เลือกสัญลักษณ์ที่รับรู้และเข้าใจได้ทันที
- การออกแบบเนื้อหา: ต้องจัดเนื้อหาให้เกี่ยวข้องกับชีวิตประจำวัน แบ่งหัวข้อชัดเจน เช่น สุขภาพ การดำรงชีวิต จิตใจ และความสัมพันธ์

2.4 พื้นฐานและการออกแบบฐานข้อมูล

การออกแบบฐานข้อมูลที่ดี [4] มีความสำคัญอย่างยิ่งต่อความถูกต้องและความสมบูรณ์ของข้อมูล และช่วยให้สามารถเรียกใช้หรือปรับปรุงข้อมูลได้อย่างมีประสิทธิภาพ โดยหลักการออกแบบฐานข้อมูลที่ดี มีดังนี้

1. ลดความซ้ำซ้อนของข้อมูล (Data Redundancy)
2. เพิ่มความถูกต้องสมบูรณ์ของข้อมูล (Data Integrity)
3. สามารถปรับปรุงหรือเปลี่ยนแปลงโครงสร้างได้ง่าย

2.5 Optical Character Recognition (OCR)

การรู้จำอักขระด้วยแสง [5] คือเทคโนโลยีที่ใช้ในการแปลงข้อความจากรูปแบบภาพ (เช่น รูปถ่าย, ภาพสแกน, ไฟล์ PDF ที่เป็นภาพ) ให้กลายเป็นข้อความดิจิทัลที่คอมพิวเตอร์สามารถอ่าน, แก้ไข, ค้นหา หรือนำไปใช้ประมวลผลต่อได้

ประเภทของ OCR

1. ซอฟต์แวร์รู้จำอักขระด้วยแสงอย่างง่าย : ใช้อัลกอริทึมการจับคู่รูปแบบเพื่อเปรียบเทียบรูปภาพข้อความกับฐานข้อมูลภายในแบบอักขระทีละตัว
2. ซอฟต์แวร์รู้จำอักขระแบบอัจฉริยะ (Intelligent Character Recognition: ICR) : ใช้ Machine Learning และ Neural Network เพื่ออ่านลายมือหรือตัวอักษรที่ซับซ้อน
3. การรู้จำคำแบบอัจฉริยะ (Intelligent Word Recognition) : วิเคราะห์คำทั้งคำแทนการประมวลผลทีละตัวอักษร
4. การรู้จำเครื่องหมายด้วยแสง (Optical Mark Recognition: OMR) : ใช้ระบุสัญลักษณ์หรือเครื่องหมาย เช่น โลโก้ หรือตัวเลือกในแบบฟอร์มสอบ

ข้อดีของ OCR

1. แปลงเอกสารกระดาษเป็นข้อมูลดิจิทัลที่สามารถค้นหาได้
2. สามารถประมวลผลข้อมูลตัวอักษรโดยอัตโนมัติ เพื่อใช้ในการวิเคราะห์และประมวลผลความรู้เพิ่มเติม

3. ลดเวลาการประมวลผลเอกสารและการป้อนข้อมูลด้วยตนเอง
4. OCR มักถูกรวมเข้ากับเทคโนโลยี AI เพื่อประยุกต์ใช้งานขั้นสูง เช่น อ่านป้ายทะเบียนและป้ายจราจรในรถยนต์ไร้คนขับ (OCR การรู้จำอักขระด้วยแสง) คืออะไร โดย AWS)

2.6 PostgreSQL

PostgreSQL [4] เป็นระบบจัดการฐานข้อมูลเชิงสัมพันธ์ (Relational Database Management System: RDBMS) ระดับองค์กรที่มีความเสถียรและเชื่อถือได้สูง โดยรองรับการทำงานแบบ ACID (Atomicity, Consistency, Isolation, Durability) อย่างสมบูรณ์ ซึ่งหมายถึงคุณสมบัติพื้นฐานที่ทำให้การทำธุรกรรม (Transaction) ในฐานข้อมูลมีความถูกต้องและปลอดภัย ได้แก่:

1. Atomicity คือ การทำธุรกรรม จะต้องสำเร็จทั้งหมดหรือไม่สำเร็จเลย หากเกิดข้อผิดพลาด ระบบจะยกเลิก (Rollback) การทำงานทั้งหมดเพื่อให้ข้อมูลคงสภาพเดิม
2. Consistency คือ หลังการทำธุรกรรม ข้อมูลในฐานข้อมูลจะต้องยังคงถูกต้องตามกฎและข้อจำกัด (Constraints) ที่กำหนดไว้เสมอ
3. Isolation คือ การทำธุรกรรมหลายรายการที่เกิดขึ้นพร้อมกันจะไม่ส่งผลกระทบต่อกัน ทำให้ผลลัพธ์เทียบเท่ากับการทำงานแบบลำดับ (Serial Execution)
4. Durability คือ เมื่อทำธุรกรรมเสร็จสิ้น ข้อมูลที่เปลี่ยนแปลงแล้วจะถูกบันทึกถาวรในระบบ แม้ระบบจะปิดตัวหรือเกิดความขัดข้อง

2.6.1 ประโยชน์ของ PostgreSQL

1. สามารถใช้งานได้โดยไม่จำเป็นต้องชำระเงินและเป็น Open Source ไม่มีค่าใช้จ่ายด้านลิขสิทธิ์
2. มีความน่าเชื่อถือสูง รองรับการทำงานต่อเนื่อง (Fault Tolerant)
3. รองรับหลายภาษาโปรแกรม เช่น Python, Java, C/C++, PHP, Ruby, Perl, Erlang, Lua, Scala, Haskell, etc.
4. รองรับการทำงานข้ามแพลตฟอร์ม

2.7 สถาปัตยกรรมแบบ Client-Server (Client-Server Architecture)

สถาปัตยกรรมแบบ Client-Server เป็นรูปแบบการออกแบบระบบคอมพิวเตอร์ที่แบ่งการทำงานออกเป็น 2 ส่วนหลัก คือ Client (ฝั่งผู้ใช้) และ Server (ฝั่งเซิร์ฟเวอร์) โดยทั้งสองส่วนจะสื่อสารกันผ่านเครือข่าย (Network) เพื่อแลกเปลี่ยนข้อมูลและประมวลผลงานร่วมกัน

2.7.1 องค์ประกอบหลัก

Client คือแอปพลิเคชันหรือโปรแกรมที่ทำงานบนอุปกรณ์ของผู้ใช้ เช่น มือถือ คอมพิวเตอร์ หรือแท็บเล็ต

Server คือระบบที่ทำงานบนเครื่องแม่ข่าย มีหน้าที่รับคำขอจาก Client และประมวลผลตามคำขอนั้น

2.7.2 ข้อดีของสถาปัตยกรรม Client-Server

1. การแยกหน้าที่ชัดเจน โดย Client รับผิดชอบส่วนแสดงผล ขณะที่ Server จัดการข้อมูลและตรรกะทางธุรกิจ
2. ข้อมูลสำคัญและตรรกะทางธุรกิจถูกเก็บไว้ที่ Server ไม่ถูกเปิดเผยบน Client ทำให้มีความปลอดภัย
3. สามารถอัปเดต Server โดยไม่ต้องแก้ไข Client และในทางกลับกันเมื่อแก้ไข Client ก็ไม่จำเป็นต้องแก้ไข Server ทำให้ง่ายต่อการบำรุงรักษา
4. รองรับผู้ใช้หลายคน โดย Server สามารถให้บริการ Client หลายเครื่องพร้อมกันได้
5. สามารถพัฒนา Client หลายแพลตฟอร์ม (iOS, Android, Web) ที่ใช้ Server เดียวกันทำให้เกิดความยืดหยุ่นในการขยายการทำงานได้ในอนาคต
6. เป็นการจัดการข้อมูลแบบรวมศูนย์ เนื่องจากข้อมูลทั้งหมดถูกจัดเก็บและจัดการที่ Server ทำให้ง่ายต่อการสำรองข้อมูลและการรักษาความปลอดภัย

2.8 Spring Boot Framework

Spring Boot [6] เป็น Framework ที่พัฒนาโดย Pivotal (ปัจจุบันเป็นส่วนหนึ่งของ VMware) สำหรับการสร้างแอปพลิเคชัน Java แบบ standalone ที่พร้อมใช้งานได้ทันที โดย Spring Boot ช่วยลดความซับซ้อนในการตั้งค่าและพัฒนาแอปพลิเคชัน Spring แบบดั้งเดิม ทำให้นักพัฒนาสามารถสร้างแอปพลิเคชันได้รวดเร็วขึ้น

2.8.1 คุณสมบัติหลัก

1. Auto-Configuration: ตั้งค่าอัตโนมัติตาม dependencies ที่มีในโปรเจกต์
2. Embedded Server: มี Web Server ในตัว เช่น Tomcat, Jetty ไม่ต้องติดตั้งแยก
3. Production-Ready: มีเครื่องมือสำหรับ monitoring, health checks, และ metrics
4. Microservices Support: รองรับการพัฒนา Microservices Architecture

2.9 Flutter Framework

Flutter [7] เป็น Framework Open Source เปิดตัวในปี 2018 ที่พัฒนาโดย Google ใช้สร้างอินเทอร์เฟซผู้ใช้ (UI) ของแอปพลิเคชันสำหรับแพลตฟอร์ม iOS, Android, Web, Windows, macOS และ Linux โดยใช้ Codebase เดียวในการสร้างแอปมือถือ

ถือ เว็บ และเดสก์ท็อป (Windows, macOS, Linux) ได้ ซึ่ง Flutter ช่วยลดความซับซ้อนของการสร้าง UI ที่สวยงามและสอดคล้องกัน บนหลายแพลตฟอร์ม

2.9.1 ข้อดีของ Flutter

1. ประสิทธิภาพใกล้เคียงแบบ Native แต่มีค่าใช้จ่ายที่ต่ำกว่า
2. Render UI เร็วและสม่ำเสมอ เนื่องจากใช้ Engine Graphic ของตัวเอง แสดงผล UI สอดคล้องกันข้ามแพลตฟอร์ม
3. รองรับการทำงานขนาน (Parallel Processing)
4. มีการสนับสนุนการพัฒนาโดยทีมงานของ Google ที่สนับสนุนการพัฒนาและรองรับการใช้งาน

2.9.2 Flutter Text-to-Speech (TTS) Plugin

Flutter TTS (flutter_tts) เป็น public package ที่ช่วยให้นักพัฒนาสามารถนำความสามารถในการแปลงข้อความเป็นเสียงพูด (Text-to-Speech: TTS) มาใช้ภายในแอปพลิเคชันได้ โดยรองรับหลายแพลตฟอร์ม ได้แก่ Android, iOS, Web, Windows และ macOS ทำให้เหมาะกับการพัฒนาแอปข้ามแพลตฟอร์มด้วย Flutter

คุณสมบัติหลัก

1. พูดข้อความ (speak)
2. ดึงรายการภาษาที่รองรับ (getLanguages)
3. ตรวจสอบว่าภาษานั้นใช้งานได้หรือไม่ (isLanguageAvailable)
4. กำหนดภาษา (setLanguage)
5. เลือกเสียง (voice) ที่เหมาะสมกับภาษาและสำเนียง
6. ความเร็วในการพูด (setSpeechRate)
7. ระดับเสียง (setVolume)
8. ระดับโทนเสียง (setPitch)

ข้อดี

1. ลดภาระการอ่านข้อความยาว ๆ โดยเฉพาะในเอกสารกำกับยา
2. ช่วยผู้ใช้ที่อ่านภาษาได้ไม่คล่อง แต่สามารถที่จะฟังและเข้าใจ

2.9.3 flutter_local_notifications

เป็นปลั๊กอินของ Flutter ที่ใช้สำหรับ การแสดงผลการแจ้งเตือน (Local Notifications) บนอุปกรณ์ต่าง ๆ โดยไม่ต้องมีการเชื่อมต่ออินเทอร์เน็ต เหมาะสำหรับการแจ้งเตือนที่เกิดขึ้นภายในเครื่อง และสามารถนำมาใช้เพื่อสร้างการแจ้งเตือนที่ซับซ้อน

คุณสมบัติหลัก

1. แสดงการแจ้งเตือนพื้นฐาน (basic notifications)
2. การตั้งเวลาแจ้งเตือน (Scheduled Notification) เช่น แจ้งเตือนรายวัน รายสัปดาห์ หรือแจ้งเตือนซ้ำเป็นช่วงเวลา

2.9.4 google_mlkit_text_recognition

เหมาะสำหรับแอปที่ต้องการ สแกนและแปลงข้อความจากรูปภาพอย่างแม่นยำ, รองรับหลายภาษา, และ ทำงานออฟไลน์

คุณสมบัติเด่น

1. ใช้โมเดล Machine Learning ของ Google ทำให้การอ่านตัวอักษรจากภาพแม่นยำ แม้ภาพไม่ชัดเจนหรือมีเงา
2. มี API ที่เข้าใจง่าย สามารถรวมเข้ากับ Flutter ได้สะดวก
3. ทำงานแบบ On-device ไม่ต้องส่งรูปภาพขึ้นไปประมวลผลบนเซิร์ฟเวอร์ ทำให้มีความรวดเร็ว, ประหยัดข้อมูล, และปลอดภัยต่อข้อมูลส่วนตัว
4. สามารถสแกนข้อความจากกล้องได้ทันที (Live Camera OCR)

2.9.5 mobile_scanner plugin

เป็น Flutter plugin สำหรับสแกน Barcode ด้วยกล้องของอุปกรณ์ โดยรองรับทั้ง Android, iOS, macOS, Web (แต่ไม่รองรับ Linux และ Windows) จุดเด่นของปลั๊กอินนี้คือ ประสิทธิภาพสูง, น้ำหนักเบา, และปรับแต่งได้ง่าย

คุณสมบัติเด่น

1. สแกนบาร์โค้ดได้เร็ว รองรับการตรวจจับแบบ real-time
2. รองรับหลายรูปแบบของบาร์โค้ด เช่น QR Code, Code128, EAN-13 เป็นต้น
3. ปรับแต่งกล้องและตัวสแกนได้ เช่น ความละเอียดของกล้อง, ความเร็วการตรวจจับ, เปิด/ปิดแฟลช, การกลับภาพ, การซูมอัตโนมัติ

2.10 Java

Java [8] เป็นภาษาโปรแกรมมิ่งเชิงวัตถุ (Object-Oriented Programming - OOP) ระดับสูงที่พัฒนาโดย James Gosling และทีมงานที่ Sun Microsystems (ปัจจุบันเป็นส่วนหนึ่งของ Oracle) เปิดตัวครั้งแรกในปี 1995 โดยมีสโลแกนสำคัญคือ "Write Once, Run Anywhere" (WORA)

2.10.1 คุณสมบัติเด่นของ Java

1. เป็นภาษาเชิงวัตถุ (Object-Oriented - OOP)

ซึ่งรวมหลักการสำคัญสี่ข้อคือ Encapsulation, Inheritance, Polymorphism, และ Abstraction การเขียนโปรแกรมแบบ OOP ช่วยให้โค้ดสามารถนำกลับมาใช้ใหม่ได้ง่ายขึ้นและมีความยืดหยุ่นสูง

2. แพลตฟอร์มอิสระ (Platform Independent)

