



รายงานฉบับสมบูรณ์

โครงการการพัฒนาาระบบเชื่อมโยงโปรแกรมที่เกี่ยวข้อง
กับบริการเพื่อการดูแลภาวะฉุกเฉินด้านการแพทย์
และสาธารณสุขจังหวัดเชียงใหม่

Consolidation databases of emergency care systems
for public health emergency surveillance in Chiang Mai

จัดทำโดย

นายแพทย์วรัญญู จ่านองประสาทพร
นายสินสมุทร จันทรืทอง
นายนคร ธรรมโก๊ะ
นายยุทธนา ตาสุกา

สำนักงานสาธารณสุขจังหวัดเชียงใหม่
สำนักงานสาธารณสุขจังหวัดเชียงใหม่
สำนักงานสาธารณสุขจังหวัดเชียงใหม่
สำนักงานสาธารณสุขจังหวัดเชียงใหม่

สนับสนุนทุนอุดหนุนการวิจัย แผนงานยุทธศาสตร์เป้าหมาย (Spearhead)
ด้านสังคม : แผนงานระบบบริการสุขภาพ โดย สำนักงานการวิจัยแห่งชาติ (วช.)

ชื่อวิจัย: โครงการพัฒนาระบบเชื่อมโยงโปรแกรมที่เกี่ยวข้องกับบริการ
เพื่อการดูแลภาวะฉุกเฉินด้านการแพทย์และสาธารณสุข จังหวัดเชียงใหม่
Consolidation databases of emergency care systems for public
health emergency surveillance in Chiang Mai

ISBN: 978-616-398-569-9

ผู้วิจัย: นายแพทย์วีรัญญ์ จำนงประสาทร
นายสินสมุทร จันทร์ทอง
นายนคร ธรรมโทะ
นายยุทธนา ตาสุภา

บรรณาธิการ: รองศาสตราจารย์ ดร.วราภรณ์ บุญเชียง
ดร.เสาวลักษณ์ เศรษฐีกุล
นางสาวสุรภี ทานเคหาสน์
นางสาวสุนิสา เสนาหวาน

ออกแบบและพิมพ์: นางสาวอรุณวดี กรรมสิทธิ์

เจ้าของและผู้จัดพิมพ์: หน่วยบริหารจัดการและส่งมอบผลลัพธ์ (ODU)
คณะสาธารณสุขศาสตร์ มหาวิทยาลัยเชียงใหม่
239 ถ.ห้วยแก้ว ต.สุเทพ อ.เมือง จ.เชียงใหม่
โทรศัพท์ 0 5394 2504

พิมพ์ครั้งแรก: พฤษภาคม 2564

พิมพ์ที่: บริษัทสยามพิมพ์นานาชาติ จำกัด โทรศัพท์ 0 5321 6962

สนับสนุนโดย: สำนักงานการวิจัยแห่งชาติ (วช.)

คำนำ

ข้อมูลสาธารณสุขปัจจุบันใช้จังหวัดเชียงใหม่ใช้ข้อมูลจาก HDC ซึ่งเป็นศูนย์ข้อมูลของกระทรวงสาธารณสุขที่รับข้อมูล 43 แห่งจากทุกสถานบริการ รวมทั้งมีรายงานต่าง ๆ อย่างมากมาย ซึ่งก็สามารถนำมาใช้ประโยชน์ในการทำงานเป็นอย่างดี แต่ก็มีบางงานที่ต้องการข้อมูลหรือรายงาน แต่ไม่มีใน HDC เนื่องจากไม่มีข้อมูลใน 43 แห่ง จึงใช้วิธีการออกแบบฟอร์มกระดาษ หรือไฟล์ Excel หรือทำโปรแกรมเพิ่มเติม ทำให้เกิดภาระผู้ปฏิบัติงานเพิ่มขึ้น อีกทั้งยังต้องใช้กำลังคนและเวลาในการรวบรวมข้อมูลประมวลผลและแสดงรายงาน

สำนักงานสาธารณสุขจังหวัดเชียงใหม่ ได้รับทุนอุดหนุนการวิจัยโครงการ “การพัฒนาระบบเชื่อมโยงโปรแกรมที่เกี่ยวข้องกับบริการเพื่อการดูแลภาวะฉุกเฉิน ด้านการแพทย์และสาธารณสุข จังหวัดเชียงใหม่” ระยะเวลาดำเนินการวิจัยตั้งแต่วันที่ 15 ตุลาคม 2562 ถึงวันที่ 31 กรกฎาคม 2563 เนื่องจากประเทศไทยประสบปัญหาการแพร่ระบาดของ COVID-19 ส่งผลกระทบต่อการดำเนินกิจกรรมโครงการทำให้โครงการไม่สามารถดำเนินกิจกรรมได้ตามแผน คณะสาธารณสุขศาสตร์ มหาวิทยาลัยเชียงใหม่ ในฐานะหน่วยบริหารจัดการและส่งมอบผลลัพธ์ จึงให้ขยายระยะดำเนินการกิจกรรมโครงการจากที่กำหนดไว้เดิมในสัญญาให้สิ้นสุดวันที่ 31 กรกฎาคม 2563 ไปเป็นวันที่ 30 พฤศจิกายน 2563

ดังนั้น เพื่อให้การดำเนินโครงการ “การพัฒนาระบบเชื่อมโยงโปรแกรมที่เกี่ยวข้องกับบริการเพื่อการดูแลภาวะฉุกเฉิน ด้านการแพทย์และสาธารณสุข จังหวัดเชียงใหม่” สามารถดำเนินงานได้ตามเป้าหมายและผลสัมฤทธิ์ที่กำหนด คือ มีการเชื่อมต่อข้อมูลระหว่างสถานพยาบาลเพื่อสามารถประเมินติดตามสถานการณ์ของโรคและภัยสุขภาพที่เกี่ยวข้องกับระบบฉุกเฉิน ด้านการแพทย์และสาธารณสุข ตลอดจนมีการวิเคราะห์ และนำข้อมูลไปใช้เพื่อวางมาตรการทางด้านสาธารณสุขของจังหวัดเชียงใหม่ เพื่อให้ประชาชนชาวเชียงใหม่ได้รับการดูแลด้วยระบบรักษาพยาบาลฉุกเฉิน อย่างทันเวลา มีประสิทธิภาพ และต่อเนื่องเป็นระบบ

สารบัญ

	หน้า
คำนำ	ก
สารบัญภาพ	ง
สารบัญตาราง	ฉ
บทคัดย่อ	ช
บทที่ 1 บทนำ	1
1. ความเป็นมาและความสำคัญของปัญหา	1
2. วัตถุประสงค์	2
3. คำถามการวิจัย	2
4. คำนียามศัพท์	3
บทที่ 2 การทบทวนวรรณกรรม	5
1. ระบบสารสนเทศการแพทย์ฉุกเฉิน	5
2. ระบบสารสนเทศด้านการสาธารณสุขไทย	8
3. ระบบ Hospital Information System (HIS) HOSxP	16
4. ระบบ Hospital Information System HospitalOS	20
5. Application Programming Interface (API)	22
6. ภาษา PHP	23
7. โปรแกรมระบบจัดการฐานข้อมูลด้วย MySQL	24
8. กรอบแนวคิด	25
บทที่ 3 วิธีดำเนินการวิจัย	26
1. ประชากรและกลุ่มตัวอย่าง	26
2. เครื่องมือที่ใช้ในการวิจัย	27
3. ขั้นตอนและวิธีการรวบรวมข้อมูล	28
4. การวิเคราะห์ข้อมูล	33
บทที่ 4 ผลการศึกษา	34
1. การพัฒนาระบบเชื่อมโยง (Application Programming Interface: API)	34
2. ระบบบริหารจัดการรายงานด้านการแพทย์ฉุกเฉินและสาธารณสุข (CM dashboard)	45
3. ผลความเป็นไปได้ของการใช้ระบบเชื่อมโยงโปรแกรมที่เกี่ยวข้องกับบริการเพื่อ การดูแลภาวะฉุกเฉินด้านการแพทย์และสาธารณสุข จังหวัดเชียงใหม่	92

สารบัญ (ต่อ)

	หน้า
บทที่ 5 สรุปผลและข้อเสนอแนะ	96
1. ข้อจำกัดในการดำเนินการวิจัย	97
2. ข้อเสนอแนะในการนำไปใช้	97
เอกสารอ้างอิง	98
ภาคผนวก แบบสอบถามความพึงพอใจของผู้ใช้ “ระบบเชื่อมโยงโปรแกรมที่เกี่ยวข้องกับ บริการเพื่อการดูแลภาวะฉุกเฉินด้านการแพทย์และสาธารณสุขจังหวัดเชียงใหม่”	99

สารบัญภาพ

	หน้า
ภาพที่ 1 ความสัมพันธ์ของข้อมูลในระบบสารสนเทศการแพทย์ฉุกเฉิน	6
ภาพที่ 2 การรับส่งข้อมูลในระบบสารสนเทศการแพทย์ฉุกเฉิน	7
ภาพที่ 3 ลักษณะการทำงานของ Application Programming Interface	23
ภาพที่ 4 ผังระบบการพัฒนาาระบบเชื่อมโยงโปรแกรมฯ	29
ภาพที่ 5 ผังไหลเวียนการเชื่อมโยงข้อมูลผ่าน API	35
ภาพที่ 6 การทำงานของ API	35
ภาพที่ 7 ผังไหลเวียนระบบการจัดเก็บข้อมูลรายงานการส่งต่อ	36
ภาพที่ 8 ตัวอย่างข้อมูลแบบ JSON Format	37
ภาพที่ 9 การทำงานของโปรแกรม CMDS	37
ภาพที่ 10 หน้าการทำงานขณะประมวลผลและจัดส่งข้อมูลของโปรแกรม CMDS	38
ภาพที่ 11 การตั้งค่าการเชื่อมต่อกับฐานข้อมูลของ CMDS	45
ภาพที่ 12 ผังไหลเวียนการทำงานของ Dashboard	46
ภาพที่ 13 ตัวอย่างตารางฐานข้อมูล ชื่อตาราง t_data_dx	48
ภาพที่ 14 ตัวอย่างข้อมูลสรุปผลที่ API ดึงข้อมูลไปแสดง	49
ภาพที่ 15 ข้อมูลที่ถูกดึงมาเก็บไว้ใน Google Sheet	49
ภาพที่ 16 แสดงข้อมูล Data Source	57
ภาพที่ 17 หน้าจอออกแบบ Dashboard ใน Google Data Studio	57
ภาพที่ 18 ผังภาพบริบท Context Diagram	74
ภาพที่ 19 Data Flow Diagram Level 0	75
ภาพที่ 20 dashboard จำนวนและร้อยละการแพทย์ฉุกเฉิน	88
ภาพที่ 21 กราฟแสดงจำนวนผู้ป่วยวิกฤติ รายปี	89
ภาพที่ 22 กราฟร้อยละผู้ป่วยวิกฤตฉุกเฉิน แยกสถานบริการและปีงบประมาณ	89
ภาพที่ 23 กราฟแสดงร้อยละประชากรที่เจ็บป่วยวิกฤตฉุกเฉิน แยกตามสี	89
ภาพที่ 24 กราฟแสดงอัตราผู้ป่วยภาวะฉุกเฉิน แยกตามกลุ่มโรค	89
ภาพที่ 25 ตารางแสดงจำนวนผู้ป่วยวิกฤตฉุกเฉิน แยกตามกลุ่มโรค	90
ภาพที่ 26 เมนูการใช้บริการการแพทย์ฉุกเฉิน	90
ภาพที่ 27 dashboard แสดงจำนวนและร้อยละการใช้บริการการแพทย์ฉุกเฉิน	90

สารบัญภาพ (ต่อ)

	หน้า
ภาพที่ 28 กราฟแสดงจำนวนการใช้บริการการแพทย์ฉุกเฉิน แยกตามปีงบประมาณ	90
ภาพที่ 29 กราฟแสดงร้อยละการใช้บริการการแพทย์ฉุกเฉิน แยกตามหน่วยบริการและปีงบประมาณ	91
ภาพที่ 30 กราฟแสดงร้อยละผู้ป่วยวิกฤตฉุกเฉิน ได้รับการผ่านระบบการแพทย์ฉุกเฉิน แยกตามสีและปีงบประมาณ	91
ภาพที่ 31 กราฟแสดงร้อยละผู้ป่วยภาวะฉุกเฉิน ได้รับการติดตามดูแลอย่างต่อเนื่อง แยกตามกลุ่มโรคและปีงบประมาณ	91

สารบัญตาราง

	หน้า
ตารางที่ 1 โครงสร้างฐานข้อมูลกลาง (data structure)	30
ตารางที่ 2 จำนวนและร้อยละผู้ป่วยภาวะฉุกเฉิน แยกตามกลุ่มโรคและปีงบประมาณ	92
ตารางที่ 3 จำนวนและร้อยละของข้อมูลทั่วไปของกลุ่มตัวอย่างในระยะดำเนินการ (n=34)	92
ตารางที่ 4 ค่าเฉลี่ยและส่วนเบี่ยงเบนมาตรฐานของคะแนนความเป็นไปในการใช้ระบบ เชื่อมโยงโปรแกรมฯ (n = 34)	93
ตารางที่ 5 จำนวนและร้อยละของกลุ่มตัวอย่าง จำแนกตามความคิดเห็นที่มีต่อความเป็นไป ในด้านความง่ายในการนำไปใช้ (n = 34)	93
ตารางที่ 6 จำนวนและร้อยละของกลุ่มตัวอย่าง จำแนกตามความคิดเห็นที่มีต่อความเป็นไป ในด้านความปลอดภัยของข้อมูล (n = 34)	94
ตารางที่ 7 จำนวนและร้อยละของกลุ่มตัวอย่าง จำแนกตามความคิดเห็นที่มีต่อความเป็นไป ในด้านประสิทธิภาพการทำงาน (n = 34)	95

บทคัดย่อ

โครงสร้างฐานข้อมูลของแต่ละระบบสารสนเทศสาธารณสุขของแต่ละหน่วยบริการสาธารณสุขในจังหวัดเชียงใหม่ มีความแตกต่างกัน การใช้ระบบเชื่อมโยง (Application Programming Interface) จึงเป็นตัวกลางที่สำคัญในการเชื่อมโยงระบบข้อมูลที่แตกต่างกัน การศึกษาครั้งนี้เป็นการศึกษาแบบวิจัยและพัฒนา มีวัตถุประสงค์เพื่อพัฒนาระบบเชื่อมโยงโปรแกรมที่เกี่ยวข้องกับบริการเพื่อการดูแลภาวะฉุกเฉินด้านการแพทย์และสาธารณสุขในจังหวัดเชียงใหม่ กระบวนการในการพัฒนาระบบแบ่งออกเป็น 3 ระยะ คือ 1) ระยะวิเคราะห์สถานการณ์ 2) ระยะดำเนินการ และ 3) ระยะประเมินผล กลุ่มตัวอย่างในระยะวิเคราะห์สถานการณ์ คือ บุคลากรในสำนักงานสาธารณสุขจังหวัดเชียงใหม่และบุคลากรที่ดูแลระบบสารสนเทศในโรงพยาบาลจำนวน 30 คน และกลุ่มตัวอย่างในระยะดำเนินการและระยะประเมินผล คือ บุคลากรในสำนักงานสาธารณสุขจังหวัดเชียงใหม่และบุคลากรที่ใช้ระบบสารสนเทศในโรงพยาบาลจำนวน 34 คน โดยทำการคัดเลือกกลุ่มตัวอย่างแบบเจาะจง (Purposive sampling) เครื่องมือที่ใช้ในการศึกษาครั้งนี้ประกอบด้วย แบบสอบถามข้อมูลทั่วไป แนวคำถามในการสนทนากลุ่ม และแบบประเมินความเป็นไปได้ของการใช้ระบบเชื่อมโยงโปรแกรมที่เกี่ยวข้องกับบริการเพื่อการดูแลภาวะฉุกเฉินด้านการแพทย์และสาธารณสุขมีการวิเคราะห์ข้อมูลด้วยวิธีวิเคราะห์เชิงเนื้อหาและสถิติเชิงพรรณนา

ผลการศึกษาทำให้ได้ระบบเชื่อมโยงโปรแกรมที่เกี่ยวข้องกับบริการ เพื่อการดูแลภาวะฉุกเฉินด้านการแพทย์และสาธารณสุขในจังหวัดเชียงใหม่ ซึ่งเป็นระบบที่ทำการเชื่อมโยงข้อมูลด้านการรักษาพยาบาลเกี่ยวกับ PM2.5 และการแพทย์ฉุกเฉินจากแต่ละโรงพยาบาลหรือหน่วยงานที่เกี่ยวข้องเพื่อนำมาใช้ในการเฝ้าระวัง ติดตาม และวางแผนการดำเนินงานด้านสาธารณสุขในระดับจังหวัด เมื่อประเมินความเป็นไปได้ของการนำระบบที่พัฒนาขึ้นไปใช้ พบว่า ระบบเชื่อมโยงโปรแกรมที่เกี่ยวข้องกับบริการเพื่อการดูแลภาวะฉุกเฉินด้านการแพทย์และสาธารณสุขมีความง่าย ความปลอดภัย และประสิทธิภาพในการทำงานในระดับปานกลาง (Mean = 2.49 2.24 และ 2.42 ตามลำดับ) ซึ่งสามารถนำระบบนี้ไปใช้เป็นแนวทางในการพัฒนาต่อยอดให้ครอบคลุมส่วนงานด้านอื่นที่มีบริบทใกล้เคียงกันได้

บทที่ 1

บทนำ

1. ความเป็นมาและความสำคัญของปัญหา

ระบบการแพทย์ฉุกเฉิน (Emergency Care System: ECS) เป็นระบบการบริหารจัดการเพื่อให้ผู้เจ็บป่วยฉุกเฉินได้รับการดูแลรักษาที่มีคุณภาพและป้องกันภาวะทุพพลภาพที่อาจเกิดขึ้นทั้งในภาวะปกติและภาวะภัยสุขภาพ ประกอบด้วย การดูแลผู้ป่วยก่อนถึงโรงพยาบาล (Emergency medical services: EMS) การดูแลในห้องฉุกเฉิน (Emergency room) การส่งต่อระหว่างสถานพยาบาล (Referral System) และการจัดการสาธารณภัยด้านการแพทย์และสาธารณสุข (Disaster) แนวคิดการจัดการบริการ ECS คือ การพัฒนาห่วงโซ่คุณภาพ (Chain of Quality) ได้แก่ EMS คุณภาพ ER คุณภาพ Refer คุณภาพ และ Disaster คุณภาพ เพื่อสร้างห่วงโซ่แห่งการรอดชีวิต (Chain of Survival) ของผู้เจ็บป่วยวิกฤตฉุกเฉิน โดยมีเป้าประสงค์เพื่อพัฒนาระบบ ECS ที่มีคุณภาพและได้มาตรฐานอย่างครบวงจร (กองยุทธศาสตร์และแผนงาน สำนักงานปลัดกระทรวงสาธารณสุข, 2562)

แผนหลักการแพทย์ฉุกเฉินแห่งชาติ ฉบับที่ 2 (พ.ศ. 2556 - 2559) ยุทธศาสตร์ข้อที่ 5 กำหนดให้มีการพัฒนาระบบสารสนเทศและการสื่อสารด้านการแพทย์ฉุกเฉิน เพื่อรองรับการตัดสินใจเชิงนโยบาย การบริหารจัดการ การปฏิบัติการ และการประเมินผล ในขณะที่การพัฒนาฐานข้อมูลด้านการแพทย์ฉุกเฉิน ปัจจุบันของประเทศไทยใช้ระบบสารสนเทศการแพทย์ฉุกเฉิน (Information Technology for Emergency Medical Service system; ITEMS) เป็นฐานข้อมูลที่ใช้ในภาวะปกติ สำหรับการบันทึกข้อมูลการให้บริการผู้ป่วยฉุกเฉินทุกราย ตั้งแต่การรับแจ้งเหตุและสั่งการ การคัดกรองระดับความรุนแรงของผู้ป่วย การปฏิบัติการดูแลผู้ป่วยนอกโรงพยาบาล จนกระทั่งการนำส่งผู้ป่วยถึงห้องฉุกเฉิน นอกจากนี้ยังมีความพยายามที่จะนำฐานข้อมูลด้านสุขภาพที่มีอยู่แล้วมาใช้ประโยชน์ร่วมกัน ตัวอย่างเช่น การเชื่อมโยงฐานข้อมูลเกี่ยวกับการบาดเจ็บมาใช้ในการกำหนดทิศทางการป้องกัน การป้องกันการบาดเจ็บ พัฒนาระบบบริการดูแลรักษาและการส่งต่อผู้ป่วย โดยเชื่อมโยงฐานข้อมูล 6 ฐาน ได้แก่ ระบบ Injury Surveillances (IS) ระบบรายงานการบาดเจ็บ 19 สาเหตุ ระบบข้อมูลสาเหตุการตาย ระบบข้อมูลพฤติกรรมเสี่ยง BRFS ระบบข้อมูล 43 แฟ้มและระบบ ITEMS อย่างไรก็ตาม ระบบดังกล่าวยังไม่ตอบสนองต่อการบันทึกข้อมูลด้านการแพทย์ฉุกเฉินในสถานการณ์ภัยพิบัติมากนัก (การสาธารณสุขไทย, 2558)

จังหวัดเชียงใหม่ มีระบบข้อมูลโรงพยาบาล (Hospital Information System: HIS) ประกอบด้วย ระบบ SSB ระบบ HOS XP ระบบ Hospital OS ระบบ DataSoft ระบบ JHOS ระบบ JHCIS และระบบที่พัฒนาขึ้นเอง ความหลากหลายของระบบทำให้โครงสร้างข้อมูลแตกต่างกัน ภายใต้

โครงการปฏิรูประบบข้อมูลสุขภาพภาครัฐฯ เพื่อให้ทุกหน่วยบริการสุขภาพสามารถส่งออกข้อมูลจากฐานข้อมูลในรูปแบบโครงสร้างมาตรฐานเดียวกัน สำนักงานปลัดกระทรวงสาธารณสุข ได้ออกแบบชุดโครงสร้างมาตรฐานข้อมูลด้านการแพทย์และสุขภาพ โดยบูรณาการโครงสร้างมาตรฐานข้อมูลเป็น 43 แฟ้มในปัจจุบัน อย่างไรก็ตาม ข้อมูลตามโครงสร้าง 43 แฟ้มมาตรฐานก็ยังไม่เพียงพอที่จะนำมาใช้ประโยชน์ในการประเมิน ติดตามสถานการณ์ของโรคและภัยสุขภาพที่เกี่ยวข้องกับระบบการแพทย์ฉุกเฉิน ถึงแม้ว่าจะมีการบูรณาการเชื่อมโยงฐานข้อมูล 6 ฐาน ได้แก่ ระบบ IS ระบบรายงานการบาดเจ็บ 19 สาเหตุ ระบบข้อมูลสาเหตุการตาย ระบบข้อมูลพฤติกรรมเสี่ยง BRFS 43 แฟ้มมาตรฐาน และระบบ ITEMS ก็ตาม ที่สำคัญแต่ละระบบยังมีข้อมูลซ้ำซ้อนไม่สมบูรณ์และเป็นภาระของหน่วยบริการที่ต้องทำการบันทึกข้อมูลในหลายระบบ

ด้วยเหตุนี้ สำนักงานสาธารณสุขจังหวัดเชียงใหม่จึงมีแนวคิดที่จะนำเทคโนโลยีสารสนเทศ ได้แก่ ระบบเชื่อมโยง (Application Programming Interface: API) มาช่วยในการจัดการและการบูรณาการเชื่อมต่อระบบ HIS ของหน่วยบริการและระบบต่าง ๆ ได้แก่ ระบบฐานข้อมูล Refer ระบบฐานข้อมูล IS ระบบฐานข้อมูล ITEMS และระบบข้อมูล 43 แฟ้มมาตรฐานเพื่อให้สามารถประเมิน ติดตามสถานการณ์ของโรคและภัยสุขภาพที่เกี่ยวข้องกับระบบการแพทย์ฉุกเฉิน ตลอดจนนำไปวิเคราะห์และใช้ประโยชน์ในการวางมาตรการทางด้านสาธารณสุขของจังหวัดเชียงใหม่ เพื่อให้ประชาชนชาวเชียงใหม่ ได้รับการดูแลด้วยระบบรักษาพยาบาลฉุกเฉิน อย่างทันเวลา มีประสิทธิภาพ และต่อเนื่องอย่างเป็นระบบ

2. วัตถุประสงค์

1. เพื่อพัฒนาระบบเชื่อมโยงโปรแกรมที่เกี่ยวข้องกับบริการเพื่อการดูแลภาวะฉุกเฉินด้านการแพทย์และสาธารณสุขในจังหวัดเชียงใหม่
2. เพื่อศึกษาความเป็นไปได้ของการใช้ระบบเชื่อมโยงโปรแกรมที่เกี่ยวข้องกับบริการเพื่อการดูแลภาวะฉุกเฉินด้านการแพทย์และสาธารณสุขในจังหวัดเชียงใหม่

3. คำถามการวิจัย

1. ระบบเชื่อมโยงโปรแกรมที่เกี่ยวข้องกับบริการเพื่อการดูแลภาวะฉุกเฉินด้านการแพทย์และสาธารณสุขในจังหวัดเชียงใหม่ ที่พัฒนาขึ้นมีลักษณะเป็นอย่างไร
2. ระบบเชื่อมโยงโปรแกรมที่เกี่ยวข้องกับบริการเพื่อการดูแลภาวะฉุกเฉินด้านการแพทย์และสาธารณสุข ในจังหวัดเชียงใหม่มีความเป็นไปได้หรือไม่ อย่างไร

4. คำนิยามศัพท์

ระบบเชื่อมโยง (Application Programming Interface: API) หมายถึง เครื่องมือเชื่อมต่อเพื่อแลกเปลี่ยนข้อมูลต่าง ๆ ระหว่างองค์กรหรือเพื่อแลกเปลี่ยนข้อมูลจากระบบหนึ่งไปสู่ระบบอื่น ๆ

โปรแกรม หมายถึง ระบบจัดเก็บข้อมูลการให้บริการของหน่วยบริการ ได้แก่ ระบบ SSB ระบบ HOSxP ระบบ HospitalOS และระบบ JHCIS

ฐานข้อมูล หมายถึง ฐานข้อมูลในระบบ Refer ระบบ Item ระบบ IS ระบบ R506 ระบบ IMC และระบบ COC

ระบบสารสนเทศในโรงพยาบาล (Hospital Information System) หมายถึง เครื่องมือสำคัญในการขับเคลื่อนระบบการดูแลรักษาผู้ป่วยและระบบบริหารจัดการทรัพยากรต่าง ๆ ภายในโรงพยาบาล วิเคราะห์ข้อมูลแสดงผลเชิงสถิติเพื่อการตัดสินใจในการบริหารและการวางแผนงาน

หน่วยบริการ หมายถึง หน่วยงานที่ให้บริการด้านการแพทย์และสาธารณสุข สังกัดสำนักงานปลัดกระทรวงสาธารณสุขในจังหวัดเชียงใหม่ ได้แก่ โรงพยาบาลศูนย์ โรงพยาบาลทั่วไป โรงพยาบาลชุมชน โรงพยาบาลส่งเสริมสุขภาพตำบล และศูนย์สุขภาพชุมชนในจังหวัดเชียงใหม่

โรงพยาบาลศูนย์ หมายถึง หน่วยบริการสุขภาพระดับจังหวัด ที่มีขนาดและจำนวนเตียงผู้ป่วยเกิน 500 เตียงขึ้นไป ได้แก่ โรงพยาบาลนครพิงค์ มีจำนวนเตียงผู้ป่วย 609 เตียง ตั้งอยู่ที่อำเภอแมริม จังหวัดเชียงใหม่

โรงพยาบาลทั่วไป หมายถึง หน่วยบริการสุขภาพระดับอำเภอขนาดใหญ่ ที่มีขนาดและจำนวนเตียงผู้ป่วย ตั้งแต่ 200 – 500 เตียง ได้แก่ โรงพยาบาลฝางและโรงพยาบาลจอมทอง มีจำนวนเตียงผู้ป่วย 210 เตียง

โรงพยาบาลชุมชน (รพช.) หมายถึง หน่วยบริการสุขภาพระดับอำเภอ ที่มีเตียงสำหรับผู้ป่วย ตั้งแต่ 10 เตียงไปจนถึง 150 เตียง ได้แก่ โรงพยาบาลสันป่าตอง โรงพยาบาลสันทราย โรงพยาบาลเชียงดาว โรงพยาบาลหางดง โรงพยาบาลดอยสะเก็ด โรงพยาบาลแม่เมาะ โรงพยาบาลพร้าว โรงพยาบาลฮอด โรงพยาบาลแม่แตง โรงพยาบาลเทพรัตนเวชชานุกูลเฉลิมพระเกียรติ 60 พรรษา โรงพยาบาลสันกำแพง โรงพยาบาลสารภี โรงพยาบาลอมก๋อย โรงพยาบาลไชยปราการ โรงพยาบาลดอยเต่า โรงพยาบาลแม่วาง โรงพยาบาลแม่ออน โรงพยาบาลดอยหล่อ โรงพยาบาลสะเมิง โรงพยาบาลเวียงแหง และโรงพยาบาลวัดจันทร์เฉลิมพระเกียรติ 80 พรรษา

โรงพยาบาลส่งเสริมสุขภาพตำบล (รพ.สต.) หมายถึง หน่วยบริการสุขภาพระดับตำบลหรือหมู่บ้านในจังหวัดเชียงใหม่ มีจำนวนทั้งหมด 267 แห่ง

ภาวะฉุกเฉินด้านการแพทย์ หมายถึง ภาวะการณ์เจ็บป่วยฉุกเฉินจนถึงการดำเนินการให้ผู้ป่วยฉุกเฉินได้รับการบำบัดรักษาให้พ้นภาวะฉุกเฉิน จำแนกเป็นการปฏิบัติการในชุมชน การปฏิบัติการต่อผู้ป่วยฉุกเฉินทั้งนอกโรงพยาบาลและในโรงพยาบาล

ภาวะฉุกเฉินด้านสาธารณสุข หมายถึง เหตุการณ์การเกิดโรคและภัยคุกคามสุขภาพซึ่งมีลักษณะเข้าได้กับเกณฑ์อย่างน้อย 2 ใน 4 ประการ คือ 1) ทำให้เกิดผลกระทบทางสุขภาพอย่างรุนแรง 2) เป็นเหตุการณ์ที่ผิดปกติหรือไม่เคยพบมาก่อน 3) มีโอกาสที่จะแพร่ไปสู่พื้นที่อื่น และ 4) ต้องจำกัดการเคลื่อนที่ของผู้คนหรือสินค้า

บทที่ 2

การทบทวนวรรณกรรม

การศึกษาครั้งนี้เป็นการวิจัยและการพัฒนา (Research and Development) เพื่อพัฒนาระบบเชื่อมโยงโปรแกรมที่เกี่ยวข้องกับบริการเพื่อการดูแลภาวะฉุกเฉินด้านการแพทย์และสาธารณสุข จังหวัดเชียงใหม่ โดยแบ่งการศึกษาเป็น 3 ระยะ คือ 1) ระยะวิเคราะห์สถานการณ์ 2) ระยะดำเนินการ และ 3) ระยะประเมินผล การทบทวนวรรณกรรมครอบคลุมเนื้อหาที่เกี่ยวข้อง ดังนี้

1. ระบบสารสนเทศการแพทย์ฉุกเฉิน
2. ระบบสารสนเทศด้านการสาธารณสุขไทย

1. ระบบสารสนเทศการแพทย์ฉุกเฉิน

ระบบรับแจ้งเหตุและสั่งการภาวะฉุกเฉินทางการแพทย์ เป็นระบบที่ใช้ในการรับแจ้งเหตุการเกิดภาวะฉุกเฉินทางการแพทย์จากช่องทางสื่อสารต่าง ๆ เช่น โทรศัพท์ วิทยุ หรือช่องทางอื่น ๆ ที่ผู้พบเหตุการณ์สามารถแจ้งเข้ามาได้ โดยมีเจ้าหน้าที่ศูนย์รับแจ้งเหตุและสั่งการทำหน้าที่ในการรับแจ้งภาวะฉุกเฉินนั้น ซึ่งจะช่วยแนะนำการบันทึกข้อมูลและสอบถามเพื่อคัดกรองระดับภาวะฉุกเฉินตามอาการของผู้ป่วยที่ได้รับแจ้งเข้ามา จากนั้นจะทำการสอบถามถึงสถานที่เกิดเหตุ เพื่อระบุพิกัดในการเลือกชุดปฏิบัติการที่เหมาะสมและอยู่ใกล้่ออกปฏิบัติการ โดยจะใช้เวลาในการสอบถามทั้งหมดประมาณ 15 วินาที สำหรับภาวะฉุกเฉินที่เร่งด่วนที่สุดและอาจจะมากกว่านั้นขึ้นอยู่กับอาการที่ได้รับแจ้ง หลังจากนั้นเจ้าหน้าที่สั่งการภาวะฉุกเฉินทางการแพทย์จะสั่งให้ชุดปฏิบัติการที่เหมาะสมออกปฏิบัติการ และติดตามการปฏิบัติงานจนกระทั่งนำส่งผู้ป่วยถึงโรงพยาบาลปลายทาง หรือจนกว่าผู้ป่วยจะพ้นจากภาวะฉุกเฉิน (สถาบันการแพทย์ฉุกเฉินแห่งชาติ, 2553)

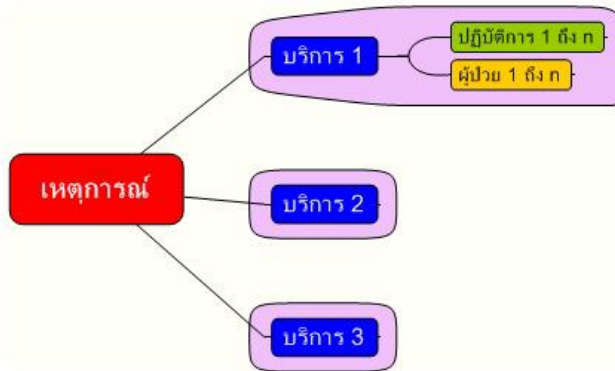
ระบบสารสนเทศการแพทย์ฉุกเฉินได้ทำการออกแบบใหม่ในทุกมุมมอง เพื่อสามารถนำข้อมูลไปใช้งานได้อย่างมีประสิทธิภาพครบถ้วนและตรงตามความต้องการมากที่สุด ดังนั้น จึงมีความจำเป็นอย่างยิ่งที่จะต้องมีการออกแบบด้านความสัมพันธ์ของข้อมูล และการรับส่งข้อมูลของระบบสารสนเทศการแพทย์ฉุกเฉินให้ตอบสนองตามโจทย์ที่วางไว้ให้มากที่สุด ดังรายละเอียดต่อไปนี้