โค้ด Java จะถูกคอมไพล์เป็น Bytecode ซึ่งไม่ได้ผูกติดกับสถาปัตยกรรมของเครื่องใดเครื่องหนึ่ง ไบต์โค้ดนี้จะถูกรันโดย Java Virtual Machine (JVM) ทำให้โค้ดที่เขียนบนระบบหนึ่งสามารถรันบนระบบอื่นได้โดยไม่ต้องแก้ไข

3. ประสิทธิภาพสูง (High Performance)

แม้ว่า Java จะรันบน JVM แต่ก็มีการใช้เทคนิค Just-In-Time (JIT) Compiler เพื่อแปลงไบต์โค้ดที่รันบ่อยครั้งให้เป็นโค้ดเครื่อง (Native Code) ในขณะรัน ทำให้ประสิทธิภาพการทำงานใกล้เคียงกับภาษาที่คอมไพล์โดยตรง

4. Robust

Java เน้นการตรวจสอบข้อผิดพลาดตั้งแต่ขั้นตอนการคอมไพล์ (Compile-Time) และการจัดการกับข้อผิดพลาดที่เกิดขึ้นในขณะรัน (Run-Time) ด้วยกลไก Exception Handling

5. Multithreaded

Java มีคุณสมบัติในการสร้างโปรแกรมที่สามารถรันหลายส่วนพร้อมกัน (Threads) ได้ กลไกนี้จำเป็นอย่างยิ่งสำหรับการพัฒนาแอปพลิเคชันที่มีการตอบสนองสูง (Responsive)

2.11 Dart

Dart [9] เป็นภาษาโปรแกรมที่ออกแบบและพัฒนาโดย Google โดยมีเป้าหมายเพื่อรองรับการพัฒนา Client Application บนหลายแพลตฟอร์ม ทั้ง มือถือ, เว็บ และ เดสก์ท็อป (Windows, macOS, Linux) โดยเน้นที่ความเร็ว, มีประสิทธิภาพ, และง่ายต่อการพัฒนา และยังสามารถในการ คอมไพล์ไปยังหลายแพลตฟอร์ม ได้แก่ Dart Native และ Dart Web

2.11.1 คุณสมบัติเด่นของ Dart

1. Type Safety (ระบบความปลอดภัยของชนิดข้อมูล)

Dart ใช้ Static Type Checking เพื่อตรวจสอบความถูกต้องของชนิดข้อมูลตั้งแต่ตอนคอมไพล์ ทำให้ลดข้อผิดพลาดขณะรันโปรแกรม

2. Null Safety (ระบบป้องกันค่า Null)

Dart มีระบบป้องกัน Null ที่บังคับใช้ได้อย่างเคร่งครัด ตัวแปรที่ไม่ได้ระบุว่าสามารถเป็น Null ได้ จะไม่สามารถเก็บค่า Null ได้เลย ป้องกัน Null Reference Error ซึ่งเป็นปัญหาคลาสสิกในภาษาอื่น ๆ

3. รองรับการเขียนโปรแกรมแบบ Asynchronous

โดยมีคำสำคัญ เช่น async, await, และชนิดข้อมูล Future รวมถึง Stream ที่ช่วยให้นักพัฒนาจัดการงานแบบไม่ประสานเวลา (เช่น ดึงข้อมูลจาก API หรือทำงาน I/O) ได้อย่างสะดวก

2.12 Agile methodology

การทำงานแบบ Agile [10] คือแนวทางการบริหารโครงการและการพัฒนาซอฟต์แวร์ที่เน้น การทำงานแบบยืดหยุ่น ปรับตัวได้เร็ว และมีการส่งมอบอย่างต่อเนื่อง แทนที่จะรอทำงานทั้งหมดเสร็จทีเดียวตามแบบ Waterfall แบบเดิม

2.12.1 หลักการของ Agile

1. Individuals Interactions: คนและปฏิสัมพันธ์มากกว่า กระบวนการและเครื่องมือ (Processes Tools)
2. Working software/value: ผลลัพธ์ที่ใช้งานได้มากกว่าการเอกสารที่สมบูรณ์แบบ (Comprehensive documentation)
3. Customer collaboration: ความร่วมมือกับลูกค้ามากกว่าการเจรจาสัญญา (Contract negotiation)
4. Responding to change: การตอบสนองต่อการเปลี่ยนแปลงมากกว่าการยึดติดแผน (Following a plan)

2.12.2 ลักษณะการทำงานของ Agile

1. แบ่งงานออกเป็นช่วงสั้น ๆ หรือ iteration / sprint ซึ่งในทุก sprint จะส่งมอบชิ้นงานที่ใช้งานได้
2. มีการทบทวน (review) และปรับปรุง (retrospective) หลังจากแต่ละ sprint เพื่อให้ทีมเรียนรู้และพัฒนาอย่างต่อเนื่อง
3. เน้นทีมที่สามารถทำได้หลากหลายหน้าที่ (cross-functional) และสามารถตัดสินใจได้ด้วยตนเอง
4. ออกแบบให้สามารถรับมือกับความไม่แน่นอน (uncertainty) และการเปลี่ยนแปลงของความต้องการได้ดี

2.12.3 ประโยชน์ของการใช้ Agile

1. ลดความเสี่ยงจากการที่ต้องวางแผนยาวและส่งมอบทีเดียว เพราะส่งมอบทีละน้อยและเก็บ feedback ได้เร็ว
2. ทำให้สามารถปรับเปลี่ยนได้ตามความต้องการของลูกค้าหรือผู้ใช้งานที่อาจเปลี่ยนแปลงได้
3. ส่งมอบคุณค่าให้ลูกค้าได้เร็วขึ้นและสม่ำเสมอ
4. สนับสนุนการทำงานเป็นทีมและการสื่อสาร

2.13 คำศัพท์

1. แอปพลิเคชัน (Application): โปรแกรมคอมพิวเตอร์ที่ถูกสร้างขึ้นมาเพื่อทำงานบางอย่างบนอุปกรณ์
2. เฟรมเวิร์ก (Framework): ชุดเครื่องมือและโครงสร้างสำเร็จรูปที่ช่วยให้นักพัฒนาสามารถสร้างโปรแกรมได้รวดเร็วและเป็นระบบมากขึ้น
3. Backend: ส่วนการทำงานเบื้องหลังของแอปพลิเคชัน ทำหน้าที่จัดการข้อมูล, ประมวลผลคำสั่ง, และสื่อสารกับฐานข้อมูล
4. Frontend: ส่วนที่ผู้ใช้งานมองเห็นและใช้งานโดยตรง เช่น หน้าจอแอปพลิเคชัน, ปุ่มกด, หรือรูปภาพต่าง ๆ
5. API (Application Programming Interface): ตัวกลางที่ช่วยให้โปรแกรมหรือระบบที่แตกต่างกันสามารถสื่อสารและแลกเปลี่ยนข้อมูลกันได้
6. RESTful API: รูปแบบมาตรฐานในการออกแบบ API ที่ใช้สื่อสารผ่านอินเทอร์เน็ต โดยมีคำสั่งหลักๆ เช่น GET (ขอข้อมูล), POST (เพิ่มข้อมูล), PUT (แก้ไขข้อมูล) และ DELETE (ลบข้อมูล)
7. Notification: การแจ้งเตือนที่ปรากฏบนหน้าจออุปกรณ์เพื่อบอกให้ผู้ใช้ทราบถึงข้อมูลหรือกิจกรรมบางอย่าง
8. Text-to-Speech (TTS): เทคโนโลยีที่แปลงข้อความตัวอักษรให้เป็นเสียงพูด
9. การรู้จำอักขระด้วยแสง (Optical Character Recognition - OCR): เทคโนโลยีที่ใช้ในการแปลงข้อความจากรูปภาพให้กลายเป็นข้อความที่คอมพิวเตอร์สามารถเข้าใจและแก้ไขได้

บทที่ 3

ผลงานที่เกี่ยวข้อง

3.1 แอปพลิเคชันช่วยอ่านฉลากยาโดยเฉพาะฟังก์ชันที่ช่วยให้อ่านข้อมูลยาได้ง่ายขึ้นอย่างชัดเจน

ผู้วิจัย: ญัฐกรณ์ ศรีบุรมย์ [11]

ที่มา: ผู้สูงอายุที่มีอายุมากกว่า 60 ปีขึ้นไป มีปัญหาด้านสายตา ความจำ และไม่คุ้นเคยกับเทคโนโลยี ซึ่งส่งผลให้เกิดความยากลำบากในการอ่านฉลากยาและปฏิบัติตามคำแนะนำในการใช้ยาได้อย่างถูกต้อง

วิธีการ: พัฒนาแอปพลิเคชันที่เป็นตัวช่วยในการอ่านฉลากยา โดยใช้เทคโนโลยีการรู้จำอักขระด้วยแสง (OCR) เพื่อแปลงข้อความบนฉลากยาให้เป็นข้อมูลดิจิทัล และนำเสนอข้อมูลในรูปแบบที่เข้าใจง่าย มีตัวอักษรขนาดใหญ่ เพื่อส่งเสริมพฤติกรรมการใช้ยาที่ถูกต้อง

ผลลัพธ์: จากการทดสอบกับกลุ่มอาสาสมัคร พบว่าอาสาสมัครสามารถตอบคำถามเกี่ยวกับวิธีการรับประทานยา จำนวนครั้งต่อวัน จำนวนเม็ด และต้องใช้อีกก่อนหรือหลังมีอาหารหรือเวลามีอาการ ได้ถูกต้องทั้งหมดทุกข้อและในทุกฉลาก แสดงให้เห็นว่าแอปพลิเคชันสามารถช่วยให้ผู้สูงอายุเข้าใจข้อมูลยาได้ดีขึ้น

ข้อดี:

1. เป็นแอปพลิเคชันที่ตอบโจทย์ผู้สูงอายุได้อย่างตรงจุด
2. ช่วยลดความผิดพลาดในการใช้ยาอย่างมีนัยสำคัญ
3. ใช้งานง่าย เหมาะสมกับผู้สูงอายุที่ไม่คุ้นเคยกับเทคโนโลยี

ข้อจำกัด:

1. การสแกนข้อมูลบางครั้งมีความหน่วง (delay) เพื่อให้การอ่านฉลากยาในครั้งก่อนหน้าเสร็จเรียบร้อยก่อนดำเนินการสแกนครั้งต่อไป
2. อาจมีข้อจำกัดในการรองรับฉลากยาที่มีรูปแบบแตกต่างกัน

3.2 แอปพลิเคชันบน LINE LIFF ร่วมกับ Google Apps Script สำหรับติดตามและประเมินผลความสม่ำเสมอในการรับประทานยาของผู้ป่วยวัณโรค โรงพยาบาลชัชวาทนเรนทร

ผู้วิจัย: วิศว์สดี ปาริยะประเสริฐ [12]

ที่มา: ปัญหาของผู้ป่วยที่ขาดความสม่ำเสมอในการทานยา โดยเฉพาะในกลุ่มโรคเรื้อรังเช่นวัณโรค และผู้สูงอายุที่มักจะลืมหรือสับสนเรื่องการทานยา ซึ่งส่งผลกระทบต่อประสิทธิภาพการรักษาและอาจทำให้เกิดภาวะแทรกซ้อนได้

วิธีการ: ใช้แอปพลิเคชันขนาดเล็กบนแพลตฟอร์ม LINE LIFF (LINE Front-end Framework) เพื่อติดตามการทานยาของผู้ป่วยวัณโรค โดยแอปพลิเคชันจะทำหน้าที่ดังนี้

1. ส่งข้อความแจ้งเตือนผู้ป่วยเมื่อถึงเวลาทานยา
2. ให้ผู้ป่วยส่งภาพถ่ายยืนยันการทานยาแต่ละครั้ง
3. บันทึกตำแหน่งที่ผู้ใช้ทานยาได้
4. จัดเก็บข้อมูลลงในฐานข้อมูลเพื่อการติดตามผลและวิเคราะห์

ผลลัพธ์: หลังจากผู้ป่วยใช้แอปพลิเคชันเป็นเวลา 4 เดือน พบว่าค่าเฉลี่ยความสม่ำเสมอในการทานยาเพิ่มขึ้นจาก 90.66% เป็น 98.38% อย่างมีนัยสำคัญทางสถิติ ซึ่งแสดงให้เห็นว่าการแจ้งเตือนและการติดตามผลอย่างเป็นระบบช่วยให้ผู้ป่วยมีวินัยในการทานยามากขึ้น และสามารถลดโอกาสการล้มเหลวในการรักษาได้อย่างมีประสิทธิภาพ

ข้อดี:

1. ใช้แพลตฟอร์ม LINE ซึ่งเป็นแอปพลิเคชันที่ผู้ป่วยคุ้นเคยและใช้งานอยู่แล้ว ทำให้เรียนรู้และเข้าถึงได้ง่าย
2. ต้นทุนในการพัฒนาระบบไม่สูงมาก
3. ผลลัพธ์ที่ได้มีความชัดเจนและวัดผลได้
4. สามารถติดตามและวิเคราะห์พฤติกรรมการทานยาได้อย่างเป็นระบบ

ข้อเสนอแนะ:

1. เนื่องจากจะมีการยุติการให้บริการ LINE Notify ตั้งแต่วันที่ 31 มีนาคม พ.ศ. 2568 เป็นต้นไป จึงจำเป็นต้องใช้โปรแกรมอื่นมาใช้ทดแทน LINE Notify เช่น Google Chat หรือ Telegram สำหรับการแจ้งเตือนต่อไป
2. ควรมีการพัฒนาเว็บไซต์รองรับการทำงานแบบ standalone มากขึ้น เพื่อลดการพึ่งพาแพลตฟอร์มเฉพาะ

3.3 โครงการวิจัยเรื่อง เทคโนโลยีสารสนเทศแสดงข้อมูลฉลากยาเอกสารกำกับยาแบบอัตโนมัติสำหรับ บริหารจัดการคลังยาปฎิชีวนะเพื่อความปลอดภัย

ผู้วิจัย: วิรุฬห์ ศรีบริรักษ์ [13]

ที่มา: ผู้บริโภคขาดความรู้ความเข้าใจถึงอันตรายที่เกี่ยวข้องกับยา อีกทั้งยังไม่มีแหล่งข้อมูลที่เข้าถึงได้ง่าย ข้อมูลบนฉลากยาจากคลินิกหรือโรงพยาบาลมักไม่เพียงพอต่อการทำความเข้าใจ นอกจากนี้ยังมีปัญหาในการบริหารจัดการคลังยาที่ต้องอาศัยการบันทึกข้อมูลด้วยมือซึ่งใช้เวลานานและเสี่ยงต่อความผิดพลาด

วิธีการ: พัฒนาระบบที่ใช้เทคโนโลยี Optical Character Recognition (OCR) ในการแปลงไฟล์ภาพเอกสารที่ได้รับการสแกนให้กลายเป็นไฟล์ข้อความดิจิทัล และจัดเก็บในฐานข้อมูลในเครื่องแม่ข่าย ระบบนี้ทำหน้าที่เป็นเครื่องมือที่ช่วยในการ

1. ลดความคลาดเคลื่อนทางยา (Medication Error)
2. ป้องกันความผิดพลาดในการจัดการคลังยา
3. ให้ข้อมูลยาที่ถูกต้องและครบถ้วนแก่ผู้ใช้