ความสัมพันธ์ของข้อมูลในระบบสารสนเทศการแพทย์ฉุกเฉิน

การออกแบบความสัมพันธ์ของข้อมูลจะอ้างอิงตามเหตุการณ์ที่ได้รับแจ้งเหตุเข้ามาเป็นหลัก โดยที่หนึ่งเหตุการณ์จะประกอบไปด้วย

- 1) เหตุการณ์ เป็นข้อมูลการรับแจ้งจากผู้แจ้ง ซึ่งภายในเหตุการณ์จะประกอบไปด้วยบริการ และภายในหนึ่งเหตุการณ์ยังสามารถมีได้มากกว่า 1 บริการ

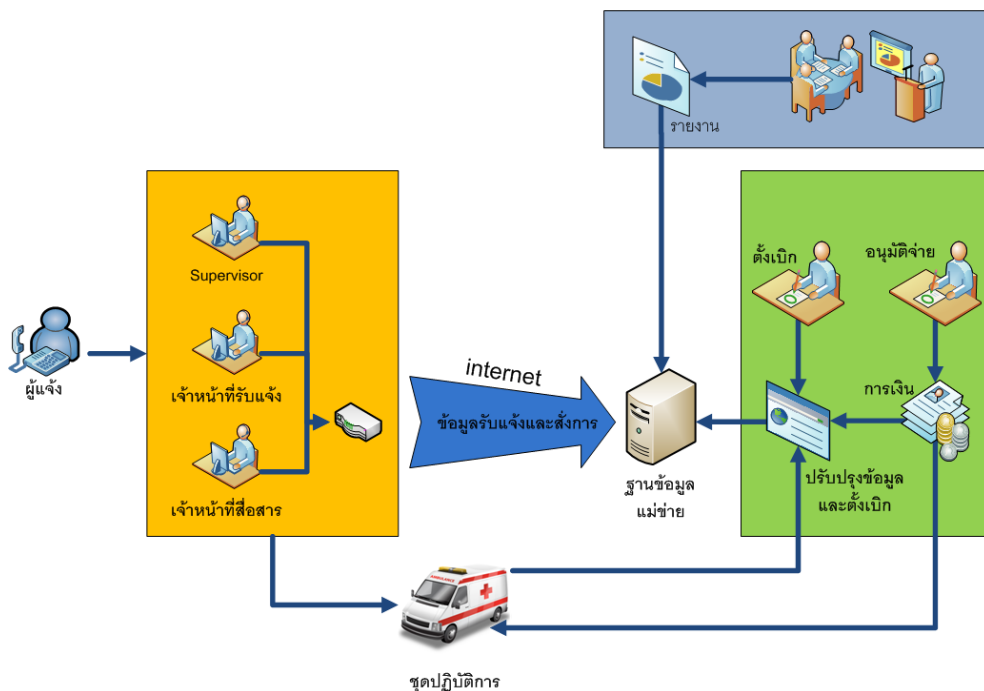
- 2) บริการ ซึ่งเราจะเปรียบเสมือนให้เป็นเส้นทางหรือเที่ยวในการออกให้บริการผู้ป่วย ภายในบริการจะประกอบไปด้วยปฏิบัติการซึ่งสามารถมีได้มากกว่า 1 ปฏิบัติการ และผู้ป่วยซึ่งสามารถมีได้มากกว่า 1 คน
- 3) ปฏิบัติการ เป็นข้อมูลการออกให้บริการของชุดปฏิบัติการ
- 4) ผู้ป่วย เป็นข้อมูลของผู้ป่วยที่ชุดปฏิบัติการที่อยู่ภายใต้บริการเดียวกันให้การดูแลซึ่งสามารถแสดงได้ตามรูปที่แสดงด้านล่าง



ภาพที่ 1 ความสัมพันธ์ของข้อมูลในระบบสารสนเทศการแพทย์ฉุกเฉิน

การรับส่งข้อมูลของระบบสารสนเทศการแพทย์ฉุกเฉิน

ระบบรับแจ้งเหตุและสั่งการภาวะฉุกเฉินทางการแพทย์เป็นระบบที่พัฒนาขึ้นมาให้สามารถทำงานได้แม้จะไม่มีสัญญาณอินเทอร์เน็ต (Internet) ใช้งาน นั่นหมายถึง สามารถทำงานในขณะ offline mode ได้นั่นเอง ศูนย์รับแจ้งเหตุและสั่งการการแพทย์ฉุกเฉินจะสามารถปฏิบัติงานได้ตามปกติ เพียงแต่ข้อมูลจะถูกเก็บไว้ที่เครื่องที่อยู่ฝั่งศูนย์สั่งการ ทั้งนี้ ข้อมูลจะถูกส่งขึ้นมาที่ server แม้ข้ายในภายหลังที่ internet ใช้งานได้หรืออยู่ใน online mode ทำให้สะดวกแก่การใช้งานและช่วยในการจัดการทรัพยากรได้อย่างมีประสิทธิภาพ



ภาพที่ 2 การรับส่งข้อมูลในระบบสารสนเทศการแพทย์ฉุกเฉิน

ในสหรัฐอเมริกา สำนักงานประสานงานแห่งชาติเพื่อเทคโนโลยีสารสนเทศด้านสุขภาพ (Office of the National Coordinator for Health Information Technology: ONC) เป็นหน่วยงานหลักที่รับผิดชอบการพัฒนาสารสนเทศสุขภาพและเทคโนโลยีสารสนเทศสุขภาพของรัฐบาลสหรัฐอเมริกา ได้กำหนดคุณสมบัติของระบบสารสนเทศสุขภาพที่ดี (Meaningful Use) กล่าวคือ ต้องเป็นระบบที่ทำหน้าที่และสามารถตอบสนองการทำงานและความต้องการของผู้ใช้ข้อมูลได้ทุกระดับ (สถาบันการแพทย์ฉุกเฉินแห่งชาติ, 2553) ได้แก่

1. ตอบสนองความต้องการในระบบบริการสุขภาพ (Health Care Delivery) เช่น ระบบความปลอดภัยและคุณภาพในการดูแลผู้ป่วย การเบิกจ่ายค่าบริการ การบริหารจัดการ การให้บริการ เป็นต้น
 2. ตอบสนองความต้องการในระบบงานด้านสาธารณสุข (Population and Public Health) เช่น ระบบงานด้านระบาดวิทยา ระบบงานด้านการป้องกันควบคุมโรค
 3. ตอบสนองความต้องการในระบบข้อมูลสำหรับประชาชน (Personal Health)
- นอกจากนี้ ระบบสารสนเทศสุขภาพและเทคโนโลยีสารสนเทศสุขภาพต้องเป็นระบบที่มีการบูรณาการ (Integrated Health Information) ของข้อมูลจากแหล่งต่าง ๆ ระบบข้อมูลสุขภาพต่าง

ระบบกันต้องสามารถแลกเปลี่ยนและเชื่อมโยงกันได้ระบบข้อมูลสุขภาพที่บูรณาการ และแลกเปลี่ยนเชื่อมโยงกันระหว่างหน่วยงาน (Integrated Health Information Systems) จะทำให้เกิดประโยชน์ระบบสุขภาพ ดังนี้

- 1) คุณภาพการบริการดีขึ้น (Improve Health Care Quality)
- 2) ป้องกันความผิดพลาดจากการให้บริการทางการแพทย์ (Prevent Medical Errors)
- 3) เพิ่มประสิทธิภาพของการบริการสุขภาพ (Increase the Efficiency of Care Provision)
- 4) ลดความสูญเปล่าจากการให้บริการสุขภาพเกินจำเป็น (Reduce Unnecessary Health Care Costs)
- 5) เพิ่มประสิทธิภาพการบริหารจัดการบริการสุขภาพ (Increase Administrative Efficiencies)
- 6) ลดงานด้านเอกสาร (Decrease Paperwork)
- 7) ทำให้ประเทศสามารถขยายการให้บริการด้านสุขภาพได้ทั่วถึง (Expand Access to Affordable Care)
- 8) งานสาธารณสุขได้รับการพัฒนาและมีประสิทธิภาพ (Improve Population Health)

2. ระบบสารสนเทศด้านการสาธารณสุขไทย

ระบบสารสนเทศด้านสาธารณสุขของประเทศไทย เริ่มต้นครั้งแรกด้วยพระมหากษัตริย์คุณของ “สมเด็จพระยา” ของปวงชนชาวไทย โดยหน่วยแพทย์อาสาสมเด็จพระราชชนนี (พอ.สว.) เป็นหน่วยแพทย์ที่สมเด็จพระศรีนครินทราบรมราชชนนีทรงก่อตั้งเมื่อ พ.ศ. 2512 ปฏิบัติงานโดยอาสาสมัครประกอบด้วย แพทย์ทันตแพทย์ เภสัชกร พยาบาล เจ้าหน้าที่สาธารณสุข และอาสาสมัครสายสนับสนุน ออกไปให้การ รักษาพยาบาลประชาชนในพื้นที่ทุรกันดาร หลังจากที่เสด็จฯ เยี่ยมราษฎรตามพื้นที่ชนบททางไกล นับตั้งแต่ พ.ศ. 2507 เป็นต้นมา ทรงพบว่า ราษฎรเหล่านั้นเมื่อเจ็บป่วยไม่มีโอกาสได้รับการรักษาจาก แพทย์แผนปัจจุบัน จึงทรงพระกรุณาโปรดเกล้าฯ ให้หน่วยแพทย์เคลื่อนที่โดยเสด็จฯ เสมอ ต่อมาในปี พ.ศ. 2516 ได้ทรงเริ่มนำระบบการสื่อสารทางวิทยุ รับ-ส่ง มาใช้ในการให้คำปรึกษาและรักษาผู้ป่วย ซึ่งเรียกว่า “แพทย์ทางอากาศ” หรือที่ต่อมาเรียกว่า “แพทย์ ทางวิทยุ” สมเด็จพระศรีนครินทราบรมราช ชนนีทรงเล็งเห็นว่า กิจการและการดำเนินการให้ความช่วยเหลือประชาชนในด้านการแพทย์และ การสาธารณสุขประสบผลและเป็นคุณประโยชน์อย่างมหาศาล จึงทรงพระกรุณาโปรดเกล้าฯ พระราชทาน ทรัพย์ส่วนพระองค์จำนวน 1 ล้านบาท เป็นทุนแรกเริ่มจดทะเบียน ตั้งเป็น “มูลนิธิแพทย์อาสาสมเด็จพระ ศรีนครินทราบรมราชชนนี (พอ.สว.)” (คณะอนุกรรมการการสาธารณสุข วุฒิสภา, 2554)

เมื่อวันที่ 21 พฤษภาคม พ.ศ. 2517 โดยพระองค์เป็นนายกิตติมศักดิ์หลังจากเสด็จสวรรคต เมื่อวันที่ 18 กรกฎาคม พ.ศ. 2538 สมเด็จพระเจ้าพี่นางเธอเจ้าฟ้ากัลยาณิวัฒนา กรมหลวงนราธิวาสราชนครินทร์ ทรงดำรงตำแหน่งประธานกิตติมศักดิ์สืบต่อมาตราบจนสิ้นพระชนม์เมื่อวันที่ 2 มกราคม

พ.ศ. 2551 ระบบสื่อสารด้านสาธารณสุขมีการพัฒนาตั้งแต่นั้นเป็นต้นมา โดยเฉพาะระบบสารสนเทศด้านสาธารณสุขของโรงพยาบาล ที่ได้พัฒนาและจัดทำกันเองในแต่ละโรงพยาบาล ซึ่งยังไม่มีระบบที่จะสามารถเชื่อมโยงข้อมูลสุขภาพของประชาชนให้เป็นมาตรฐานเดียวกันได้ทั้งหมด และในช่วงนี้ รัฐบาลได้มีการให้ความสำคัญกับเรื่องเมดิคัล ฮับ (Medical Hub) อาจจะมีผลทำให้ประชาชนผู้ด้อยโอกาสส่วนใหญ่จะถูกมองข้ามและจากระบบสื่อสารได้พัฒนาเป็นระบบสารสนเทศที่ให้ความสะดวกชัดเจน ในข้อมูลอย่างมาก จึงมีแนวความคิดที่จะนำระบบสารสนเทศเพื่อเข้ามาช่วยในการดูแลผู้ป่วยหรือผู้ด้อยโอกาสที่อยู่ในพื้นที่ที่ห่างไกลความเจริญในทุกภูมิภาคโดยมีแนวความคิดที่ว่าควรมีการฝึกอบรมพยาบาลหรือเจ้าหน้าที่ที่จะปฏิบัติงานในโรงพยาบาลส่งเสริมสุขภาพตำบล และโรงพยาบาลชุมชนที่อยู่ในพื้นที่ห่างไกลให้มีความรู้ในการใช้ระบบสารสนเทศเพื่อใช้ในการติดต่อประสานงานในด้านการรักษาให้แพทย์ในท้องถิ่นทุรกันดารได้มีการใช้เทคโนโลยีสารสนเทศเพื่อปรึกษาแพทย์ในโรงพยาบาลศูนย์ในกรณีที่ไม่มีแพทย์เฉพาะทาง เป็นการช่วยเหลือผู้ป่วยฉุกเฉินเพื่อให้มีการรักษาเบื้องต้นได้ทันต่อเหตุการณ์และช่วยลดความแออัดของโรงพยาบาลขนาดใหญ่ รวมทั้งหากมีระบบสารสนเทศด้านการสาธารณสุขที่ดีจะสามารถช่วยในด้านการส่งต่อผู้ป่วย ข้อมูลทะเบียนประวัติที่มีการเชื่อมโยง (link) และแลกเปลี่ยนกันได้โรงพยาบาลจะสามารถทราบประวัติของผู้ป่วยได้ทุกที่ทุกเวลา ผ่านระบบสารสนเทศที่ออกแบบให้นำประวัติผู้ป่วยมาใช้โดยไม่ผิดกฎหมาย อีกทั้งยังเป็นการเพิ่มประสิทธิภาพ การใช้สิทธิในการเบิกจ่ายที่บูรณาการกับกองทุนต่าง ๆ และกระทรวงการคลังและการบริหารการเงิน การคลัง และบุคลากรของหน่วยงานต่าง ๆ รวมถึงการตรวจสอบการเบิกจ่ายยาของผู้ป่วยในกรณีที่มีการเวียนขอเบิกยาตามโรงพยาบาลต่าง ๆ ได้ด้วย อีกทั้งจะช่วยให้การรักษาผู้ป่วยด้วยเวลาน้อยลง โดยแพทย์สามารถเรียกดูข้อมูลได้อย่างรวดเร็ว แม่นยำ ด้วยระบบที่เชื่อมต่อกันทั้งภายในและภายนอกโรงพยาบาล ผู้ป่วยไม่ต้องเสียเวลามากที่ต้องไปโรงพยาบาล รวมทั้งช่วยให้มีการติดตามในกรณีมีโรคระบาด โรคติดต่อได้อย่างรวดเร็ว ทันเวลาซึ่งจะช่วยให้การเป็น Medical Hub ได้ด้วย

คณะกรรมการการสาธารณสุข วุฒิสภา โดยคณะอนุกรรมการพิจารณาศึกษาระบบสารสนเทศด้านสาธารณสุขไทย ได้ตระหนักถึงความสำคัญของปัญหาดังกล่าว จึงได้ทำการศึกษาระบบสารสนเทศด้านการสาธารณสุขไทย ผลการศึกษาพบว่า ระบบบริการสุขภาพ (Service Delivery) โดยทั่วไปจะหมายถึงการบริการ ๔ ด้านหลัก ได้แก่ การรักษาพยาบาล การป้องกันและควบคุมโรค บริการส่งเสริมสุขภาพ และงาน บริการการฟื้นฟูสภาพ งานบริการสุขภาพเป็นงานสหวิทยาการ ซึ่งการสื่อสารกับคนที่มีอาชีพต่างกัน เป็นเรื่องที่จะต้องให้ความสำคัญเรื่องการบริหารจัดการข้อมูล และสื่อสารระหว่างวิชาชีพผู้ให้บริการ ระหว่างผู้ให้บริการและผู้รับบริการ ประเทศไทยจัดระบบบริการสุขภาพเป็นลำดับขั้น มีระบบการส่งต่อทำให้บริการ สุขภาพจากระดับต้นซึ่งให้บริการสุขภาพขั้นพื้นฐาน (Primary Care) ไปจนถึงการบริการที่ซับซ้อน ต้องใช้ผู้เชี่ยวชาญและเทคโนโลยีขั้นสูง (Tertiary & Super Tertiary Care) คือ 1) บริการสุขภาพระดับปฐมภูมิ (Primary Care) 2) บริการสุขภาพระดับทุติยภูมิ (Secondary Care) 3) บริการสุขภาพระดับตติยภูมิ (Tertiary Care) และ 4) บริการสุขภาพระดับสูงกว่าตติยภูมิ (Super-Tertiary Care)

ประเทศไทยเป็นหนึ่งในไม่กี่ประเทศในโลกที่มีระบบหลักประกันสุขภาพถ้วนหน้า ซึ่งประกอบด้วยหน่วยงานที่ดูแลกองทุนประกันสุขภาพใหญ่อยู่ 3 หน่วยงาน ได้แก่ สำนักงานหลักประกันสุขภาพถ้วนหน้า กรมบัญชีกลางซึ่งดูแลสวัสดิการค่ารักษาพยาบาลของข้าราชการและครอบครัว และสำนักงานประกันสังคม กองทุนค่ารักษาพยาบาลของทั้ง 3 หลักประกันสุขภาพมีวิธีการจ่ายเงินและกระบวนการจ่ายที่แตกต่างกัน ซึ่งโมเดลของการจ่ายเงินและระบบข้อมูลที่สนับสนุนกิจกรรมของทั้ง 3 กองทุนที่แตกต่างกันนี้เป็นส่วนสำคัญที่ทำให้ระบบข้อมูลสารสนเทศของประเทศไม่มีความเป็นเอกภาพ การออกแบบข้อมูลสารสนเทศสุขภาพของประเทศจึงต้องคำนึงถึงกลไกด้าน การเงินการคลังสุขภาพ และระบบบริการสุขภาพของประเทศด้วย ระบบสารสนเทศสุขภาพและสาธารณสุข เป็นองค์ประกอบที่สำคัญที่จำเป็นทำให้เกิดระบบสุขภาพที่ดี โดยเฉพาะอย่างยิ่งระบบข้อมูลสุขภาพ ระบบบริการสุขภาพ และระบบการคลังสุขภาพเป็นระบบย่อยที่ต้องมีปฏิสัมพันธ์กันอย่างใกล้ชิด การออกแบบระบบย่อยหนึ่งต้องคำนึงถึงระบบที่เกี่ยวข้องให้ทำงานสอดคล้องกัน ไม่สามารถออกแบบระบบย่อยเดี่ยว ๆ โดยไม่สนใจระบบอื่นได้ ดังนั้น การพัฒนาระบบสารสนเทศสุขภาพ (Health Information Technology) จะต้องมีการพิจารณาควบคู่ไปกับระบบการเงินการคลังสุขภาพ (Health Care Financing) และระบบบริการสุขภาพ (Health Care Delivery) ทั้งนี้เพราะทั้ง ๓ ระบบมีความสัมพันธ์กันอย่างใกล้ชิด มีผลกระทบต่อกัน การพัฒนาระบบข้อมูลสุขภาพจึงต้องมีความสอดคล้องกับบริบทของระบบการบริการสุขภาพ และกลไกทางการเงินการคลังด้านสุขภาพของประเทศโดยแบ่งระดับของระบบสารสนเทศสุขภาพ ได้เป็น 3 ระดับ (คณะอนุกรรมการพิจารณาศึกษาระบบสารสนเทศ ด้านการสาธารณสุขไทย ในคณะกรรมการการสาธารณสุข วุฒิสภา, 2554)

1. ระบบสารสนเทศในระดับปฏิบัติงาน (Operational or Transactional Information System) เช่น ระบบสารสนเทศการให้บริการในสถานอนามัย ระบบสารสนเทศการให้บริการผู้ป่วยนอกในโรงพยาบาล เป็นระบบที่สนับสนุนการทำงาน และการตัดสินใจของผู้ปฏิบัติการ เช่น แพทย์พยาบาล เจ้าหน้าที่สถานอนามัยที่จุดให้การรักษายาบาล หรือจุดให้บริการ (Point of Care)

2. ระบบสารสนเทศเพื่อการจัดการ (Management Information System) เป็นระบบสารสนเทศที่มีวัตถุประสงค์เพื่อสนับสนุนการตัดสินใจของการทำงานในระดับผู้จัดการเพื่อบริหารจัดการองค์กร เช่น หัวหน้าฝ่าย และผู้บริหารโรงพยาบาล ระบบสารสนเทศระดับนี้จะนำข้อมูลจากระบบสารสนเทศในระดับปฏิบัติงานขององค์กรมาจัดการจัดระเบียบให้เหมาะสมกับการตัดสินใจของคนทำงานในระดับผู้จัดการองค์กร

3. ระบบสารสนเทศเพื่อการบริหาร (Executive Information System) เป็นระบบสารสนเทศที่มีวัตถุประสงค์สนับสนุนการตัดสินใจการทำงานของผู้บริหารเพื่อการวางแผนยุทธศาสตร์ขององค์กรเพื่อการตัดสินใจในระดับสูง เป็นระบบสารสนเทศที่มีแหล่งข้อมูลที่กว้างขวางทั้งในและนอกองค์กร มีการนำข้อมูลสารสนเทศในระดับล่างมาจัดการให้เหมาะสมกับผู้บริหารที่ต้อง ตัดสินใจในระดับสูง ไม่ใช่

การแก้ปัญหาในระดับปฏิบัติการ แหล่งข้อมูลหน่วยบริการและหน่วยงานที่เกี่ยวข้อง (Institution-Based Data Sources) ข้อมูลด้านสุขภาพส่วนใหญ่มีแหล่งข้อมูลจากหน่วยบริการและหน่วยบริหารจัดการ

ข้อเสนอทางนโยบายการพัฒนาาระบบสารสนเทศสุขภาพและสาธารณสุขของประเทศ มีดังนี้คือ (คณะอนุกรรมการการสาธารณสุข วุฒิสภา, 2554)

1. ประเทศไทยควรมีองค์กรระดับประเทศที่ทำหน้าที่กำหนดทิศทางการพัฒนา กำหนด ยุทธศาสตร์วางนโยบายและแผนหลักการพัฒนาาระบบสารสนเทศและเทคโนโลยี สาธารณสุขของประเทศ องค์กรหรือหน่วยงานกลางนี้ควรประกอบด้วยผู้คนจากทุกหน่วยงาน ที่เกี่ยวข้องกับระบบสารสนเทศและเทคโนโลยีสารสนเทศสุขภาพทั้งภาครัฐ และเอกชน ควรเป็นหน่วยงานที่มีความเป็นอิสระในการบริหารงาน เช่น เป็นองค์กรมหาชน

2. กำหนดนโยบาย และยุทธศาสตร์การพัฒนาาระบบสารสนเทศและเทคโนโลยี สาธารณสุขสุขภาพของประเทศ ให้เป็นส่วนหนึ่งของกรอบนโยบายและยุทธศาสตร์ด้านเทคโนโลยี สาธารณสุขของประเทศในภาพรวม ควบคู่ไปกับยุทธศาสตร์และการพัฒนา งานด้านอื่น ๆ eGovernment, eEducation, eIndustry, eSociety and eCommerce

3. พัฒนามาตรฐานข้อมูลสุขภาพในทุกมิติรวมถึงกลไกในการดูแลมาตรฐาน เพื่อให้ระบบสารสนเทศต่าง ๆ ทำงานร่วมกันได้ สามารถแลกเปลี่ยนข้อมูลกันได้แบบไร้รอยต่อด้วยความปลอดภัยและเป็นประโยชน์

4. ดำเนินการออกกฎหมาย กฎระเบียบที่เกี่ยวข้องกับการรักษาความปลอดภัยและ ความลับส่วนบุคคลของข้อมูลสุขภาพ โดยพิจารณาประโยชน์ทั้งการป้องกันส่วนบุคคล และประโยชน์ที่เกิดขึ้นกับสังคมในกรณีที่ต้องละเมิดความเป็นส่วนตัวของบุคคล

5. พัฒนากลไกอย่างเป็นระบบในการผลิตและพัฒนาคนที่ทำงานเกี่ยวข้องกับระบบ สาธารณสุขและเทคโนโลยีสารสนเทศสุขภาพทั้งระดับปฏิบัติการและระดับบริหารจัดการให้เพียงพอกับการนำระบบสารสนเทศและเทคโนโลยีสารสนเทศสุขภาพมาใช้งานอย่างมีประสิทธิภาพ

นอกจากข้อเสนอทางนโยบายข้างต้น คณะผู้ศึกษาได้เสนอกระบวนการดำเนินงานเพื่อพัฒนาาระบบสารสนเทศสุขภาพและสาธารณสุขของประเทศตามข้อเสนอตามนโยบาย ดังนี้ (คณะอนุกรรมการการสาธารณสุข วุฒิสภา, 2554)

1. ควรมีคณะทำงานเฉพาะกิจ (Ad Hoc Committee) โดยมีผู้อำนวยการสำนักนโยบายและยุทธศาสตร์ กระทรวงสาธารณสุขเป็นประธาน มีผู้แทนจากกองหรือฝ่ายนโยบายแต่ละกรมของกระทรวงสาธารณสุข หน่วยงานต่าง ๆ ที่มีส่วนเกี่ยวข้องกับงานด้านสาธารณสุข รวมทั้งจาก ภาครัฐ รัฐวิสาหกิจ และ

เอกชน เป็นกรรมการทำงานร่วมกับที่ปรึกษา (อาจเป็นบริษัทเอกชน) จากในหรือนอกประเทศที่มีความรู้ ความชำนาญการพัฒนาระบบสารสนเทศสุขภาพดำเนินการ

- 1.1 ศึกษาโครงสร้างองค์กรและหน้าที่ของเจ้าหน้าที่ตามโครงสร้างเพื่อเสนอให้ มีการจัดตั้งขึ้น
- 1.2 ทำ Pre-Feasibility Study ของการพัฒนาระบบสารสนเทศสุขภาพของประเทศ
- 1.3 สสำรวจ ประเมินระบบสารสนเทศที่ใช้อยู่ของกระทรวงสาธารณสุข และ หน่วยงานที่เกี่ยวข้องทั้งหมดครอบคลุมตามที่รวบรวมไว้ในรายงานนี้

1.4 สามารถกำหนดนโยบายยุทธศาสตร์การพัฒนาระบบสารสนเทศและเทคโนโลยีสารสนเทศสุขภาพของประเทศ เพื่อให้เป็นส่วนหนึ่งของกรอบนโยบาย และยุทธศาสตร์ด้าน เทคโนโลยีสารสนเทศของประเทศในภาพรวม และควบคู่ไปกับยุทธศาสตร์และการพัฒนางานด้าน อื่น ๆ เช่น eGovernment eEducation eIndustry eSociety eCommerce เป็นต้น

1.5 สามารถพัฒนามาตรฐานข้อมูลสุขภาพในทุกมิติ รวมถึงกลไกในการดูแล มาตรฐาน เพื่อให้ระบบสารสนเทศต่าง ๆ ทำงานร่วมกันได้สามารถแลกเปลี่ยนข้อมูลกันได้แบบ ไร้รอยต่อด้วยความปลอดภัย และเป็นประโยชน์ในการดูแลรักษาประชาชนที่มีคุณภาพ และมี ประสิทธิภาพ รวมทั้งด้านอื่น ๆ ที่มีระบุไว้ในหลักการและเหตุผลในรายงานตอนต้น

- 1.6 จัดทำแนวทางหัวข้อที่จะใช้ทำ Feasibility Study เพื่อใช้ประมวลจัดทำบริษัทที่ปรึกษา

2. ประเทศไทยมีองค์กรระดับประเทศที่ทำหน้าที่กำหนดทิศทางการพัฒนา กำหนด ยุทธศาสตร์ วางนโยบายและแผนหลักการพัฒนาระบบสารสนเทศและเทคโนโลยีสารสนเทศสุขภาพของประเทศ

2.1 องค์กรระดับประเทศ นี้ควรประกอบด้วยเจ้าหน้าที่จากทุกหน่วยงานที่เกี่ยวข้องกับระบบสารสนเทศและเทคโนโลยีสารสนเทศสุขภาพทั้งภาครัฐ รัฐวิสาหกิจและเอกชน

2.2 องค์กรระดับประเทศนี้ควรเป็นหน่วยงานอิสระในการบริหารงานและเพื่อความคล่องตัวให้เป็นองค์การมหาชน (โดยมีกฎหมายรองรับ)

3. เมื่อมีองค์กรระดับประเทศนี้แล้วองค์กรนี้ควรทำงานร่วมกับที่ปรึกษามีความเชี่ยวชาญด้านระบบข้อมูลสารสนเทศสุขภาพ (อาจเป็นบริษัทเอกชน) โดยทำงานร่วมกันตั้งแต่ศึกษาความ เหมาะสม (Feasibility Study) จัดการทำ Term of Reference (TOR) การพัฒนาระบบข้อมูล สารสนเทศสุขภาพของประเทศเพื่อใช้ในการจ้างเมื่อจะมี Implement งาน และควรทำหน้าที่กำกับ ติดตามการ Implementer ในการติดตั้ง Hardware Software การพัฒนาบุคลากร และงานด้าน อื่น ๆ โดยใช้ผลการศึกษา Pre-feasibility Study เป็นกรอบในการศึกษา ทั้งนี้ใน Feasibility Study และ TOR ควรมีการกำหนดกรอบ ดังนี้

3.1 กำหนดกรอบเวลาที่ Implementer จะใช้ในการ Implement รวมทั้งขั้นตอนต่าง ๆ เป็นตารางและศึกษาทำประมาณการงบประมาณที่จะต้องใช้อย่างละเอียด

3.2 กำหนดรายละเอียดด้านเทคนิค Specification ของทั้ง Hardware Communication Software และการพัฒนาบุคลากรที่จำเป็นและเหมาะสมกับระบบบริการสุขภาพของไทยตั้งแต่ระดับปฐมภูมิ ทุติยภูมิ ตติยภูมิ และระดับสูงกว่าตติยภูมิเพื่อให้ระบบสามารถเชื่อมโยงกันทั้งในและนอกสถานพยาบาล และใช้ประโยชน์ได้สูงสุด

3.3 กำหนดหลักสูตรการฝึกอบรมทางทฤษฎีและ On the Job แก่เจ้าหน้าที่ที่จะปฏิบัติงาน บริหารจัดการระบบและการใช้บำรุงรักษาอุปกรณ์เหล่านี้

4. ในการ Implement องค์กรระดับประเทศที่เป็นองค์กรความร่วมมือที่สร้างขึ้นนี้ ต้องมีส่วนร่วมในการ Implementation โดยต้องกำกับติดตามบริษัทที่ประมูลได้ดำเนินการให้เป็นไปตามข้อกำหนด

4.1 การจัดวางระบบ Hardware และ Communication ที่เหมาะสมกับระบบบริการสุขภาพของไทยตั้งแต่ระดับปฐมภูมิ ทุติยภูมิ ตติยภูมิ และระดับสูงกว่าตติยภูมิ

4.2 การจัดหา Software Package ที่เหมาะสมกับระบบบริการสุขภาพของไทย ตั้งแต่ระดับปฐมภูมิ ทุติยภูมิ ตติยภูมิ และระดับสูงกว่าตติยภูมิ

4.3 การทำให้ให้ระบบต่าง ๆ ใช้ได้และเชื่อมโยงกันได้อย่างมีประสิทธิภาพ

4.4 ทดสอบระบบและปรับปรุงแก้ไขระบบให้ใช้งานเชื่อมโยงกันได้ตรงเป้าหมาย

4.5 จัดอบรมเจ้าหน้าที่ตามหลักสูตรที่กำหนด

5. องค์กรระดับประเทศที่สร้างขึ้นต้องดำเนินการให้มีการศึกษาและดำเนินการออกกฎหมาย กฎระเบียบต่าง ๆ ที่เกี่ยวข้องกับการรักษาความปลอดภัยและความลับส่วนบุคคลของข้อมูลสุขภาพ โดยพิจารณาประโยชน์ทั้งการป้องกันส่วนบุคคลและประโยชน์ที่เกิดขึ้นกับสังคมในกรณี ที่ต้องละเมิดความเป็นส่วนตัวของบุคคล

6. องค์กรระดับประเทศที่สร้างขึ้นควรทำงานร่วมกับสถานศึกษาในระดับต่าง ๆ เช่น วิทยาลัย และ มหาวิทยาลัย สนับสนุนให้มีการเรียนการสอนด้านเทคโนโลยีสารสนเทศสุขภาพ และระบบข้อมูลสารสนเทศสุขภาพในหลักสูตรวิชาชีพด้านสุขภาพ เช่น แพทย์ พยาบาล เภสัช เป็นต้น และควรสนับสนุนให้เกิดหลักสูตรสาขาวิชา Biomedical and Health Informatics สำหรับบุคลากรด้านสุขภาพและสาธารณสุข บุคลากรด้านเทคโนโลยีคอมพิวเตอร์และนักเรียนนักศึกษาทั่วไป

ระบบข้อมูลข่าวสารและสารสนเทศสุขภาพเป็นเครื่องมือสำคัญที่ช่วยในการพัฒนางานสาธารณสุข ทั้งในด้านการรักษาพยาบาล การส่งเสริมสุขภาพ การควบคุม ป้องกันโรค และการฟื้นฟูสุขภาพสารสนเทศต่าง ๆ โดยมากได้มาจากการเก็บรวบรวมข้อมูล (data) ทั้งจากแหล่งปฐมภูมิและแหล่งทุติยภูมิ ซึ่งข้อมูลที่ได้จะต้องนำมาจัดระบบสารสนเทศ (Management Information System: MIS) ให้เป็นหมวดหมู่ที่ง่ายและสะดวกในการนำไปใช้ในการบริหารงาน การควบคุม กำกับ ติดตาม และประเมินผลการดำเนินงาน ตลอดจนใช้ในการตัดสินใจเชิงนโยบายเพื่อพัฒนางานสาธารณสุขให้ดำเนินไปอย่างมี

ประสิทธิภาพ ระบบข้อมูลข่าวสารและสารสนเทศด้านสุขภาพมีความสำคัญต่อการพัฒนางานสาธารณสุขอย่างยิ่ง ซึ่งสามารถสรุปได้ ดังนี้ (สำนักงานปลัดกระทรวงสาธารณสุข, 2553)