ผลลัพธ์: ระบบที่พัฒนาขึ้นมีความแม่นยำในการแปลงข้อมูลสูงถึงร้อยละ 96.61 และสามารถช่วยลดเวลาของการเก็บข้อมูลลงฐานข้อมูลได้อย่างมีนัยสำคัญ เมื่อเทียบกับการบันทึกข้อมูลด้วยมือ

ข้อดี:

1. ช่วยให้ผู้ป่วยมีความเข้าใจในการใช้ยามากขึ้น
2. ช่วยให้การใช้ยามีความปลอดภัยและลดปัญหาของการใช้ยาที่ผิดกับโรค
3. ลดเวลาและแรงงานในการบันทึกข้อมูลยา
4. มีความแม่นยำสูงในการอ่านข้อมูล

ข้อจำกัด:

1. ยังคงต้องพัฒนาความยืดหยุ่นของระบบให้รองรับรูปแบบเอกสารที่แตกต่างกัน เช่น ฉลากยาจากโรงพยาบาลต่างๆ ที่อาจมีรูปแบบไม่เหมือนกัน
2. ความแม่นยำอาจลดลงเมื่อฉลากยามีคุณภาพภาพต่ำหรือข้อมูลไม่ชัดเจน
3. ต้องมีการปรับปรุงและฝึกฝนโมเดล OCR อย่างต่อเนื่องเพื่อรองรับฟอนต์และรูปแบบตัวอักษรที่หลากหลาย

3.4 การเปรียบเทียบแอปพลิเคชันที่มีอยู่ในตลาด

การวิเคราะห์แอปพลิเคชันที่อยู่ในตลาดช่วยให้เห็นภาพรวมของฟีเจอร์และแนวทางการออกแบบที่ได้รับความนิยม ซึ่งสามารถนำมาประยุกต์ใช้ในการพัฒนาแอปพลิเคชันได้

โดยได้ทำการเปรียบเทียบฟีเจอร์หลักของแอปพลิเคชันยอดนิยม 5 ตัว ได้แก่ CapYaDoo, PharmaSee [14], MyYaAndYou [15], Medisafe [16] และ RDU รู้เรื่องยา [17] ดังตารางต่อไปนี้

3.4.1 ตารางเปรียบเทียบแอปพลิเคชันในตลาด

หมวดฟีเจอร์	CapYaDoo	PharmaSee	MyYaAndYou	Medisafe	RDU รู้เรื่องยา
ค้นหาข้อมูลยา	ค้นหา ด้วยข้อความ / สแกน / แปลภาษาได้	ค้นหา จาก ภาพยา (AI ตรวจรูป)	ค้นหาด้วยชื่อยา	ค้นหา ด้วย ชื่อ ยาเท่านั้น	ค้นหา ด้วยข้อความ / QR Code
เสียง อ่าน ข้อมูล ยา (Text-to-Speech)	มีเสียงอ่านข้อมูลยาให้ผู้ฟัง	ไม่มี	ไม่มี	ไม่มี	มีเสียงอ่านข้อมูลยาให้ผู้ฟัง
บันทึก อาการ หลัง/การแพ้	มี ระบบ บันทึกอาการ และ ผลข้างเคียง	ไม่มี	มี บันทึก ยา และบันทึกอาการ	ไม่มี	ไม่มี
แจ้งเตือนการทานยา	ตั้ง เวลา เตือนตาม วัน เวลา ที่ต้องการได้	ไม่มี	มีระบบแจ้งเตือน	มีระบบแจ้งเตือนครบถ้วน	ไม่มีระบบเตือน
สแกนบาร์โค้ด / QR Code / ตัวหนังสือ	สแกน ได้ ทั้ง "บาร์ โค้ด" และ "ข้อความ บนฉลากยา"	สแกน รูป ยา (AI Image Recognition)	ไม่มี สแกน บาร์โค้ด	ไม่มีสแกน	มี สแกน QR Code
ข้อมูลยา	TMT และ ออย.	ขึ้นกับฐานข้อมูลของ AI	อ้างอิง จาก ฐานข้อมูล ออย.	ข้อมูลทั่วไป	อ้างอิง จาก ฐานข้อมูล ออย.
เหมาะกับผู้สูงอายุ / ใช้งานง่าย	UI เรียบง่าย พัฒนา เพื่อ ผู้ สูงอายุ	UI ใช้ งาน ยาก และไม่เหมาะกับบุคคลทั่วไป	UI ใช้งานง่าย แต่ไม่อัปเดต	ระบบ แจ้ง เตือนมีความ ซับ ซ้อนและใช้งานยาก	UI ใช้ งาน ยาก และ ยัง ไม่มี การอัปเดตเพิ่มเติม
บันทึก ประวัติ ยา ในเครื่อง	มี ระบบ เก็บประวัติ ยา และอาการย้อนหลัง	เก็บ เฉพาะ ภาพยา	บันทึก ข้อมูล ยาได้	เก็บข้อมูลยาและเวลาเตือน	ไม่มีระบบบันทึก
เทคโนโลยี AI / OCR	ใช้ OCR อ่านฉลาก ยา และตรวจบาร์โค้ด	ใช้ AI แยกภาพยา	ไม่มี	ไม่มี	ไม่มี

ตาราง 3.1: เปรียบเทียบแอปพลิเคชันในตลาด

จากการเปรียบเทียบข้างต้น จะเห็นว่า CapYaDoo มีข้อได้เปรียบในหลายด้าน ได้แก่

ข้อได้เปรียบของ CapYaDoo

1. สามารถสแกนได้ทั้งบาร์โค้ดและข้อความ ทำให้ง่ายต่อการใช้งาน
2. สามารถจัดการข้อมูลยาและอาการได้อย่างครบถ้วน ได้แก่ เพิ่ม แก้ไข และลบข้อมูลยาที่กำลังใช้อยู่
3. รองรับการอ่านข้อมูลยาให้ผู้ฟัง (Text-to-Speech)
4. สามารถตั้งแจ้งเตือนการทานยาได้ และสามารถแจ้งเตือนผู้ใช้เมื่อถึงเวลา

ข้อจำกัดของ CapYaDoo

1. รองรับเพียงระบบปฏิบัติการ Android อย่างเดียว
2. เนื่องจากยังไม่มีความร่วมมือกับหน่วยงานใด ทำให้มีข้อจำกัดในการพัฒนาแอปพลิเคชันให้ครอบคลุมทุกฟีเจอร์

3.4.2 สรุปการเปรียบเทียบ

จากการเปรียบเทียบแอปพลิเคชันที่มีอยู่ในปัจจุบัน พบว่าแต่ละแอปพลิเคชันมีฟีเจอร์ที่โดดเด่นแตกต่างกันไป ส่งผลให้ผู้ใช้ต้องสลับใช้งานหลายแอปพลิเคชันจึงจะสามารถจัดการข้อมูลการทานยาได้อย่างครบถ้วน แอปพลิเคชัน CapYaDoo จึงถูกพัฒนาขึ้นเพื่อตอบโจทย์ปัญหานี้ โดยรวบรวมฟังก์ชันสำคัญทั้งหมด ไม่ว่าจะเป็นการแจ้งเตือน การติดตามประวัติการทานยา การจัดเก็บข้อมูลยา และการบันทึกอาการ มาไว้ภายในแอปเดียว ช่วยให้ผู้ใช้สามารถจัดการด้านการทานยาได้สะดวกและมีประสิทธิภาพมากยิ่งขึ้น

นอกจากนี้ แอปพลิเคชันจำนวนมากยังไม่ได้ออกแบบโดยคำนึงถึงความต้องการเฉพาะของผู้สูงอายุ เช่น การอ่านตัวอักษรยาก การใช้งานซับซ้อน หรือองค์ประกอบบนหน้าจอที่ไม่เป็นมิตรต่อผู้ใช้งาน ทำให้ผู้สูงอายุประสบปัญหาในการใช้งานจริง รวมทั้งบางแอปพลิเคชันไม่มีการอัปเดตอย่างต่อเนื่อง ทำให้ดีไซน์และประสบการณ์ใช้งานล้าสมัย ไม่สอดคล้องกับพฤติกรรมผู้ใช้ในปัจจุบัน

ด้วยเหตุนี้ CapYaDoo จึงมุ่งเน้นการออกแบบที่เหมาะสมกับผู้สูงอายุ ควบคู่กับการรวมฟังก์ชันที่จำเป็นทั้งหมดเข้าด้วยกัน เพื่อให้เป็นแอปพลิเคชันที่ตอบโจทย์การจัดการการทานยาอย่างครบวงจร

3.5 ประสบการณ์การพัฒนาที่ผ่านมา

เพื่อให้เห็นถึงประสบการณ์และความสามารถของทีมผู้พัฒนาในการสร้างแอปพลิเคชัน จึงได้นำเสนอโครงการที่ผ่านมาที่เกี่ยวข้องกับการพัฒนาแอปพลิเคชันด้วย Flutter Framework โดยมีรายละเอียดดังนี้

3.5.1 Project 1: Application Chat Music Player

วัตถุประสงค์:

1. เพื่อให้ผู้ใช้งานเกิดความผ่อนคลายและสร้างความสุขระหว่างการใช้งานให้กับผู้ใช้ได้
2. เพื่อให้ผู้ใช้งานได้พูดคุยและแลกเปลี่ยนความคิดเห็นกับผู้ใช้คนอื่น ๆ ได้

บทบาทและสิ่งที่ทำ:

1. นาย ฤทธิช พลราช: มีหน้าที่ในการออกแบบ UX/UI และทำการพัฒนาแอปพลิเคชัน
2. นางสาว จิรัชญา ราชพลแสน: มีหน้าที่ในการจัดทำรายงานและพัฒนาแอปพลิเคชัน

เทคโนโลยีที่ใช้:

1. ใช้ภาษา Dart และ Flutter framework
2. Backend ใช้ Firebase Firestore, Firebase Authentication
3. Package ที่ใช้คือ audioplayer

3.5.2 Project 2: Application Location Discovery

วัตถุประสงค์หลัก:

1. เพื่อให้ผู้ใช้งานเลือก/ไม่เลือก สถานที่พักผ่อน อ่านหนังสือภายในมหาวิทยาลัย
2. เพื่อให้ผู้ใช้งานใช้ข้อมูลในการตัดสินใจเลือก เช่น สถานที่ที่อยู่ใกล้เรา สิ่งอำนวยความสะดวกในสถานที่นั้น

บทบาทและสิ่งที่ทำ:

1. นางสาว ปารรณาสุภาวรงค์: ทำหน้าที่ในการออกแบบแอปพลิเคชัน Logo, UI และพัฒนาแอปพลิเคชัน

เทคโนโลยีที่ใช้:

1. ใช้ภาษา Dart และ Flutter framework
2. Backend ใช้ Firebase Firestore, Firebase Authentication

3.5.3 บทเรียนที่ได้จากประสบการณ์การพัฒนา

จากการพัฒนาแอปพลิเคชันทั้งสองโครงการ ทีมผู้พัฒนาได้เรียนรู้และสะสมประสบการณ์ในด้านต่างๆ ดังนี้

ด้านการออกแบบ UI/UX

1. การออกแบบส่วนติดต่อผู้ใช้ที่ใช้งานง่ายและสวยงาม
2. การเลือกใช้สีและฟอนต์ที่เหมาะสมกับกลุ่มผู้ใช้
3. การจัดวางองค์ประกอบบนหน้าจอให้เป็นระเบียบและเข้าถึงได้ง่าย

ด้านการพัฒนาแอปพลิเคชัน

1. การใช้ Flutter Framework ในการพัฒนาแอปพลิเคชันข้ามแพลตฟอร์ม
2. การจัดการ State Management ในแอปพลิเคชัน
3. การใช้งาน Package ต่างๆ เพื่อเพิ่มฟังก์ชันให้กับแอปพลิเคชัน

ด้านการทำงานเป็นทีม

1. การแบ่งหน้าที่และความรับผิดชอบในทีม
2. การสื่อสารและประสานงานระหว่างสมาชิกในทีม
3. การจัดการเวลาและกำหนดการในการพัฒนา

ประสบการณ์เหล่านี้จะเป็นประโยชน์อย่างมากในการพัฒนาโครงการ CapYaDoo เนื่องจากทีมผู้พัฒนาได้มีความคุ้นเคยกับ Flutter Framework และการทำงานร่วมกันมาแล้ว

บทที่ 4

ระเบียบวิธีดำเนินโครงการ

4.1 การวิเคราะห์และออกแบบระบบ

4.1.1 Persona

หมายถึง ตัวตนสมมติที่เป็นตัวแทนกลุ่มลูกค้าเป้าหมายของธุรกิจ โดยอิงจากข้อมูลการวิจัยและวิเคราะห์พฤติกรรม ความต้องการ และแรงจูงใจของกลุ่มลูกค้าจริง เพื่อให้ธุรกิจเข้าใจลูกค้าได้อย่างลึกซึ้ง [18]

ความสำคัญของ Persona

1. ช่วยให้ธุรกิจเข้าใจลูกค้าได้อย่างลึกซึ้ง
2. ปรับกลยุทธ์ทางการตลาดให้เหมาะสมกับพฤติกรรมของกลุ่มเป้าหมาย
3. เพื่อให้สามารถสื่อสารได้อย่างตรงใจลูกค้า


โดยมีตัวอย่างของ Persona ดังนี้

	คุณยามาลี (อายุ 75 ปี) อาชีพ: แม่บ้าน ผู้สูงอายุที่ต้องการความช่วยเหลือ	Short description ผู้สูงอายุที่มีปัญหาด้านสุขภาพและการมองเห็น ไม่ถนัดการใช้เทคโนโลยี ทำให้ต้องพึ่งพาลูกหลานในการดูแลจัดการเรื่องยาและสุขภาพ
	Key Attribute ต้องการการดูแลและความปลอดภัย: ให้ความสำคัญกับการมีคนช่วยจัดการสุขภาพและยาให้เป็นไปอย่างปลอดภัย	Challenges <ul style="list-style-type: none">• ไม่สามารถอ่านฉลากยาได้ด้วยตัวเอง• หลงลืมเรื่องการทานยาบ่อยครั้ง• ไม่คุ้นเคยและกลัวการใช้เทคโนโลยี
	Needs <ul style="list-style-type: none">• ความมั่นใจว่าจะมีคนคอยช่วยจัดการยาให้ถูกต้องและปลอดภัย• วิธีการรับข้อมูลยาที่ง่ายและชัดเจน เช่น การฟังเสียง• การแจ้งเตือนที่ช่วยให้ไม่ลืมทานยา	Opportunities <ul style="list-style-type: none">• พัฒนาฟังก์ชันแปลง ข้อความเป็นเสียง (Text-to-Speech) สำหรับข้อมูลยา• ใช้การแจ้งเตือนด้วย เสียงที่ดังและชัดเจน

รูป 4.1: ตัวตนสมมติที่เป็นตัวแทนของกลุ่มผู้ใช้เป้าหมายคนที่ 1

	ชื่อ: คุณหมอชัย (อายุ 58 ปี) อาชีพ: แพทย์เกษียณ ผู้สูงอายุที่เน้นความเป็นอยู่ที่ดีและสุขภาพ	Short description ผู้สูงอายุที่เป็นแพทย์เกษียณ มีความรู้เรื่องสุขภาพเป็นอย่างดี แต่ยังคงต้องการเครื่องมือที่ช่วยให้การติดตามสุขภาพเป็นเรื่องง่ายและครบวงจร
Key Attribute ต้องการความรู้และการบูรณาการ: ให้ความสำคัญกับความถูกต้องของข้อมูลและการจัดการสุขภาพแบบครบวงจร		Challenges <ul style="list-style-type: none"> ข้อมูลในแอปพลิเคชันที่น่าเชื่อถือ หรือไม่เป็นไปตามหลักการแพทย์ แอปพลิเคชันที่ซับซ้อนและมีข้อมูลที่ไม่จำเป็นมากเกินไป ความกังวลว่าข้อมูลสุขภาพส่วนตัวจะไม่ปลอดภัย
Needs <ul style="list-style-type: none"> เครื่องมือที่ช่วยจัดระเบียบข้อมูลสุขภาพและยาได้อย่างแม่นยำ ฟังก์ชันที่สามารถบันทึกข้อมูลสุขภาพอื่น ๆ ได้ เช่น การออกกำลังกาย หรืออาหาร ข้อมูลที่ละเอียดและน่าเชื่อถือจากแหล่งอ้างอิงที่เชื่อถือได้ 		Opportunities <ul style="list-style-type: none"> แอปพลิเคชันที่เน้นการบันทึกข้อมูลสุขภาพได้ จัดทำฐานข้อมูลยาที่ละเอียดพร้อมแหล่งอ้างอิง

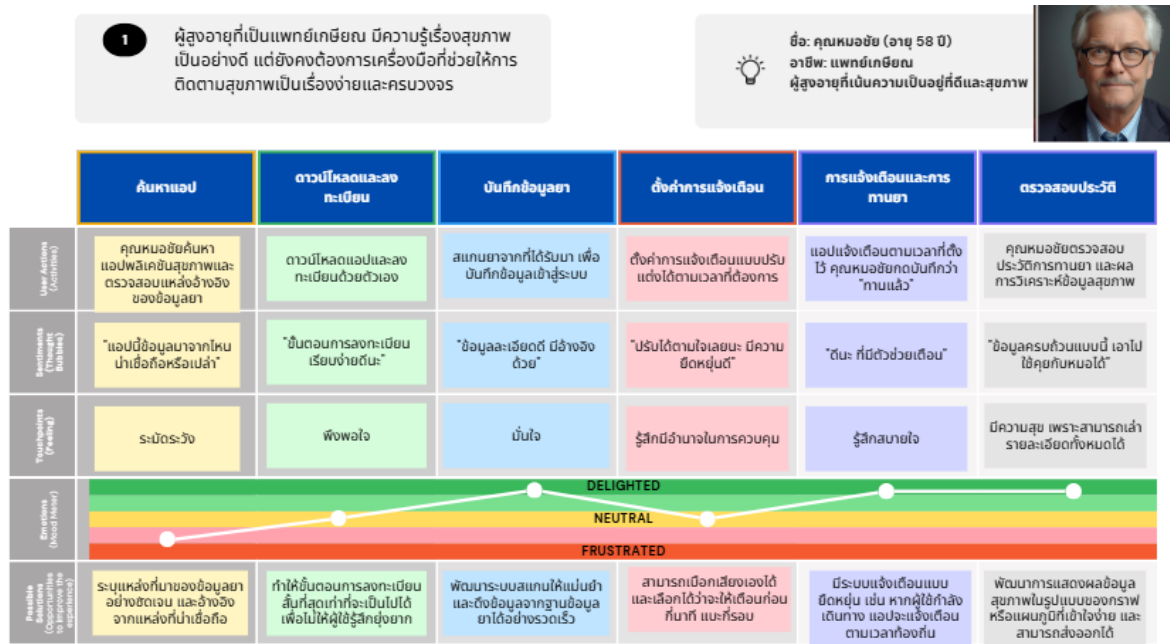
รูป 4.2: ตัวตนสมมติที่เป็นตัวแทนของกลุ่มผู้ใช้เป้าหมายคนที่ 2

	ชื่อ: คุณนารี (อายุ 58 ปี) อาชีพ: พนักงานบริษัท ผู้ป่วยเรื้อรัง (Chronic Patient)	Short description พนักงานบริษัทที่ป่วยเป็นโรคความดันโลหิตสูงและเบาหวาน ต้องการจัดการยาหลายชนิดอย่างเป็นระบบเพื่อให้ไม่สับสนและลืมทานยา
Key Attribute ต้องการความแม่นยำและเป็นระบบ: ต้องการเครื่องมือที่ช่วยจัดการชีวิตประจำวันให้ง่ายขึ้น		Challenges <ul style="list-style-type: none"> ลืมทานยาบ่อยครั้งเพราะต้องทานหลายเวลาในแต่ละวัน สับสนเรื่องชนิดของยาและเวลาที่ควรทาน ไม่แน่ใจว่ายาตัวไหนมีผลข้างเคียงอย่างไร
Needs <ul style="list-style-type: none"> เครื่องมือที่ช่วยจัดการยาที่ต้องทานหลายชนิดในแต่ละวันอย่างเป็นระบบ ข้อมูลที่ถูกต้องเกี่ยวกับยาที่กำลังจะทาน เช่น ผลข้างเคียง หรือคำแนะนำพิเศษ ระบบแจ้งเตือนที่ช่วยให้ไม่ลืมทานยา 		Opportunities <ul style="list-style-type: none"> พัฒนาระบบแจ้งเตือนที่ใช้งานง่ายและปรับแต่งได้ จัดทำฐานข้อมูลที่ให้ข้อมูลครบถ้วนและเข้าใจง่าย ออกแบบหน้าจอให้แสดงตารางการทานยาแต่ละวันอย่างชัดเจน

รูป 4.3: ตัวตนสมมติที่เป็นตัวแทนของกลุ่มผู้ใช้เป้าหมายคนที่ 3

4.1.2 User Journey Map

หรือ แผนภาพเส้นทางผู้ใช้ เป็นเครื่องมือที่ใช้อธิบายขั้นตอนและประสบการณ์ของผู้ใช้เมื่อมีปฏิสัมพันธ์กับแอปพลิเคชันของเรา ว่าผู้ใช้ได้ตอบอย่างไร พบปัญหาตรงไหน และจะปรับปรุงจุดใดได้บ้าง เพื่อให้ผู้ใช้ได้รับประสบการณ์ที่ดีขึ้น ซึ่งจะแสดงภาพรวมตั้งแต่จุดเริ่มต้นที่ผู้ใช้เริ่มสนใจไปจนถึงการบรรลุเป้าหมายโดยแบ่งเป็นช่วงเวลาและเหตุการณ์สำคัญต่าง ๆ [19] โดยมีตัวอย่างของ User Journey Map ดังนี้



รูป 4.4: ประสบการณ์ของผู้ใช้เมื่อมีปฏิสัมพันธ์กับแอปพลิเคชันของเราคนที่ 1



รูป 4.5: ประสบการณ์ของผู้ใช้เมื่อมีปฏิสัมพันธ์กับแอปพลิเคชันของเราคนที่ 2



รูป 4.6: ประสบการณ์ของผู้ใช้เมื่อมีปฏิสัมพันธ์กับแอปพลิเคชันของเราคนที่ 3

4.1.3 ปัญหาและแนวทางการแก้ไข

1. ปัญหาด้านการจัดการยาหลายชนิด: ผู้สูงอายุมักต้องทานยาหลายชนิดในแต่ละวัน ทำให้สับสนและลืมรับประทาน

แนวทางแก้ไข:

 - พัฒนาแอปที่ช่วยจัดการยาตามเวลาอย่างเป็นระบบ
 - มีฟังก์ชันบันทึกและตรวจสอบการทานยา
2. ปัญหาด้านการขาดข้อมูลที่ครบถ้วนและน่าเชื่อถือ: ผู้ใช้ต้องการข้อมูลเกี่ยวกับยา เช่น ผลข้างเคียง

แนวทางแก้ไข:

 - นำเข้าข้อมูลยาจากแหล่งอ้างอิงที่เชื่อถือได้
 - แสดงข้อมูลอย่างชัดเจน และสามารถเข้าถึงง่าย เช่น ผ่านข้อความหรือเสียงอ่าน
3. ปัญหาการแจ้งเตือนและการติดตามการทานยา: ผู้สูงอายุต้องการระบบแจ้งเตือนที่ช่วยให้ไม่ลืมทานยา และสามารถบันทึกข้อมูลการทานยาได้

แนวทางแก้ไข:

 - ระบบแจ้งเตือนยาที่ใช้งานง่าย พร้อมสัญลักษณ์หรือเสียง
 - ระบบที่สามารถบันทึกการทานยาได้

4.1.4 การวิเคราะห์ความต้องการ (Requirement Analysis)

ความต้องการของผู้ใช้ (User Requirements)

1. ระบบจัดการข้อมูลยา (เพิ่ม/แก้ไข/ลบ)
2. ระบบแจ้งเตือนการทานยาที่ตั้งเวลาได้
3. ระบบการสแกนข้อมูลยาจากฉลาก (Barcode & OCR)
4. ระบบบันทึกอาการหลังการใช้ยา
5. ระบบอ่านข้อมูลยาให้ผู้พิการ (Text-to-Speech)

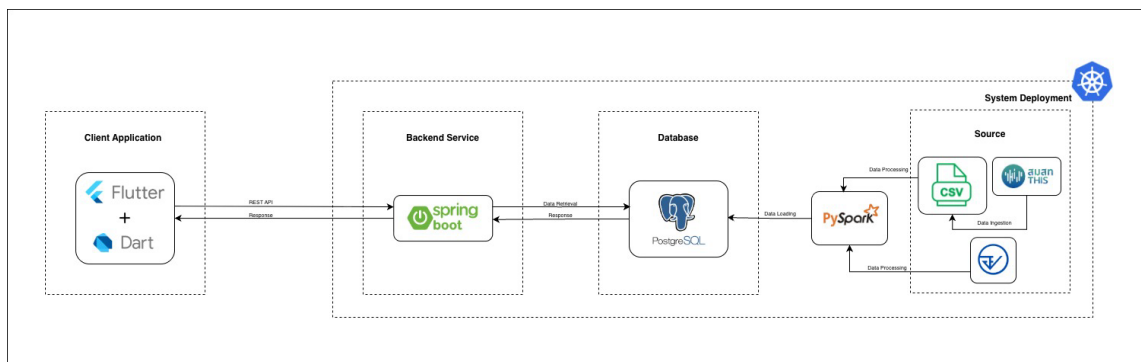
ความต้องการของระบบ (System Requirements)

1. สามารถทำงานได้บน Android
2. มีความเร็วและความแม่นยำในการสแกนข้อมูล
3. มีการจัดเก็บข้อมูลผู้ใช้และข้อมูลยาอย่างปลอดภัย
4. สามารถแสดงข้อมูลได้อย่างครบถ้วน ไม่ตกหล่น

4.1.5 การออกแบบสถาปัตยกรรม (System Architecture Design)

1. Source: ข้อมูลต้นทางมาจาก สำนักพัฒนามาตรฐานระบบข้อมูลสุขภาพไทย ซึ่งเผยแพร่ข้อมูลในรูปแบบไฟล์ Excel และจากเว็บไซต์ของทางสำนักงานคณะกรรมการอาหารและยา โดยข้อมูลดังกล่าวจะถูกนำมาใช้เป็นแหล่งข้อมูลหลักของระบบ
2. Data Processing: ข้อมูลจากไฟล์ Excel จะถูกนำเข้าสู่กระบวนการประมวลผลด้วย PySpark ซึ่งเป็นเครื่องมือที่เหมาะสมสำหรับการจัดการและประมวลผลข้อมูลขนาดใหญ่ (Big Data Processing) โดยจะทำการจัดรูปแบบข้อมูลให้อยู่ในโครงสร้างที่เหมาะสม จากนั้นนำข้อมูลที่ได้นำมาทำการอ้างอิงเพื่อค้นหาข้อมูลเพิ่มเติมจากเว็บไซต์ของสำนักงานคณะกรรมการอาหารและยา เพื่อใช้เป็นข้อมูลประกอบสำหรับแอปพลิเคชัน
3. Database: หลังจากผ่านการประมวลผลแล้ว ข้อมูลจะถูกโหลดเข้าสู่ฐานข้อมูล PostgreSQL เพื่อจัดเก็บอย่างเป็นระบบและสามารถเข้าถึงได้อย่างมีประสิทธิภาพ
4. Backend Service: ใช้ Spring Boot Framework ในการพัฒนาเป็นบริการฝั่งเซิร์ฟเวอร์ (Backend Service) ทำหน้าที่เป็นตัวกลางระหว่างฐานข้อมูลและฝั่งผู้ใช้งาน โดยจัดการคำขอ (Request) จากฝั่ง Client รวมถึงประมวลผลและส่งข้อมูลกลับไปยังผู้ใช้งานในรูปแบบ API

5. Client Application: ส่วนของผู้ใช้งาน (Frontend) ถูกพัฒนาด้วย Flutter ซึ่งเป็นเครื่องมือสำหรับพัฒนาแอปพลิเคชันแบบ Cross-platform โดยในโครงการนี้จะพัฒนาแอปพลิเคชันแบบ Android เท่านั้น
6. System Deployment: สำหรับการทดสอบและจำลองสภาพแวดล้อมการทำงานของระบบบน Kubernetes โดยมีการใช้เครื่องมือ kind (Kubernetes IN Docker) เพื่อสร้าง Kubernetes cluster จำลองบนเครื่องพัฒนา (Local Environment) โดย cluster ที่สร้างขึ้นจะประกอบด้วย container หลัก ได้แก่ Spring Boot Application Container, PySpark Container และ PostgreSQL Database Container



รูป 4.7: System Architecture Design

4.1.6 ขั้นตอนการออกแบบระบบ

1. Entity Relationship Diagram (ERD) : อธิบายโครงสร้างระหว่างฐานข้อมูลในรูปแบบแผนภาพ โดยในโครงการนี้ จะใช้ในแนวคิด ERD แบบ Crow's Foot Notation [20] ซึ่งเป็นรูปแบบการวาด ERD ที่นิยมใช้เนื่องจากอ่านง่าย และแสดงจำนวนความสัมพันธ์ (Cardinality) ระหว่าง Entities ชัดเจน ซึ่งมีองค์ประกอบดังนี้