1. การใช้ข้อมูลข่าวสารและสารสนเทศเพื่อการพัฒนาประสิทธิภาพของงานสาธารณสุข การมีระบบข้อมูลและสารสนเทศในงานสาธารณสุขที่ดีนั้นจะช่วยในการปฏิบัติภารกิจประจำได้ดีและรวดเร็วยิ่งขึ้น ทั้งในเรื่องการรักษาพยาบาล การส่งเสริมสุขภาพ การควบคุมป้องกันโรคและการฟื้นฟูสภาพสถานบริการและหน่วยงานสาธารณสุขจำเป็นต้องมีข้อมูลที่ถูกต้อง ครบถ้วน ทันเวลา เพื่อนำมาพัฒนาประสิทธิภาพงานสาธารณสุขให้ดียิ่งขึ้น เพื่อนำไปสู่เป้าหมายให้ประชาชนมีสุขภาพดีรวมถึงการให้บริการด้านต่าง ๆ ต้องครอบคลุมพื้นที่และประชากรในเขตรับผิดชอบ ดังนี้

1.1 ด้านการรักษาพยาบาล ประชาชนที่มาใช้บริการบริการทุกคนควรจะได้รับบริการที่มีประสิทธิภาพ คือ รวดเร็วไม่รอนาน ผู้ให้บริการทุกคนควรมีข้อมูลสำหรับให้บริการประชาชน เช่น ประวัติ ผู้มารับบริการ ข้อมูลการรักษาพยาบาล ข้อมูลการเจ็บป่วยฯ เพื่อให้การรักษาพยาบาลมีความต่อเนื่องไม่ต้องทำประวัติใหม่ การให้ความเสมอภาคแก่ผู้มารับบริการและการเข้าถึงบริการ เช่น การให้บริการโดยหน่วยแพทย์เคลื่อนที่ ประชาชนที่อยู่ในพื้นที่ห่างไกลจะได้รับการดูแลอย่างทั่วถึงเช่นกัน เป็นต้น

1.2 ด้านการส่งเสริมสุขภาพ เป็นการให้บริการทางด้านอนามัยเด็ก อนามัยมารดา หญิงมีครรภ์ ผู้สูงอายุ การวางแผนครอบครัวฯ โดยมีเป้าหมายเพื่อต้องการส่งเสริมสุขภาพให้ประชาชน ทุกคนทุกกลุ่มอายุในเขตพื้นที่รับผิดชอบ รวมถึงผู้มารับบริการที่สถานพยาบาลนั้น ๆ มีสุขภาพแข็งแรง มีสุขภาพจิตที่ดีปราศจากโรคภัยไข้เจ็บมาเบียดเบียน ข้อมูลข่าวสารที่จำเป็น เช่น ประวัติครอบครัว ข้อมูลการเจ็บป่วย ข้อมูลการดูแลสุขภาพ จำนวนประชากรกลุ่มเป้าหมาย เป็นต้น

1.3 ด้านการควบคุมป้องกันโรค เป็นการให้บริการทางด้านป้องกันโรคภัย ไข้เจ็บ โดยการให้วัคซีนต่าง ๆ ป้องกันให้ไม่เกิดโรคที่ป้องกันได้ หรือทำให้ร่างกายมีภูมิคุ้มกันต้านทาน ข้อมูล ข่าวสารที่ต้องการ เช่น ประวัติการให้ภูมิคุ้มกันโรค ข้อมูลการเจ็บป่วยต่าง ๆ และการได้รับการ รักษาพยาบาล จำนวนประชากรกลุ่มเป้าหมายในเขตพื้นที่รับผิดชอบ รวมถึงผู้มารับบริการในสถานพยาบาลนั้น ๆ เป็นต้น

1.4 ด้านการฟื้นฟูสภาพ หลังจากได้รับการรักษาแล้ว ผู้ป่วยจะต้องได้รับการฟื้นฟู สภาพทั้งร่างกายและจิตใจ เช่น ผู้ป่วยโรคอัมพาต ผู้ป่วยโรคเอดส์ ผู้ป่วยโรคหัวใจ ข้อมูลที่จำเป็น เช่น ประวัติครอบครัว ประวัติการเจ็บป่วย ข้อมูลการรักษาพยาบาล จำนวนผู้ป่วยที่จะต้องได้รับการฟื้นฟูสภาพ ทั้งในสถานพยาบาลและในเขตพื้นที่รับผิดชอบ เป็นต้น

2. การใช้ข้อมูลข่าวสารและสารสนเทศเพื่อการบริหารงานสาธารณสุข การพัฒนาระบบข้อมูลข่าวสารและสารสนเทศ เพื่อการบริหารงานสาธารณสุข มีวัตถุประสงค์เพื่อตอบสนองความต้องการของผู้บริหารทุกระดับ สำหรับใช้ในการบริหารงาน กำกับ ควบคุม ติดตาม และประเมินผลด้านสุขภาพ ซึ่งระบบข้อมูลข่าวสารและสารสนเทศเพื่อการบริหารงานสาธารณสุข จะช่วยให้นักบริหารทราบถึงปัญหาสาธารณสุขรวมถึงการเปลี่ยนแปลงที่เกิดขึ้นในแต่ละช่วงเวลา ซึ่งจะต้องตัดสินใจแก้ไขปัญหานั้น ๆ ให้สอดคล้องกับสภาพที่เป็นอยู่ กระทรวงสาธารณสุขมีการใช้ระบบสารสนเทศในงานสาธารณสุข ดังนี้

2.1 ด้านการวางแผนและกำหนดนโยบายจะต้องนำข้อมูลข่าวสารที่ต้องการมา ประยุกต์อย่างมีหลักเกณฑ์เพื่อนำไปสู่การตัดสินใจเชิงนโยบายและใช้เป็นแนวทางในการปฏิบัติงานที่มีประสิทธิภาพ ข้อมูลที่จะนำมาใช้ในการวางแผนและกำหนดนโยบายนั้นจะต้องมีการประมวลผลและวิเคราะห์ข้อมูลที่เกี่ยวข้อง ซึ่งจะต้องมีการกำหนดความต้องการข้อมูลก่อน หลังจากนั้นจึงไปทบทวนจากระบบข้อมูลที่มีอยู่ หากไม่มีในระบบจึงทำการพัฒนาระบบสำหรับการจัดเก็บข้อมูลขึ้นใหม่ เพื่อให้ได้มาซึ่งข้อมูลที่สามารถนำมาใช้ประโยชน์ได้อย่างมีประสิทธิภาพ

2.2 ด้านการควบคุมกำกับงาน โดยทั่วไปองค์กรจะใช้ผังปฏิบัติงานประจำการตรวจราชการ การนิเทศและติดตามการปฏิบัติงาน ซึ่งสามารถแก้ไขปัญหาได้ทันเวลา รวมทั้งมีการประเมินผลการปฏิบัติงานด้วย

2.3 ด้านการบริหารงานบุคคล ผู้บริหารจะต้องมีข้อมูลข่าวสารเพื่อใช้ในการประกอบการพิจารณาตัดสินใจ เช่น การกระจายบุคลากร การเลื่อนตำแหน่ง พิจารณาความดีความชอบ การสอบคัดเลือก การดำเนินการทางวินัย การลงโทษ เป็นต้น

2.4 ด้านการบริหารงบประมาณ ต้องมีข้อมูลด้านการเงิน การบัญชีและการพัสดุที่เป็นปัจจุบัน สำหรับใช้ในการติดตามการใช้จ่ายเงิน และควบคุมการใช้จ่ายเงินให้เป็นไปตามระเบียบของทางราชการ

3. การใช้ข้อมูลข่าวสารและสารสนเทศเพื่อการตรวจสอบการบริหารงานสาธารณสุขระบบข้อมูลข่าวสารและสารสนเทศเพื่อใช้ในการตรวจสอบการบริหารงานสาธารณสุขจะต้องเป็นสารสนเทศที่มีความถูกต้องมีคุณภาพและสอดคล้องกับการบริหารงานสาธารณสุข ซึ่งอาจเป็นทั้งเชิงคุณภาพและเชิงปริมาณที่เกี่ยวข้องกับปัจจัยแวดล้อมที่มีผลกระทบต่อสุขภาพอนามัยของประชาชนโดยรวมการบริหารงานสาธารณสุขจำเป็นต้องมีสารสนเทศเพื่อตรวจสอบให้มีประสิทธิภาพ ดังนี้

3.1 ด้านการวางแผน เช่น กฎหมายหรือระเบียบราชการที่มีผลกระทบต่อการทำงานโดยตรง แนวโน้มของประชากร สภาพเศรษฐกิจและสังคม เป็นต้น

3.2 ด้านการควบคุมกำกับงาน ต้องเป็นสารสนเทศที่ใช้ในการควบคุม กำกับและตรวจสอบดูแลการปฏิบัติงานของผู้ปฏิบัติให้เป็นไปตามแผนการดำเนินงาน

3.3 ด้านการบริหารงานบุคคล เช่น สารสนเทศเกี่ยวกับกฎเกณฑ์การปฏิบัติงานกฎระเบียบต่าง ๆ เป็นต้น

3.4 ด้านการบริหารงบประมาณ เช่น ระเบียบการเงินการคลังและการพัสดุ รายงานการรับจ่ายเงินแผ่นดิน เป็นต้น

จากการศึกษาและรับฟังข้อมูลชี้แจงของหน่วยงานต่าง ๆ ที่เกี่ยวข้องสามารถสรุปปัญหาของระบบข้อมูลสารสนเทศและเทคโนโลยีสารสนเทศสุขภาพของไทย (National Health Information Systems & eHealth) ดังนี้ (สำนักงานปลัดกระทรวงสาธารณสุข, 2553)

1. ไทยมีระบบข้อมูลสารสนเทศสุขภาพมากมายหลายระบบ ส่วนใหญ่ตอบสนองการ บริหารจัดการและการเบิกจ่ายค่ารักษาพยาบาล หน่วยงานต่าง ๆ มีข้อมูลตัวชี้วัดโครงการฯ ต่าง ๆ มากมาย ซ้ำซ้อน ระบบสารสนเทศที่มีตอบสนองความต้องการของผู้ใช้ที่เป็นผู้ให้บริการ และประชาชนมีน้อย นอกจากนั้น ระบบสารสนเทศสุขภาพที่มีเหล่านี้ไม่สามารถแลกเปลี่ยนข้อมูลกันได้ ข้อมูลสุขภาพไม่มีการบูรณาการ

2. ขาดการดำเนินการเพื่อวางรากฐานด้านระบบสารสนเทศสุขภาพ โดยเฉพาะอย่างยิ่ง ด้านมาตรฐานข้อมูลสุขภาพ มาตรฐานกฎหมายที่เกี่ยวข้องกับความปลอดภัยและความเป็นส่วนตัวของการ ใช้ข้อมูลสุขภาพ

3. ขาดหน่วยงานในระดับประเทศที่จะกำหนดยุทธศาสตร์วางกรอบการพัฒนาและชี้นำการ ระบบสารสนเทศสุขภาพและสาธารณสุขบุคลากรทางการแพทย์และสาธารณสุขต้องใช้เวลาถึงกว่าร้อยละ 40 ของเวลาทำงานมาเพื่อจัดการรายงานและข้อมูลที่คนอื่นต้องการมากกว่าจะได้รับประโยชน์จากระบบ ข้อมูลและเทคโนโลยีสารสนเทศที่ใช้อยู่ ทำให้การให้บริการสุขภาพแก่ ประชาชนไม่มีประสิทธิภาพตามที่ ควร

4. ประชาชนไม่สามารถเข้าถึงและใช้ประโยชน์ข้อมูลสุขภาพของตน คุณสมบัติที่พึงประสงค์ของ ระบบสารสนเทศสุขภาพของประเทศ

จากการทบทวนวรรณกรรมที่ผ่านมา พบว่า ระบบสารสนเทศในระดับปฏิบัติงานที่นิยมนำมาใช้ ในโรงพยาบาลมี 2 ระบบ คือ ระบบ HIS HOSxP และ ระบบ HIS HospitalOS ดังรายละเอียดต่อไปนี้

3. ระบบ Hospital Information System (HIS) HOSxP

โปรแกรมที่พัฒนาขึ้นใช้สำหรับสถานพยาบาล สถานีอนามัยและโรงพยาบาล เพื่อเก็บข้อมูล ผู้ป่วยและช่วยบุคลากรทางการแพทย์ในการให้บริการผู้ป่วย มีลักษณะเป็น Hospital Information System ถูกพัฒนาขึ้นเมื่อปี พ.ศ. 2542 และเป็นโปรแกรมที่สามารถเชื่อมโยงโครงสร้างพื้นฐานด้าน เครือข่ายทั้งสำนักงานสาธารณสุขจังหวัด โรงพยาบาลและสถานีอนามัย เพื่อยกระดับคุณภาพชีวิตของคน ไทยและสังคมไทย ปัจจุบันโปรแกรม HOSxP ถูกนำไปใช้ในโรงพยาบาลมากกว่า 500 แห่งทั่วประเทศไทย ทั้งโรงพยาบาลรัฐบาลและโรงพยาบาลเอกชน

เป็นการนำระบบคอมพิวเตอร์เครือข่าย (Computer Network) มาใช้ในการบันทึกการ ให้บริการผู้ป่วยในโรงพยาบาล มีใช้กันอย่างแพร่หลายเนื่องจากเครื่องคอมพิวเตอร์มีราคาถูกระบบ เครือข่ายสามารถติดตั้งง่าย มีความสะดวกรวดเร็ว แต่ระบบ Software ที่นำมาใช้งานในปัจจุบันได้รับการ สนับสนุนจากกระทรวงสาธารณสุข ซึ่งมีหลายโปรแกรม แต่โปรแกรมที่ได้รับความนิยมในปัจจุบันมีชื่อว่า STAT เป็นระบบที่ทำงานใน DOS ซึ่งทีมผู้พัฒนาได้หยุดการพัฒนาแล้ว ถึงแม้ว่าโปรแกรมต่าง ๆ จะได้รับการ สนับสนุนจากกระทรวงสาธารณสุข แต่โปรแกรมเหล่านี้กลับมีการทำงานที่เป็นระบบของตัวเองไม่มี

การเชื่อมโยงถึงกัน เนื่องจากมีการพัฒนาจากหลายหน่วยงาน หลายบริษัท และที่สำคัญไม่มีการรับประกัน หรือการสนับสนุน (Support) การใช้งาน ส่งผลให้การนำไปใช้ไม่เกิดประสิทธิผลดีเท่าที่ควร หรือการจัดซื้อ Software จากบริษัทเอกชนที่ได้ทำออกมาขายก็มีราคาค่อนข้างแพงมาก ทั้งค่าโปรแกรมครั้งแรก และค่าดูแลรักษาโปรแกรมรายปีที่อาจมีราคาแพงหลักแสนบาทต่อปี ที่สำคัญกระทรวงสาธารณสุขมีความต้องการรายงานที่เปลี่ยนไปทุกปี โปรแกรมที่ใช้งานต่าง ๆ ควรจะรองรับการทำรายงานที่มีขึ้นมาใหม่เรื่อย ๆ มิเช่นนั้นแล้วจะเป็นภาระอย่างมากกับผู้ดูแลระบบข้อมูลข่าวสารต่าง ๆ ของโรงพยาบาล (Bangkok Medical Software, 2021)

ลักษณะเด่นของ ระบบ BMS HOSxP มีคุณลักษณะ ดังนี้

1. ถูกออกแบบให้ทำงานโดยใช้คำสั่ง SQL ที่เป็นมาตรฐาน
2. มีระบบเก็บรูปภาพของผู้ป่วยหรือลายนิ้วมือ เพื่อใช้ในการ Identify ผู้ป่วย และมีระบบเรียกชื่อผู้ป่วยอัตโนมัติ (ตามลำดับคิว)
3. แสดงรูปทางการแพทย์ โดยได้มีการประยุกต์ใช้ระบบแสดงรูป DICOM
4. รองรับการวาดรูปบาดแผลหรือรูปแบบอื่น ๆ บนจอคอมพิวเตอร์ ช่วยลดการใช้กระดาษลง
5. แสดงประวัติของผู้ป่วยผ่าน EMR (Electronic Medication Record) ซึ่งแพทย์สามารถเข้าถึงข้อมูลประวัติผู้ป่วยได้
6. สามารถส่งออกข้อมูลได้หลายรูปแบบ เช่น EXCEL, DBASE, XML, HTML โดยการเข้าถึงข้อมูลจะใช้คำสั่งภาษา SQL
7. มีระบบ Replication ภายในของตนเอง ซึ่งรองรับการทำ Asynchronous Replication ในแบบ Real Time และแบบ Offline
8. สามารถแก้ไขหรือสร้างรายงานใหม่ได้ โดยผู้ดูแลระบบ
9. มีการคำนวณค่า DRGs ในขั้นตอนสรุปผลการวินิจฉัย
10. รองรับการทำงานเชื่อมต่อกับระบบ LIS
11. รองรับการลงบันทึกข้อมูลแพทย์แผนไทย
12. รองรับการเก็บข้อมูลแบบ Universal Form Entry (UE Form)
13. มีระบบตรวจสอบสิทธิการรักษาผ่าน Web NHSO และจากฐานข้อมูล HIPDATA
14. มีระบบส่งออกข้อมูลภายนอก เช่น 12 แฟ้ม, 18 แฟ้ม, 21 แฟ้ม, Text File สกส. ส่งข้อมูลไปยัง SSN_Data ประกันสังคม ส่งออก EXCEL File แพทย์แผนไทย
15. มีระบบควบคุมความปลอดภัย โดยใช้ USER ID และ PASSWORD และกำหนดสิทธิ์การใช้งานในระบบฐานข้อมูลของผู้ใช้งานแต่ละราย
16. มีระบบสำรองข้อมูลแบบ Hot Back up โดยไม่ต้องหยุดระบบ ซึ่งสามารถทำได้โดยระบบ Auto และ Manual แล้วสามารถนำข้อมูลสำรอง จนถึง ณ จุดที่ทำการ Back up ครั้งสุดท้าย
17. รองรับการแปลงข้อมูลจากระบบปัจจุบันมาสู่ระบบใหม่

ระบบ BMS HOSxP รองรับระบบงาน

1. ระบบประชาสัมพันธ์ สามารถดูและสืบค้นข้อมูลพื้นฐานและคุณลักษณะทั่วไป และการเรียกดูสรุปข้อมูล
2. ระบบเวชระเบียน บันทึกข้อมูลประวัติผู้ป่วย การส่งตรวจผู้ป่วย การเยี่ยม-คืนแฟ้มเวชระเบียน การพิมพ์ และการบันทึกการส่งตรวจ
3. ระบบตรวจสอบสิทธิการรักษา มีการกำหนดข้อมูลพื้นฐาน การตรวจสอบสิทธิ สามารถตรวจสอบสิทธิจากบริษัทกลางได้
4. ระบบซักประวัติ อาการเจ็บป่วย น้ำหนัก, ส่วนสูง, อุณหภูมิ, รอบเอว, อัตราเต้นชีพจร, อัตราหายใจ, ความดันโลหิต, FBS, เป็นมากี่วันแล้ว, BMI (คำนวณให้อัตโนมัต เมื่อใส่น้ำหนักและส่วนสูง), Chief complaint (CC), History of present illness (HPI), Past medical history (PMH), Family history (FH), Social history (SH), Review of system (ROS) และสามารถระบุได้ว่าเป็นผู้ป่วยกำลังตั้งครรภ์หรือกำลังให้นมบุตรซึ่งมีผลเชื่อมโยงกับการให้ยา การดูประวัติย้อนหลัง การบันทึกการแพ้ยา
5. ระบบนัดหมาย ประกอบด้วย คุณสมบัติทั่วไปและข้อมูลพื้นฐาน การตรวจสอบข้อมูลการนัดหมาย การลงทะเบียนนัดหมาย การเลื่อนการนัดหมาย การยกเลิกการนัดหมาย การส่งตรวจผู้ป่วยล่วงหน้า
6. ระบบห้องทำงานแพทย์ บันทึกการตรวจ รักษา วินิจฉัยโรค การทำหัตถการ การส่งตรวจทางห้องปฏิบัติการ การตรวจทางรีซีและการตรวจทางพยาธิ การส่งยา การดูประวัติการรักษา การส่งต่อ การ admit การ consult การนัดหมาย
7. ระบบห้องปฏิบัติการ สามารถกำหนดข้อมูลรายการ LAB การออกแบบใบสั่ง LAB มีการเชื่อมโยงกับหมวดคำรักษาพยาบาลตามกลุ่มค่าตรวจวินิจฉัยทางเทคนิคการแพทย์และพยาธิวิทยา การลงและรายงานผล LAB การดูประวัติผล LAB
8. ระบบคลินิกพิเศษ เป็นการลงทะเบียนผู้ป่วยโรคเรื้อรัง เบาหวาน ความดัน การบันทึกภาวะแทรกซ้อนทางตา ทางเท้า ไต หลอดเลือดหัวใจ หลอดเลือดสมอง หลอดเลือดส่วนปลายและปริทันต์ การจัดทำแผนการดูแลผู้ป่วยต่อเนื่อง
9. ระบบคัดกรองกลุ่มเสี่ยงเรื้อรัง สามารถค้นหากลุ่มประชากรกลุ่มเป้าหมายที่มีอายุ 15 ปีขึ้นไป โดยไม่การป่วยเป็นโรคหรือมีสถานะเสียชีวิตมาแสดงเพื่อคัดกรองกลุ่มเสี่ยงโรคเรื้อรัง บันทึกการคัดกรองแบบสัมภาษณ์ ข้อมูลประวัติครอบครัว ประวัติการเจ็บป่วย การดื่มแอลกอฮอล์ การสูบบุหรี่ การออกกำลังกาย การรับประทานอาหาร บันทึกการคัดกรองระดับน้ำตาลในเลือด และค่าการวัดความดันได้ สามารถระบุ น้ำหนัก ส่วนสูง รอบเอว ซึ่งจะหาคำนวนค่า BMI โดยอัตโนมัติ แสดงผลการประเมินความเสี่ยงในแต่ละด้านให้อัตโนมัต

10. ระบบทันตกรรม บันทึกทะเบียนผู้ป่วยทันตกรรม ลงทะเบียนเข้ารับการรักษาทางทันตกรรม สามารถเรียกดูข้อมูลทั่วไปและประวัติการมารับบริการของผู้ป่วย บันทึกข้อมูลการซักประวัติ บันทึกข้อมูลการรักษาและการให้บริการทางทันตกรรม บันทึกการจ่ายยา บันทึกข้อมูลการตรวจร่างกาย บันทึกข้อมูลการนัดหมาย
11. ระบบรังสีรักษา สามารถกำหนดข้อมูลรายการ X-Ray เชื่อมโยงข้อมูลกับหมวดหมู่ค่าบริการ ตามกลุ่มการรักษาพยาบาลทางรังสีวิทยา กำหนดรูปแบบการชำระเงิน รองรับการจัดเก็บข้อมูล การลงทะเบียนตรวจรักษาทางรังสีวิทยา เรียกดูข้อมูลประวัติการตรวจย้อนหลังได้ เชื่อมระบบ PACs รองรับการยืนยันผลการตรวจอัตโนมัติจากระบบ LIS ซึ่งตรวจสอบประวัติการรายงานผลการทำ X-Ray แต่ละครั้งของผู้ป่วย รังสีแพทย์สามารถบันทึกข้อมูลผลการอ่านฟิล์ม และลงผลวินิจฉัยด้วยรหัสโรคและชื่อโรค โดยใช้รหัส ICD ขององค์การอนามัยโลก (WHO) และของประเทศไทย
12. ระบบเวชศาสตร์ฟื้นฟู สามารถลงทะเบียนและดูประวัติการทำกายภาพ บันทึกการตรวจรักษา การตรวจร่างกาย การสั่งยา การลงวินิจฉัยโรค การทำหัตถการ สั่ง LAB XRAY การส่งต่อ
13. ระบบเภสัชกรรม สามารถบันทึกรายการจ่ายยาให้ผู้ป่วย แสดงจำนวน ราคาและคำนวณราคารวม และสั่งพิมพ์สติ๊กเกอร์ติดของยา สำหรับนำไปจัดยาได้ เก็บเงื่อนไขการสั่งจ่าย สิทธิราคาขาย และชื่อยกเว้นต่าง ๆ ได้หลายรูปแบบ การบันทึก แก้ไขข้อมูลเกี่ยวกับรายละเอียดยา และเวชภัณฑ์แต่ละรายการ มีการกำหนดวิธีใช้ยาหลัก และคำแนะนำเพิ่มเติมของยาแต่ละรายการทั้งภาษาไทยและอังกฤษ กำหนดให้มีการแจ้งเตือนการสั่งยาและเวชภัณฑ์ที่อาจเป็นอันตรายต่อสตรีมีครรภ์, หญิงให้นมบุตรและเด็กในช่วงอายุ 0-5 ปี บันทึกและแก้ไข ข้อมูล ADR ประวัติแพ้ยา มีฐานข้อมูล Major Drug Interaction และสามารถบันทึก เพิ่มเติม แก้ไข Import/export ข้อมูล Drug Interaction ได้
14. ระบบห้องผ่าตัด และ วิสัญญี สามารถลงทะเบียนสั่งผ่าตัดผู้ป่วยเองได้ เพิ่มและดูรายละเอียด รายการนัดผ่าตัดที่มีการ set ไว้ คีย์ค่าใช้จ่าย ค่ายา บันทึกข้อมูลการผ่าตัด การลงข้อมูลของ Airway Problem เป็นการลงข้อมูลส่วนที่เป็นปัญหาทางเดินหายใจ และวิธีการแก้ปัญหา
15. ระบบ Admission Center เก็บข้อมูลหอผู้ป่วย ประเภทห้อง/เตียง จำนวนเตียงและราคาในแต่ละหอผู้ป่วย กำหนดราคาเตียงตามประเภทและหอผู้ป่วย ลงทะเบียนผู้ป่วยใน โดยการเชื่อมโยงข้อมูลทะเบียนประวัติ ผู้ป่วยและข้อมูลอื่น ๆ ที่เกี่ยวข้องจากระบบทะเบียนผู้ป่วยนอกเข้ามายังระบบทะเบียนผู้ป่วยในสามารถบันทึกประเภทการ Admit ของคนไข้ ได้แก่ ปกติ อุบัติเหตุ เด็กแรกเกิด คลอดบุตร
16. ระบบผู้ป่วยใน สามารถลงทะเบียนรับผู้ป่วยใหม่ โดยแยกรับเป็นรายหอผู้ป่วย บันทึก/การแก้ไข ประวัติส่วนตัวและการรักษา ลงผลการวินิจฉัยตามมาตรฐานรหัส ICD10 คำนวณค่า relative weight ตามระบบ DRGs ได้ในขั้นตอนการลงผลวินิจฉัย ลงข้อมูล DRGs ตามโครงสร้าง 12 แพ้มมาตรฐาน บันทึกข้อมูลการทำหัตถการของผู้ป่วยใน บันทึกการให้การักษา การให้ยา

ในผู้ป่วยแต่ละราย ส่งใบสั่งยา ใบตรวจ Lab, X-ray, MRI, CT, U/S, ใบส่งตรวจพิเศษต่าง ๆ
ใบ set ผ่าตัดผ่านระบบ

17. ระบบห้องคลอด การลงทะเบียนผู้ป่วยในของห้องคลอด บันทึกข้อมูลการฝากครรภ์ ข้อมูลระยะรอคลอด บันทึกข้อมูลการคลอดของทารกแต่ละคนบันทึกข้อมูลภาวะแทรกซ้อน
18. ระบบงานโภชนาการ การบันทึกข้อมูลการสั่งอาหาร การแก้ไขและยกเลิกการสั่งอาหาร ดูสรุปยอดข้อมูลการสั่งอาหารจากหอผู้ป่วยใน
19. ระบบสำรองข้อมูล แบบ Local Backup แบบ Daily Backup, ตั้งเวลาได้ Undo Restore ได้ Upgrade Replication Log, Clear Replication Log
20. ระบบงานเจ้าหน้าที่ผู้ดูแลระบบ

4. ระบบ Hospital Information System HospitalOS

HospitalOS เป็นโปรแกรม Open Source ที่รองรับการให้บริการทั้งระบบคนไข้นอก และระบบคนไข้ใน เช่น ระบบงานเวชระเบียน คัดกรองผู้ป่วย ห้องตรวจ เกสซกรรม คลังยา การเงิน ทันตกรรม ห้องตรวจทางห้องปฏิบัติการ และห้องรังสีวิทยา โดยโปรแกรมสามารถส่งออกรายงานตัวชี้วัดต่าง ๆ ในระดับประเทศได้ เช่น รายงาน 21 แฟ้ม, รายงาน 43+7 แฟ้ม รองรับการนำเข้าข้อมูลสู่โปรแกรม E-Claim เป็นต้น พร้อมต่อยอดเข้าระบบวิเคราะห์ข้อมูลสำหรับผู้บริหาร (BI) โดยมีคุณสมบัติ ดังนี้ (Opensource technology, 2014)

1. โปรแกรม Hospital OS Community. เป็นโปรแกรมฟรี ไม่มีค่าลงทะเบียนลิขสิทธิ์
2. รองรับการรักษาผู้ป่วยนอกและผู้ป่วยใน (IPD, OPD)
3. รองรับการเชื่อมต่ออุปกรณ์ LAB และ X-Ray
4. รองรับการออกรายงานด้วย SQL QUERY
5. สามารถส่งออกข้อมูลได้ทุกมาตรฐานของกระทรวงสาธารณสุขและสำนักงานหลักประกันสุขภาพแห่งชาติ
6. สามารถปรับแต่งหน้าจอการใช้งานให้เหมาะสมกับหน้าที่การทำงานของบุคลากรแผนกต่าง ๆ ในโรงพยาบาล
7. สามารถแก้ไข ฟอรัม การออกรายงานเพื่อความเหมาะสมของแต่ละโรงพยาบาล
8. Hospital OS Community พัฒนาคือมาตรฐาน CMMI ML3, HL7 และมีระบบสำรองข้อมูล Solony-I

ระบบ HIS HospitalOS ประกอบด้วย

1. ระบบเวชระเบียน การลงทะเบียน เลือกสิทธิการรักษาและส่งผู้ป่วยเข้าสู่จุดบริการต่าง ๆ ด้วยระบบคิว
2. ระบบการรักษาผู้ป่วย ER, OPD/IPD
3. การซักประวัติ บันทึกอาการสำคัญ อุบัติเหตุฉุกเฉิน ส่งตรวจ LAB/X-Ray และวินิจฉัย
4. ระบบตารางเวรแพทย์ [Add-On]
5. ระบบบริหารจัดการ การปฏิบัติงานของแพทย์ประจำจุดบริการต่าง ๆ ของสถานพยาบาล สามารถกำหนดการลาพัก วันหยุด และจองคิวการรักษาล่วงหน้าโดยแสดงจำนวนที่รับผู้ป่วยในแต่ละจุดบริการ
6. ระบบส่งเสริมสุขภาพผู้ป่วย
7. ระบบงานส่งเสริมสุขภาพชุมชน บันทึกข้อมูลตามตัวชี้วัด ส่งออกชุดข้อมูล ตามมาตรฐาน Data Set ของกระทรวงสาธารณสุข และสำนักงานหลักประกันสุขภาพแห่งชาติ.
8. ระบบคัดกรองผู้ป่วย
9. ระบบงานคัดกรองผู้ป่วยในกลุ่มเสี่ยงต่อ โรคระบาดและโรคเรื้อรัง (ส่งออกข้อมูลโรคเรื้อรังสู่ ส่วนกลาง)
10. ระบบทันตกรรม [Add-On] บันทึกข้อมูลทันตกรรมตามมาตรฐานกระทรวงสาธารณสุขด้วยภาพ แสดงชื่อและสีฟัน
11. ระบบงานเภสัชกรรม
 - 11.1 การสั่งจ่ายยาและเวชภัณฑ์ ตามสิทธิการรักษา สามารถเชื่อมต่อกับระบบ Inventory (สินค้าคงคลัง) ได้
 - 11.2 ระบบคลังยา
 - 11.3 ระบบ Inventory บริหารจัดการยาโดยไม่จำกัดจำนวนคลัง รองรับการออกไปขอซื้อยา และสั่งซื้อยา
12. ระบบการเงินรองรับการเบิกจ่ายผู้ป่วยสิทธิประกันชีวิต รองรับการเบิกจ่ายแบบหลายแผนและหลายบริษัทประกันชีวิต
13. ระบบรายงานและการส่งออกข้อมูล ส่งออกรายงานตามมาตรฐาน กระทรวงสาธารณสุข, สำนักนโยบายและแผนงาน กระทรวงสาธารณสุข และ สำนักงานหลักประกันสุขภาพแห่งชาติ (รองรับ 21, 43 แพ้ม และรายงานมาตรฐานของกระทรวงสาธารณสุข)

เครื่องมือที่นำมาใช้ในการพัฒนาระบบเชื่อมโยงโปรแกรมที่เกี่ยวข้องกับบริการเพื่อการดูแลภาวะฉุกเฉินด้านการแพทย์และสาธารณสุข จังหวัดเชียงใหม่ มีดังนี้

5. Application Programming Interface (API)

คือ ระบบบริการข้อมูลกลางระหว่าง client และการทำงานฝั่ง Server Side หน้าที่หลักของ API คือคอยรับ คำสั่งจากฝั่ง client ซึ่งก็คือแอปพลิเคชันต่าง ๆ แอปพลิเคชันในที่นี้หมายถึงทั้งเว็บ แอปพลิเคชันแอปพลิเคชันมือถือและอื่น ๆ เรียกคำสั่งที่ได้จากฝั่ง client ว่าการ request เมื่อเกิดคำสั่ง หรือการร้องขอใด ๆ ตัว API จะรับคำสั่งนั้น ๆ นำไปประมวลผลและสรุปเป็นก้อนข้อมูลที่ตรงกับกรร้องขอและส่งข้อมูลเหล่านั้น กลับไปที่ส่วนของ client หรือ แอปพลิเคชันอีกครั้ง เรียกการทำงานในขั้นตอนนี้ว่า response API ส่วนการทำงานของ client Response การคืนข้อมูลภาษาโปรแกรมมิ่งที่ใช้พัฒนา API จะเป็นภาษาแบบ back-end languages ตัวอย่างเช่น PHP, Python, Ruby ข้อมูลที่ถูกส่งค่ากลับมาจาก API มักจะอยู่ในรูปแบบ JSON, XML, CSV (บริษัท โค้ดปี จำกัด, 2014)

Representational State Transfer Application Programming Interface (REST API)