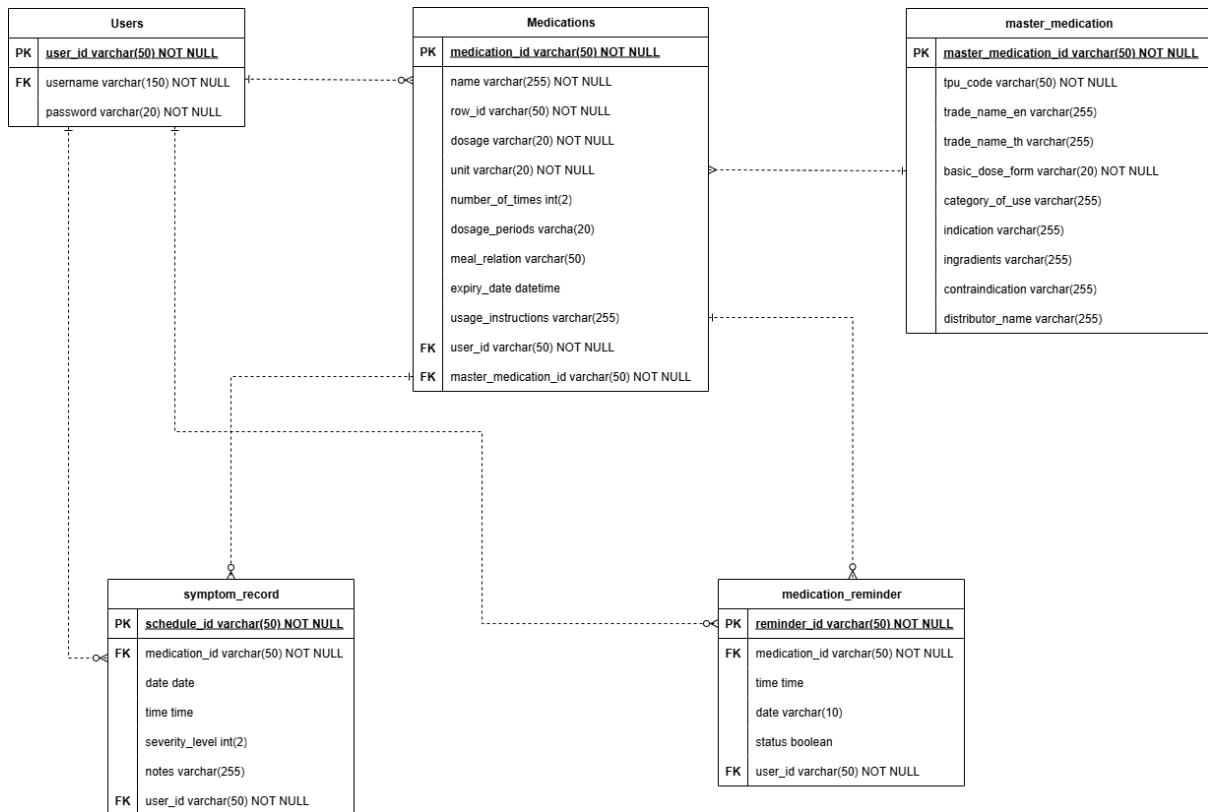
- (a) Entity: สิ่งของ บุคคล เหตุการณ์ หรือแนวคิดที่สามารถระบุได้ชัดเจน
- (b) Relationship: ความสัมพันธ์ระหว่างเอนทิตี
- (c) Attributes: คุณลักษณะของเอนทิตีหรือความสัมพันธ์
- (d) Primary Key: ใช้เพื่อระบุเอนทิตีอย่างไม่ซ้ำกัน
- (e) Cardinality: จำนวนขั้นต่ำและสูงสุดของความสัมพันธ์ระหว่างเอนทิตี

โดยฐานข้อมูลถูกออกแบบบน PostgreSQL ซึ่งเป็นฐานข้อมูลเชิงสัมพันธ์ (Relational Database) ที่มีความน่าเชื่อถือ และเหมาะสมกับการจัดการข้อมูลที่มีโครงสร้างชัดเจน โครงสร้างหลัก (Tables) ที่ออกแบบไว้ประกอบด้วย:

- (a) users: จัดเก็บข้อมูลของผู้ใช้
- (b) master_medications: ฐานข้อมูลยาหลัก
- (c) medications: จัดเก็บข้อมูลยาที่ผู้ใช้ทำการบันทึกไว้

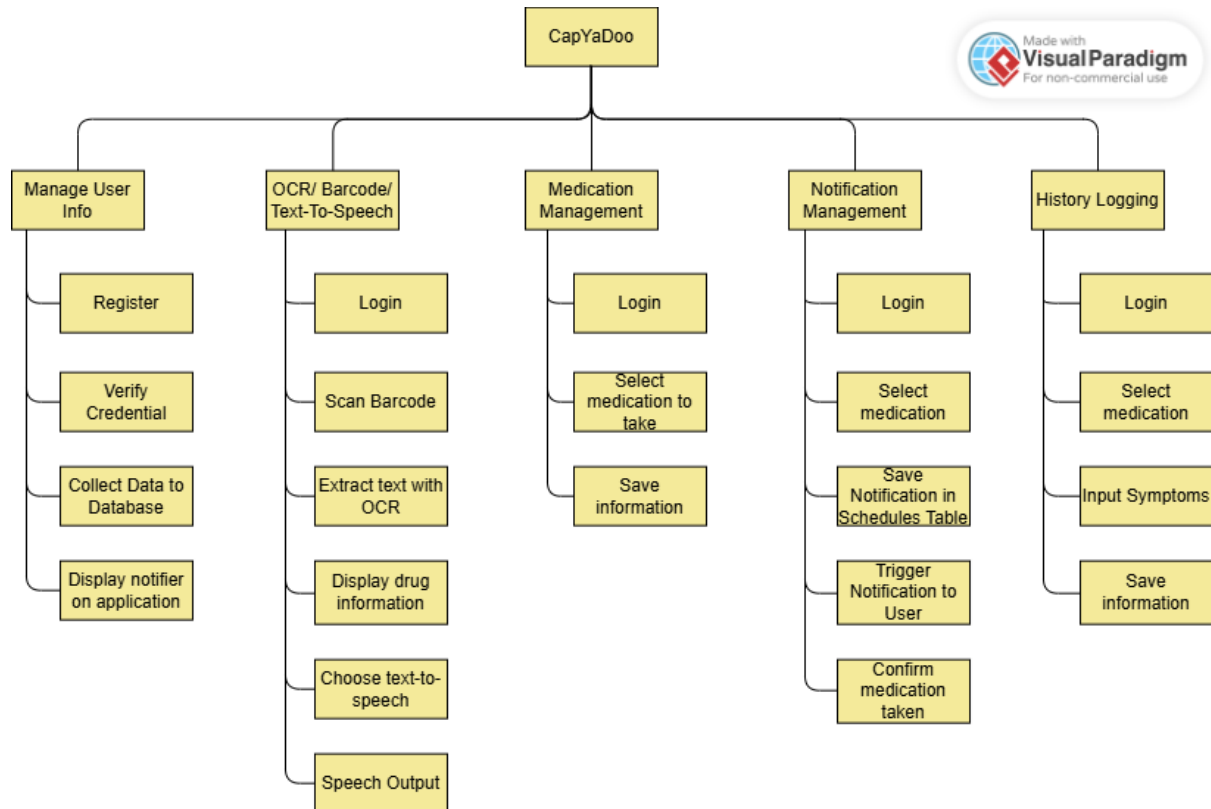
(d) medicatin_reminders: จัดเก็บตารางเวลาการแจ้งเตือนการทานยา

(e) symptom_recorde: จัดเก็บประวัติการทานยาและอาการของผู้ใช้แต่ละคน



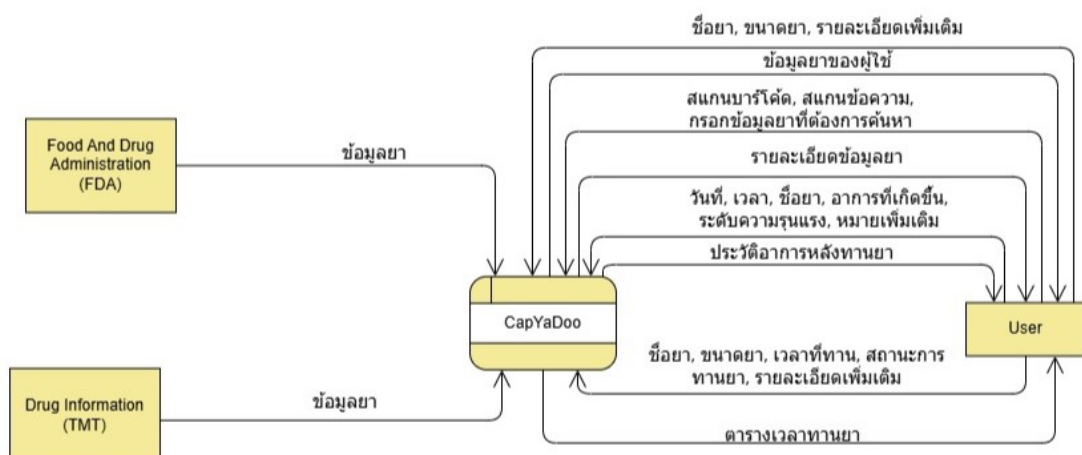
รูป 4.8: Entity Relationship Diagram (ERD)

2. Functional Decomposition: ขั้นตอนการแบ่งระบบออกเป็นส่วนย่อย ๆ เพื่อให้ง่ายต่อการทำความเข้าใจ การวิเคราะห์ การพัฒนา และการนำไปใช้งาน



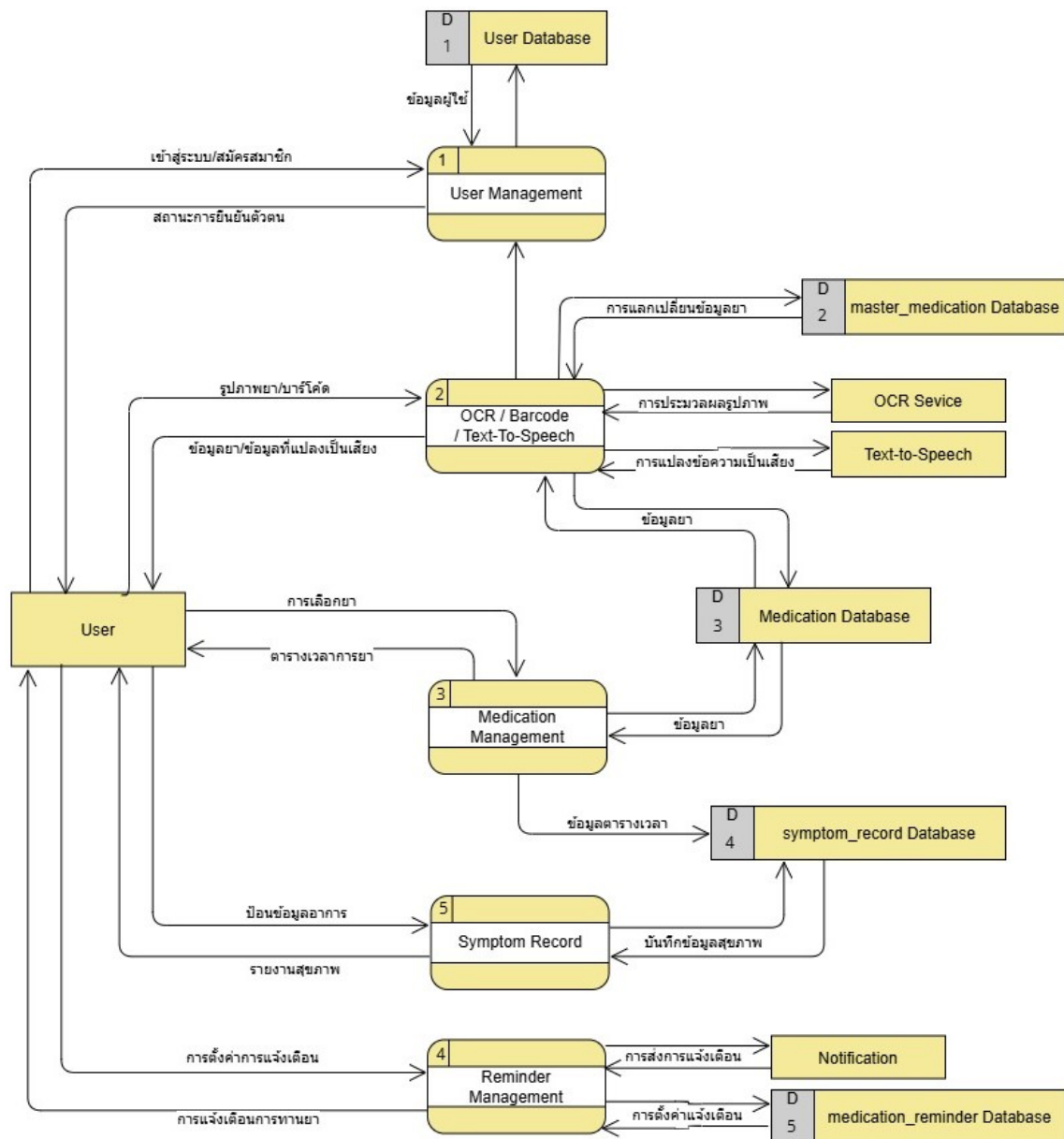
รูป 4.9: Functional Decomposition

3. Data Flow Diagram Level 0: หรือ Context Diagram ใช้แสดงภาพรวมของระบบทั้งหมด



รูป 4.10: Data Flow Diagram Level 0

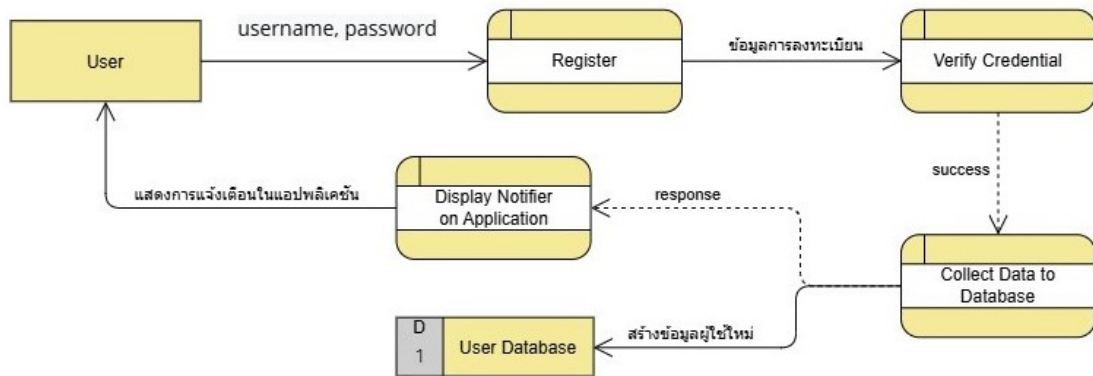
4. Data Flow Diagram Level 1 : ใช้แสดงกระบวนการย่อย ๆ ภายในระบบ



รูป 4.11: Data Flow Diagram Level 1

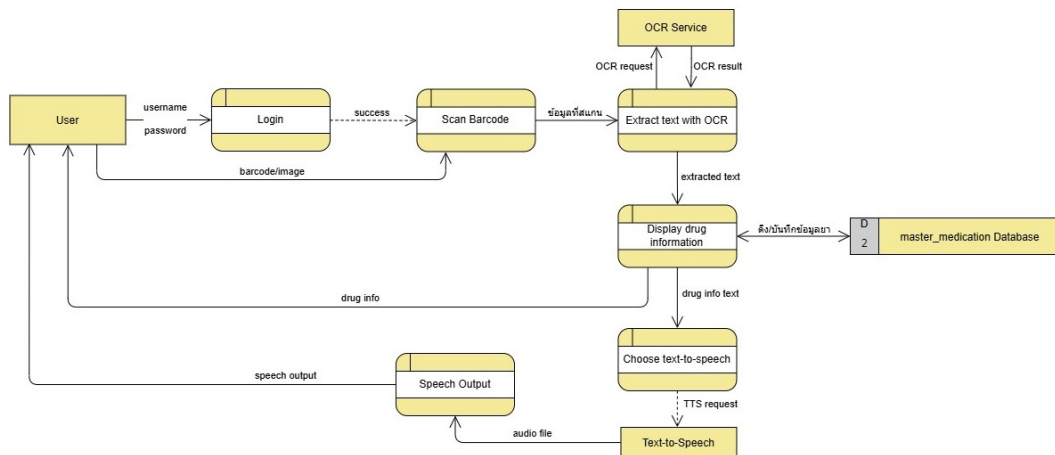
5. Data Flow Diagram Level 2: ใช้ในการขยายข้อมูลจาก Data Flow Diagram ระดับ 1 ให้ละเอียดมากยิ่งขึ้น