เป็นวิธีในการสร้าง Web Service รูปแบบหนึ่งที่อาศัย HTTP Method อันได้แก่ GET POST PUT และ DELETE ในการทำงาน และ ส่งกลับมาในรูปแบบของ JSON หรือ XML ซึ่ง XML หรือ Extensible Markup Language เป็นภาษาหนึ่งที่ใช้ในการแสดงผลข้อมูล เมื่อเทียบกับภาษา HTML จะต่างกันคือ HTML จะถูกออกแบบมาเพื่อการแสดงผล อย่างเดียวเท่านั้นแต่ XML นั้นถูกออกแบบมาเพื่อเก็บข้อมูลโดยทั้งข้อมูลและโครงสร้างของข้อมูลนั้น ๆ ไว้ด้วยกัน ส่งผลให้สามารถรับ-ข้อมูลไปมาข้าม Platform ได้อย่างสะดวกมากขึ้น เพราะเป็นการเรียกผ่าน HTTP Protocol และเนื่องจาก REST ส่งค่ากลับมาในรูปแบบของ JSON หรือ XML ซึ่งมีขนาดเล็กทำให้เมื่อต้องการใช้ข้อมูลก็สามารถเข้าถึงได้สะดวกมากขึ้น

API ย่อมาจาก Application Programming Interface ในภาพรวมการทำงานของ REST API จะทำโดยรองรับการเรียกของแอปพลิเคชันว่าแอปพลิเคชันต้องการข้อมูลอะไรแล้วหลังจากนั้น API ก็จะนำข้อมูลจากฐานข้อมูลออกตามที่แอปพลิเคชันต้องการ และกลับไปในรูปแบบของ XML หรือ JSON Method กระบวนการทำงานหลัก ๆ ของ API จะมีกระบวนการทำงานหลัก ๆ ในการรับส่งข้อมูลกันโดยแบ่งกระบวนการทำงานได้ดังนี้

1. POST Method ใช้สำหรับการสร้างข้อมูลใหม่ในฐานข้อมูล
2. GET Method ใช้สำหรับร้องขอข้อมูลในรูปแบบต่าง ๆ
3. PUT Method ใช้สำหรับแทนที่ข้อมูลเดิม
4. PATCH Method ใช้สำหรับปรับปรุงแก้ไขข้อมูลเดิม
5. DELETE Method ใช้สำหรับการลบข้อมูล



ภาพที่ 3 ลักษณะการทำงานของ Application Programming Interface

6. ภาษา PHP

PHP ย่อมาจาก PHP Hypertext Preprocessor แต่เดิมนย่อมาจาก Personal Home Page Tools คือ ภาษาคอมพิวเตอร์จำพวก scripting language ภาษาจำพวกนี้คำสั่งต่าง ๆ จะเก็บอยู่ในไฟล์ที่เรียกว่า script และเวลาใช้งานต้องอาศัยตัวแปลชุดคำสั่ง ตัวอย่างของภาษาสคริปก็เช่น JavaScript, Perl เป็นต้น ลักษณะของ PHP ที่แตกต่างจากภาษาสคริปต์แบบอื่น ๆ คือ PHP ได้รับการพัฒนาและออกแบบมาเพื่อใช้งานในการสร้างเอกสารแบบ HTML โดยสามารถสอดแทรกหรือแก้ไขเนื้อหาได้โดยอัตโนมัติ ดังนั้นจึงกล่าวว่า PHP เป็นภาษาที่เรียกว่า server-side หรือ HTML-embedded scripting language นั่นคือในทุก ๆ ครั้งก่อนที่เครื่องคอมพิวเตอร์ซึ่งให้บริการเป็น Web server จะส่งหน้าเว็บเพจที่เขียนด้วย PHP ให้เรา มันจะทำการประมวลผลตามคำสั่งที่มีอยู่ให้เสร็จเสียก่อน แล้วจึงค่อยส่งผลลัพธ์ที่ได้ให้เรา ผลลัพธ์ที่ได้นั้นก็คือเว็บเพจที่เราเห็นนั่นเอง ถือได้ว่า PHP เป็นเครื่องมือที่สำคัญชนิดหนึ่งที่ช่วยให้เราสามารถสร้าง Dynamic Web pages (เว็บเพจที่มีการโต้ตอบกับผู้ใช้) ได้อย่างมีประสิทธิภาพและมีลูกเล่นมากขึ้น (mindphp, 2542)

PHP เป็นผลงานที่เติบโตมาจากกลุ่มของนักพัฒนาในเชิงเปิดเผยรหัสต้นฉบับหรือ Opensource ดังนั้น PHP จึงมีการพัฒนาไปอย่างรวดเร็ว และแพร่หลายโดยเฉพาะอย่างยิ่งเมื่อใช้ร่วมกับ Apache Web server ระบบปฏิบัติการอย่างเช่น Linux หรือ FreeBSD เป็นต้น ในปัจจุบัน PHP สามารถใช้ร่วมกับ Web Server หลาย ๆ ตัวบนระบบปฏิบัติการอย่างเช่น Windows 95/98/NT เป็นต้น

ลักษณะเด่นของ PHP คือ

1. ใช้ได้ฟรี
2. PHP เป็นโปรแกรมวิ่งข้าง Sever ดังนั้นขีดความสามารถไม่จำกัด
3. Conlatfun นั่นคือPHP วิ่งบนเครื่อง UNIX, Linux, Windows ได้หมด
4. เรียนรู้ง่าย เนื่องจาก PHP ผีงเข้าไปใน HTML และใช้โครงสร้างและไวยากรณ์ภาษาง่าย ๆ
5. เร็วและมีประสิทธิภาพ โดยเฉพาะเมื่อใช้กับ Apach Xerve เพราะไม่ต้องใช้โปรแกรมจากภายนอก
6. ใช้ร่วมกับ XML ได้ทันที
7. ใช้กับระบบแฟ้มข้อมูลได้
8. ใช้กับข้อมูลตัวอักษรได้อย่างมีประสิทธิภาพ
9. ใช้กับโครงสร้างข้อมูล แบบ Scalar, Array, Associative array
10. ใช้กับการประมวลผลภาพได้

7. โปรแกรมระบบจัดการฐานข้อมูลด้วย MySQL

คือ โปรแกรมระบบจัดการฐานข้อมูล ที่พัฒนาโดยบริษัท MySQL AB มีหน้าที่เก็บข้อมูลอย่างเป็นระบบ รองรับคำสั่ง SQL เป็นเครื่องมือสำหรับเก็บข้อมูล ที่ต้องใช้ร่วมกับเครื่องมือหรือโปรแกรมอื่นอย่างบูรณาการ เพื่อให้ได้ระบบงานที่รองรับ ความต้องการของผู้ใช้ เช่นทำงานร่วมกับเครื่องบริการเว็บ (Web Server) เพื่อให้บริการแก่ภาษาสคริปต์ที่ทำงานฝั่งเครื่องบริการ (Server-Side Script) เช่น ภาษา php ภาษา asp.net หรือภาษาเจเอสพี เป็นต้น หรือทำงานร่วมกับโปรแกรมประยุกต์ (Application Program) เช่น ภาษาวิซวลเบสิกคอตเน็ต ภาษาจาวา หรือภาษาซีชาร์ป เป็นต้น โปรแกรมถูกออกแบบให้สามารถทำงานได้บนระบบปฏิบัติการที่หลากหลาย และเป็นระบบฐานข้อมูลโอเพนซอร์ซ (Open Source) ที่ถูกนำไปใช้งานมากที่สุด (networks365, 2015)

MySQL ถือเป็นระบบจัดการฐานข้อมูล Database Management System (DBMS) ฐานข้อมูลมีลักษณะเป็นโครงสร้างของการเก็บรวบรวมข้อมูล การที่จะเพิ่มเติม เข้าถึงหรือประมวลผลข้อมูลที่เก็บในฐานข้อมูลจำเป็นจะต้องอาศัยระบบจัดการ ฐานข้อมูล ซึ่งจะทำหน้าที่เป็นตัวกลางในการจัดการกับข้อมูลในฐานข้อมูลทั้งสำหรับการ ใช้งานเฉพาะ และรองรับการทำงานของแอปพลิเคชันอื่น ๆ ที่ต้องการใช้งานข้อมูลในฐานข้อมูล เพื่อให้ได้รับความสะดวกในการจัดการกับข้อมูลจำนวนมาก MySQL ทำหน้าที่เป็นทั้งตัวฐานข้อมูลและระบบจัดการฐานข้อมูล

MySQL เป็นระบบจัดการฐานข้อมูลแบบ relational จะทำการเก็บข้อมูลทั้งหมดในรูปแบบของตารางแทนการเก็บข้อมูลทั้งหมดลงในไฟล์เพียงไฟล์เดียว ทำให้ทำงานได้รวดเร็วและมีความยืดหยุ่น นอกจากนั้น แต่ละตารางที่เก็บข้อมูลสามารถเชื่อมโยงเข้าหากันทำให้สามารถรวมหรือจัดกลุ่มข้อมูลได้

ตามต้องการ โดยอาศัยภาษา SQL ที่เป็นส่วนหนึ่งของโปรแกรม MySQL ซึ่งเป็นภาษามาตรฐานในการเข้าถึงฐานข้อมูล

MySQL แจกจ่ายให้ใช้งานแบบ Open Source นั่นคือ ผู้ใช้งาน MySQL ทุกคนสามารถใช้งานและปรับแต่งการทำงานได้ตามต้องการ สามารถดาวน์โหลดโปรแกรม MySQL ได้จากอินเทอร์เน็ตและนำมาใช้งานโดยไม่มีค่าใช้จ่ายใด ๆ

8. กรอบแนวคิด

การศึกษาครั้งนี้เป็นการศึกษาเพื่อพัฒนาระบบเชื่อมโยงโปรแกรมที่เกี่ยวข้องกับบริการเพื่อการดูแลภาวะฉุกเฉินด้านการแพทย์และสาธารณสุข จังหวัดเชียงใหม่ โดยใช้กระบวนการวิจัยและการพัฒนาซึ่งแบ่งออกเป็น 3 ระยะ ได้แก่ ระยะที่ 1 ระยะการวิเคราะห์สถานการณ์ โดยศึกษาปัญหาและความต้องการเพื่อให้ได้ระบบเชื่อมโยงโปรแกรมฯ ตามความต้องการของผู้ใช้งาน ระยะที่ 2 ระยะดำเนินการ เป็นการทดลองใช้ระบบเชื่อมโยงโปรแกรมที่พัฒนาขึ้น และระยะที่ 3 ระยะประเมินผล โดยประเมินความเป็นไปได้ของการนำระบบเชื่อมโยงโปรแกรมที่พัฒนาขึ้นไปใช้ เพื่อเป็นแนวทางในการขยายผลต่อไป

บทที่ 3

วิธีดำเนินการวิจัย

การศึกษาครั้งนี้เป็นการศึกษาแบบวิจัยและพัฒนา (Research and Development) เพื่อพัฒนาระบบเชื่อมโยงโปรแกรมที่เกี่ยวข้องกับบริการเพื่อการดูแลภาวะฉุกเฉินด้านการแพทย์และสาธารณสุข จังหวัดเชียงใหม่ และศึกษาความเป็นไปได้ของระบบเชื่อมโยงโปรแกรมฯ โดยแบ่งการศึกษาออกเป็น 3 ระยะ คือ 1) ระยะวิเคราะห์สถานการณ์ 2) ระยะดำเนินการ และ 3) ระยะประเมินผล โดยทำการศึกษาในโรงพยาบาลในจังหวัดเชียงใหม่ 15 ตุลาคม พ.ศ. 2562 ถึง 30 พฤศจิกายน พ.ศ. 2563

1. ประชากรและกลุ่มตัวอย่าง

ประชากรและกลุ่มตัวอย่างที่ใช้ในการศึกษาวิจัยครั้งนี้ แบ่งออกตามกระบวนการในการดำเนินการวิจัย ดังนี้

ระยะที่ 1 ระยะวิเคราะห์สถานการณ์

ประชากร แบ่งเป็น 2 กลุ่ม ประกอบด้วย

1. บุคลากร ในกลุ่มงานยุทธศาสตร์ สำนักงานสาธารณสุขจังหวัดเชียงใหม่ ที่ทำหน้าที่ในการบริหารและวางแผนพัฒนาระบบสาธารณสุข
2. บุคลากรในโรงพยาบาลที่ดูแลระบบเทคโนโลยีสารสนเทศของโรงพยาบาล ในจังหวัดเชียงใหม่

กลุ่มตัวอย่าง คัดเลือกแบบเฉพาะเจาะจง (purposive sampling) โดยแบ่งเป็น 2 กลุ่ม ประกอบด้วย

1. บุคลากรในสำนักงานสาธารณสุขจังหวัดเชียงใหม่ จำนวน 5 คน ที่มีคุณสมบัติดังนี้
 - 1.1 เป็นบุคคลที่ทำงานอยู่ในกลุ่มงานยุทธศาสตร์ สำนักงานสาธารณสุขจังหวัดเชียงใหม่
 - 1.2 ทำหน้าที่เชียงใหม่ ที่ทำหน้าที่ในการบริหารและวางแผนพัฒนาระบบสาธารณสุข
 - 1.3 เป็นผู้ยินยอมให้ความร่วมมือในการศึกษา
2. บุคลากรในโรงพยาบาล จำนวน 10 คน ที่มีคุณสมบัติ ดังนี้
 - 2.1 ทำหน้าที่ดูแลระบบเทคโนโลยีสารสนเทศของโรงพยาบาล ในจังหวัดเชียงใหม่
 - 2.2 เป็นผู้ยินยอมให้ความร่วมมือในการศึกษา

ระยะที่ 2-3 ระยะดำเนินการและการประเมินผล

ประชากร แบ่งเป็น 2 กลุ่ม ประกอบด้วย

1. บุคลากร ในกลุ่มงานยุทธศาสตร์ สำนักงานสาธารณสุขจังหวัดเชียงใหม่ ที่ทำหน้าที่ในการบริหารและวางแผนพัฒนาระบบสาธารณสุข
2. บุคลากรในโรงพยาบาลที่ใช้ระบบเทคโนโลยีสารสนเทศของโรงพยาบาล ในจังหวัดเชียงใหม่

กลุ่มตัวอย่าง คัดเลือกแบบเฉพาะเจาะจง โดยแบ่งเป็น 2 กลุ่ม ประกอบด้วย

1. บุคลากรในสำนักงานสาธารณสุขจังหวัดเชียงใหม่ จำนวน 3 คน ที่มีคุณสมบัติดังนี้
 - 1.1 เป็นบุคคลที่ทำงานอยู่ในกลุ่มงานยุทธศาสตร์ สำนักงานสาธารณสุขจังหวัดเชียงใหม่
 - 1.2 ทำหน้าที่เชียงใหม่ ที่ทำหน้าที่ในการบริหารและวางแผนพัฒนาระบบสาธารณสุข
 - 1.3 เป็นผู้ยินยอมให้ความร่วมมือในการศึกษา
2. บุคลากรในโรงพยาบาล ในจังหวัดเชียงใหม่ จำนวน 31 คน ที่มีคุณสมบัติดังนี้
 - 2.1 ใช้ระบบเทคโนโลยีสารสนเทศของโรงพยาบาล ในจังหวัดเชียงใหม่
 - 2.2 เป็นผู้ยินยอมให้ความร่วมมือในการศึกษา

2. เครื่องมือที่ใช้ในการวิจัย

เครื่องมือที่ใช้ในการศึกษาวิจัยครั้งนี้ ประกอบด้วย

1. แบบสอบถามข้อมูลทั่วไป ได้แก่ หน่วยงาน และงาน/กลุ่มงาน
2. แบบสอบถามความเป็นไปได้ของการใช้ระบบเชื่อมโยงโปรแกรมที่เกี่ยวข้องกับบริการเพื่อการดูแลภาวะฉุกเฉินด้านการแพทย์และสาธารณสุข ประกอบด้วยข้อคำถามทั้งหมด 3 ด้าน ได้แก่
 - 1) ความง่ายในการนำไปใช้
 - 2) ความปลอดภัยของข้อมูล
 - 3) ประสิทธิภาพการใช้งาน ซึ่งลักษณะคำถามเป็นแบบมาตราส่วนประมาณค่า (Rating scale) เลือกตอบได้เพียง 1 คำตอบ โดยใส่เครื่องหมาย (/) ลงในช่องว่างที่กำหนดและต้องตอบให้ครบทุกข้อ ซึ่งแบ่งระดับคะแนนตามรายด้านทั้ง 3 ด้าน ดังนี้

ระดับคะแนนด้านความง่ายในการนำไปใช้ แบ่งออกเป็น 3 ระดับ โดยกำหนดเกณฑ์การประเมินดังนี้

- 1 คะแนน หมายถึง โปรแกรมฯ มีความง่ายในการนำไปใช้อยู่ในระดับน้อย
- 2 คะแนน หมายถึง โปรแกรมฯ มีความง่ายในการนำไปใช้อยู่ในระดับปานกลาง
- 3 คะแนน หมายถึง โปรแกรมฯ มีความง่ายในการนำไปใช้อยู่ในระดับมาก

ระดับคะแนนด้านความปลอดภัยของข้อมูล แบ่งออกเป็น 3 ระดับ โดยกำหนดเกณฑ์การประเมินดังนี้

- 1 คะแนน หมายถึง โปรแกรมา มีความปลอดภัยของข้อมูลอยู่ในระดับน้อย
- 2 คะแนน หมายถึง โปรแกรมา มีความปลอดภัยของข้อมูลอยู่ในระดับปานกลาง
- 3 คะแนน หมายถึง โปรแกรมา มีความปลอดภัยของข้อมูลอยู่ในระดับมาก

ระดับคะแนนด้านประสิทธิภาพการใช้งาน แบ่งออกเป็น 3 ระดับ โดยกำหนดเกณฑ์การประเมินดังนี้

- 1 คะแนน หมายถึง โปรแกรมา มีประสิทธิภาพการใช้งานอยู่ในระดับน้อย
- 2 คะแนน หมายถึง โปรแกรมา มีประสิทธิภาพการใช้งานอยู่ในระดับปานกลาง
- 3 คะแนน หมายถึง โปรแกรมา มีประสิทธิภาพการใช้งานอยู่ในระดับมาก

การแปลความหมายค่าคะแนนเฉลี่ยใช้เกณฑ์การแปลความหมายแบ่งออกเป็น 3 ระดับ ดังนี้

- คะแนนเฉลี่ยระหว่าง 2.5-3.0 หมายถึง มีความเป็นไปได้ในระดับมาก
คะแนนเฉลี่ยระหว่าง 1.5-2.49 หมายถึง มีความเป็นไปได้ในระดับปานกลาง
คะแนนเฉลี่ยระหว่าง 0-1.49 หมายถึง มีความเป็นไปได้ในระดับน้อย

3. ขั้นตอนและวิธีการรวบรวมข้อมูล

ในการรวบรวมข้อมูล ผู้วิจัยดำเนินการรวบรวมข้อมูลด้วยตนเองทั้งหมดตามขั้นตอนดังนี้

3.1 ขั้นตอนเตรียมการดำเนินการวิจัย

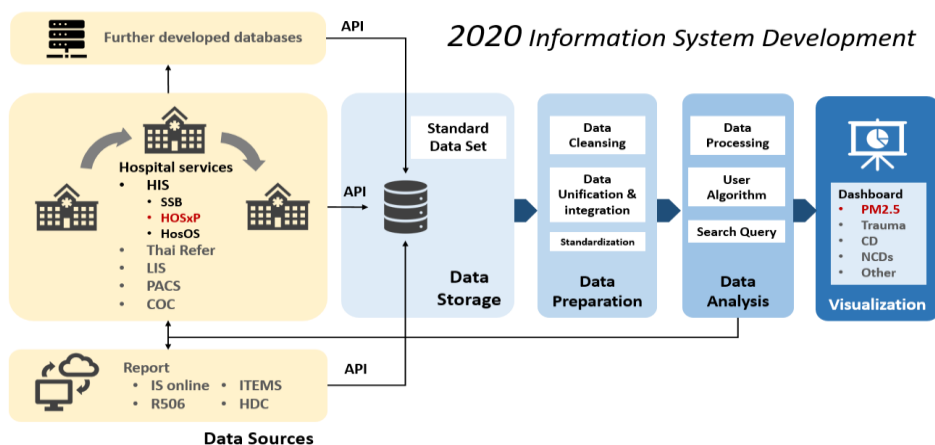
- 3.1.1 ทำหนังสือจากสำนักงานสาธารณสุขจังหวัดเชียงใหม่ ถึง โรงพยาบาลในสังกัดสำนักงานสาธารณสุขจังหวัดเชียงใหม่ เพื่อขออนุญาตเข้าดำเนินการวิจัยกับกลุ่มตัวอย่าง
- 3.1.2 ภายหลังจากได้รับการอนุญาตจากผู้อำนวยการโรงพยาบาล ผู้วิจัยเข้าชี้แจงวัตถุประสงค์การวิจัยและรายละเอียดต่าง ๆ ต่อเจ้าหน้าที่ในหน่วยงานเทคโนโลยีสารสนเทศเพื่อขอความร่วมมือในการดำเนินการวิจัย
- 3.1.3 ผู้วิจัยคัดเลือกกลุ่มตัวอย่างแบบเฉพาะเจาะจงตามคุณสมบัติที่กำหนดไว้ จากบัญชีรายชื่อของผู้ที่เกี่ยวข้องกับการใช้ระบบสารสนเทศในโรงพยาบาล
- 3.1.4 จัดเตรียมสถานที่ วัสดุ และอุปกรณ์ต่าง ๆ ที่จำเป็นสำหรับ ได้แก่ เครื่องคอมพิวเตอร์ เครื่องฉาย ทีวี โปรแกรมสำหรับประชุมออนไลน์

3.2 ขั้นตอนการวิจัย แบ่งออกเป็น 3 ระยะ ได้แก่

ระยะที่ 1 ระยะวิเคราะห์สถานการณ์

3.2.1 ประชุมคณะทำงานพัฒนาระบบเชื่อมโยงโปรแกรมที่เกี่ยวข้องกับการบริการเพื่อการดูแลภาวะฉุกเฉินด้านการแพทย์และสาธารณสุข ซึ่งแจ้งโครงการพัฒนาระบบเชื่อมโยงโปรแกรมฯ กำหนดความต้องการ การแสดงผลข้อมูล และระบุผู้ที่มีส่วนเกี่ยวข้องของแต่ละโรงพยาบาลพร้อมทั้งคัดเลือกกลุ่มตัวอย่างแบบเฉพาะเจาะจงตามคุณสมบัติที่กำหนดไว้จำนวน 15 คน

3.2.2 ศึกษาฐานข้อมูลที่เกี่ยวข้องกับการดูแลภาวะฉุกเฉินด้านการแพทย์และสาธารณสุข ได้แก่ ระบบฐานข้อมูล Refer โปรแกรม Thai Refer ระบบฐานข้อมูล Injury Surveillances (IS) โปรแกรม IS Online ระบบฐานข้อมูล ITEMS ระบบโปรแกรมโรงพยาบาล Hospital Information System (HIS) ได้แก่ HOSxP, HospitalOS, SSB และ JHCIS



ภาพที่ 4 ผังระบบการพัฒนาเชื่อมโยงโปรแกรมฯ

3.2.3 ประชุมกลุ่มตัวอย่างเพื่อวิเคราะห์ปัญหาและความต้องการในการใช้สารสนเทศที่เกี่ยวข้องกับการดูแลภาวะฉุกเฉินด้านการแพทย์และสาธารณสุขของจังหวัดเชียงใหม่

3.2.4 ออกแบบระบบเชื่อมโยง (Application Programming Interface: API) เพื่อเป็นช่องทางรับ-ส่งข้อมูล ระหว่างโปรแกรมของหน่วยบริการกับฐานข้อมูลกลาง พร้อมทั้งออกแบบโครงสร้างข้อมูลกลาง (Data Structure) เพื่อเก็บข้อมูลจากหน่วยบริการที่โครงสร้างข้อมูลที่แตกต่างกัน ดังแสดงในตารางที่ 1

ตารางที่ 1 โครงสร้างฐานข้อมูลกลาง (data structure)

ลำดับ	รายการข้อมูล	คำนิยาม	ชื่อข้อมูล
1	รหัสสถานพยาบาล	รหัสสถานพยาบาล ตามมาตรฐานกอง ยุทธศาสตร์และแผนงาน	HCODE
2	เลขบัตรประจำตัวผู้ป่วย	เลขทะเบียนการมารับบริการ (สามารถกำหนด ได้ตั้งแต่ 1-15 หลัก)	HN
3	วันเดือนปีที่เข้ารับการ รักษาแผนกผู้ป่วยนอก	วันเดือนปีที่มารับบริการ กำหนดเป็น ค.ศ. (YYYYMMDD) YYYY =ปีค.ศ., MM=เดือน 2 หลัก 01-12, DD=วันที่ 2 หลัก 01-31 หมายเหตุ : กรณีที่บันทึกข้อมูลย้อนหลังให้ เปลี่ยนวันกลับเป็นวันที่รับบริการจริง	DATEOPD
4	ชื่อ	ชื่อ	FNAME
5	นามสกุล	นามสกุล	SNAME
6	เลขบัตรประจำตัว ประชาชน	เลขประจำตัวประชาชน ตามกรมการปกครอง กำหนดเป็นรหัสประจำตัวบุคคลหมายเหตุ : เป็นเลขจำนวนเต็ม 13 หลัก (ประชาชน สัญชาติไทยต้องบันทึกทุกราย)	CID
7	วันเดือนปีเกิด	วันเดือนปีเกิด กำหนดเป็น ค.ศ. (YYYYMMDD) (หากไม่ทราบวัน เดือนที่เกิด แต่ทราบ ค.ศ. เกิด ให้กำหนดวันเกิดเป็นวันที่ 1 มกราคมของปี ค.ศ.นั้น ๆ) หมายเหตุ : YYYY =ปีค.ศ., MM=เดือน 2 หลัก 01-12, DD=วันที่ 2 หลัก 01-31และสามารถคำนวณ เป็นวันที่ตามมาตรฐานได้	DOB
8	อายุนับเป็นจำนวนปี	อายุนับเป็นจำนวนปี	AGE
9	อายุนับเป็นจำนวนวัน	อายุนับเป็นจำนวนวัน	AGEDAY
10	เพศ	1 = ชาย, 2 = หญิง	SEX
11	รหัสพื้นที่	รหัสมาตรฐานตามกรมการปกครอง	AREA
12	สัญชาติ	รหัสมาตรฐานตามกรมการปกครอง ถ้าไม่ ทราบให้ระบุ 999 ตามรหัสมาตรฐาน	NATION
13	การวินิจฉัยโรค	(DIAG1 - DIAG5) และ DIAG1 คือ การ วินิจฉัยโรคหลัก (Principle Diagnosis)	DIAG

ลำดับ	รายการข้อมูล	คำนิยาม	ชื่อข้อมูล
14	รหัสสถานพยาบาลที่รับย้ายผู้ป่วย	รหัสสถานพยาบาลที่รับย้ายผู้ป่วยจากสถานพยาบาลอื่นเพื่อมารับการตรวจหรือรักษาต่อ	REFERIN
15	รหัสสถานพยาบาลที่ส่งต่อผู้ป่วย	รหัสสถานพยาบาลที่ส่งต่อผู้ป่วยเพื่อไปรับการตรวจหรือรักษาต่อยังสถานพยาบาลอื่น	REFEROUT
16	สิทธิการรักษาพยาบาล	รหัสสิทธิมาตรฐานที่กำหนดโดยหน่วยงานที่เกี่ยวข้อง	INSURE
17	วันนัดหมาย	วันเดือนปีทีนัดครั้งต่อไป กำหนดเป็น ค.ศ. (YYYYMMDD) YYYY = ปี ค.ศ., MM = เดือน 2 หลัก 01-12, DD=วันที่ 2 หลัก 01-31	APPOINT
18	สถานะเมื่อรับบริการเสร็จแล้ว	<ol style="list-style-type: none"> 1 จำหน่ายกลับบ้าน 2 รับไว้รักษาต่อในแผนกผู้ป่วยใน 3 ส่งต่อไปยังสถานพยาบาลอื่น 4 เสียชีวิต 5 เสียชีวิตก่อนมาถึงสถานพยาบาล 6 เสียชีวิตระหว่างส่งต่อไปยังสถานพยาบาลอื่น 7 ปฏิเสธการรักษา 8 หนีกลับ 9 การให้บริการโดยไม่มีคำวินิจฉัยโรค 	TYPEOUT
20	ห้องฉุกเฉิน	ห้องฉุกเฉิน	ER
21	การคัดแยกผู้ป่วย	<ol style="list-style-type: none"> 1 การช่วยฟื้นคืนชีพ (Resuscitation) 2 ผู้ป่วยฉุกเฉิน (Emergency) 3 ผู้ป่วยเร่งด่วน (Urgent) 4 ผู้ป่วยกึ่งเร่งด่วน/เร่งด่วนน้อย (Semi/Less urgent) 5 ผู้ป่วยไม่เร่งด่วน (Non-urgent) 	TRIAGE

3.2.5 จัดทำโปรแกรมประยุกต์ CMDS (ChiangMai Data Sync) ด้วย NET Framework เพื่อเชื่อมฐานข้อมูลหน่วยบริการ และจัดส่งข้อมูลเข้าสู่ฐานข้อมูลกลาง ซึ่งจะนำไปติดตั้งที่เครื่องคอมพิวเตอร์ของแต่ละโรงพยาบาล

3.2.6 อบรมเชิงปฏิบัติการผู้รับผิดชอบงานเทคโนโลยีสารสนเทศของระบบ HOSxP, HospitalOS, SSB และ JHCIS เพื่อเขียนคำสั่ง ภาษา SQL Script ส่งออกข้อมูลด้านภาวะฉุกเฉินด้านการแพทย์ และการวินิจฉัยโรค ของโรงพยาบาล ผ่านโปรแกรม CMDS สู่มูลนิธิข้อมูลกลาง (API datacenter)

3.2.7 จัดอบรมเชิงปฏิบัติการผู้รับผิดชอบงานข้อมูลและเทคโนโลยีสารสนเทศ จากทุกโรงพยาบาล และหน่วยงานที่เกี่ยวข้อง เพื่อพัฒนาศักยภาพเครือข่าย ให้สามารถติดตั้งโปรแกรม CMDS การเขียนชุดคำสั่ง เพื่อออกรายงานด้วย MySQL

3.2.8 ออกแบบระบบบริหารจัดการการแสดงผลข้อมูล รายงาน และ Dashboard ต่าง ๆ (CM Dashboard center)

3.2.9 ใช้ Google data studio ในการการสร้างรายงานและ Dashboard ภาวะฉุกเฉินทางการแพทย์และสาธารณสุข เพื่อใช้ประโยชน์ในการเฝ้าระวัง และการพัฒนาบริการ

3.2.10 พัฒนาเว็บแอปพลิเคชันในการแสดงผลรายงานและ Dashboard เพื่อเปรียบเทียบกับกลุ่มโรคระบบทางเดินหายใจและฝุ่น PM2.5

ระยะที่ 2 ระยะดำเนินการ

1. คัดเลือกกลุ่มตัวอย่างแบบเฉพาะเจาะจงตามคุณสมบัติที่กำหนดไว้จำนวน 34 คน
2. จัดประชุมกลุ่มตัวอย่างเพื่ออธิบายและสาธิตวิธีการใช้โปรแกรมที่เกี่ยวข้องกับบริการเพื่อการดูแลภาวะฉุกเฉินด้านการแพทย์และสาธารณสุข
3. กลุ่มตัวอย่างนำโปรแกรมที่เกี่ยวข้องกับบริการเพื่อการดูแลภาวะฉุกเฉินด้านการแพทย์และสาธารณสุขที่พัฒนาขึ้นไปทดลองใช้ในโรงพยาบาล 24 แห่ง ประกอบไปด้วย
 - โรงพยาบาลศูนย์ จำนวน 1 แห่ง ได้แก่ โรงพยาบาลนครพิงค์
 - โรงพยาบาลทั่วไป จำนวน 2 แห่ง ได้แก่ โรงพยาบาลฝางและโรงพยาบาลจอมทอง
 - โรงพยาบาลชุมชน จำนวน 21 แห่ง ได้แก่ โรงพยาบาลสันป่าตอง โรงพยาบาลสันทรายโรงพยาบาลเชียงดาว โรงพยาบาลหางดง โรงพยาบาลดอยสะเก็ด โรงพยาบาลแม่ฮ้อย โรงพยาบาลพร้าว โรงพยาบาลฮอด โรงพยาบาลแม่แตง โรงพยาบาลเทพรัตนเวชชานุกูล โรงพยาบาลสันกำแพง โรงพยาบาลสารภี โรงพยาบาลลอมก๋อย โรงพยาบาลไชยปราการ โรงพยาบาลดอยเต่า โรงพยาบาลแม่วาง โรงพยาบาลแม่ฮ่องสอน โรงพยาบาลดอยหล่อ โรงพยาบาลสะเมิง โรงพยาบาลเวียงแหง และโรงพยาบาลวัดจันทร์เฉลิมพระเกียรติ 80 พรรษา
4. ลงพื้นที่เพื่อติดตาม

ระยะที่ 3 ระยะประเมินผล

ประเมินความเป็นไปได้ของการใช้ระบบเชื่อมโยงโปรแกรมเกี่ยวข้องกับบริการเพื่อการดูแลภาวะฉุกเฉินด้านการแพทย์และสาธารณสุข จังหวัดเชียงใหม่ โดยใช้แบบสอบถามความเป็นไปได้ของการใช้ระบบเชื่อมโยงโปรแกรมฯ ซึ่งประเมินผลหลังจากได้นำระบบเชื่อมโยงโปรแกรมฯ ทดลองใช้เป็นระยะเวลา 1 เดือน โดยให้บุคลากรจากสำนักงานสาธารณสุขจังหวัดเชียงใหม่และโรงพยาบาล จำนวน 34 คน ตอบแบบสอบถาม

4. การวิเคราะห์ข้อมูล

1. วิเคราะห์ข้อมูลเชิงคุณภาพ โดยนำข้อมูลที่ได้จากการประชุมมาจำแนกเนื้อหาและสรุปเป็นประเด็นต่าง ๆ เพื่อใช้ในการพัฒนาโปรแกรมฯ
2. วิเคราะห์ข้อมูลทั่วไป และความเป็นไปได้ของการใช้ระบบเชื่อมโยงโปรแกรมที่เกี่ยวข้องกับบริการเพื่อการดูแลภาวะฉุกเฉินด้านการแพทย์และสาธารณสุขจังหวัดเชียงใหม่ โดยใช้สถิติเชิงพรรณนา ได้แก่ การแจกแจงความถี่ ร้อยละ ค่าเฉลี่ย และส่วนเบี่ยงเบนมาตรฐาน

บทที่ 4

ผลการศึกษา

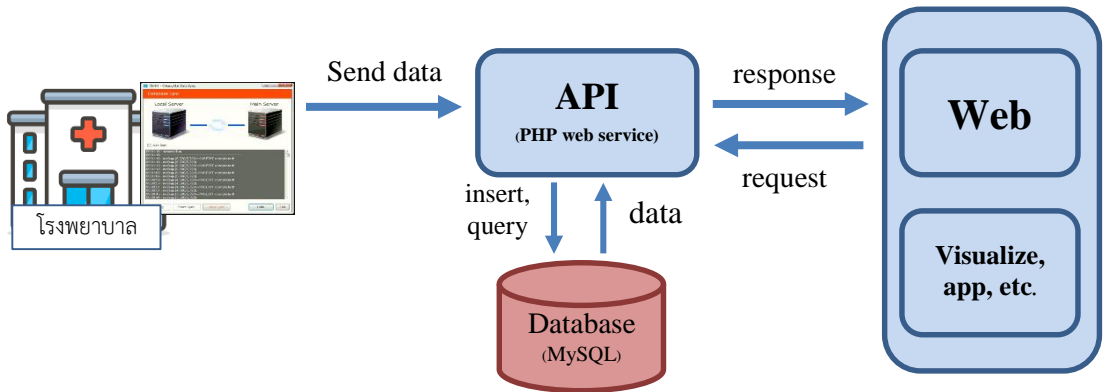
การศึกษาค้นคว้าครั้งนี้เป็นการศึกษาเพื่อพัฒนาระบบเชื่อมโยงโปรแกรมที่เกี่ยวข้องกับบริการเพื่อดูแลภาวะฉุกเฉินด้านการแพทย์และสาธารณสุข จังหวัดเชียงใหม่ ผลการวิเคราะห์ข้อมูลแบ่งเป็น 3 ส่วน ดังนี้

- ส่วนที่ 1 การพัฒนาระบบเชื่อมโยง (Application Programming Interface)
- ส่วนที่ 2 ระบบบริหารจัดการรายงานด้านการแพทย์ฉุกเฉินและสาธารณสุข
- ส่วนที่ 3 ผลความเป็นไปได้ของการใช้ระบบเชื่อมโยงโปรแกรมที่เกี่ยวข้องกับบริการเพื่อดูแลภาวะฉุกเฉินด้านการแพทย์และสาธารณสุขจังหวัดเชียงใหม่

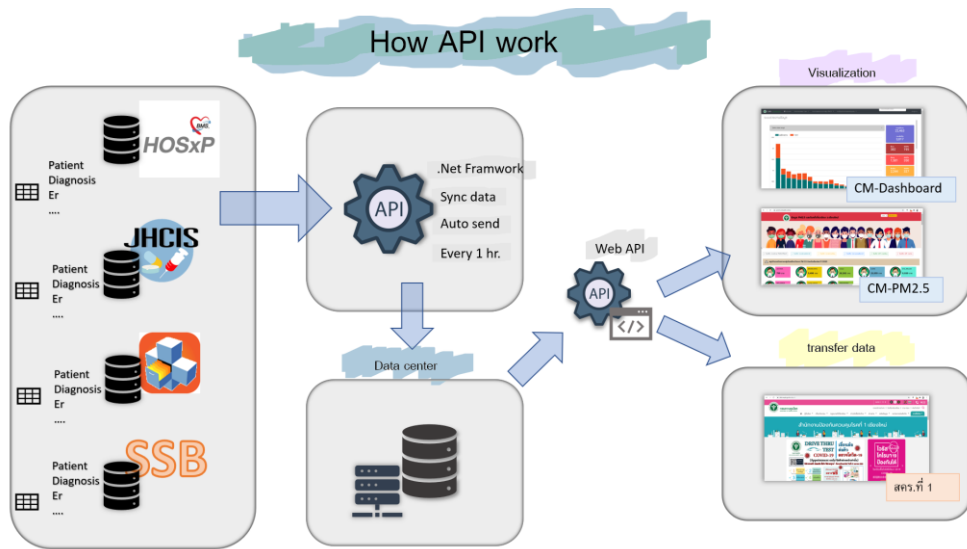
1. การพัฒนาระบบเชื่อมโยง (Application Programming Interface: API)

จากการวิเคราะห์ปัญหาด้านการรับ-ส่งข้อมูลด้านการแพทย์และสาธารณสุข ที่ผ่านมานั้นยังถือว่าไม่ครอบคลุมสำหรับการนำมาใช้วิเคราะห์สถานการณ์สุขภาพของประชาชนในจังหวัดเชียงใหม่ในปัจจุบันได้ ถึงแม้ว่าตามนโยบายของกระทรวงสาธารณสุขจัดให้มีการนำส่งฐานข้อมูลด้านการแพทย์และสาธารณสุข ในรูปแบบโครงสร้างมาตรฐานข้อมูลหรือที่เรียกกันว่าโครงสร้าง 43 แฟ้มแล้วก็ตาม ยังมีข้อมูลอื่น ๆ อีกมากที่ยังต้องการการเฉพาะเจาะจงรายละเอียด เช่น ข้อมูลด้านการแพทย์ฉุกเฉิน ข้อมูลด้านระบาดวิทยา และอื่น ๆ รวมทั้งปัญหาการจัดส่งข้อมูลที่ล่าช้าของหน่วยบริการสุขภาพ (โรงพยาบาล และโรงพยาบาลส่งเสริมสุขภาพตำบล) ที่ไม่ทันต่อสถานการณ์

จากการศึกษาเครื่องมือ (Tools) และเทคนิคทางคอมพิวเตอร์แล้ว พบว่า สามารถนำหลักการทางคอมพิวเตอร์มาประยุกต์ใช้เพื่อแก้ไขปัญหาดังกล่าวได้ โดยใช้หลักการทำงานของภาษาคอมพิวเตอร์ PHP (PHP Hypertext Preprocessor) API (Application Programming Interface) และ MySQL โดยแต่ละเครื่องมือจะทำหน้าที่ในการรับ-ส่งข้อมูลจากฐานข้อมูลสู่แอปพลิเคชันหรือเว็บไซต์ที่ร้องขอข้อมูล เพื่อนำข้อมูลที่ได้ไปวิเคราะห์สถานการณ์สุขภาพของประชาชนได้อย่างเป็นปัจจุบัน ทันต่อสถานการณ์ของโรค

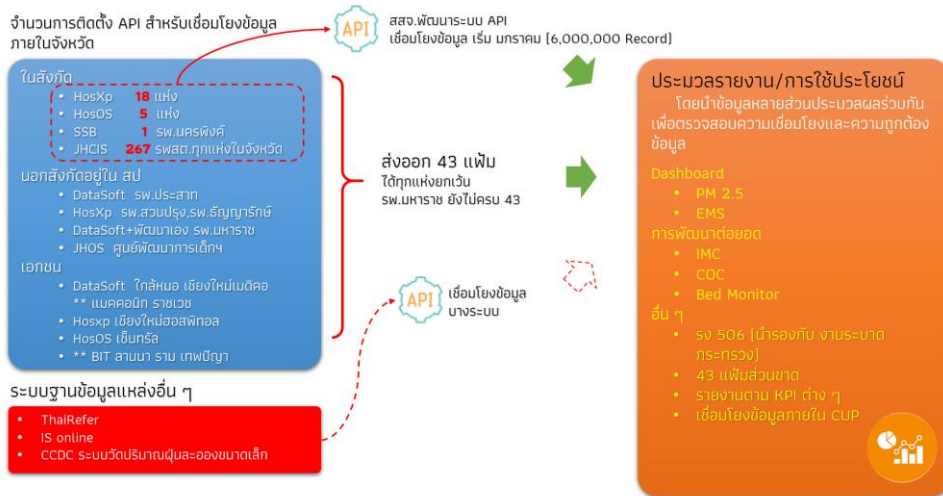


ภาพที่ 5 ผังไหลเวียนการเชื่อมโยงข้อมูลผ่าน API



ภาพที่ 6 การทำงานของ API

ระบบการจัดเก็บข้อมูลรายงานการส่งต่อ



ภาพที่ 7 ผังไหลเวียนระบบการจัดเก็บข้อมูลรายงานการส่งต่อ

ผลที่ได้จากการที่ระบบ API ทำงาน จะมีการรับส่งข้อมูลกันในระบบด้วยรูปแบบโครงสร้างภาษา JSON (JavaScript Object Notation) ซึ่งสามารถนำผลที่ได้ไปเก็บรวบรวมเป็นตารางข้อมูล ใน API data center หรือสามารถใช้ web application แสดงผลของ JSON ในรูปแบบ dashboard หรือรายงานได้เลย

```

1 // 28281109234636
2 // https://www.chiangmaihealth.go.th/cmpho_web/api/index.php?service_type=t_ems_kpi
3
4 * [
5 * {
6   "hoscode": 10713,
7   "hosname": "โรงพยาบาลนครพิงค์",
8   "y": 2562,
9   "HCODE": 10713,
10  "tiage": 4,
11  "Not_er": 0,
12  "ER": 4,
13  "EMS": 0,
14  "Stroke": 0,
15  "Brain_Injury": 0,
16  "Spinal_Cord_Injury": 0,
17  "dath_24": 0
18  },
19  * {
20  "hoscode": 10713,
21  "hosname": "โรงพยาบาลนครพิงค์",
22  "y": 2563,
23  "HCODE": 10713,
24  "tiage": 4,
25  "Not_er": 11,
26  "ER": 109,
27  "EMS": 0,
28  "Stroke": 0,
29  "Brain_Injury": 0,
30  "Spinal_Cord_Injury": 0,
31  "dath_24": 0
32  }
33 ]

```

ภาพที่ 8 ตัวอย่างข้อมูลแบบ JSON Format

จากนั้น จึงพัฒนาโปรแกรม CMDS เพื่อกำหนดชุดคำสั่งประมวลผลข้อมูลเวลาจัดส่ง เพื่อติดตั้งที่หน่วยบริการและส่งข้อมูลเข้า API data center

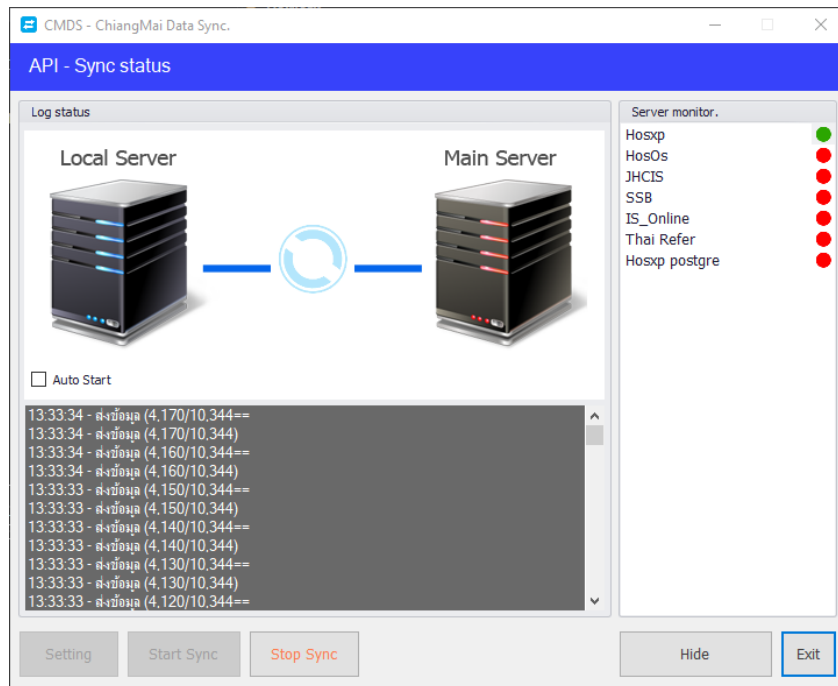
1.1 การพัฒนาโปรแกรม CMDS-Chiangmai Data Sync

ในการจัดส่งข้อมูลเข้าสู่ API ของจังหวัดเชียงใหม่ สามารถจัดส่งโดยหน่วยบริการจัดทำชุดคำสั่งและส่งข้อมูลโดยใช้เครื่องมือในการจัดส่งข้อมูลตามกรอบโครงสร้างของข้อมูล แต่ในการจัดทำชุดคำสั่งและการจัดส่งนั้น ไม่สามารถกำหนดเวลาในการจัดส่งและไม่สามารถกำหนดรหัสชุดคำสั่งในการประมวลผลข้อมูลให้เหมือนกันได้ในทุกสถานบริการ ทำให้เกิดข้อมูลที่ไม่ถูกต้องตามเงื่อนไข ดังนั้น จึงได้จัดทำโปรแกรม CMDS สำหรับจัดส่งข้อมูลจากหน่วยบริการเข้าสู่ API ซึ่งมีวิธีการดังภาพ



ภาพที่ 9 การทำงานของโปรแกรม CMDS

จากภาพจะเห็นว่าโปรแกรม CMDS จะทำการเรียก Scripts (ชุดคำสั่งสำหรับการประมวลผลจากฐานข้อมูล) จาก API ที่รวบรวมชุดคำสั่งไว้ของแต่ละ HIS (Hospital Information System) และทำการประมวลข้อมูล จัดส่งข้อมูลเข้าสู่ DataCenter ผ่าน API ตามช่วงเวลาที่ได้กำหนดไว้ ดังแสดงในภาพที่ 10



ภาพที่ 10 หน้าการทำงานขณะประมวลผลและจัดส่งข้อมูลของโปรแกรม CMDS

ในการประมวลผลข้อมูลจากหน่วยบริการใช้ชุดคำสั่ง Scripts ที่จัดทำขึ้นแยกตามประเภทของฐานข้อมูลที่ใช้ ดังนี้

1.2 ชุดคำสั่งสำหรับดึงและแปลงข้อมูลของโรงพยาบาลที่ใช้ HOSxP

```
SELECT
(SELECT hospitalcode FROM opdconfig LIMIT 1) HCODE,
v.hn,
v.vstdate DATEOPD,
v.vn,
p.fname FNAME,
p.lname SNAME,
p.cid,
p.birthday DOB,
v.age_y AGE,
TIMESTAMPDIFF(day,p.birthday,v.vstdate) AGEDAY,
p.sex as sex,
p.nationality NATION,
CONCAT(p.chwpart,p.ampart,p.tmbpart,LPAD(p.moopart,2,'0')) AREA,
v.pdx DIAG1,
v.dx0 DIAG2,
v.dx1 DIAG3,
v.dx2 DIAG4,
v.dx3 DIAG5,
IF(ri.refer_hospcode IS NULL,'NULL',ri.refer_hospcode) REFERIN,
IF(r.refer_hospcode IS NULL,'NULL',r.refer_hospcode) REFEROUT,
v.pcode INSURE,
IF(ap.vn IS NOT NULL,'Y','N') APPOINT,
'NULL' TYPEOUT,
IF(e.vn IS NOT NULL,'Y','N') ER,
IF(e.er_emergency_level_id IS NULL,'NULL',e.er_emergency_level_id) TRIAGE
FROM vn_stat AS v
INNER JOIN patient p ON p.hn = v.hn
LEFT JOIN er_regist e ON e.vn = av.vn
LEFT JOIN referout r ON r.vn = v.vn
```



```

LEFT JOIN referin ri ON ri.vn = v.vn
LEFT JOIN oapp ap ON ap.hn = v.hn AND ap.nextdate = v.vstdate
WHERE v.vstdate BETWEEN subdate(now(),interval 10 day) AND now()
GROUP BY v.vn

```

ผลลัพธ์จากการประมวลผลของ HOSXP

HCODE	hn	DATEOPD	vn	FNAME	SNAME	cid	DOB	AGE	AGEDAY	sex	NATION	AREA	DIAG1	DIAG2	DIAG3	DIAG4	DIAG5	REFERIN	REFEROUT	INSURE	APPOINT	TYPEOUT	ER
11122	5500229	2020-10-01	631001C				1966-09-25	54	19730	2	99	50051306						NULL	NULL	AJ	Y	NULL	N
11122	5503339	2020-10-01	631001C				1963-10-22	56	20799	2	99	50050102						NULL	NULL	AJ	Y	NULL	N
11122	5605139	2020-10-01	631001C				1969-03-13	51	18830	1	99	50050102						NULL	NULL	AB	Y	NULL	N
11122	5605140	2020-10-01	631001C				1970-07-18	48	18877	2	99	50050102						NULL	NULL	AB	Y	NULL	N
11122	5704852	2020-10-01	631001C				1971-07-30	48	17961	1	99	50051302						NULL	NULL	UC	Y	NULL	N
11122	5709831	2020-10-01	631001C				1960-12-26	59	21829	1	99	50050208						NULL	NULL	AB	Y	NULL	N
11122	5710750	2020-10-01	631001C				1961-02-10	59	21783	1	99	50051002						NULL	NULL	AB	Y	NULL	N
11122	5821799	2020-10-01	631001C				1965-09-19	55	20101	2	99	50230408						NULL	NULL	UC	Y	NULL	N
11122	5503795	2020-10-01	631001C				1955-07-29	65	23896	2	99	50531207						NULL	NULL	AG	Y	NULL	N
11122	5502789	2020-10-01	631001C				1961-01-25	59	21798	2	99	50050902						NULL	NULL	UC	Y	NULL	N
11122	5503371	2020-10-01	631001C				1950-02-24	70	25787	1	99	50050704						NULL	NULL	AG	Y	NULL	N
11122	5503544	2020-10-01	631001C				1971-06-27	49	17994	2	99	50050802						NULL	NULL	AJ	Y	NULL	N
11122	5508823	2020-10-01	631001C				1968-10-06	51	18988	2	99	50051302						NULL	NULL	AB	Y	NULL	N
11122	5609060	2020-10-01	631001C				1972-08-05	48	17589	1	99	50050802						NULL	NULL	AJ	Y	NULL	N
11122	5607792	2020-10-01	631001C				1971-01-23	49	18149	2	99	50050803						NULL	NULL	UC	Y	NULL	N
11122	5702621	2020-10-01	631001C				1968-05-29	52	19118	1	99	50051203						NULL	NULL	UC	Y	NULL	N
11122	5705413	2020-10-01	631001C				1979-07-02	41	15087	1	99	50050901						NULL	NULL	UC	Y	NULL	N
11122	5705416	2020-10-01	631001C				1964-09-30	56	20455	1	99	50050504						NULL	NULL	AE	Y	NULL	N
11122	5710753	2020-10-01	631001C				1971-09-30	49	17999	2	99	50050107						NULL	NULL	UC	Y	NULL	N
11122	5710961	2020-10-01	631001C				1983-01-16	37	13773	1	99	50050207						NULL	NULL	UC	Y	NULL	N
11122	5802332	2020-10-01	631001C				1971-12-08	48	17830	2	99	50051304						NULL	NULL	UC	Y	NULL	N
11122	5805704	2020-10-01	631001C				1973-03-10	45	16642	2	99	50110302						NULL	NULL	UC	Y	NULL	N
11122	5817580	2020-10-01	631001C				1962-02-09	58	21419	1	99	50110507						NULL	NULL	AB	Y	NULL	N
11122	5901123	2020-10-01	631001C				1970-01-26	50	18511	2	99	50050310						NULL	NULL	AB	Y	NULL	N
11122	5901123	2020-10-01	631001C				1971-09-12	49	17917	2	99	50050201						NULL	NULL	UC	Y	NULL	N
11122	5311631	2020-10-06	631006C				1975-10-12	44	16401	1	99	50050402						NULL	NULL	UC	Y	NULL	N
11122	5503541	2020-10-06	631006C				1965-01-22	55	20346	1	99	50051003						NULL	NULL	AB	Y	NULL	N

1.3 ชุดคำสั่งสำหรับดึงและแปลงข้อมูลของโรงพยาบาลที่ใช้ HospitalOS

```

select distinct
  b_site.b_visit_office_id as hcode ,
  t_patient.patient_hn as hn
  ,substr(t_visit.visit_begin_visit_time,1,4)::int -
543||substr(t_visit.visit_begin_visit_time,6,2)||substr(t_visit.visit_begin_visit_time,9,2) as
dateopd
  ,t_patient.patient_firstname as fname
  ,t_patient.patient_lastname as sname
  ,t_patient.patient_pid as cid
  ,substr(t_patient.patient_birthday,1,4)::int -
543||substr(t_patient.patient_birthday,6,2)||substr(t_patient.patient_birthday,9,2) as dob

```

```

-- ,to_number(to_char(current_date,'YYYY'),'9999')-
(to_number(substr(t_patient.patient_birthday,1,4),'9999')-543 ) as age
/* ,(case when (substr(' ' || age(to_date(t_patient.patient_birthday,'YYYY-MM-DD')) -
interval '543 year') from '(...)year') IS NOT NULL)
    then substr(' ' || age(to_date(t_patient.patient_birthday,'YYYY-MM-DD')) -
interval '543 year') from '(...)year') || ' ปี '
    else "
end
||
case when (substr(' ' || age(to_date(t_patient.patient_birthday,'YYYY-MM-DD')) -
interval '543 year') from '(...)mon') IS NOT NULL)
    then substr(' ' || age(to_date(t_patient.patient_birthday,'YYYY-MM-DD')) - interval
'543 year') from '(...)mon') || ' เดือน '     else "
end
||
case when (substr(' ' || age(to_date(t_patient.patient_birthday,'YYYY-MM-DD')) -
interval '543 year') from '(...)day') IS NOT NULL)
    then substr(' ' || age(to_date(t_patient.patient_birthday,'YYYY-MM-DD')) -
interval '543 year') from '(...)day') || ' วัน '
    else "
end) as age */
,t_patient.f_sex_id as sex
,t_patient.patient_tambon as area
,f_patient_nation.r_rp1853_nation_id as nation
,t_diag_icd10.diag_icd10_number as ICD10
," as referin
,t_visit_refer_in_out.visit_refer_in_out_refer_hospital as referout
,r_rp1853_instype.maininscl as insure
,substr(t_patient_appointment.patient_appointment_date,1,4)::int -
543||substr(t_patient_appointment.patient_appointment_date,6,2)||substr(t_patient_ap
pointment.patient_appointment_date,9,2) as appoint
,t_visit.f_visit_opd_discharge_status_id as typeout

```

```

," as er
,t_visit.f_emergency_status_id as triage
from t_visit
  inner join t_diag_icd10 on t_visit.t_visit_id = t_diag_icd10.diag_icd10_vn and
t_diag_icd10.f_diag_icd10_type_id = '1'
  left join t_visit_refer_in_out on t_visit.t_visit_id = t_visit_refer_in_out.t_visit_id
  inner join t_patient on t_visit.t_patient_id = t_patient.t_patient_id
  inner join t_visit_payment on t_visit.t_visit_id = t_visit_payment.t_visit_id and
t_visit_payment.visit_payment_active = '1'
  inner join b_contract_plans on t_visit_payment.b_contract_plans_id =
b_contract_plans.b_contract_plans_id
  left join t_patient_appointment on t_visit.t_visit_id =
t_patient_appointment.t_visit_id
  inner join f_patient_nation on t_patient.f_patient_nation_id =
f_patient_nation.f_patient_nation_id
  inner join b_map_rp1853_instype on b_contract_plans.b_contract_plans_id =
b_map_rp1853_instype.b_contract_plans_id
  inner join r_rp1853_instype ON b_map_rp1853_instype.r_rp1853_instype_id =
r_rp1853_instype.id
                                -- and r_rp1853_instype.maininscl in
('UCS','OFC','SSS','LGO','SSI')
  cross join b_site
where t_visit.f_visit_status_id <> '4'
  and t_visit.visit_begin_visit_time like '2563%'
  and t_visit.f_visit_type_id = '0'

```

ผลลัพธ์จากการประมวลผลของ HospitalOS

HCODE	HN	DATEOPD	VN	FNAME	SNAME	CID	DOB	AGE	AGEDAY	SEX	NATION	AREA	DIAG1	DIAG2	DIAG3	DIAG4	DIAG5	REFERN	REFEROUT	INSURE	APPOINT	TYPEOUT	ER
11122	5300229	2020-10-01	631001C				1966-09-25	54	19720	2	99	50051204						NULL	NULL	AJ	Y	NULL	N
11122	5303359	2020-10-01	631001C				1963-10-22	56	20799	2	99	50050102						NULL	NULL	AJ	Y	NULL	N
11122	5605130	2020-10-01	631001C				1969-03-13	51	18830	1	99	50050102						NULL	NULL	AB	Y	NULL	N
11122	5605140	2020-10-01	631001C				1974-07-18	46	16877	2	99	50050102						NULL	NULL	AB	Y	NULL	N
11122	5704852	2020-10-01	631001C				1971-07-30	49	17961	1	99	50051302						NULL	NULL	UC	Y	NULL	N
11122	5709831	2020-10-01	631001C				1960-12-26	59	21629	1	99	50050208						NULL	NULL	AB	Y	NULL	N
11122	5712792	2020-10-01	631001C				1961-02-10	59	21702	1	99	50051002						NULL	NULL	AB	Y	NULL	N
11122	5821759	2020-10-01	631001C				1965-09-19	55	20101	2	99	50230408						NULL	NULL	UC	Y	NULL	N
11122	5300781	2020-10-01	631001C				1955-07-29	65	23806	2	99	50051207						NULL	NULL	AG	Y	NULL	N
11122	5300780	2020-10-01	631001C				1961-01-25	59	21799	2	99	50050902						NULL	NULL	UC	Y	NULL	N
11122	5303371	2020-10-01	631001C				1950-02-24	70	23707	1	99	50050704						NULL	NULL	AG	Y	NULL	N
11122	5303544	2020-10-01	631001C				1971-06-27	49	17994	2	99	50050802						NULL	NULL	AJ	Y	NULL	N
11122	5306823	2020-10-01	631001C				1968-10-06	51	18988	2	99	50051302						NULL	NULL	AB	Y	NULL	N
11122	5606006	2020-10-01	631001C				1972-05-03	48	17350	1	99	50050802						NULL	NULL	AJ	Y	NULL	N
11122	5607792	2020-10-01	631001C				1971-01-23	49	18149	2	99	50050803						NULL	NULL	UC	Y	NULL	N
11122	5702621	2020-10-01	631001C				1968-05-29	52	19118	1	99	50051203						NULL	NULL	UC	Y	NULL	N
11122	5705413	2020-10-01	631001C				1979-07-02	41	15067	1	99	50050901						NULL	NULL	UC	Y	NULL	N
11122	5705443	2020-10-01	631001C				1964-09-30	56	20455	1	99	50050504						NULL	NULL	AE	Y	NULL	N
11122	5710753	2020-10-01	631001C				1971-09-30	49	17899	2	99	50050107						NULL	NULL	UC	Y	NULL	N
11122	5710981	2020-10-01	631001C				1983-01-16	37	13773	1	99	50050207						NULL	NULL	UC	Y	NULL	N
11122	5802232	2020-10-01	631001C				1971-12-08	48	17336	2	99	50051804						NULL	NULL	UC	Y	NULL	N
11122	5805704	2020-10-01	631001C				1975-03-10	45	16642	2	99	57110302						NULL	NULL	UC	Y	NULL	N
11122	5817580	2020-10-01	631001C				1962-02-09	58	21419	1	99	50110507						NULL	NULL	AB	Y	NULL	N
11122	5801125	2020-10-01	631001C				1970-01-26	50	18511	2	99	50050310						NULL	NULL	AB	Y	NULL	N
11122	5801126	2020-10-01	631001C				1971-09-12	49	17917	2	99	50050201						NULL	NULL	UC	Y	NULL	N
11122	5311631	2020-10-06	631006C				1975-10-12	44	16431	1	99	50050402						NULL	NULL	UC	Y	NULL	N
11122	5303541	2020-10-06	631006C				1965-01-22	55	20346	1	99	50051003						NULL	NULL	AB	Y	NULL	N

1.4 ชุดคำสั่งสำหรับดึงและแปลงข้อมูลของโรงพยาบาลส่งเสริมสุขภาพตำบลที่ใช้ JHCIS

Select

```

visit.pcucode as HCODE,
person.pid as HN,
visit.visitdate as DATEOPD,
visit.visitno as VN,
person.fname as FNAME,
person.lname as SNAME,
person.idcard as CID,
person.birth as DOB,
timestampdiff(YEAR,person.birth,visit.visitdate) as AGE,
0 as AGEDAY,
person.sex as SEX,
'' as AREA,
person.nation as NATION,
if(visitdiag.d xtype=1,replace(visitdiag.diagcode,',',''),NULL) as DIAG1,
if(visitdiag.d xtype=4,replace(visitdiag.diagcode,',',''),NULL) as DIAG2,
'' as DIAG3,
'' as DIAG4,
'' as DIAG5,
'' REFERIN,
'' as REFEROUT,
'' as INSURE,
'N' as APPOINT,
'O' as TYPEOUT,
'N' as ER,
0 as TRIAGE

```

FROM

```
visit
```

```
INNER JOIN visitdiag ON visit.visitno = visitdiag.visitno AND visitdiag.pcucode = visit.pcucode
```

INNER JOIN person ON visit.pid = person.pid

WHERE visit.visitdate BETWEEN subdate(now(),interval 10 day) AND now()

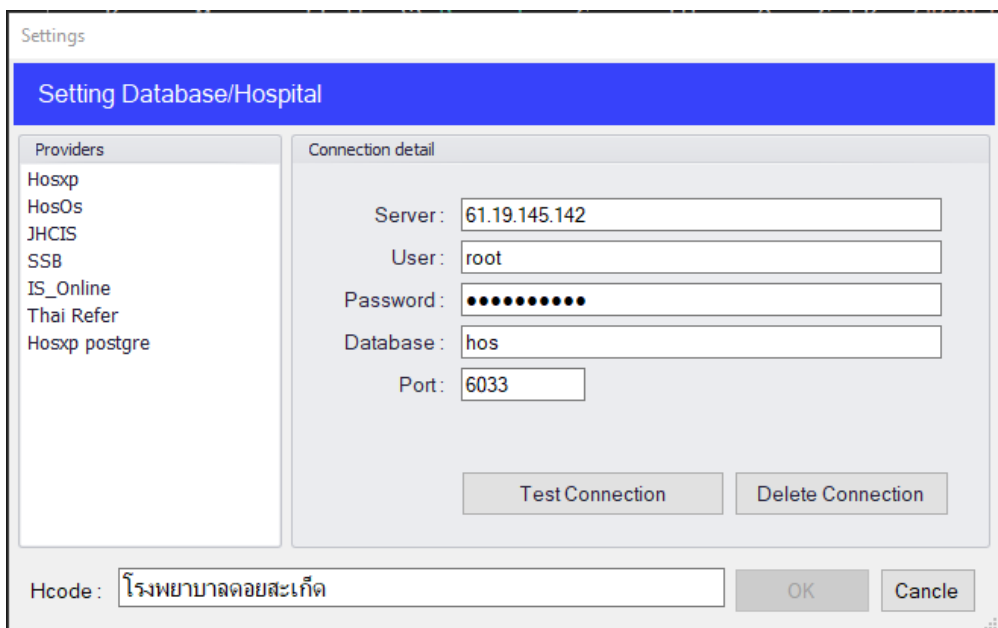
ผลลัพธ์จากการประมวลผลของ JHCIS

HCODE	hr	DATEPDI	vn	FNAME	SNAME	cid	DOB	AGE	AGEDAY	sex	NATION	AREA	DIAG1	DIAG2	DIAG3	DIAG4	DIAG5	REFERRIN	REFROUT	INSURE	APPOINT	TYPEOUT	ER
11122	530223	2020-10-01	631001C				1966-09-25	54	19730	2	99	50051358						NULL	NULL	AJ	Y	NULL	N
11122	530339	2020-10-01	631001C				1963-10-22	56	20799	2	99	50050102						NULL	NULL	AJ	Y	NULL	N
11122	5605199	2020-10-01	631001C				1969-03-13	51	18830	1	99	50050102						NULL	NULL	AB	Y	NULL	N
11122	5605140	2020-10-01	631001C				1974-07-18	46	16877	2	99	50050102						NULL	NULL	AB	Y	NULL	N
11122	570462	2020-10-01	631001C				1971-07-30	49	17661	1	99	50051302						NULL	NULL	UC	Y	NULL	N
11122	5709831	2020-10-01	631001C				1960-12-26	59	21829	1	99	50050208						NULL	NULL	AB	Y	NULL	N
11122	5710790	2020-10-01	631001C				1981-02-10	39	21783	1	99	50051002						NULL	NULL	AB	Y	NULL	N
11122	5821799	2020-10-01	631001C				1965-09-19	55	20101	2	99	50230408						NULL	NULL	UC	Y	NULL	N
11122	5900785	2020-10-01	631001C				1965-07-29	55	23062	2	99	50051007						NULL	NULL	UC	Y	NULL	N
11122	5900789	2020-10-01	631001C				1981-01-25	39	21799	2	99	50050902						NULL	NULL	UC	Y	NULL	N
11122	5903371	2020-10-01	631001C				1950-02-24	70	25787	1	99	50050704						NULL	NULL	AG	Y	NULL	N
11122	5903544	2020-10-01	631001C				1971-06-27	49	17994	2	99	50050802						NULL	NULL	AJ	Y	NULL	N
11122	5906823	2020-10-01	631001C				1966-10-06	51	18666	2	99	50051302						NULL	NULL	AB	Y	NULL	N
11122	5906866	2020-10-01	631001C				1972-08-05	48	17589	1	99	50050802						NULL	NULL	AJ	Y	NULL	N
11122	5607792	2020-10-01	631001C				1971-01-23	49	18149	2	99	50050803						NULL	NULL	UC	Y	NULL	N
11122	5702621	2020-10-01	631001C				1968-05-29	52	19118	1	99	50051203						NULL	NULL	UC	Y	NULL	N
11122	5702415	2020-10-01	631001C				1979-07-02	41	15267	1	99	50050901						NULL	NULL	UC	Y	NULL	N
11122	5704416	2020-10-01	631001C				1964-09-30	56	20453	1	99	50050904						NULL	NULL	AB	Y	NULL	N
11122	5710753	2020-10-01	631001C				1971-09-30	49	17899	2	99	50050107						NULL	NULL	UC	Y	NULL	N
11122	5710981	2020-10-01	631001C				1983-01-16	37	13773	1	99	50050207						NULL	NULL	UC	Y	NULL	N
11122	5802132	2020-10-01	631001C				1971-12-08	48	17830	2	99	50051304						NULL	NULL	UC	Y	NULL	N
11122	5805706	2020-10-01	631001C				1975-03-10	45	16642	2	99	57110302						NULL	NULL	UC	Y	NULL	N

การเชื่อมโยงข้อมูลฐานข้อมูล CMDS

ความต้องการในการใช้ข้อมูลที่หลากหลายและต้องการความถูกต้องของข้อมูลจึงมีความจำเป็นในการเชื่อมโยงข้อมูลจากหลาย ๆ แหล่งเพื่อนำข้อมูลมาเทียบเคียงและตรวจสอบข้อมูล ดังนั้น CMDS จึงพัฒนาให้สามารถเชื่อมโยงข้อมูลได้หลาย ๆ ฐาน ดังนี้