(a) Manage User Information



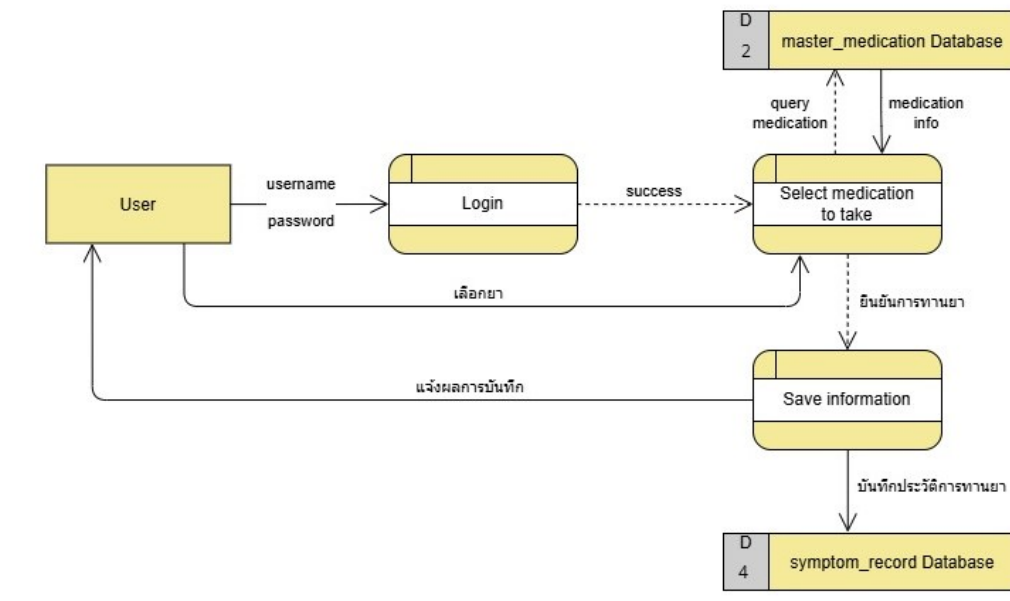
รูป 4.12: Data Flow Diagram Level 2.1 - Manage User Information

(b) OCR/ Barcode/ Text-To-Speech



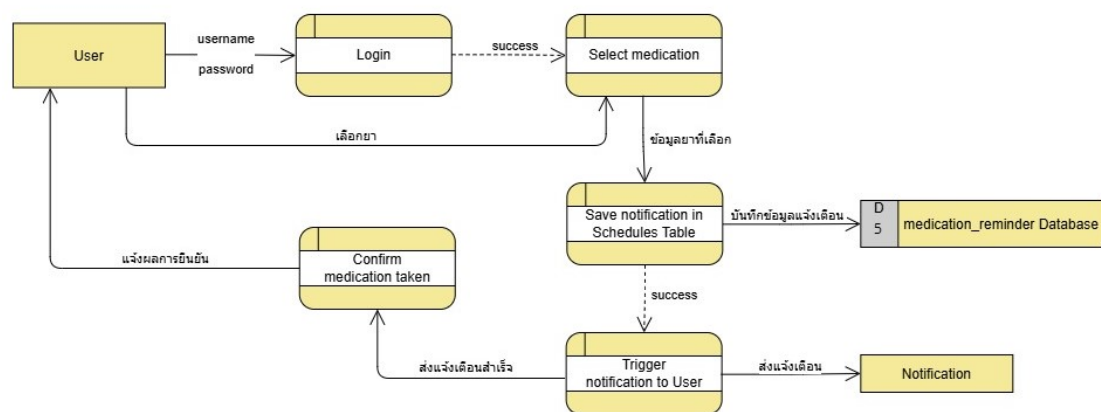
รูป 4.13: Data Flow Diagram Level 2.2 - OCR/ Barcode/ Text-To-Speech

(c) Medication Management



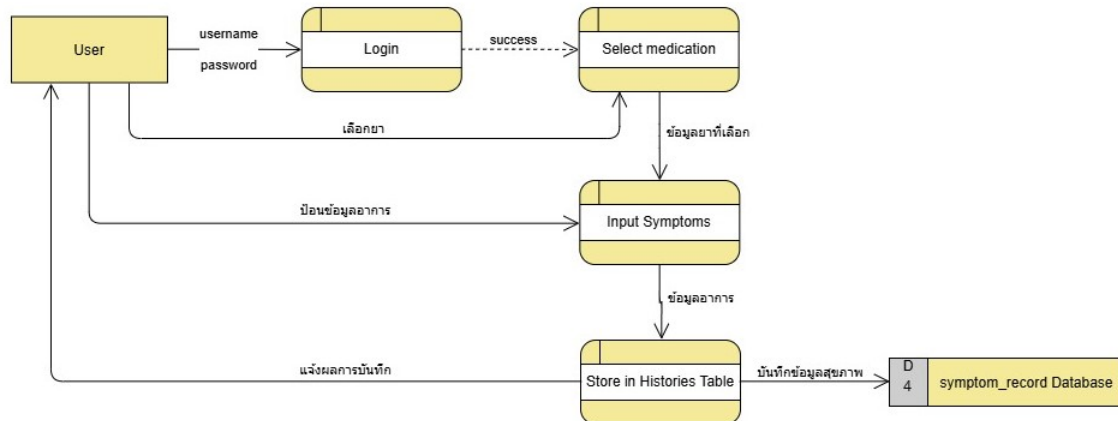
รูป 4.14: Data Flow Diagram Level 2.3 - Medication Management

(d) Notification Management



รูป 4.15: Data Flow Diagram Level 2.4 - Notification Management

(e) History Logging

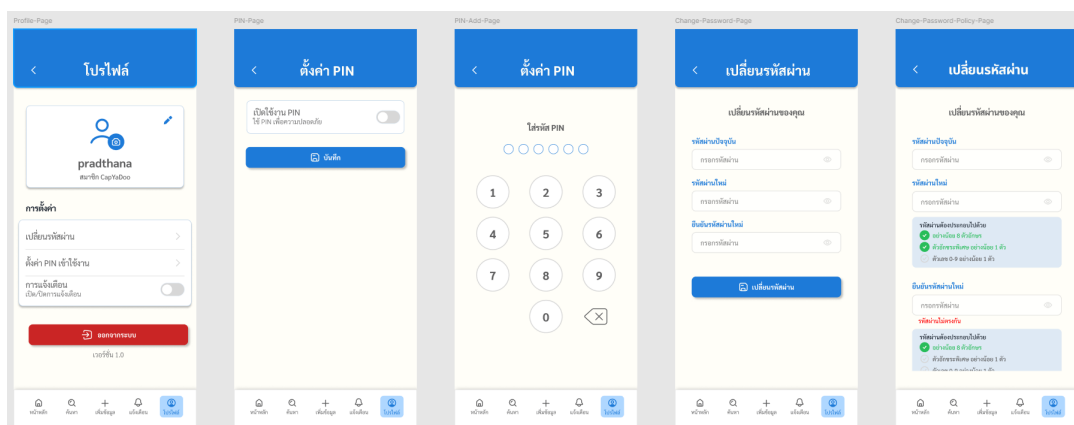


รูป 4.16: Data Flow Diagram Level 2.5 - History Logging

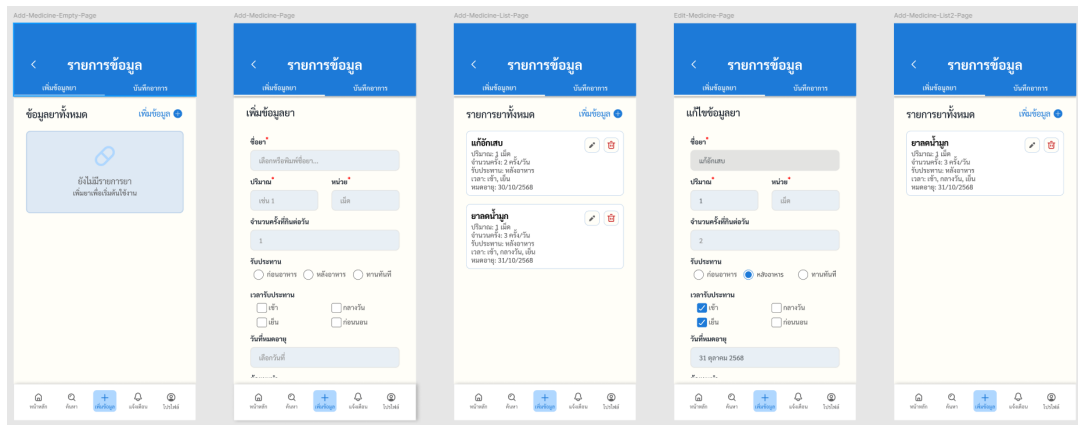
4.1.7 การพัฒนา UX/UI:

ในการออกแบบส่วนติดต่อผู้ใช้ (User Interface: UI) และประสบการณ์ผู้ใช้ (User Experience: UX) ของแอปพลิเคชัน ได้เลือกใช้เครื่องมือ Figma [21] ซึ่งเป็นแพลตฟอร์มออกแบบแบบออนไลน์ (Cloud-based Design Tool) ที่ได้รับความนิยมอย่างแพร่หลาย เนื่องจากรองรับการออกแบบแบบร่วมมือกัน (Collaborative Design) และสามารถแก้ไขแบบร่างได้แบบเรียลไทม์ผ่านเว็บเบราว์เซอร์โดยไม่ต้องติดตั้งโปรแกรมเพิ่มเติม

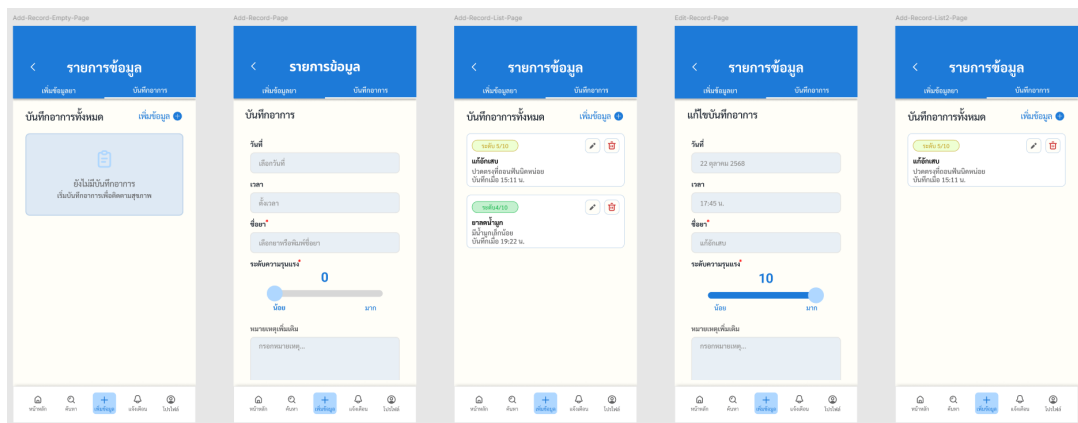
Figma ช่วยให้สามารถสร้างต้นแบบ (Prototype) ของหน้าจอแอปพลิเคชันได้อย่างสะดวก โดยรองรับการสร้างองค์ประกอบ UI เช่น ปุ่ม เมนู ตัวอักษร รวมถึงการเชื่อมโยงลำดับหน้าจอเพื่อจำลองการใช้งานจริง ทำให้ง่ายต่อการนำเสนอและทดสอบการใช้งานก่อนเริ่มพัฒนา โดยมีหน้าแอปพลิเคชันที่ได้จากการออกแบบด้วย Figma ดังแสดงต่อไปนี้