- HosXP เป็นโปรแกรมที่ถูกใช้งานภายในจังหวัดเชียงใหม่ ในสังกัด 18 แห่ง นอกสังกัดรวมเอกชน 2 แห่ง
- HosOS เป็นโปรแกรมที่ถูกใช้งานภายในจังหวัดเชียงใหม่ ในสังกัด 5 แห่ง นอกสังกัดรวมเอกชน 1 แห่ง
- JHCIS เป็นโปรแกรมสำหรับใช้งานใน รพสต. และหน่วยบริการระดับปฐมภูมิ ในจังหวัด 277 แห่ง
- SSB เป็นโปรแกรมที่ใช้งานที่โรงพยาบาลนครพิงค์
- IS_Online เป็นโปรแกรมที่ใช้ในการจัดเก็บข้อมูลอุบัติเหตุ
- Thai Refer เป็นโปรแกรมที่ใช้บันทึกข้อมูลการส่งต่อ
- Hosxp Postgre เป็นโปรแกรมที่ถูกใช้งานภายในจังหวัดเชียงใหม่ 1 แห่ง ที่ รพ.สวนปรุง



ภาพที่ 11 การตั้งค่าการเชื่อมต่อกับฐานข้อมูลของ CMDS

2. ระบบบริหารจัดการรายงานด้านการแพทย์ฉุกเฉินและสาธารณสุข (CM dashboard)

การดึงข้อมูลจากหน่วยบริการสาธารณสุขมารวมกันในฐานข้อมูลเดียวกันทำให้ปริมาณข้อมูลมีจำนวนมากขึ้นเรื่อย ๆ จึงทำให้เกิดปัญหาในการแสดงผลข้อมูลล่าช้า ดังนั้น จึงต้องมีวิธีการประมวลผลข้อมูลเป็นรอบ ๆ โดยเก็บข้อมูลผลรวมนั้นไว้ในแฟ้มพักข้อมูล และนำเสนอข้อมูลผ่าน visualization ซึ่งเป็นวิธีการสนับสนุนขอบเขตของข้อมูลหรือการวิเคราะห์ข้อมูลที่สนใจโดยผ่านรูปแบบของการค้นหาข้อมูลด้วยวิธีการของ Visual ซึ่งได้นำเอาทฤษฎีในการดึงคุณสมบัติของโครงสร้างที่สำคัญของข้อมูลมาร่างร่วมกับคุณสมบัติ Visual เช่น การแสดงภาพ กราฟต่าง ๆ ซึ่งจุดมุ่งหมายก็คือการจัดการหรือการแสดงผลข้อมูลในปริมาณที่สูงให้สามารถแสดงผลและสร้างความเข้าใจให้แก่ผู้ที่อ่านค่าได้

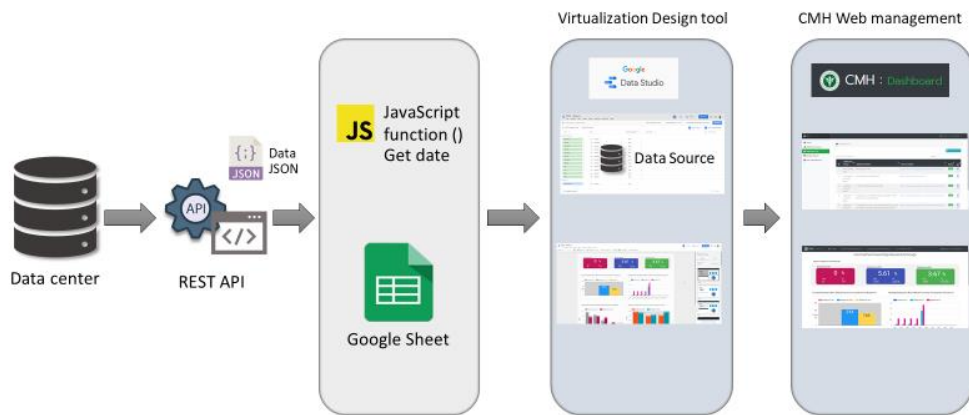
ประโยชน์ของ visualization มีดังนี้

- ช่วยในการแสดงข้อมูลที่มีปริมาณสูงให้เป็นรูปแบบที่เข้าใจง่าย
- ช่วยในการรับรู้หรือคาดคะเนสิ่งที่อาจเกิดขึ้นในอนาคต
- visualization ไม่เพียงแต่แสดงรายละเอียดข้อมูลในตัวเองเท่านั้นยังสามารถแสดงปัญหาที่เกิดขึ้นได้ด้วย
- ช่วยให้ผู้ใช้สามารถเข้าใจหรือวิเคราะห์ข้อมูลได้สะดวกขึ้นไม่ว่าขอบเขตข้อมูลนั้นจะมีขนาดใหญ่หรือเล็ก

ช่วยให้ผู้ใช้สามารถสันนิษฐานข้อมูลได้สะดวกขึ้น ซึ่ง Visualization เป็นมากกว่า วิธีการทาง Computer Visualization เป็นการนำเสนอข้อมูลแบบหนึ่งที่ทำให้การแสดงผลข้อมูลให้อยู่ในรูปแบบของ Visual Form ซึ่งอาจจะเป็นการแสดงในรูปแบบของรูปภาพ กราฟ หรือ แผนภาพ ซึ่งผลลัพธ์ของการทำ Visualization คือ การนำข้อมูลต่าง ๆ ที่ซ่อนอยู่ในตัวของข้อมูลเองออกมาให้ผู้ใช้สามารถสัมผัสได้ ซึ่งกระบวนการต่าง ๆ ในการแสดงหรือนำข้อมูลต่าง ๆ ออกมา อาจจะถูกแอบซ่อนอยู่หรือผู้ใช้ไม่ทันสังเกต แต่อย่างไรก็ตาม Visualization เป็นสิ่งที่จำเป็นในการค้นหาข้อมูล หรือ ใช้ในการวิเคราะห์ข้อมูล โดยจุดมุ่งหมายของ Visualization ก็คือ การถ่ายทอดข้อมูลไปสู่ระบบการรับรู้โดยภาพของผู้ใช้ระบบเพื่อช่วยในการลดช่องว่างระหว่างผู้ใช้และตัวข้อมูล และยังช่วยผู้ใช้สามารถเรียนรู้ ได้อย่างมีประสิทธิภาพ

Data Studio เป็นโซลูชันการรายงานของ Google สำหรับผู้ใช้ที่ต้องการก้าวไปไกลกว่าข้อมูล และแดชบอร์ดของ Google Analytics วิดเจ็ตข้อมูลใน Data Studio มีความโดดเด่นในด้านความหลากหลายตัวเลือกการปรับแต่งข้อมูลและการควบคุมแบบโต้ตอบ (เช่นการเรียงคอลัมน์และการแบ่งหน้าต่าง) แหล่งข้อมูล ได้แก่ ผลิตภัณฑ์ Google (Analytics, AdWords, Search Console, Sheet, MySQL, YouTube ฯลฯ) ตัวเชื่อมต่อฐานข้อมูลและการอัปโหลดไฟล์

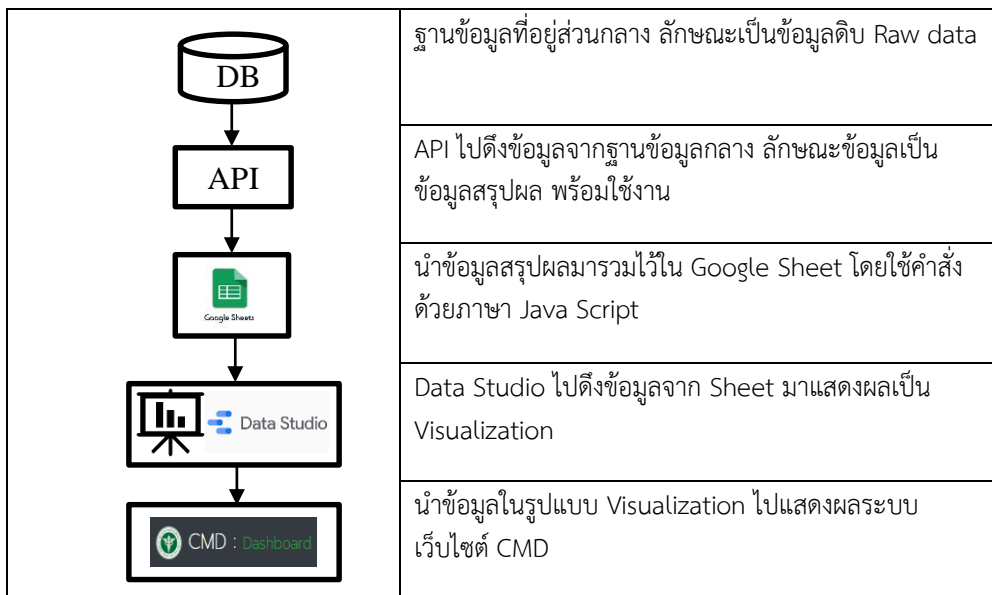
How Dashboard created



ภาพที่ 12 ฟังไหลเวียนการทำงานของ Dashboard

2.1 การสร้างการแสดงผล Dashboard หรือรายงาน ด้วย Google Data Studio

ในการนี้การเชื่อมต่อข้อมูลมาแสดงผล จะใช้วิธีการให้ Google Sheet รับค่าจาก API datacenter ซึ่ง เป็นแหล่งเก็บข้อมูล สำหรับสร้าง Dashboard เพื่อแสดงผลข้อมูลที่เกี่ยวข้องกับการดูแลภาวะฉุกเฉิน ด้านการแพทย์และสาธารณสุขในจังหวัดเชียงใหม่ มีหลักการดังนี้



ข้อมูลดิบ (Raw data) จาก API datacenter

คือ ฐานข้อมูลที่ดึงมาจาก API ตามกรอบโครงสร้างฐานข้อมูลและชุดข้อมูล (Data set) ที่เชื่อมต่อบระบบ HIS ของหน่วยบริการ เช่น ตาราง t_data_dx

HCODE	HN	DATEOPD	VN	FNAME	SNAME	CID	DOB	AGE	AGEDAY	SEX	AREA	NATION	DIAG1	DIAG2
10713	47	2020-06-04	0001			35	1943-09-05	77	28033	1	501401	99	I259	E785
10713	52	2020-04-26	0001			35	1981-03-17	39	14286	1	500711	99	H103	(Null)
10713	53	2020-05-01	0001			35	1940-10-01	80	29068	2	501705	99	I500	J960
10713	55	2020-07-20	0001			35	1951-01-01	70	25404	2	501412	99	I219	(Null)
10713	56	2019-10-22	0001			15	1991-03-24	29	10516	2	500407	99	J029	(Null)
10713	56	2020-01-23	0001			35	1965-01-01	55	20111	1	500502	99	I255	Z019
10713	56	2020-10-15	0001			15	1998-05-24	22	8181	2	500608	99	J46	J46
10713	57	2019-11-11	0001			15	2014-02-20	6	2147	1	570410	99	J00	(Null)
10713	57	2019-11-23	0001			33	1937-01-01	83	30322	2	500110	99	I500	(Null)
10713	58	2019-11-28	0001			34	1962-03-31	58	21103	2	500110	99	I420	I10
10713	58	2020-03-12	0001			15	2015-03-02	5	1838	2	500603	99	J039	(Null)
10713	58	2019-11-26	0001			12	1980-12-12	40	100	1	99	99	I10	I12
10713	58	2020-05-28	0001			15	1998-07-29	22	7975	2	500407	99	H101	Z133
10713	59	2019-12-13	0001			35	1934-11-14	85	31107	1	501308	99	I469	(Null)
10713	60	2020-05-06	0001			35	1948-06-09	72	26265	1	500705	99	J441	(Null)
10713	61	2020-01-11	0001			05	2018-06-24	2	570	1	501003	98	J189	J969
10713	61	2020-08-12	0001			15	2005-01-29	16	5675	1	5403	99	J209	J209
10713	62	2020-05-03	0001			65	1954-06-02	66	24078	1	501001	98	J189	I269
10713	62	2019-10-09	0001			15	2001-01-22	19	6923	1	501005	99	J101	(Null)
10713	62	2019-10-15	0001			35	1956-06-15	64	23216	1	580305	99	I219	(Null)

DIAG1	DIAG2	DIAG3	DIAG4	DIAG5	REFERIN	REFEROUT	INSURE	APPOINT	TYPEOUT	ER	TRIAGE	DIAG6	DIAG7	DIAG8	DIAG9	DIAG10	DATA_D
I259	E785	I10	Z019	(Null)	(Null)	(Null)	A2	N	1	N	(Null)	(Null)	(Null)	(Null)	(Null)	(Null)	(Null)
H103	(Null)	(Null)	(Null)	(Null)	(Null)	(Null)	UC	N	1	Y	3	(Null)	(Null)	(Null)	(Null)	(Null)	(Null)
I500	J960	E876	J189	(Null)	11133	(Null)	UC	N	2	Y	1	(Null)	(Null)	(Null)	(Null)	(Null)	(Null)
I219	(Null)	(Null)	(Null)	(Null)	(Null)	(Null)	A1	N	2	Y	2	(Null)	(Null)	(Null)	(Null)	(Null)	(Null)
J029	(Null)	(Null)	(Null)	(Null)	(Null)	(Null)	A7	N	1	N	(Null)	(Null)	(Null)	(Null)	(Null)	(Null)	(Null)
I255	Z019	(Null)	(Null)	(Null)	11122	(Null)	UC	N	1	N	(Null)	(Null)	(Null)	(Null)	(Null)	(Null)	(Null)
J46	J46	(Null)	(Null)	(Null)	(Null)	(Null)	A7	N	1	Y	2	(Null)	(Null)	(Null)	(Null)	(Null)	(Null)
J00	(Null)	(Null)	(Null)	(Null)	(Null)	(Null)	UC	N	1	Y	3	(Null)	(Null)	(Null)	(Null)	(Null)	(Null)
I500	(Null)	(Null)	(Null)	(Null)	(Null)	(Null)	UC	N	2	Y	2	(Null)	(Null)	(Null)	(Null)	(Null)	(Null)
I420	I10	Z133	Z716	(Null)	(Null)	(Null)	A2	N	1	N	(Null)	(Null)	(Null)	(Null)	(Null)	(Null)	(Null)
J039	(Null)	(Null)	(Null)	(Null)	(Null)	(Null)	UC	N	2	N	(Null)	(Null)	(Null)	(Null)	(Null)	(Null)	(Null)
I10	I12	I13	I14	I15	2019-11-20	2019-11-26	A2	Y	1	Y	2	(Null)	(Null)	(Null)	(Null)	(Null)	(Null)
H101	Z133	(Null)	(Null)	(Null)	(Null)	(Null)	UC	N	1	N	(Null)	(Null)	(Null)	(Null)	(Null)	(Null)	(Null)
I469	(Null)	(Null)	(Null)	(Null)	(Null)	(Null)	UC	N	4	Y	1	(Null)	(Null)	(Null)	(Null)	(Null)	(Null)
J441	(Null)	(Null)	(Null)	(Null)	(Null)	(Null)	UC	N	2	Y	2	(Null)	(Null)	(Null)	(Null)	(Null)	(Null)
J189	J969	(Null)	(Null)	(Null)	(Null)	(Null)	AH	N	2	Y	1	(Null)	(Null)	(Null)	(Null)	(Null)	(Null)
J209	J209	(Null)	(Null)	(Null)	(Null)	(Null)	UC	N	1	Y	(Null)	(Null)	(Null)	(Null)	(Null)	(Null)	(Null)
J189	I269	(Null)	(Null)	(Null)	11126	(Null)	AH	N	2	Y	2	(Null)	(Null)	(Null)	(Null)	(Null)	(Null)

ภาพที่ 13 ตัวอย่างตารางฐานข้อมูล ชื่อตาราง t_data_dx

โดย API ไปดึงข้อมูลจากฐานข้อมูลกลาง (API datacenter) ประมวลผลเป็นข้อมูลลักษณะข้อมูลสรุปผล ซึ่งได้กำหนดให้ script ทำงานทุกวันตามเวลาที่กำหนดเพื่อนำไปจัดทำ visualization ด้วย Google data studio ต่อไป

```

[
  - {
    ampurcode: "07",
    ampurname: "แม่ริม",
    hoscode: "10713",
    hosname: "โรงพยาบาลนครพิงค์",
    sum: "8",
    death: "1",
    red: "2",
    pink: "0",
    yellow: "0",
    green: "0",
    white: "6",
    red_death: "1",
    pink_death: "0",
    yellow_death: "0",
    green_death: "0",
    white_death: "0"
  },
  - {
    ampurcode: "07",
    ampurname: "โชคชัย",
    hoscode: "10876",
    hosname: "โรงพยาบาลโชคชัย",
    sum: "383",
    death: "0",
    red: "5",
    pink: "3",
    yellow: "171",
    green: "75",
    white: "137",
    red_death: "0",
    pink_death: "0",
    yellow_death: "0",
    green_death: "0",
    white_death: "0"
  },
],

```

ภาพที่ 14 ตัวอย่างข้อมูลสรุปผลที่ API ดึงข้อมูลไปแสดง

การใช้ Google Sheet มาเก็บข้อมูลก่อนนำไปแสดงผลต่อไปด้วย Google Data Studio โดยจะต้องใช้คำสั่งภาษา JAVA Script เพื่อรับข้อมูลจาก API

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	Ampurcode	Ampurname	Hoscode	Hosname	Sum	Death	Red	Pink	Yellow	Green	White	Red Death	Pink Death
2	07	แม่ริม	10713	โรงพยาบาลนครพิงค์ 8	8	1	2	0	0	0	6	1	0
3	07	โชคชัย	10876	โรงพยาบาลโชคชัย 383	383	0	5	3	171	75	137	0	0
4	02	จอมทอง	11119	โรงพยาบาลจอมทอง 17809	17809	0	64	318	7711	1941	7776	0	0
5	03	แม่แจ่ม	11120	โรงพยาบาลแม่แจ่ม 3987	3987	0	17	47	621	1800	1502	0	0
6	04	เขียงเตา	11121	โรงพยาบาลเขียงเตา 6517	6517	0	14	35	216	1436	4816	0	0
7	05	คลองสะเภา	11122	โรงพยาบาลคลองสะเภา 8098	8098	0	24	92	1008	1775	5199	0	0
8	06	แม่แตง	11123	โรงพยาบาลแม่แตง 5896	5896	0	30	43	2224	2031	1568	0	0
9	08	สะเมิง	11124	โรงพยาบาลสะเมิง 2136	2136	0	3	11	111	683	1328	0	0
10	09	พร้าว	11125	โรงพยาบาลพร้าว 17239	17239	0	145	938	10900	1228	3752	0	0
11	10	แม่ฮาด	11126	โรงพยาบาลแม่ฮาด 8192	8192	0	56	136	1717	3467	2816	0	0
12	11	พร้าว	11127	โรงพยาบาลพร้าว 4903	4903	0	3	15	259	804	3822	0	0
13	12	สันป่าตอง	11128	โรงพยาบาลสันป่าตอง 17107	17107	0	115	341	4235	6835	5581	0	0
14	13	สันกำแพง	11129	โรงพยาบาลสันกำแพง 10037	10037	0	52	131	1938	2548	5374	0	0
15	15	หางดง	11131	โรงพยาบาลหางดง 16879	16879	0	104	516	6657	3856	6480	0	0
16	16	ฮอด	11132	โรงพยาบาลฮอด 4251	4251	0	25	16	384	1234	2592	0	0
17	17	คลองเตา	11133	โรงพยาบาลคลองเตา 22174	22174	0	16	37	528	1753	19840	0	0
18	18	อมก๋อย	11134	โรงพยาบาลอมก๋อย 11008	11008	0	17	119	2003	1872	7014	0	0
19	19	สารภี	11135	โรงพยาบาลสารภี 8712	8712	0	21	86	2934	585	5086	0	0
20	21	ไชยปราการ	11137	โรงพยาบาลไชยปราการ 4641	4641	0	41	57	1762	2071	710	0	0
21	22	แม่วาง	11138	โรงพยาบาลแม่วาง 4221	4221	0	8	29	258	586	3340	0	0

ภาพที่ 15 ข้อมูลที่ถูกดึงมาเก็บไว้ใน Google Sheet

คำสั่งภาษา JAVA Script ที่ใช้ดึงข้อมูลจาก API เข้าสู่ Google Sheet โดยนำไปวางในส่วนของ Script Editor ของ Google Sheet

```
function readRows() {
  var sheet = SpreadsheetApp.getActiveSheet();
  var rows = sheet.getDataRange();
  var numRows = rows.getNumRows();
  var values = rows.getValues();

  for (var i = 0; i <= numRows - 1; i++) {
    var row = values[i];
    Logger.log(row);
  }
};

function onOpen() {
  var sheet = SpreadsheetApp.getActiveSpreadsheet();
  var entries = [{
    name : "Read Data",
    functionName : "readRows"
  }];
  sheet.addMenu("Script Center Menu", entries);
};

function ImportJSON(url, query, options) {
  return ImportJSONAdvanced(url, query, options, includeXPath_,
  defaultTransform_);
}

function ImportJSONAdvanced(url, query, options, includeFunc, transformFunc) {
  var jsondata = UrlFetchApp.fetch(url);
  var object = JSON.parse(jsondata.getContentText());
}
```

```

return parseJSONObject_(object, query, options, includeFunc, transformFunc);
}

function URLEncode(value) {
return encodeURIComponent(value.toString());
}

function parseJSONObject_(object, query, options, includeFunc, transformFunc) {
var headers = new Array();
var data = new Array();

if (query && !Array.isArray(query) && query.toString().indexOf(",") != -1) {
query = query.toString().split(",");
}

if (options) {
options = options.toString().split(",");
}

parseData_(headers, data, "", 1, object, query, options, includeFunc);
parseHeaders_(headers, data);
transformData_(data, options, transformFunc);

return hasOption_(options, "noHeaders") ? (data.length > 1 ? data.slice(1) : new
Array()) : data;
}

function parseData_(headers, data, path, rowIndex, value, query, options,
includeFunc) {
var dataInserted = false;

if (isObject_(value)) {

```

```

for (key in value) {
  if (parseData_(headers, data, path + "/" + key, rowIndex, value[key], query, options,
  includeFunc)) {
    dataInserted = true;
  }
}
} else if (Array.isArray(value) && isObjectArray_(value)) {
  for (var i = 0; i < value.length; i++) {
    if (parseData_(headers, data, path, rowIndex, value[i], query, options, includeFunc))
    {
      dataInserted = true;
      rowIndex++;
    }
  }
} else if (!includeFunc || includeFunc(query, path, options)) {
  // Handle arrays containing only scalar values
  if (Array.isArray(value)) {
    value = value.join();
  }

  if (!data[rowIndex]) {
    data[rowIndex] = new Array();
  }

  if (!headers[path] && headers[path] != 0) {
    headers[path] = Object.keys(headers).length;
  }

  data[rowIndex][headers[path]] = value;
  dataInserted = true;
}

```

```
return dataInserted;
}

function parseHeaders_(headers, data) {
  data[0] = new Array();

  for (key in headers) {
    data[0][headers[key]] = key;
  }
}

function transformData_(data, options, transformFunc) {
  for (var i = 0; i < data.length; i++) {
    for (var j = 0; j < data[i].length; j++) {
      transformFunc(data, i, j, options);
    }
  }
}

function isObject_(test) {
  return Object.prototype.toString.call(test) === '[object Object]';
}

function isObjectArray_(test) {
  for (var i = 0; i < test.length; i++) {
    if (isObject_(test[i])) {
      return true;
    }
  }

  return false;
}
```

```

function includeXPath_(query, path, options) {
  if (!query) {
    return true;
  } else if (Array.isArray(query)) {
    for (var i = 0; i < query.length; i++) {
      if (applyXPathRule_(query[i], path, options)) {
        return true;
      }
    }
  } else {
    return applyXPathRule_(query, path, options);
  }

  return false;
};

function applyXPathRule_(rule, path, options) {
  return path.indexOf(rule) == 0;
}

function defaultTransform_(data, row, column, options) {
  if (!data[row][column]) {
    if (row < 2 || hasOption_(options, "noInherit")) {
      data[row][column] = "";
    } else {
      data[row][column] = data[row-1][column];
    }
  }
}

if (!hasOption_(options, "rawHeaders") && row == 0) {
  if (column == 0 && data[row].length > 1) {

```

```

removeCommonPrefixes_(data, row);
}

data[row][column] = toTitleCase_(data[row][column].toString().replace(/[\V_]/g, "
"));
}

if (!hasOption_(options, "noTruncate") && data[row][column]) {
data[row][column] = data[row][column].toString().substr(0, 256);
}

if (hasOption_(options, "debugLocation")) {
data[row][column] = "[" + row + "," + column + "]" + data[row][column];
}
}

function removeCommonPrefixes_(data, row) {
var matchIndex = data[row][0].length;

for (var i = 1; i < data[row].length; i++) {
matchIndex = findEqualityEndpoint_(data[row][i-1], data[row][i], matchIndex);

if (matchIndex == 0) {
return;
}
}

for (var i = 0; i < data[row].length; i++) {
data[row][i] = data[row][i].substring(matchIndex, data[row][i].length);
}
}

```



```

function findEqualityEndpoint_(string1, string2, stopAt) {
  if (!string1 || !string2) {
    return -1;
  }

  var maxEndpoint = Math.min(stopAt, string1.length, string2.length);

  for (var i = 0; i < maxEndpoint; i++) {
    if (string1.charAt(i) != string2.charAt(i)) {
      return i;
    }
  }

  return maxEndpoint;
}

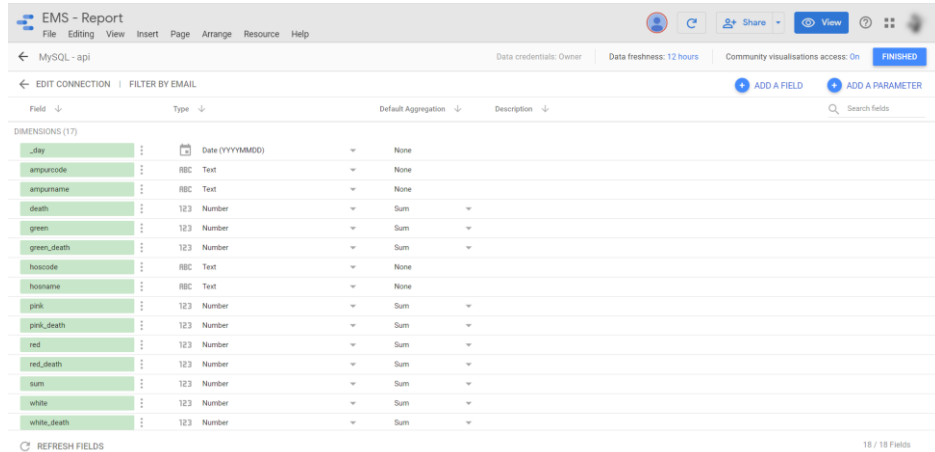
function toTitleCase_(text) {
  if (text == null) {
    return null;
  }

  return text.replace(/\w\S*/g, function(word) { return word.charAt(0).toUpperCase()
+ word.substr(1).toLowerCase(); });
}

function hasOption_(options, option) {
  return options && options.indexOf(option) >= 0;
}

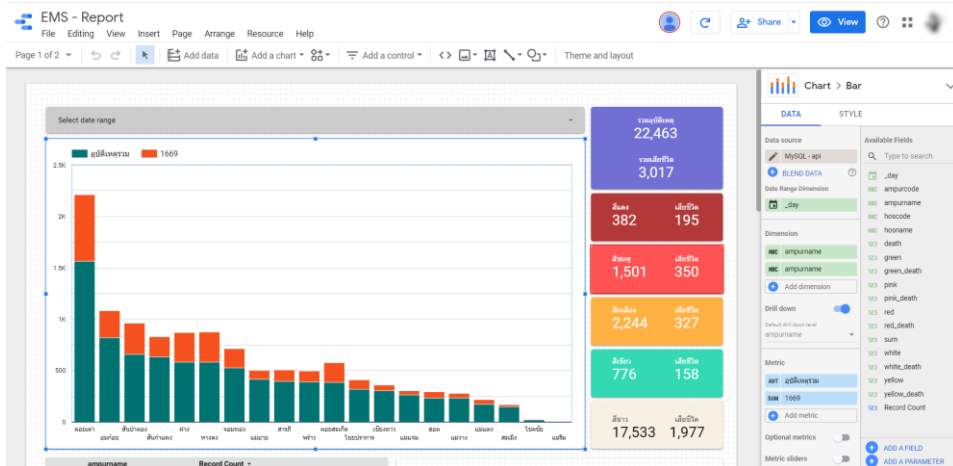
```

การนำ Google Data Studio มาใช้งานสำหรับเก็บข้อมูล
 โดยให้ Google Data Studio เชื่อมโยงกับ Sheet ดึงข้อมูลมาเก็บไว้เป็น Data Source



ภาพที่ 16 แสดงข้อมูล Data Source

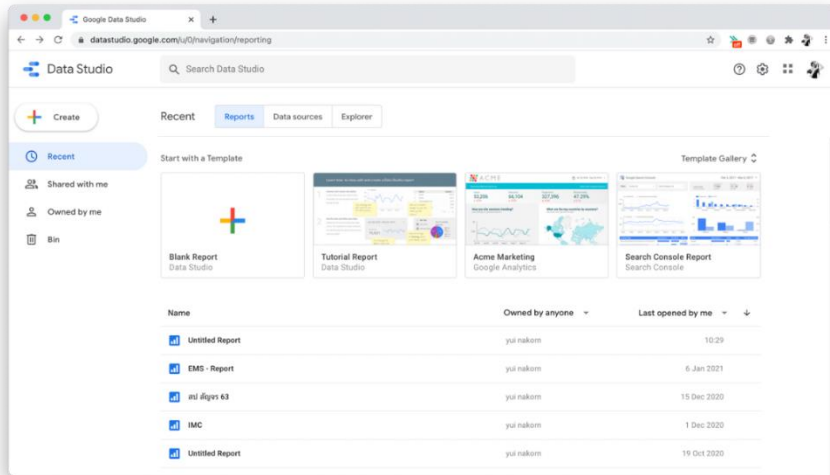
การออกแบบหน้าจอแสดงผล Dashboard ด้วย Google Data Studio โดยสามารถเลือกข้อมูล
 จาก Data Source มาแสดงผลในรูปแบบ กราฟแสดงผล เช่น กราฟแท่ง กราฟวงกลม ตารางข้อมูล และ
 อื่น ๆ



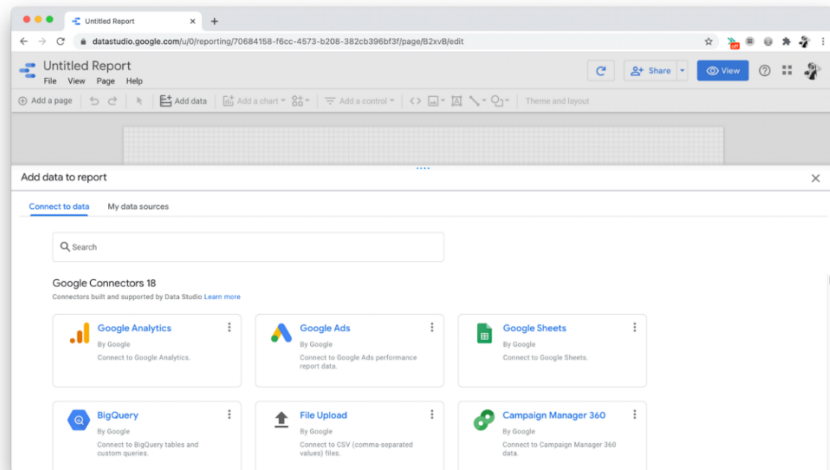
ภาพที่ 17 หน้าจอออกแบบ Dashboard ใน Google Data Studio

2.1.1 ขั้นตอน การใช้งาน Google data studio

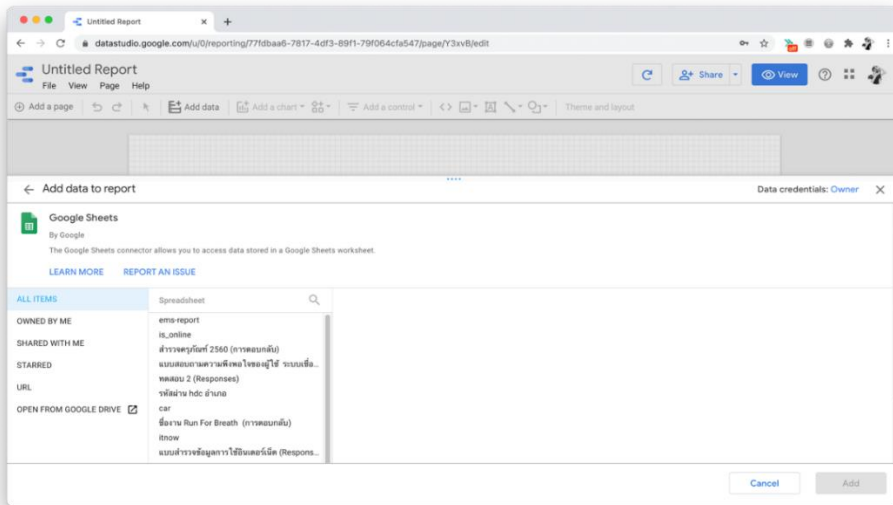
1. สมัครใช้งาน Google Data Studio โดยจะต้องสมัครใช้งานบริการของ Google Service ก่อน
2. เข้าสู่ระบบ Google Data Studio ผ่าน url <https://datastudio.google.com> กดเครื่องหมาย “+” Blank Report เพื่อสร้าง Dashboard ใหม่



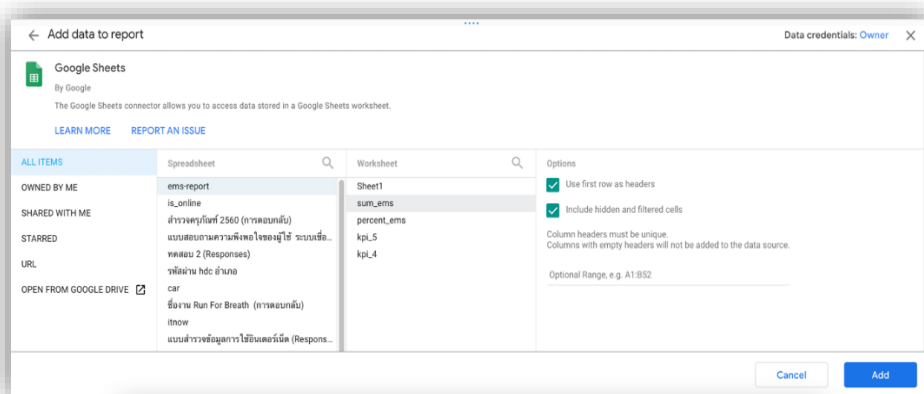
3. จะมีหน้าต่าง Add data to report เลือกเป็น Google Sheets