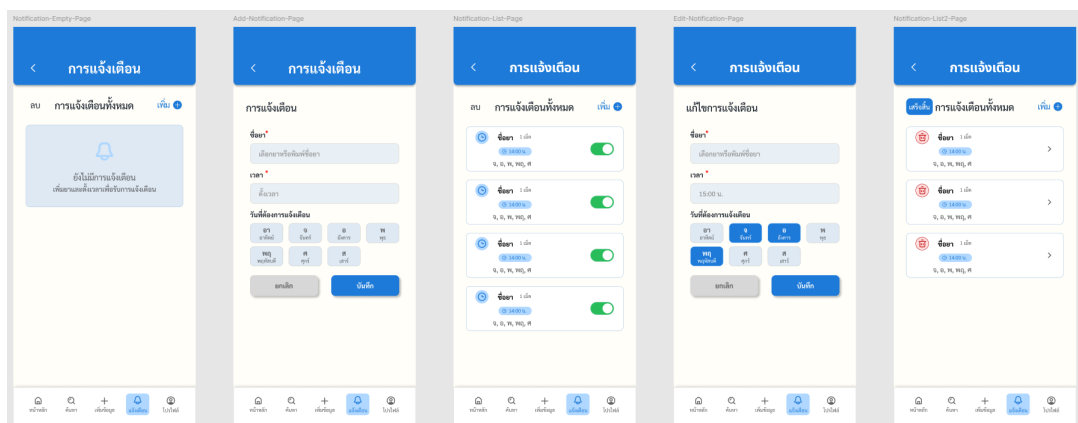
รูป 4.17: หน้าการจัดการข้อมูลผู้ใช้งาน



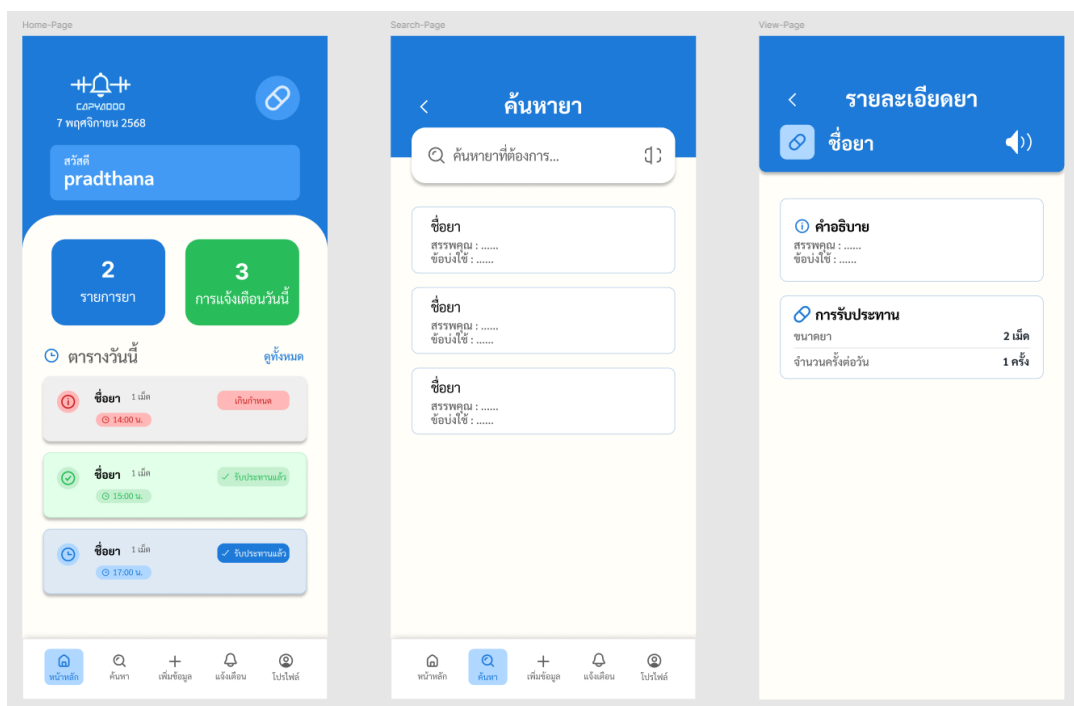
รูป 4.18: หน้าสำหรับการจัดการข้อมูลยา



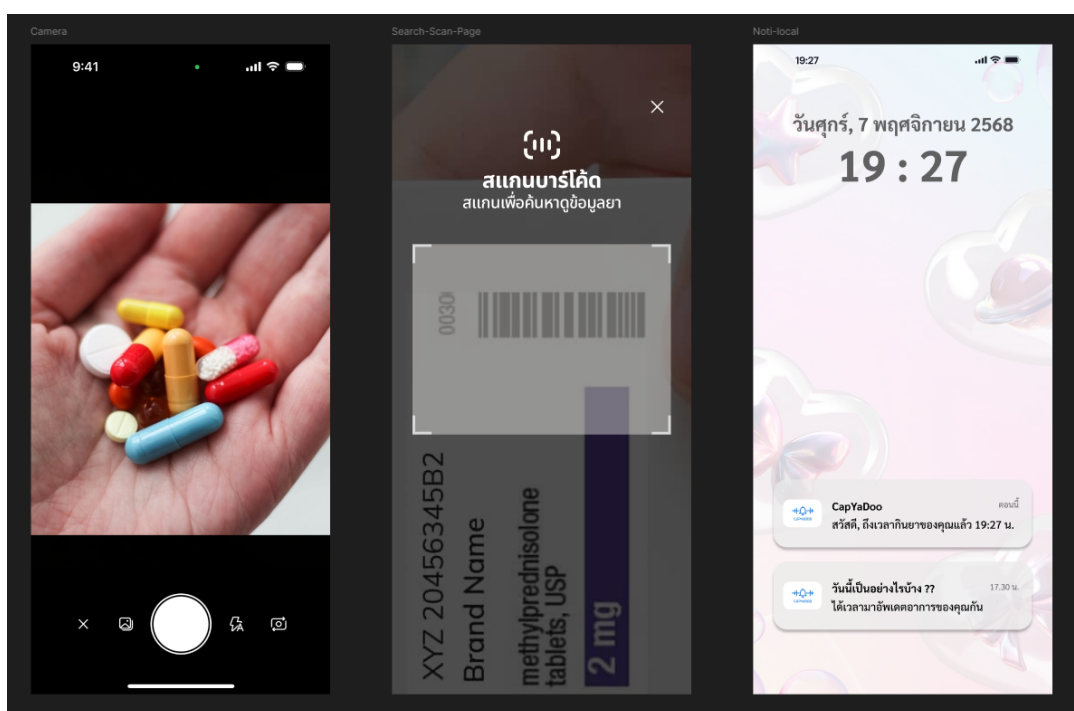
รูป 4.19: หน้าสำหรับการจัดการข้อมูลอาการหลังการไข้ยา



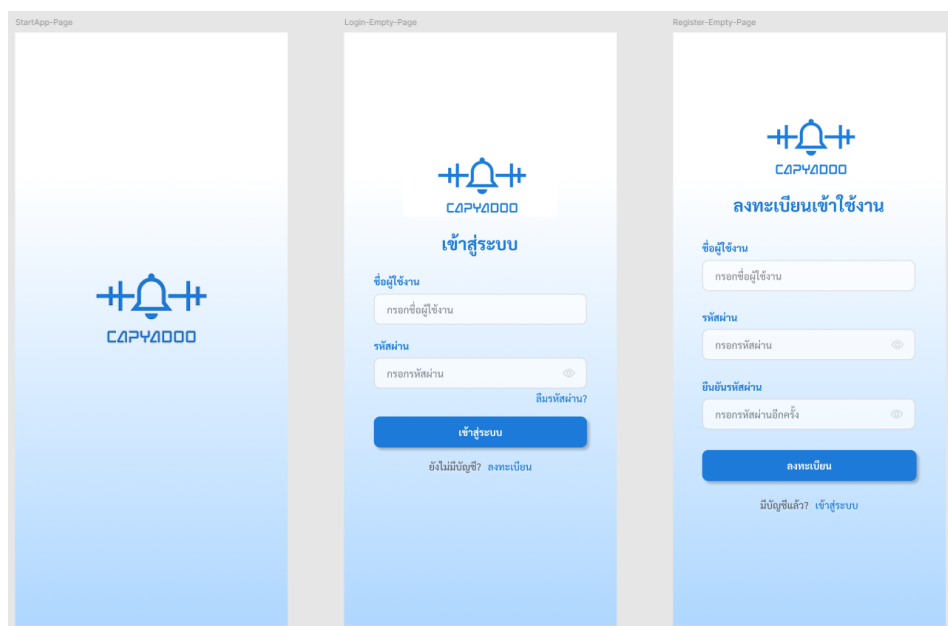
รูป 4.20: หน้าสำหรับการจัดการตั้งค่าการแจ้งเตือน



รูป 4.21: หน้าสำหรับการค้นหาข้อมูลยา



รูป 4.22: หน้าสำหรับการถ่ายภาพยาเพื่อยืนยันการทานยา การสแกนบาร์โค้ด และการแจ้งเตือนการทานยา



รูป 4.23: หน้าสำหรับการเข้าสู่ระบบ

4.2 ขั้นตอนการพัฒนา

การพัฒนาโครงการนี้ใช้แนวทางการพัฒนาแบบ Agile Framework โดยแบ่งการพัฒนาออกเป็น 8 Sprint เพื่อให้สามารถปรับปรุงและพัฒนาระบบได้อย่างต่อเนื่อง โดยแต่ละ Sprint มีระยะเวลา 2 สัปดาห์ โดยแบ่งการทำงานออกเป็น 6 ขั้นตอนหลัก ดังนี้

4.2.1 ขั้นตอนที่ 1: การรวบรวมปัญหา วิเคราะห์ความต้องการของผู้ใช้งานและการออกแบบ

ในขั้นตอนนี้ได้ทำการศึกษาและวิเคราะห์ปัญหาของกลุ่มเป้าหมาย ซึ่งเป็นผู้สูงอายุที่ต้องรับประทานยาหลายชนิดต่อวัน จากนั้นรวบรวมความต้องการของผู้ใช้งาน (User Requirement) และออกแบบระบบ (System Design) ครอบคลุมการออกแบบส่วนติดต่อผู้ใช้ (UI Design), ฐานข้อมูล, การแยกฟังก์ชันการทำงานของระบบ (Functional Decomposition) และการออกแบบกระบวนการทำงาน (Data Flow Diagram: DFD)

4.2.2 ขั้นตอนที่ 2: การศึกษาและพัฒนา Data Pipeline ด้วย PySpark

ขั้นตอนนี้เป็นการออกแบบและพัฒนากระบวนการ Data Pipeline เพื่อจัดการข้อมูลจากแหล่งข้อมูลภายนอก ได้แก่ ข้อมูลจากระบบ Thai Medicine Terminology (TMT) จากสำนักพัฒนามาตรฐานระบบข้อมูลสุขภาพไทย ซึ่งให้ข้อมูลในรูปแบบไฟล์ Excel และสำนักงานคณะกรรมการอาหารและยา โดยให้ข้อมูลผ่านเว็บไซต์ จากนั้นใช้ PySpark ในการประมวลผล ทำความสะอาดข้อมูล (Data Cleaning) และแปลงข้อมูลให้อยู่ในรูปแบบที่เหมาะสมก่อนนำเข้าฐานข้อมูล PostgreSQL เพื่อให้ระบบ Data Pipeline สามารถทำงานร่วมกับส่วน Backend และ Database ได้อย่างราบรื่น

4.2.3 ขั้นตอนที่ 3: การจัดการ Container ด้วย Kubernetes in Docker (Kind)

มีการใช้ Kind ในการควบคุมการทำงานของ Container ทั้งหมด โดยภายใน Cluster จะประกอบด้วย

1. Container ของ PySpark สำหรับประมวลผลและจัดการข้อมูล
2. Container ของ Spring Boot สำหรับให้บริการ API
3. Container ของ PostgreSQL สำหรับเก็บข้อมูลผ่านการประมวลผลแล้ว

4.2.4 ขั้นตอนที่ 4: พัฒนาระบบในส่วนของหน้าบ้าน (Frontend)

ทำการออกแบบและพัฒนาแอปพลิเคชันบนระบบปฏิบัติการ Android โดยใช้ Flutter Framework ร่วมกับภาษา Dart โดยพัฒนาหน้าจอต่าง ๆ เช่น หน้าสำหรับตั้งแจ้งเตือนการทานยา หน้าสำหรับการค้นหาข้อมูลยาฯลฯ หลังจากนั้นทำการเชื่อมต่อกับระบบ API เพื่อให้สามารถดึงและบันทึกข้อมูลจากส่วน Backend ได้อย่างถูกต้อง ซึ่งแบ่งเป็น Sprint ดังนี้

1. Sprint1: พัฒนาหน้าสำหรับค้นหาข้อมูลยาด้วยการสแกนบาร์โค้ด และการอ่านข้อความบนฉลากยา (OCR)
2. Sprint2: พัฒนาระบบการจัดการข้อมูลยา ครอบคลุมฟังก์ชันการเพิ่ม ลบ และแก้ไขข้อมูลยา
3. Sprint3: พัฒนาระบบการจัดการข้อมูลอาการหลังการใช้ยา ครอบคลุมฟังก์ชันการเพิ่ม ลบ และแก้ไขข้อมูล
4. Sprint4: ทดสอบการทำงานของระบบทั้งหมด เพื่อตรวจสอบความถูกต้องและประสิทธิภาพ
5. Sprint5: พัฒนาระบบการแจ้งเตือนการทานยา ครอบคลุมฟังก์ชันการตั้งเวลาและแจ้งเตือนการทานยา
6. Sprint7: พัฒนาระบบการจัดการข้อมูลผู้ใช้งาน ครอบคลุมฟังก์ชันการเพิ่ม ลบ และแก้ไขข้อมูลผู้ใช้งาน
7. Sprint8: ทดสอบการทำงานของระบบทั้งหมด โดยมีการทดสอบกับผู้ใช้งานจริงเพื่อรับฟังความคิดเห็น

4.2.5 ขั้นตอนที่ 5: การพัฒนาระบบในส่วนของหลังบ้าน (Backend)

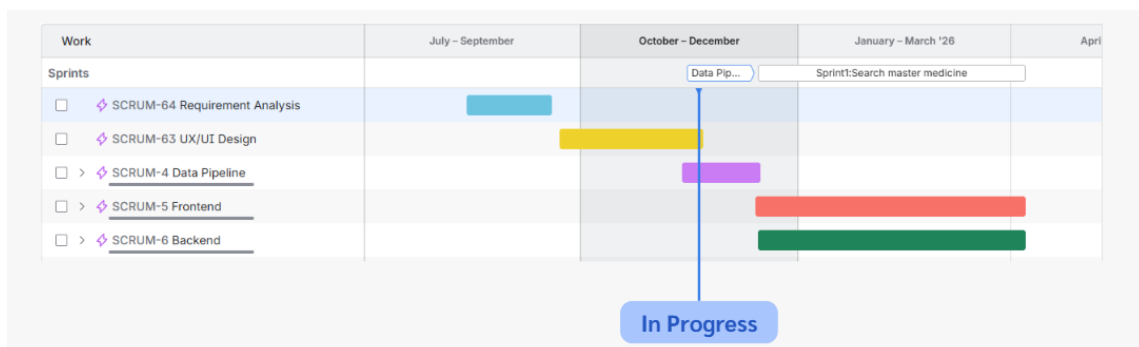
ใช้ Spring Boot Framework ในการสร้าง REST API ที่เชื่อมต่อระหว่างแอปพลิเคชันกับฐานข้อมูล PostgreSQL ซึ่งทำหน้าที่เก็บข้อมูลยา ข้อมูลผู้ใช้ ตารางการทานยา และข้อมูลการบันทึกประวัติการรับประทานยา โดยออกแบบ API endpoints ให้รองรับการทำงานต่าง ๆ เช่น การเพิ่ม แก้ไข และลบข้อมูลยา การตั้งค่าการแจ้งเตือน และการดึงข้อมูลยาจากฐานข้อมูล ซึ่งแบ่งเป็น Sprint ดังนี้

1. Sprint1: พัฒนา API สำหรับค้นหาข้อมูลยา
2. Sprint2: พัฒนา API สำหรับเพิ่ม ลบ และแก้ไขข้อมูลยา
3. Sprint3: พัฒนา API สำหรับเพิ่ม ลบ และแก้ไขอาการหลังการรับประทานยา

4. Sprint4: สร้าง Unit Test สำหรับทดสอบการทำงานของ API
5. Sprint5: พัฒนา API สำหรับจัดการการแจ้งเตือนการทานยา
6. Sprint6: พัฒนา API สำหรับจัดการหน้า Home Page
7. Sprint7: พัฒนา API สำหรับจัดการข้อมูลผู้ใช้งาน
8. Sprint8: สร้าง Unit Test สำหรับทดสอบการทำงานของ API

4.2.6 ขั้นตอนที่ 6: การทดสอบและปรับปรุงระบบ

ในขั้นตอนนี้ทำการทดสอบระบบทั้งหมดเพื่อตรวจสอบความถูกต้องและประสิทธิภาพของแอปพลิเคชัน โดยแบ่งการทดสอบออกเป็น Unit Testing สำหรับทดสอบฟังก์ชันแต่ละส่วน Integration Testing สำหรับทดสอบการทำงานร่วมกันระหว่างส่วนต่าง ๆ และ User Acceptance Testing (UAT) โดยให้กลุ่มผู้สูงอายุทดลองใช้งานจริง จากนั้นรวบรวม Feedback และข้อเสนอแนะจากผู้ทดสอบเพื่อนำมาปรับปรุงแก้ไขระบบให้มีความสมบูรณ์และตอบสนองความต้องการของผู้ใช้งานได้ดียิ่งขึ้น



รูป 4.24: แผนการดำเนินโครงการ

4.2.7 เครื่องมือและสภาพแวดล้อมในการพัฒนา

1. ภาษาโปรแกรม: Dart (สำหรับ Frontend), Java (สำหรับ Backend)
2. Framework: Flutter, Spring Boot
3. Integrated Development Environment (IDE): Android Studio และ Visual Studio Code สำหรับ Flutter, IntelliJ IDEA สำหรับ Spring Boot
4. จำลองบนเครื่องพัฒนา (Local Environment): Kubernetes in Docker (kind)
5. ฐานข้อมูล: PostgreSQL

4.3 การทดสอบและการประเมินผล

4.3.1 การทดสอบระบบ (System Testing)

ทดสอบการทำงานของฟังก์ชันหลักทั้งหมด ตั้งแต่การเข้าสู่ระบบไปจนถึงการใช้งานฟีเจอร์ต่างๆ เพื่อหาข้อผิดพลาด