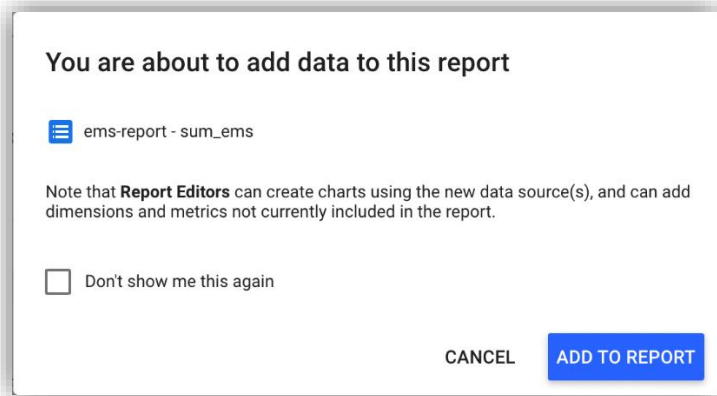
4. จะมีหน้าต่างขึ้นมาให้เลือก data source ในกรณีนี้ เลือก ems-report ซึ่งเป็น data source ที่เราได้จัดทำไว้แล้วล่วงหน้า โดยดึงข้อมูลจาก API



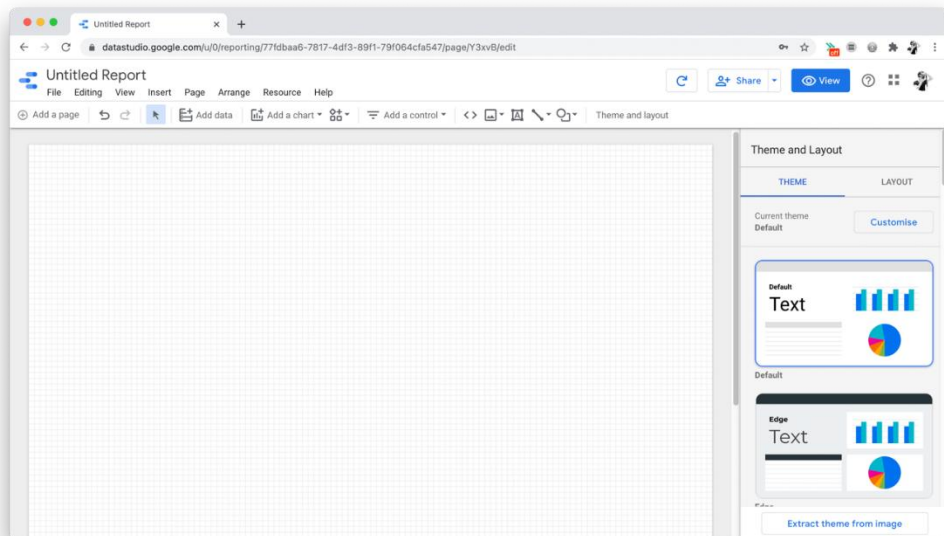
5. เลือก Sheets ที่สร้างไว้ เช่น sum_ems เป็นข้อมูลสรุป แล้วกดปุ่ม “Add”



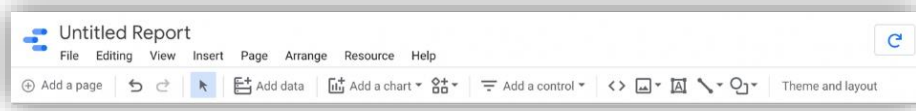
6. กด Add to report



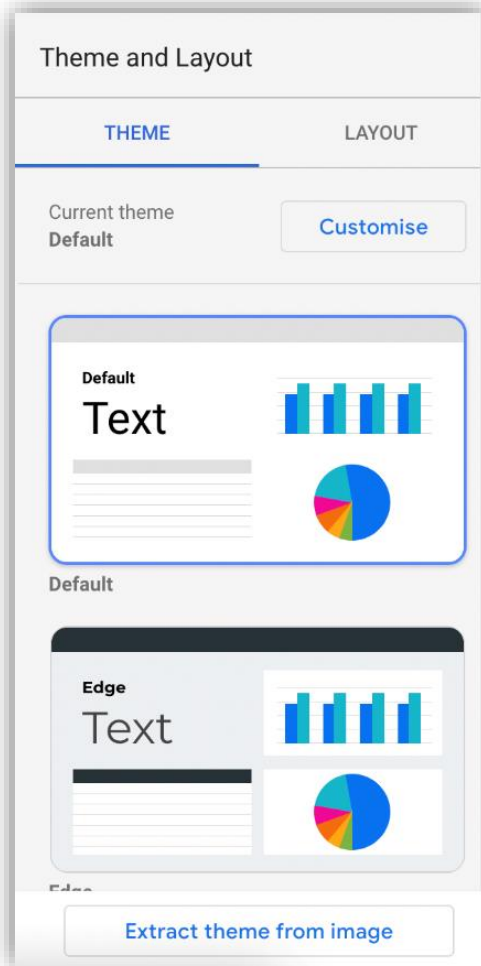
หน้าจอสั่งสร้างรายงาน



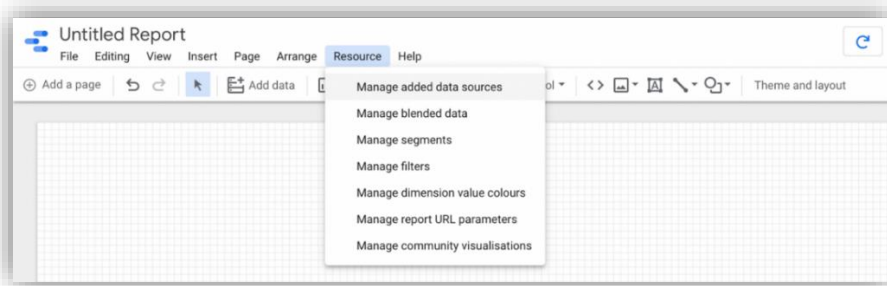
แถบเครื่องมือ เช่น เพิ่มหน้า เพิ่มกราฟ และการแทรกข้อความหรือรูปภาพ เป็นต้น



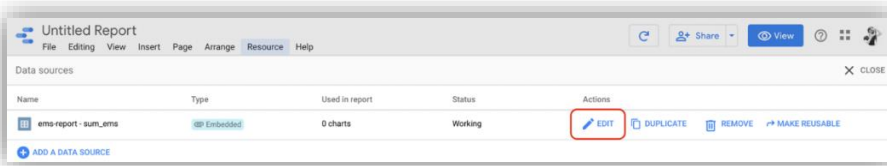
แถบเครื่องมือตกแต่งและกำหนดข้อมูลจะอยู่ด้านขวามือของหน้าจอ



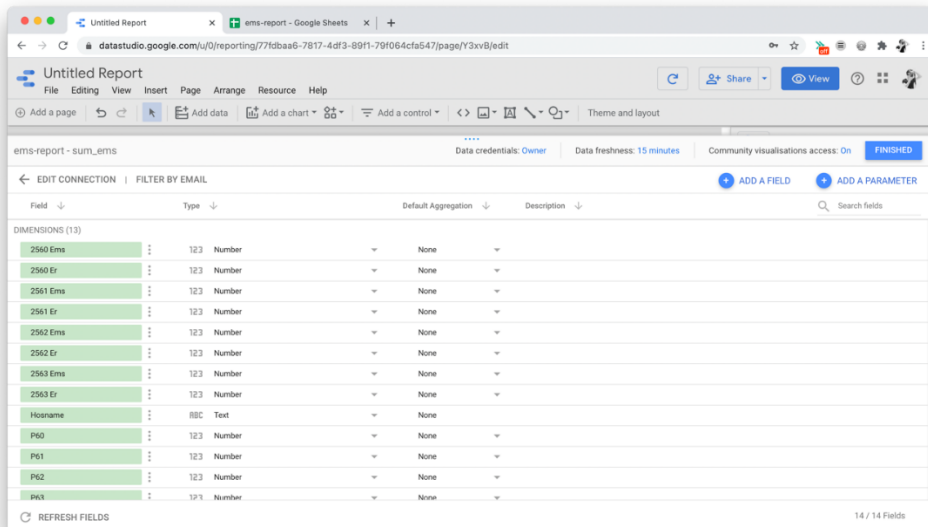
7. ตั้งค่าชนิดข้อมูล ไปที่ เมนู Resource เลือก Manage added data sources



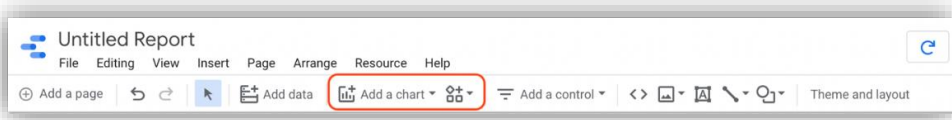
8. กด Edit เพื่อตรวจสอบชนิดข้อมูล หากพบว่าชนิดข้อมูลไม่ถูกต้องให้เปลี่ยนชนิดข้อมูลให้ตรงกับข้อมูล



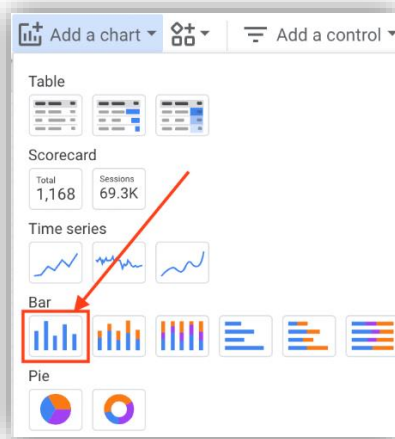
9. แปลงข้อมูลที่เป็นตัวเลขให้เป็น Number ข้อมูลที่เป็นตัวอักษรให้เลือกเป็น Text



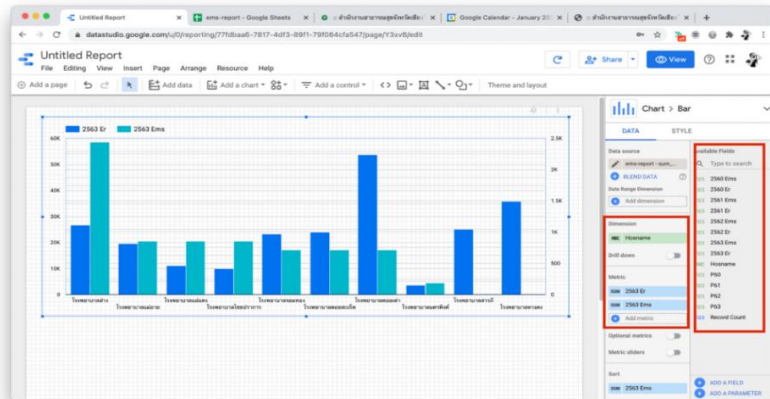
10. เริ่มสร้างกราฟแรกโดยคลิกที่เมนูบาร์ด้านบน “add a chart”



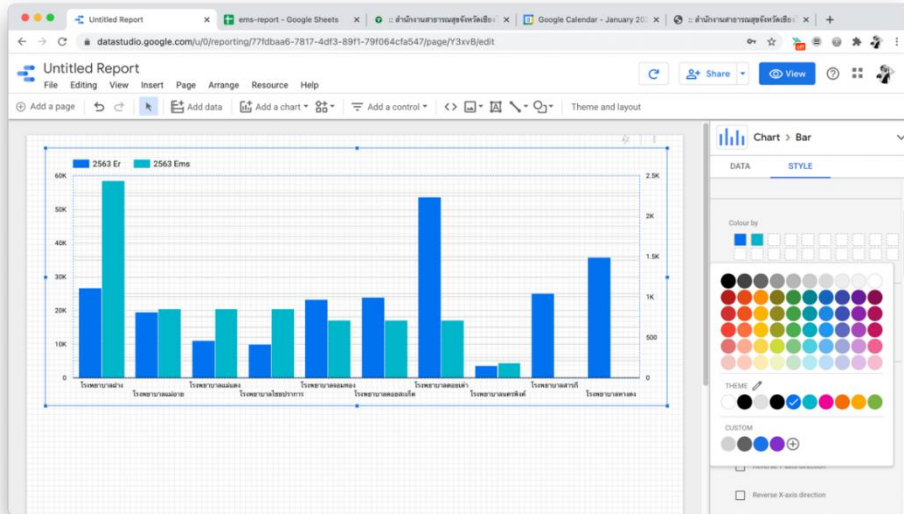
11. เลือกกราฟแท่ง



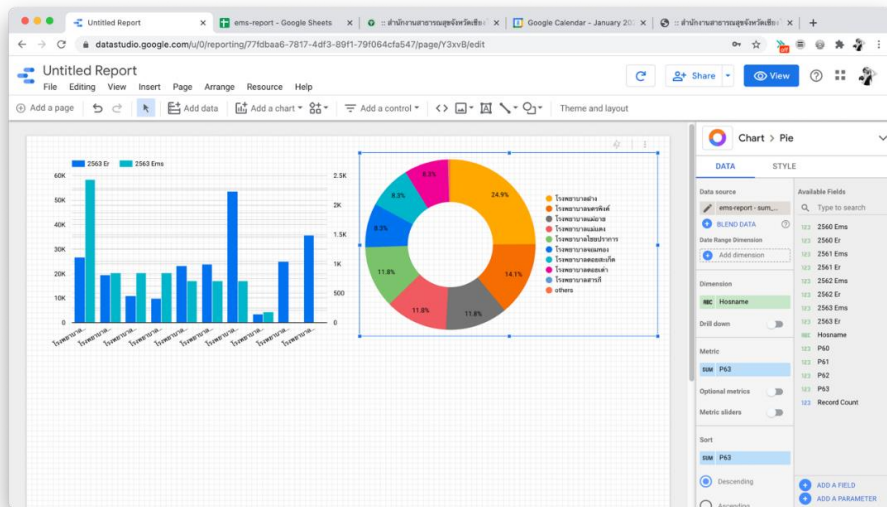
12. ในช่อง Dimension คือกราฟในแนวนแกน X ให้ทำการลากฟิลด์ข้อมูล Hosname (ชื่อหน่วยบริการ) มาวาง และในช่อง Metric เปรียบเสมือนกราฟในแนวนแกน Y ให้ลากฟิลด์ข้อมูล 2563_Er กับ 2563_Ems มาวางก็จะปรากฏเป็นกราฟแท่ง 2 แท่งขึ้นมาเปรียบเทียบกัน



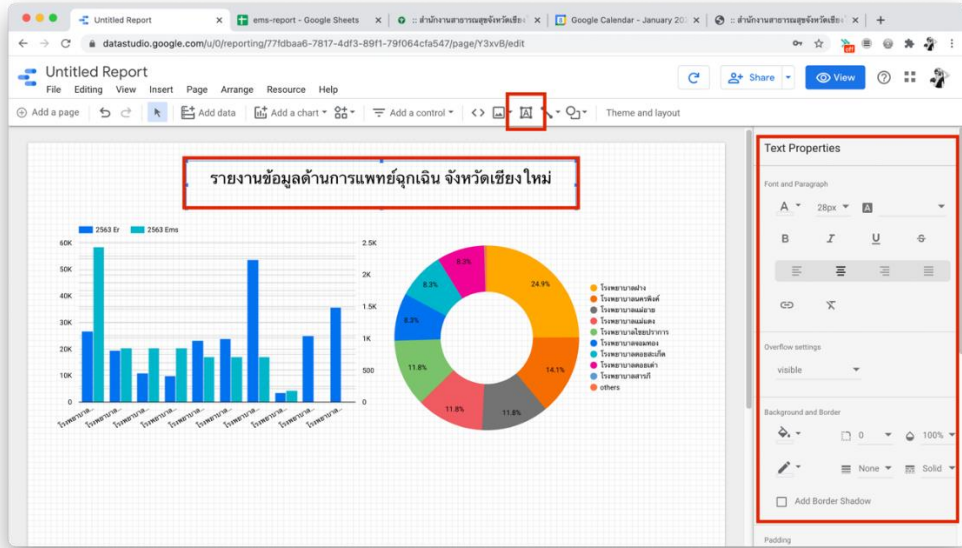
13. การปรับสีพื้นหลังหรือสีของกราฟ สามารถทำได้โดยคลิกไปที่แถบ Style



14. สร้างกราฟที่ 2 เลือกกราฟวงกลม โดยเลือกฟิลด์ Hosname มาวางในช่อง Dimension และ ลาก P63 (จำนวนผู้ป่วยในปี 2563) มาวางในช่อง Metric

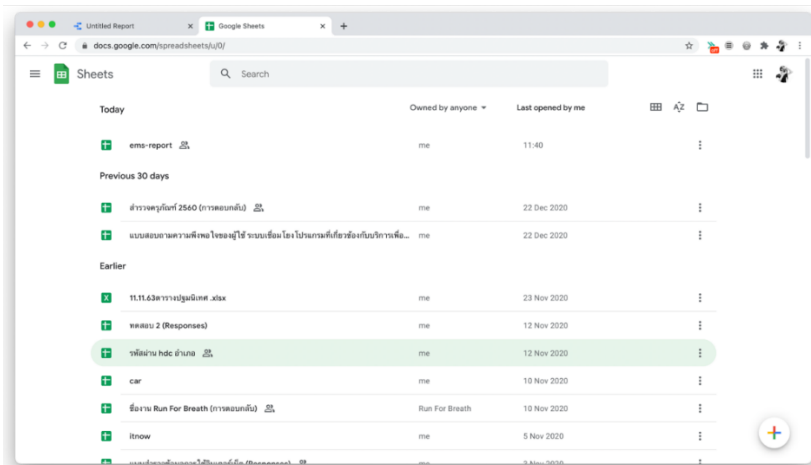


15. การแทรกข้อความลงใน Dashboard โดยคลิกที่ไอคอนรูปตัว A ที่แถบเมนูด้านบนและสามารถปรับขนาดตัวอักษรได้จากเมนูด้านขวามือ

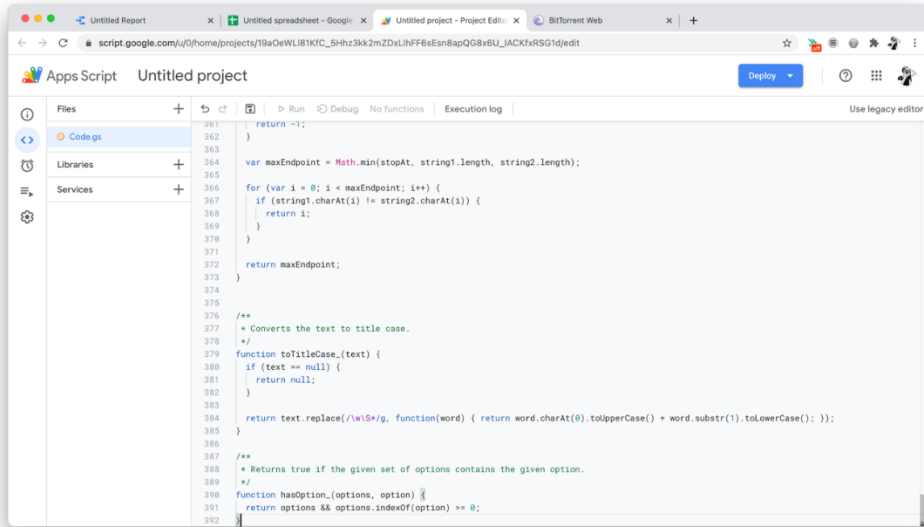


2.1.2 วิธีการจัดทำ Data Source ด้วย Google Sheets

1. เข้าเว็บผ่าน url ดังนี้ <https://docs.google.com/spreadsheets>
2. กดเครื่องหมาย “+” ที่หน้าจอบูมลงข้างขวามือ เพื่อสร้างแผ่นงานขึ้นมาใหม่



3. จากนั้นไปที่เมนู Tools เลือก script editor แล้วลบสคริปต์ในช่องออกทั้งหมด แล้ววางสคริปต์ใหม่ลงไปดังนี้



สคริปต์ที่นำไปวางแทน ดังนี้

```
function readRows() {
  var sheet = SpreadsheetApp.getActiveSheet();
  var rows = sheet.getDataRange();
  var numRows = rows.getNumRows();
  var values = rows.getValues();

  for (var i = 0; i <= numRows - 1; i++) {
    var row = values[i];
    Logger.log(row);
  }
};

function onOpen() {
  var sheet = SpreadsheetApp.getActiveSpreadsheet();
  var entries = [{}

```

```

    name : "Read Data",
    functionName : "readRows"
  }];
  sheet.addMenu("Script Center Menu", entries);
};

function ImportJSON(url, query, options) {
  return ImportJSONAdvanced(url, query, options, includeXPath_, defaultTransform_);
}

function ImportJSONAdvanced(url, query, options, includeFunc, transformFunc) {
  var jsondata = UrlFetchApp.fetch(url);
  var object = JSON.parse(jsondata.getContentText());

  return parseJSONObject_(object, query, options, includeFunc, transformFunc);
}

function URLEncode(value) {
  return encodeURIComponent(value.toString());
}

function parseJSONObject_(object, query, options, includeFunc, transformFunc) {
  var headers = new Array();
  var data = new Array();

  if (query && !Array.isArray(query) && query.toString().indexOf(",") != -1) {
    query = query.toString().split(",");
  }

  if (options) {
    options = options.toString().split(",");
  }
}

```

```

parseData_(headers, data, "", 1, object, query, options, includeFunc);
parseHeaders_(headers, data);
transformData_(data, options, transformFunc);

return hasOption_(options, "noHeaders") ? (data.length > 1 ? data.slice(1) : new Array()) :
data;
}

function parseData_(headers, data, path, rowIndex, value, query, options, includeFunc) {
  var dataInserted = false;

  if (isObject_(value)) {
    for (key in value) {
      if (parseData_(headers, data, path + "/" + key, rowIndex, value[key], query, options,
includeFunc)) {
        dataInserted = true;
      }
    }
  } else if (Array.isArray(value) && isObjectArray_(value)) {
    for (var i = 0; i < value.length; i++) {
      if (parseData_(headers, data, path, rowIndex, value[i], query, options, includeFunc)) {
        dataInserted = true;
        rowIndex++;
      }
    }
  } else if (!includeFunc || includeFunc(query, path, options)) {
    // Handle arrays containing only scalar values
    if (Array.isArray(value)) {
      value = value.join();
    }

    // Insert new row if one doesn't already exist
    if (!data[rowIndex]) {

```

```

    data[rowIndex] = new Array();
}

// Add a new header if one doesn't exist
if (!headers[path] && headers[path] != 0) {
    headers[path] = Object.keys(headers).length;
}

// Insert the data
data[rowIndex][headers[path]] = value;
dataInserted = true;
}

return dataInserted;
}

function parseHeaders_(headers, data) {
    data[0] = new Array();

    for (key in headers) {
        data[0][headers[key]] = key;
    }
}

function transformData_(data, options, transformFunc) {
    for (var i = 0; i < data.length; i++) {
        for (var j = 0; j < data[i].length; j++) {
            transformFunc(data, i, j, options);
        }
    }
}

function isObject_(test) {

```

```
    return Object.prototype.toString.call(test) === '[object Object]';  
}
```

```
function isObjectArray_(test) {  
    for (var i = 0; i < test.length; i++) {  
        if (isObject_(test[i])) {  
            return true;  
        }  
    }  
}
```

```
    return false;  
}
```

```
function includeXPath_(query, path, options) {  
    if (!query) {  
        return true;  
    } else if (Array.isArray(query)) {  
        for (var i = 0; i < query.length; i++) {  
            if (applyXPathRule_(query[i], path, options)) {  
                return true;  
            }  
        }  
    } else {  
        return applyXPathRule_(query, path, options);  
    }  
}
```

```
    return false;  
};
```

```
function applyXPathRule_(rule, path, options) {  
    return path.indexOf(rule) == 0;  
}
```

```

function defaultTransform_(data, row, column, options) {
  if (!data[row][column]) {
    if (row < 2 || hasOption_(options, "noInherit")) {
      data[row][column] = "";
    } else {
      data[row][column] = data[row-1][column];
    }
  }
}

if (!hasOption_(options, "rowHeaders") && row == 0) {
  if (column == 0 && data[row].length > 1) {
    removeCommonPrefixes_(data, row);
  }

  data[row][column] = toTitleCase_(data[row][column].toString().replace(/[\V_]/g, " "));
}

if (!hasOption_(options, "noTruncate") && data[row][column]) {
  data[row][column] = data[row][column].toString().substr(0, 256);
}

if (hasOption_(options, "debugLocation")) {
  data[row][column] = "[" + row + ", " + column + "]" + data[row][column];
}
}

function removeCommonPrefixes_(data, row) {
  var matchIndex = data[row][0].length;

  for (var i = 1; i < data[row].length; i++) {
    matchIndex = findEqualityEndpoint_(data[row][i-1], data[row][i], matchIndex);
  }

  if (matchIndex == 0) {

```



```

    return;
  }
}

for (var i = 0; i < data[row].length; i++) {
  data[row][i] = data[row][i].substring(matchIndex, data[row][i].length);
}
}

function findEqualityEndpoint_(string1, string2, stopAt) {
  if (!string1 || !string2) {
    return -1;
  }

  var maxEndpoint = Math.min(stopAt, string1.length, string2.length);

  for (var i = 0; i < maxEndpoint; i++) {
    if (string1.charAt(i) != string2.charAt(i)) {
      return i;
    }
  }

  return maxEndpoint;
}

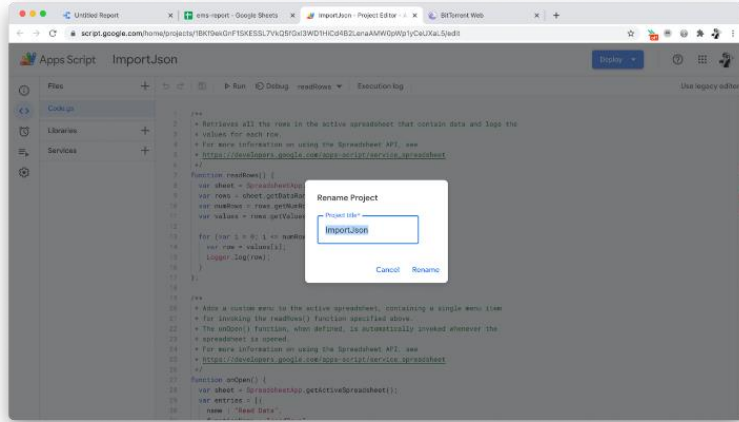
function toTitleCase_(text) {
  if (text == null) {
    return null;
  }

  return text.replace(/\w\S*/g, function(word) { return word.charAt(0).toUpperCase() +
word.substr(1).toLowerCase(); });
}

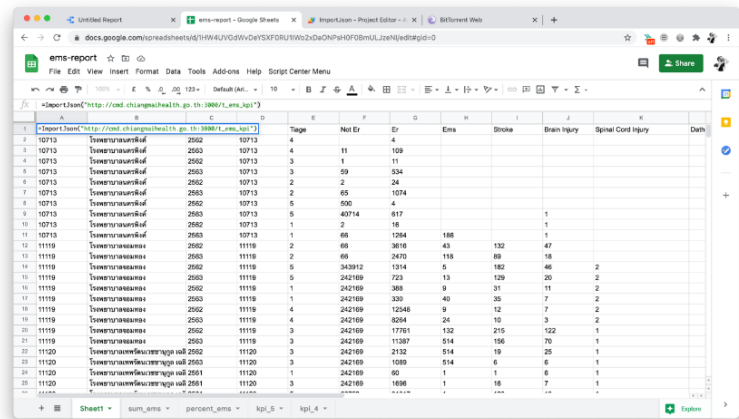
```

```
function hasOption_(options, option) {
  return options && options.indexOf(option) >= 0;
}
```

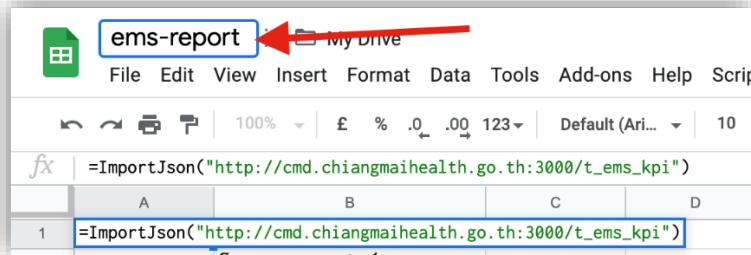
4. จากนั้นตั้งชื่อฟังก์ชันนี้ว่า ImportJson



5. กลับไปที่หน้า Sheet พิมพ์คำสั่งนี้ลงไปในเซลล์แรกของแผ่นงาน=ImportJson ("http://cmd.chiangmaihealth.go.th:3000/t_ems_kpi") จากนั้นกด Enter เพื่อให้ Google Sheet ดึงข้อมูลจาก API มาแสดงในแผ่นงาน ซึ่ง url ที่อยู่ในวงเล็บของฟังก์ชัน ImportJson(“ ”) นั้นจะเป็นข้อมูลรูปแบบ JSON format ซึ่งได้มาจาก API



6. เสร็จแล้วให้ตั้งชื่อไฟล์ ในที่นี้ชื่อ ems-report

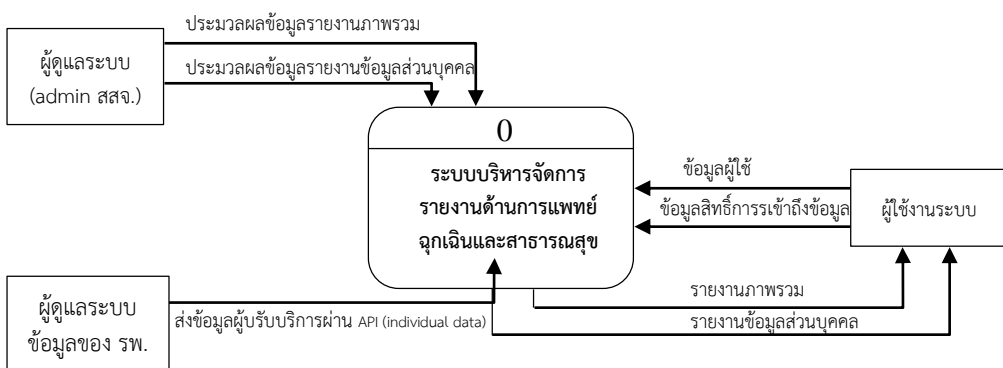


2.1.3 ระบบบริหารจัดการรายงานด้านการแพทย์ฉุกเฉินและสาธารณสุข (Admin)

เป็นระบบที่สามารถสร้างรายงานและแสดงผลรายงานชนิด dashboard และรายงานชนิดข้อมูลรายบุคคลได้ ระบบนี้พัฒนาขึ้นมาเพื่ออำนวยความสะดวกให้แก่ผู้ดูแลระบบ ซึ่งผู้ดูแลระบบสามารถบริหารจัดการส่วนหน้าแสดงผลของระบบเองได้ โดยไม่ต้องเขียนโค้ดคำสั่งใหม่เพื่อปรับปรุงหน้าเว็บไซต์โดยหลังจากที่เราสร้าง dashboard ใน Google Data Studio แล้ว จากนั้นนำ url ของ embed code มาเพิ่มในระบบบริหารจัดการรายงานด้านการแพทย์ฉุกเฉินและสาธารณสุข (CMD)

ผังภาพบริบท Context Diagram

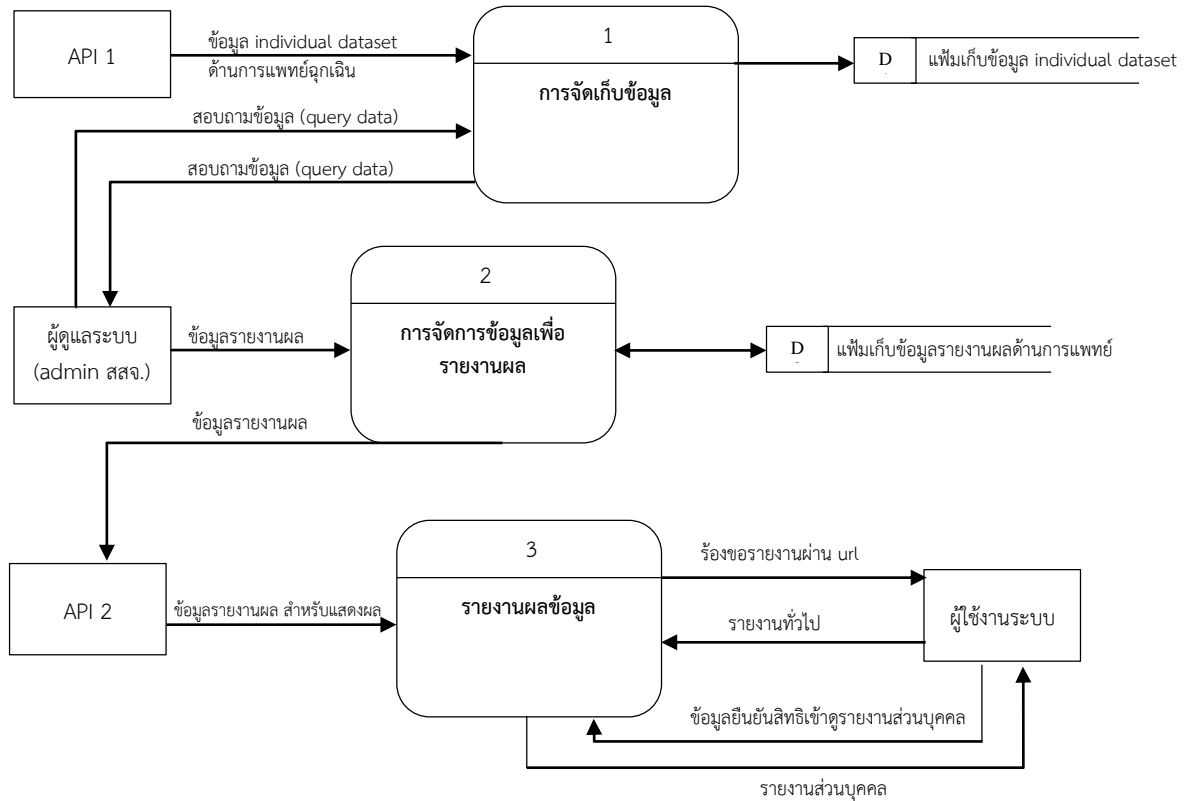
แผนภาพภาพบริบทแสดงถึงกระบวนการหรือกิจกรรมของผู้มีส่วนเกี่ยวข้องในกระบวนการของระบบบริหารจัดการรายงานด้านการแพทย์ฉุกเฉินและสาธารณสุข