4.3.2 การทดสอบประสิทธิภาพ (Performance Testing)

วัดความเร็ว ประสิทธิภาพ และความแม่นยำของฟังก์ชันการทำงาน

4.3.3 การทดสอบการยอมรับจากผู้ใช้งาน (User Acceptance Testing - UAT)

นำแอปพลิเคชันไปให้กลุ่มผู้สูงอายุทดลองใช้จริง เพื่อรวบรวมข้อเสนอแนะและปัญหาที่พบจากการใช้งาน เพื่อนำมาปรับปรุงเพิ่มเติม

4.4 ผลลัพธ์เบื้องต้นจากการพัฒนา

4.4.1 การทดสอบระบบเบื้องต้นในสภาพแวดล้อม Container ด้วย Kind

1. สร้าง Kind Cluster จำลองบนเครื่องพัฒนา (Local Environment) โดยมี container ดังนี้ Spring Boot Application, PySpark และ ฐานข้อมูล PostgreSQL

```
● PS C:\Users\jirac\OneDrive\Desktop\CapYaDoo\kind01> kubectl get pods
NAME                                READY   STATUS    RESTARTS   AGE
backend-5d75bb88f4-v757g            1/1     Running   1 (2m27s ago)    14d
jupyterlab-b68db857d-8p9bq          1/1     Running   1 (2m27s ago)    14d
postgres-6466d87b6d-l7qq5           1/1     Running   1 (2m27s ago)    14d
spark-master-7d59c445cb-l466t       1/1     Running   1 (2m27s ago)    14d
spark-worker-1-8599c5565c-d9rcr     1/1     Running   23 (2m27s ago)   14d
spark-worker-2-5f95b54cf9-6g6gl     1/1     Running   22 (2m27s ago)   14d
```

รูป 4.25: รูปผลลัพธ์ container ที่สร้างขึ้นจาก Kind

4.4.2 กระบวนการ Data Pipeline

1. สามารถอ่านข้อมูลจากไฟล์ Excel ของ Thai Medicine Terminology (TMT) ได้
2. PySpark สามารถประมวลผลข้อมูลที่ได้จากการอ่านไฟล์ Excel ของ Thai Medicine Terminology (TMT) ได้
3. สามารถโหลดข้อมูลที่อ่านได้ไปยังฐานข้อมูล PostgreSQL ได้

	A2 tpu_code	A2 active_ingredient	A2 strength	A2 dosageform	A2 cont_value	A2 cont_unit	A2 disp_unit	A2 trade_name	A2 manufacturer	A2 fsn
1	100005	alginate acid + dried aluminium	200 mg + 30 mg +	chewable tablet	[NULL]	[NULL]	tablet	ALGYCON	TAI YU CHEMICAL & PHARMACEU	ALGYCON (TAI YU CHEMICAL & PHARMACEU)
2	100014	bismuth subsalicylate	262 mg	tablet	[NULL]	[NULL]	tablet	GASTRO-BISMOL 262	นางเอกเคมี แอนด์ คอสมอล	GASTRO-BISMOL 262 (นางเอกเคมี แอนด์ คอสมอล)
3	100022	bismuth subsalicylate	262 mg	tablet	[NULL]	[NULL]	tablet	BESMAL	บริษัทเบสแมล จำกัด	BESMAL (บริษัทเบสแมล จำกัด) (bismuth subsalicylate)
4	100033	bismuth subsalicylate	262 mg	tablet	[NULL]	[NULL]	tablet	BRISTROL	บริษัทบริสตอล จำกัด	BRISTROL (บริษัทบริสตอล จำกัด) (bismuth subsalicylate)
5	100046	bismuth subsalicylate	524 mg	tablet	[NULL]	[NULL]	tablet	GASTRO-BISMOL 524	นางเอกเคมี แอนด์ คอสมอล	GASTRO-BISMOL 524 (นางเอกเคมี แอนด์ คอสมอล)
6	100051	bismuth subsalicylate	262 mg/15 mL	oral suspension	240	mL	bottle	GASMOL	อีลิคเอด	GASMOL (อีลิคเอด) (bismuth subsalicylate 262 mg
7	100080	bismuth subsalicylate	262 mg/15 mL	oral suspension	240	mL	bottle	GASTRO-BISMOL (ORA)	นางเอกเคมี แอนด์ คอสมอล	GASTRO-BISMOL (ORANGE) (นางเอกเคมี แอนด์ คอ
8	100098	bismuth subsalicylate	262 mg/15 mL	oral suspension	240	mL	bottle	GASTRO-BISMOL	นางเอกเคมี แอนด์ คอสมอล	GASTRO-BISMOL (นางเอกเคมี แอนด์ คอสมอล) (bisi
9	100108	bismuth subsalicylate	262 mg/15 mL	oral suspension	240	mL	bottle	TUMMY-MATE	ห้าแย่นิโ	TUMMY-MATE (ห้าแย่นิโ) (bismuth subsalicylate
10	100120	bismuth subsalicylate	524 mg/15 mL	oral suspension	240	mL	bottle	GASTRO-BISMOL FORTE	นางเอกเคมี แอนด์ คอสมอล	GASTRO-BISMOL FORTE (นางเอกเคมี แอนด์ คอสม
11	100131	cimetidine	200 mg	film-coated tablet	[NULL]	[NULL]	tablet	ALSERINE 200	ซานิโรฟีน	ALSERINE 200 (ซานิโรฟีน) (cimetidine 200 mg) fi
12	100149	cimetidine	200 mg	tablet	[NULL]	[NULL]	tablet	BEMETIDINE-200	ห้าแย่นิโ	BEMETIDINE-200 (ห้าแย่นิโ) (cimetidine 200 mg)
13	100154	cimetidine	200 mg	film-coated tablet	[NULL]	[NULL]	tablet	C.M.D.	โชนเซฟเอด	C.M.D. (โชนเซฟเอด) (cimetidine 200 mg) film-c
14	100165	cimetidine	200 mg	tablet	[NULL]	[NULL]	tablet	C.R. MET	ซี.อาร์.เม็ท	C.R. MET (ซี.อาร์.เม็ท) (cimetidine 200 m
15	100177	cimetidine	200 mg	tablet	[NULL]	[NULL]	tablet	CANCIMET 200	นางเอกเคมี แอนด์ คอสมอล	CANCIMET 200 (นางเอกเคมี แอนด์ คอสมอล) (cimet
16	100183	cimetidine	200 mg	film-coated tablet	[NULL]	[NULL]	tablet	CENCAMET W	ฟ้านิยามเอด	CENCAMET W (ฟ้านิยามเอด) (cimetidine 200 mg) film-co
17	100196	cimetidine	200 mg	film-coated tablet	[NULL]	[NULL]	tablet	CIAMET-200	ปัปเม็ท	CIAMET-200 (ปัปเม็ท) (cimetidine 200 mg) film-co
18	100203	cimetidine	200 mg	tablet	[NULL]	[NULL]	tablet	CIDIMET 200	บุคคิมเม็ท	CIDIMET 200 (บุคคิมเม็ท) (cimetidine 200 mg) t
19	100219	cimetidine	200 mg	film-coated tablet	[NULL]	[NULL]	tablet	CIGAMET	แนนเอจเอด	CIGAMET (แนนเอจเอด) (cimetidine 200 m

รูป 4.26: รูปผลลัพธ์จากฐานข้อมูล PostgreSQL ที่ได้จากการประมวลผลข้อมูลด้วย PySpark

4.4.3 Backend

1. สามารถเชื่อมต่อกับฐานข้อมูล PostgreSQL ได้
2. สามารถพัฒนา API สำหรับดึงข้อมูลยาออกมาได้

```
{
  "tpuCode": "1049338",
  "activeIngredient": "diphenylcyclopropenone",
  "strength": "1 mg/100 mL",
  "dosageform": "cutaneous solution",
  "contValue": "5",
  "contUnit": "mL",
  "dispUnit": "bottle",
  "tradeName": "0.001% DPCP",
  "manufacturer": "สถาบันโรคผิวหนัง",
  "fsn": "0.001% DPCP (F 12249) (diphenylcyclopropenone 1 mg/100 mL) cutaneous solution, 5 mL bottle (TPU)"
}
```

รูป 4.27: รูปผลลัพธ์ข้อมูลจาก API ที่ได้พัฒนาด้วย Spring Boot

เอกสารอ้างอิง

- [1] กรมอนามัย. “สถานการณ์ สติติกลุ่มโรคไม่ติดต่อเรื้อรัง (Non-Communicable diseases)”; 2022, สืบค้นเมื่อ: 17 สิงหาคม 2568, [ออนไลน์], [Online]. Available: <https://share.google/Dsz4woehlYvHsfYik>.
- [2] ผศ. ดร. พญ. มยุรี หอมสนิท. “NCD โรคไม่ติดต่อ”; 2557, สืบค้นเมื่อ 29 สิงหาคม 2568, [ออนไลน์], [Online]. Available: <https://share.google/fLVqWdcNTidgS22mZ>.
- [3] พิชญา นิลรุ่งรัตน และ อติเทพ แจ้ดนาลาว. “Human-Computer Interaction”; 2022, VOL.9-NO.1 JANUARY–JUNE 2022; สืบค้นเมื่อ: 20 กันยายน 2568, [ออนไลน์], [Online]. Available: <https://share.google/mikVx1SaFMp3w5haR>.
- [4] Appmaster. “Fundamentals of Database Systems”; 10 2022, สืบค้นเมื่อ: 1 กันยายน 2568, [ออนไลน์], [Online]. Available: <https://appmaster.io/th/blog/postgresql-khuue-aair>.
- [5] AWS. “OCR (การรู้จำอักขระด้วยแสง) คืออะไร”; 2025, สืบค้นเมื่อ: 20 กันยายน 2568, [ออนไลน์], [Online]. Available: <https://share.google/brT7qg1wnMVyyZ9S9>.
- [6] VMware, Inc. . “Spring Boot: Build anything with Spring Boot”; 2024, สืบค้นเมื่อ: 20 กันยายน 2568, [ออนไลน์], [Online]. Available: <https://spring.io/projects/spring-boot>.
- [7] Google LLC. “Flutter: Build apps for any screen”; 2024, สืบค้นเมื่อ: 20 กันยายน 2568, [ออนไลน์], [Online]. Available: <https://flutter.dev>.
- [8] Oracle. “Java”; 2025, สืบค้นเมื่อ 6 พฤศจิกายน 2568, [ออนไลน์], [Online]. Available: <https://www.oracle.com/java/>.
- [9] Dart. “Dart”; 9 2025, สืบค้นเมื่อ: 20 กันยายน 2568, [ออนไลน์], [Online]. Available: <https://dart.dev/overview>.
- [10] Atlassian. “What is the Agile methodology?”; 6 2025, สืบค้นเมื่อ 6 พฤศจิกายน 2568, [ออนไลน์], [Online]. Available: https://www.atlassian.com/agile?utm_source=chatgpt.com.
- [11] อนุรักษ์ ศรีบุรีมย์. “การพัฒนาและการประเมินระบบโปรแกรมประยุกต์บนโทรศัพท์มือถือ เพื่อช่วยในการอ่านฉลากยาสำหรับผู้สูงอายุ”; มีนาคม 2024, สืบค้นเมื่อ: 20 กันยายน 2568, [ออนไลน์], [Online]. Available: <https://he01.tci-thaijo.org/index.php/TJPP/article/view/260634>.

- [12] วิศว์สดี ปาริยะประเสริฐ. “การพัฒนาแอปพลิเคชันบน LINE LIFF ร่วมกับ Google Apps Script สำหรับ ติดตามและประเมินผล ความสม่ำเสมอในการรับประทานยา ของผู้ป่วยวัณโรค โรงพยาบาลชัยนาทนเรนทร”; 2024, [ออนไลน์], [Online]. Available: <https://share.google/DS1rM9eBJ3zmpBa8P>.
- [13] วิรุฬห์ ศรีบริรักษ์. “โครงการวิจัยเรื่อง เทคโนโลยีสารสนเทศแสดงข้อมูลฉลากยาเอกสารกำกับยาแบบอัตโนมัติสำหรับ บริหารจัดการคลังยาปฏิชีวนะเพื่อความปลอดภัย”; 2018, [ออนไลน์], [Online]. Available: <https://share.google/JaCyThQY9ISZ5uNNf>.
- [14] สมาคมการค้าเฮลท์เทคโนโลยี. “PharmaSee”; 2020, [ออนไลน์], [Online]. Available: <https://www.thaihealthtech.com/thta-applications-and-services/pharmasee-app>.
- [15] NECTEC. “MyYaAndYou”; 2018, [ออนไลน์], [Online]. Available: <https://software.thaiware.com/4396-MyYaAndYou-App.html>.
- [16] Medisafe. “Medisafe”; 2025, [ออนไลน์], [Online]. Available: https://play.google.com/store/apps/details/Pill_Reminder_Medication_Tracker_Medisafe?id=com.medisafe.android.client&hl=th.
- [17] ผศ. ดร. จริยา วิทยะศุภกร และ ผศ. ดร. จุฬารักษ์ กวีวิรัชชัย. “RDU”; 2017, [ออนไลน์], [Online]. Available: <https://www.rama.mahidol.ac.th/nursing/sites/default/files/public/knowledge/doc/1.pdf>.
- [18] DIGITORY CONTENT. “PERSONA คืออะไร สำคัญอย่างไรต่อการสร้างแบรนด์”; 5 2023, สืบค้นเมื่อ: 10 กันยายน 2568, [ออนไลน์], [Online]. Available: <https://ditorystyle.com/what-is-persona/>.
- [19] Figma. “User journey mapping: What it is + how to do it”; 2025, [ออนไลน์], [Online]. Available: <https://www.figma.com/resource-library/user-journey-map>.
- [20] Ihechikara Abba. “Crow’s Foot Notation – Relationship Symbols And How to Read Diagrams”; 6 2022, สืบค้นเมื่อ 29 สิงหาคม 2568, [ออนไลน์], [Online]. Available: <https://www.freecodecamp.org/news/crows-foot-notation-relationship-symbols-and-how-to-read-diagrams>.
- [21] Figma. “Figma Design”; 2025, [ออนไลน์], [Online]. Available: <https://www.figma.com/design/>.

ข้อความเปิดเผยการใช้เทคโนโลยีปัญญาประดิษฐ์

ข้าพเจ้าขอรับรองว่าได้มีการใช้เทคโนโลยี Generative AI ในการจัดทำกิจกรรมหรือชิ้นงานนี้ โดยนำเครื่องมือ Generative AI (.....,,) มาใช้เพื่อวัตถุประสงค์ดังต่อไปนี้:

- ☐ ระดมความคิด
- ☐ ทบทวนวรรณกรรมและผลงานที่เกี่ยวข้อง
- ☒ สรุปและถอดความเนื้อหา
- ☐ ร่างชิ้นงานเบื้องต้น
- ☒ ตรวจสอบและแก้ไขภาษา ไวยากรณ์หรือการแปล
- ☒ แก้ไขและตรวจสอบโค้ด
- ☒ เขียนโค้ด
- ☐ ใช้ AI เป็นส่วนหนึ่งในผลงาน (เช่น การใช้ generative AI API)
- ☐ อื่น ๆ (โปรดระบุ):

นางสาว ปรรธนา สุภาวงศ์

นาย ฤทธิช พลราช

นางสาว จิรัชญา ราชพลแสน