ภาพที่ 18 ผังภาพบริบท Context Diagram

Data Flow Diagram Level 0

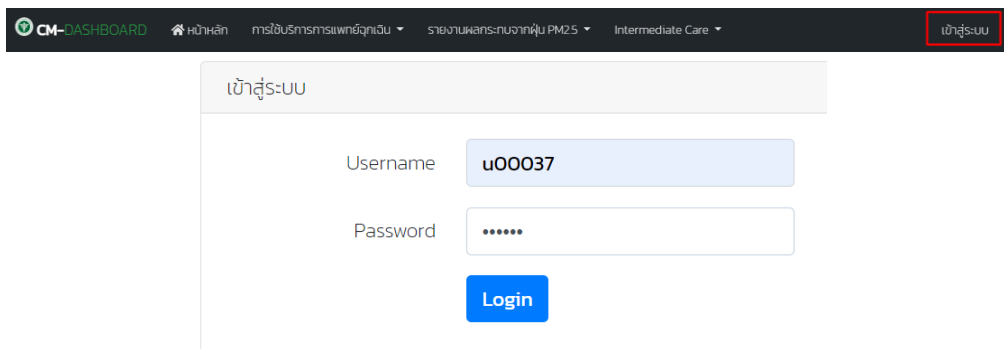
แผนภาพกระแสข้อมูล ระดับ 0 (Data Flow Diagram Level 0) แสดงถึงกระบวนการหรือกิจกรรมที่เกิดขึ้นในกระบวนการของระบบบริหารจัดการรายงานด้านการแพทย์ฉุกเฉินและสาธารณสุข



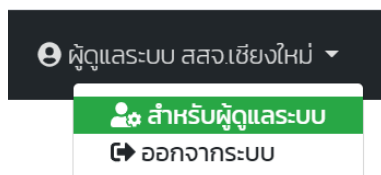
ภาพที่ 19 Data Flow Diagram Level 0

ขั้นตอนการนำ Dashboard / รายงานต่าง ๆ ที่สร้างจาก Google data studio หรือ web application เข้าสู่ระบบบริหารจัดการข้อมูลรายงาน (CM Dashboard)

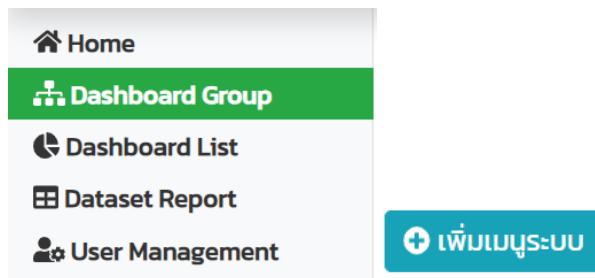
1. เปิด browser <https://cmd.chiangmaihealth.go.th/>
2. เข้าสู่ระบบ



3. คลิกเมนู ผู้ดูแลระบบ สสจ.เชียงใหม่ > สำหรับผู้ดูแลระบบ



4. สร้างเมนูที่ต้องการแสดงบน CM dashboard โดคลิกที่เมนู Dashboard group > เพิ่มเมนูระบบ



5. กรอกรายละเอียดเมนู

+ เพิ่มเมนูหลัก ×

ชื่อเมนู

ลำดับการแสดงผล

สถานะ

บันทึกรายการ ปิดหน้าต่าง

6. เมื่อกรอกเสร็จแล้วจะแสดงข้อมูลเมนูดังนี้ ซึ่งสามารถคลิกปุ่มเพื่อแก้ไขได้

Dashboard Group

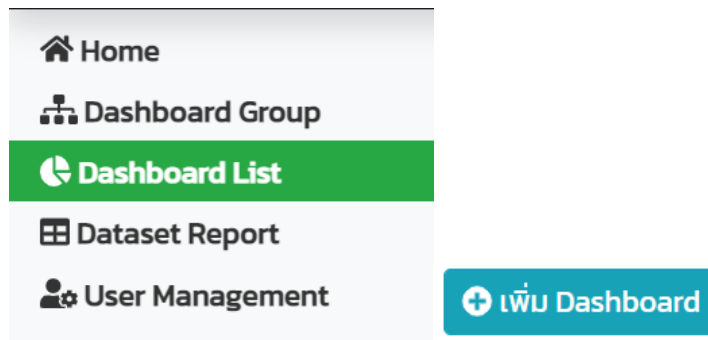
Show 10 entries Search

เพิ่มเมนูระบบ

Order	Dashboard Group	Status	
1	การใช้บริการทางการแพทย์ฉุกเฉิน	เปิด	แก้ไข
2	รายงานผลกระทบจากฝุ่น PM2.5	เปิด	แก้ไข
3	Intermediate Care	เปิด	แก้ไข

Showing 1 to 3 of 3 entries Previous 1 Next

7. การสร้างการเชื่อมต่อตามเมนูที่ได้สร้างไว้ โดยคลิกเมนู Dashboard list > เพิ่ม Dashboard



8. กรอกรายละเอียด Dashboard ที่ต้องการเชื่อม

+ เพิ่ม Dashboard ×

ชื่อเมนูหลัก

ชื่อรายงาน

Embed

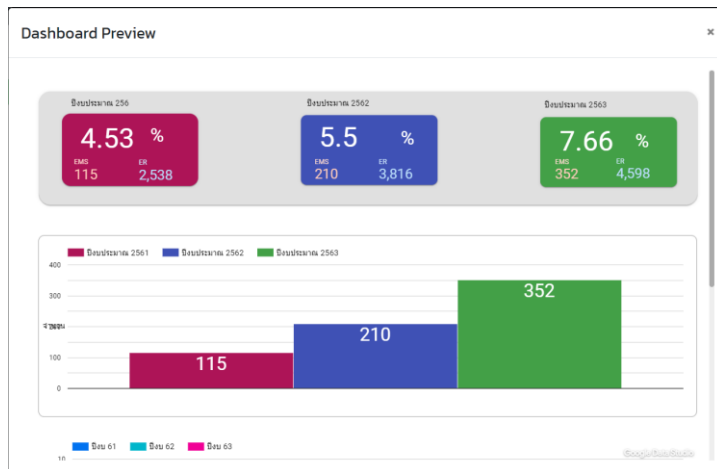
Size

สถานะ

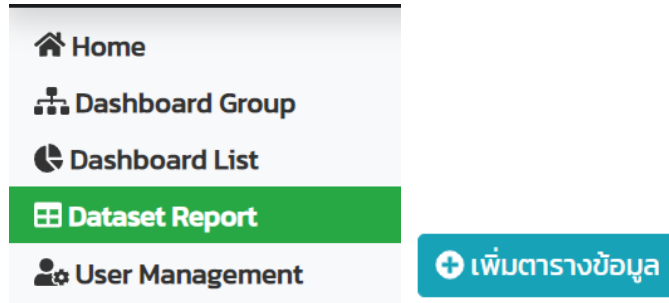
9. จะได้รายการ Dashboard ที่ได้เพิ่มไปและสามารถแก้ไขได้

ID	Dashboard Group	Dashboard Name	Source Embed	Status	
1	Intermediate Care	Intermediate Care	https://imc-cm.chiangmaihealth.go.th/web/index.php...	เปิด	แก้ไข
2	รายงานผลกระทบจากฝุ่น PM2.5	รายงานผลกระทบจากฝุ่น PM 2.5	https://pm25.chiangmaihealth.go.th...	เปิด	แก้ไข
3	การใช้บริการการแพทย์ฉุกเฉิน	3. จำนวนการใช้บริการการแพทย์ฉุกเฉิน	https://datastudio.google.com/embed/reporting/74f7...	เปิด	แก้ไข
4	การใช้บริการการแพทย์ฉุกเฉิน	4. ร้อยละของประชากรที่เจ็บป่วยวิกฤตฉุกเฉิน(สีแดง) ได้รับบริการผ่านระบบการแพทย์ฉุกเฉิน	https://datastudio.google.com/embed/reporting/74f7...	เปิด	แก้ไข

10. เมื่อคลิกแก้ไขจะสามารถ Preview ดูได้ด้วย โดยคลิกปุ่ม Dashboard preview



11. การสร้างเมนูเพื่อดูตารางข้อมูลต่าง ๆ ซึ่งจะต้องมีการ Login จึงจะสามารถดูได้เฉพาะสถาน
บริการตนเองเท่านั้น โดยคลิกเมนู Dataset report > เพิ่มตารางข้อมูล



12. กรอกรายละเอียด คำสั่ง Query SQL และออกแบบตารางตามต้องการ

A screenshot of the 'New Dataset Report' form. The form has the following sections:

- ชื่อตารางข้อมูล** (Table Name): A text input field containing 'ระบุชื่อตารางข้อมูล'.
- กำหนดสิทธิ์การเข้าถึงข้อมูล** (Set data access rights): Three radio button options:
 - ระดับ : จังหวัด
 - ระดับ : อำเภอ
 - ระดับ : sw.สต.
- สถานะ** (Status): A dropdown menu with 'เลือก...' (Select...) selected.
- คำสั่ง : Query SQL** (Query SQL Command): A text area containing the SQL query: `SELECT some_column FROM some_table`.
- กำหนดหัวตาราง** (Define table header): A rich text editor toolbar with various icons for text formatting and alignment.

13. จะได้รายการ Dataset report ต่าง ๆ จากนั้นจึงนำลิงค์ URL ไปเพิ่มใน Dashboard group และ Dashboard List

Dataset Report

เพิ่มตารางข้อมูล

Show 10 entries Search:

ID	Report Name	Query	Permission		
1	Test Datasets	select person.CID,person.NAME ...	3,4,5	แก้ไข	https://cmd.chiangmaihealth URL

Showing 1 to 1 of 1 entries Previous 1 Next

2.1.4 การแสดงผลข้อมูล PM2.5 และโรคที่เกี่ยวข้อง




เป็นการพัฒนาการแสดงผล Dashboard ด้วย php web application เพื่อเปรียบเทียบและดูแนวโน้มความสัมพันธ์ระหว่าง ค่าฝุ่น PM2.5 และ กลุ่มโรคทางเดินหายใจ ของจังหวัดเชียงใหม่ โดยใช้ข้อมูลการวินิจฉัยโรคจากทุกโรงพยาบาลของรัฐในจังหวัดเชียงใหม่ ผ่านโปรแกรมเชื่อมโยงข้อมูล CmDS (Chiangmai Data Sync) และนำเข้าค่า PM2.5/PM10 จากศูนย์ข้อมูลการเปลี่ยนแปลงสภาพภูมิอากาศ (Climate Change Data Center: CCDC) มหาวิทยาลัยเชียงใหม่ (<https://www.cmuccdc.org/api/ccdc>) โดยนำข้อมูล PM2.5/PM10 มาคำนวณหาค่าเฉลี่ยเป็นรายสัปดาห์/เดือน/ปี ของทั้งจังหวัดเชียงใหม่ จากนั้นจึงนำไปเข้าระบบบริหารจัดการรายงานฯ (CM Dashboard) เพื่อวางลิงค์แสดงผล

โครงสร้างตารางการแสดงผล PM2.5

Table t_group_dx

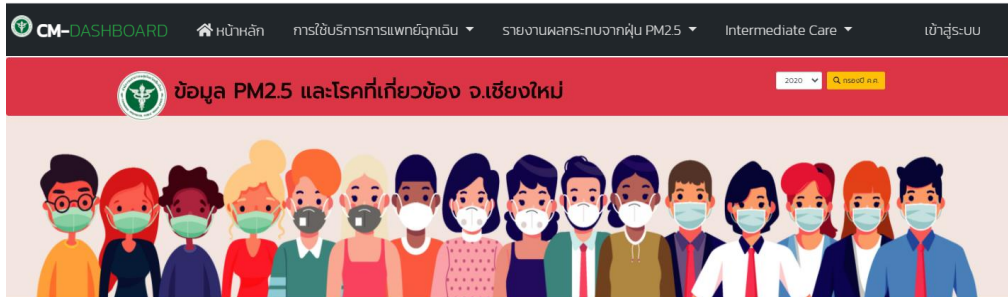
Name	Type	Length	Decimals	Not null	Virtual	Key	Comment
month	bigint	20	0	<input type="checkbox"/>	<input type="checkbox"/>		เดือน
HCODE	varchar	5	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>		รหัสสถานบริการ
hosname	varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>		ชื่อสถานบริการ
CID	varchar	13	0	<input type="checkbox"/>	<input type="checkbox"/>		เลขบัตรประชาชน
VN	varchar	15	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>		visit number
SEX	varchar	1	0	<input type="checkbox"/>	<input type="checkbox"/>		เพศ
DATEOPD	date	0	0	<input type="checkbox"/>	<input type="checkbox"/>		วันที่มารับบริการ
WEEKYEAR	int	11	0	<input type="checkbox"/>	<input type="checkbox"/>		สัปดาห์ที่ตามปีปฏิทิน
ampurcode	varchar	2	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>		รหัสอำเภอ
ampurname	varchar	255	0	<input type="checkbox"/>	<input type="checkbox"/>		ชื่ออำเภอ
DX_GROUP	varchar	11	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>		กลุ่มโรค

Table t_dustboy_ymw

Name	Type	Length	Decimals	Not null	Virtual	Key	Comment
Y	int	4	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	 1	คศ.
M	int	2	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	 2	เดือน
W	int	2	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	 3	สัปดาห์
PM10	double	19	2	<input type="checkbox"/>	<input type="checkbox"/>		ค่า PM10
PM25	double	19	2	<input type="checkbox"/>	<input type="checkbox"/>		ค่า PM2.5

คู่มือการใช้งาน

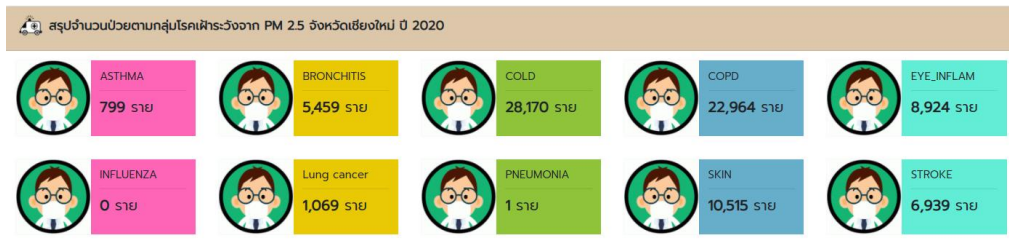
วิธีการเข้าใช้งาน ผ่าน web browser <https://cmd.chiangmaihealth.go.th/> เลือกเมนู รายงาน ผลกระทบจากฝุ่น PM2.5



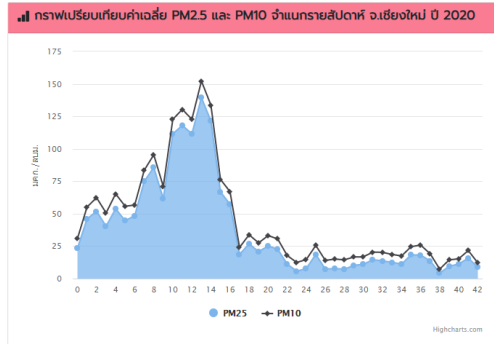
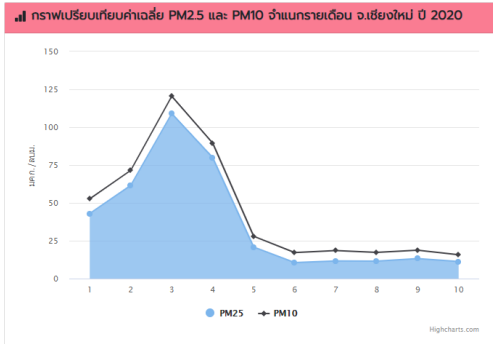
โดยผู้ใช้สามารถเลือกการแสดงผลของแต่ละปีได้ เริ่มตั้งแต่ปี คศ 2019-ปัจจุบัน เมื่อเลือกปีแล้ว จะส่งผลให้ทั้งหน้าเปลี่ยนเป็นการแสดงผลตามปีปฏิทินที่เลือก

รายละเอียดการแสดงผล มีดังนี้

1. สรุปจำนวนการป่วยด้วยกลุ่มโรคที่เฝ้าระวังจาก PM2.5 เป็นการแสดงจำนวนผู้ป่วยสะสมแยกตาม 10 กลุ่มโรคสำคัญที่ควรเฝ้าระวังจาก PM2.5 ซึ่งจะมีการประมวลผลทุกวันเวลา 23.30 น



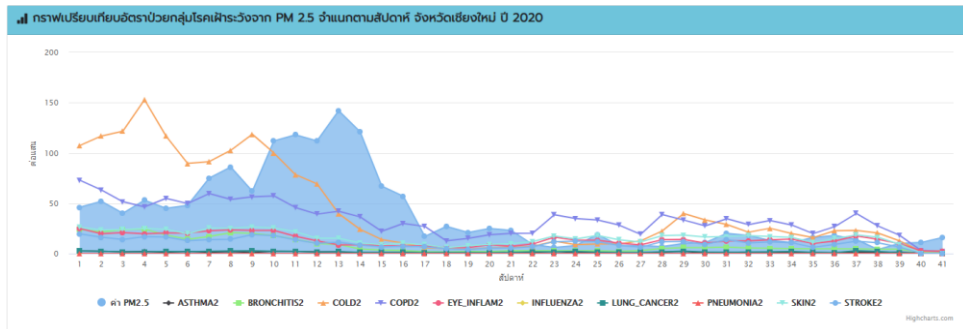
- กราฟแสดงแนวโน้มค่าเฉลี่ย PM2.5 และ PM10 จำแนกรายเดือน และสัปดาห์ เป็นข้อมูลที่นำเข้ามาจากคณะวิศวกรรมศาสตร์ มหาวิทยาลัยเชียงใหม่ ประมวลผลหาค่าเฉลี่ยของทั้งจังหวัดเชียงใหม่



ใช้คำสั่ง MySQL ดังนี้

```
SELECT Y,M,W,AVG(PM25) as PM25,AVG(PM10) as PM10 FROM t_dustboy_ymw
WHERE Y=$y GROUP BY Y,W
SELECT Y,M,AVG(PM25) as PM25,AVG(PM10) as PM10 FROM t_dustboy_ymw
WHERE Y=$y GROUP BY Y,M
```

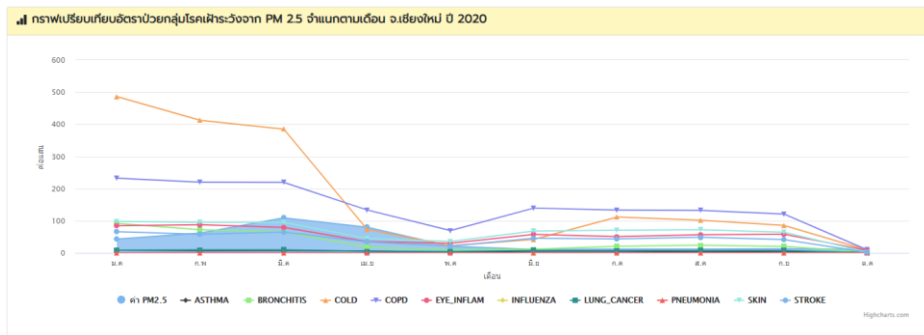
- กราฟแนวโน้มอัตราป่วยกลุ่มโรคเฝ้าระวังจาก PM2.5 โดยกราฟพื้นที่แสดงถึงค่าเฉลี่ย PM2.5 รายสัปดาห์ และมีกราฟเส้นแสดงถึงจำนวนผู้ป่วย แยกตามกลุ่มโรค รายสัปดาห์



ใช้คำสั่ง MySQL ดังนี้

```
SELECT line_week.years,line_week.WEEKYEAR, line_week.PM25,  
round((line_week.ASTHMA*100000)/ tpop.total,2) as ASTHMA,  
round((line_week.BRONCHITIS*100000)/ tpop.total,2) as BRONCHITIS,  
round((line_week.COLD*100000)/ tpop.total,2) as COLD, round((line_week.COPD*100000)/  
tpop.total,2) as COPD, round((line_week.EYE_INFLAM*100000)/ tpop.total,2) as  
EYE_INFLAM, round((line_week.INFLUENZA*100000)/ tpop.total,2) as INFLUENZA,  
round((line_week.LUNG_CANCER*100000)/ tpop.total,2) as LUNG_CANCER,  
round((line_week.PNEUMONIA*100000)/ tpop.total,2) as PNEUMONIA,  
round((line_week.SKIN*100000)/ tpop.total,2) as SKIN, round((line_week.STROKE*100000)/  
tpop.total,2) as STROKE FROM line_week INNER JOIN (SELECT pop.years as y,  
sum(pop.pop) as total FROM pop GROUP BY pop.years) tpop ON line_week.years =  
tpop.y WHERE line_week.years = $y GROUP BY line_week.years,line_week.WEEKYEAR
```

4. กราฟแสดงแนวโน้มอัตราการป่วยกลุ่มโรคเฝ้าระวังจาก PM2.5 แยกตามเดือน



ใช้คำสั่ง MySQL ดังนี้

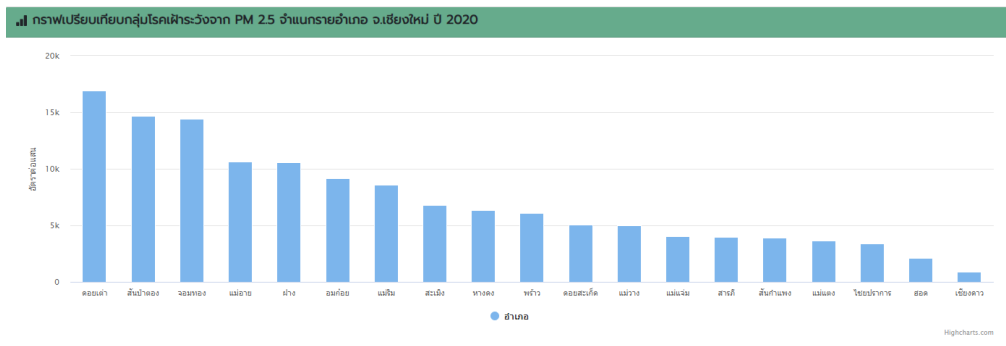
```
SELECT line_month.years,line_month.month, line_month.PM25,  
round((line_month.ASTHMA*100000)/ tpop.total,2) as ASTHMA,  
round((line_month.BRONCHITIS*100000)/ tpop.total,2) as BRONCHITIS,  
round((line_month.COLD*100000)/ tpop.total,2) as COLD,  
round((line_month.COPD*100000)/ tpop.total,2) as COPD,  
round((line_month.EYE_INFLAM*100000)/ tpop.total,2) as EYE_INFLAM,  
round((line_month.INFLUENZA*100000)/ tpop.total,2) as INFLUENZA,
```

```

round((line_month.LUNG_CANCER*100000)/ tpop.total,2) as LUNG_CANCER,
round((line_month.PNEUMONIA*100000)/ tpop.total,2) as PNEUMONIA,
round((line_month.SKIN*100000)/ tpop.total,2) as SKIN,
round((line_month.STROKE*100000)/ tpop.total,2) as STROKE FROM line_month
INNER JOIN (SELECT pop.years as y, sum(pop.pop) as total FROM pop GROUP BY
pop.years) tpop ON line_month.years = tpop.y WHERE line_month.years = $y
GROUP BY line_month.years,line_month.M

```

5. กราฟเปรียบเทียบกลุ่มโรคเฝ้าระวังจาก PM2.5 แยกตามอำเภอ โดยจะเรียงจากอัตราป่วยต่อแสนจากมากไปหาน้อย



ใช้คำสั่ง MySQL ดังนี้

```
SELECT * FROM column_ampur where years=$y
```

6. ตารางแสดงจำนวนการป่วยตามกลุ่มโรคเฝ้าระวังจาก PM2.5 แยกตามสัปดาห์

ตารางแสดงกลุ่มโรคเฝ้าระวังจาก PM 2.5 จำแนกตามสัปดาห์ จังหวัดเชียงใหม่ ปี 2020

Showing 1-9 of 9 items.

Dx	W1	W2	W3	W4	W5	W6	W7	W8	W9	W10	W11	W12	W13	W14	W15	W16	W17	W18	W19	W20	W21	W22	W23	W24	W25	W26
ASTHMA	41	36	28	34	21	33	20	31	18	38	34	28	29	12	18	16	20	5	7	5	6	17	33	15	21	10
BRONCHITIS	392	350	387	396	300	232	276	338	326	290	230	163	147	74	55	48	30	26	34	30	46	44	45	44	44	38
COLD	1748	1902	1981	2489	1900	1458	1488	1671	1931	1627	1277	1129	646	390	226	169	93	82	68	92	102	116	200	143	154	154
COPD	1187	1030	839	755	893	810	971	876	918	933	744	638	685	593	360	488	438	205	244	303	328	329	629	567	544	461
EYE_INFLAM	404	322	339	321	333	331	370	384	375	375	278	206	140	140	131	119	127	99	97	144	117	154	264	217	217	170
LUNGS_CANCER	45	37	28	28	34	32	38	42	44	36	37	22	30	20	17	20	15	17	16	13	20	27	28	33	23	17
PNEUMONIA	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
SKIN	428	382	389	409	392	322	413	409	431	415	354	247	248	171	143	168	145	97	127	149	161	194	285	236	292	223
STROKE	319	264	229	272	268	209	226	232	307	290	222	165	184	138	106	118	109	76	66	87	90	110	193	201	191	133

ใช้คำสั่ง MySQL ดังนี้

```
SELECT * FROM s_dx_w WHERE Y=$y
```

7. ตารางแสดงจำนวนการป่วยแยกตามสถานบริการที่ส่งข้อมูลเข้า API Center และเดือน

ตารางแสดงสถานบริการที่ส่งข้อมูลเข้า API Center จำแนกตามเดือน ปี 2020

Showing 1-19 of 19 items.

#	ปี	Hcode	สถานบริการ	ม.ค.	ก.พ.	มี.ค.	เม.ย.	พ.ค.	มิ.ย.	ก.ค.	ส.ค.	ก.ย.	ต.ค.	พ.ย.	ธ.ค.
1	2020	10713	โรงพยาบาลนครลำดิ่ง	5767	4599	4706	2724	2298	2436	2694	2881	3139	1603	0	0
2	2020	11119	โรงพยาบาลจอมทอง	36251	31036	32782	10159	0	25358	18522	18599	0	0	0	0
3	2020	11120	โรงพยาบาลเทพรัตนเวทยานุกูล เฉลิมพระเกียรติ ๖๐ พรรษา	6014	2617	8669	382	0	8120	7224	1135	0	0	0	0
4	2020	11121	โรงพยาบาลเชียงดาว	6095	0	4608	0	0	0	0	0	0	0	0	0
5	2020	11122	โรงพยาบาลดอยสะเก็ด	12387	11102	11913	6019	0	0	0	0	6043	349	0	0
6	2020	11123	โรงพยาบาลแม่แตง	13115	11572	3864	5896	3425	0	0	0	0	0	0	0
7	2020	11124	โรงพยาบาลสะเมิง	6033	4917	3367	0	0	0	0	0	0	0	0	0

ใช้คำสั่ง MySQL ดังนี้

```
SELECT * FROM hcode_month WHERE YEARS=$y
```


8. ตารางแสดงจำนวนการป่วยแยกตามสถานบริการที่ส่งข้อมูลเข้า API Center รายวัน

ตารางแสดงสถานบริการที่ส่งข้อมูลเข้า API Center จำนวนตามวัน 0 2020

Showing 1-137 of 137 items.

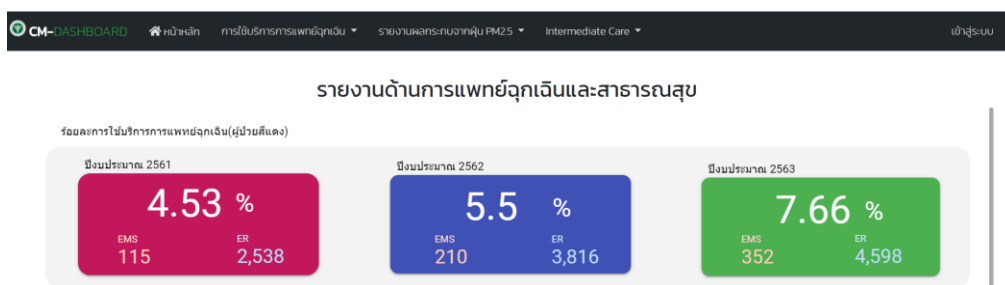
Hcode	Hname	Months	D01	D02	D03	D04	D05	D06	D07	D08	D09	D10	D11	D12	D13	D14	D15	D16	D17	D18	D19	D20	D21	D22
10713	โรงพยาบาลนครพิงค์	1	54	239	180	76	63	244	211	232	209	182	58	67	258	217	222	261	187	94	64	247	249	21
		2	72	61	267	245	232	198	157	63	65	68	263	208	241	190	56	57	243	204	201	197	127	65
		3	54	208	208	173	189	160	54	55	237	197	202	135	185	59	55	248	190	206	225	171	75	64
		4	131	143	112	39	26	32	174	129	163	150	28	13	106	58	48	118	90	19	16	145	128	13
		5	60	22	15	13	122	17	149	111	27	13	20	140	115	133	87	33	19	138	126	135	113	74
		6	110	98	15	113	0	21	15	114	100	132	112	92	26	12	132	114	120	121	119	28	9	14

ใช้คำสั่ง MySQL ดังนี้

```
SELECT * FROM hcode_day WHERE YEARS=$y
```

2.1.5 การแสดงผลข้อมูลฉุกเฉินการแพทย์

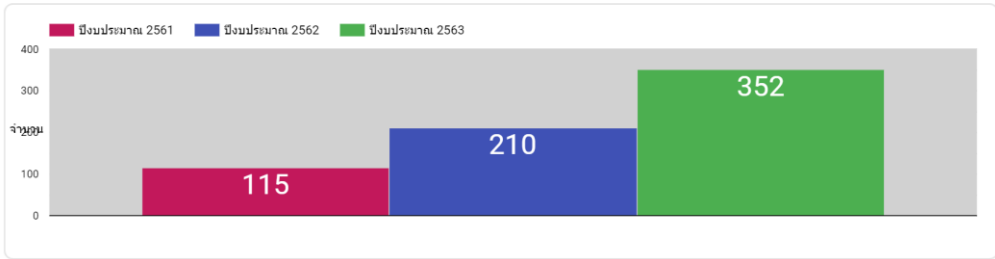
การสร้างและออกแบบการแสดงผล Dashboard จะใช้ Google data studio เป็นเครื่องมือในการสร้าง โดยใช้ข้อมูลจาก API data center ที่รวบรวมข้อมูลมาจากการเชื่อมโยงข้อมูลผ่านโปรแกรม CMDS ซึ่งได้ติดตั้งในทุกโรงพยาบาลในจังหวัดเชียงใหม่แล้ว นำมาผ่านชุดคำสั่งตามเงื่อนไขรายงานหรือข้อมูลที่ต้องการ แล้วใช้ Google data studio เชื่อมโยงข้อมูลดังกล่าว เพื่อแสดงผลรายงาน Dashboard เมื่อได้หน้าแสดงผลแล้ว จากนั้นจึงใช้โปรแกรม CM Dashboard เป็นเครื่องมือในการจัดการเพิ่มเมนู ฝั่งชุดคำสั่งในการแสดงผลรายงานต่าง ๆ



ภาพที่ 20 dashboard จำนวนและร้อยละการแพทย์ฉุกเฉิน

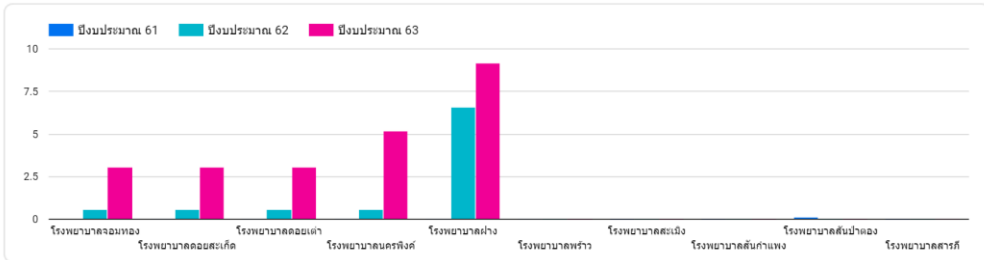
รายงานด้านการแพทย์ฉุกเฉินและสาธารณสุข

จำนวนผู้ป่วยวิกฤตฉุกเฉิน (สีแดง) ได้รับการบริการผ่านระบบการแพทย์ฉุกเฉิน รายปีงบประมาณ



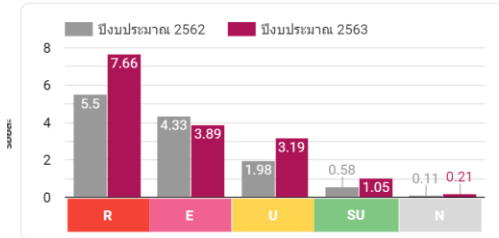
ภาพที่ 21 กราฟแสดงจำนวนผู้ป่วยวิกฤต รายปี

ร้อยละผู้ป่วยวิกฤตฉุกเฉิน (สีแดง) ได้รับการบริการผ่านระบบการแพทย์ฉุกเฉิน รายโรงพยาบาล



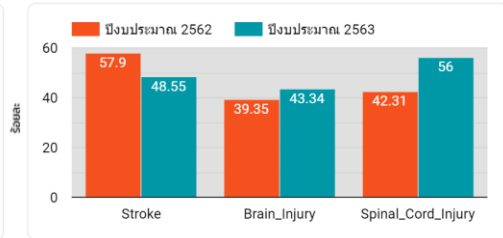
ภาพที่ 22 กราฟร้อยละผู้ป่วยวิกฤตฉุกเฉิน แยกรายสถานบริการและปีงบประมาณ

ร้อยละของประชากรที่เจ็บป่วยวิกฤตฉุกเฉิน ได้รับการบริการผ่านระบบการแพทย์ฉุกเฉิน



ภาพที่ 23 กราฟแสดงร้อยละประชากรที่เจ็บป่วยวิกฤตฉุกเฉิน แยกตามสี่

อัตราผู้ป่วยภาวะฉุกเฉิน (Stroke, Traumatic Brani Injury และ Spinal Cord Injury)



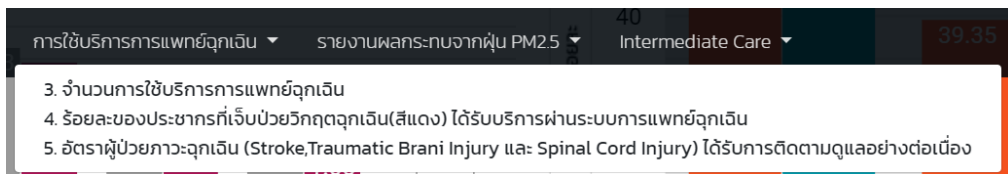
ภาพที่ 24 กราฟแสดงอัตราผู้ป่วยภาวะฉุกเฉิน แยกตามกลุ่มโรค

อัตราผู้ป่วยภาวะฉุกเฉิน (Stroke, Traumatic Brani Injury และ Spinal Cord Injury)

โรค	ปีงบประมาณ 2561			ปีงบประมาณ 2562			ปีงบประมาณ 2563		
	EMS	ER	ร้อยละ	EMS	ER	ร้อยละ	EMS	ER	ร้อยละ
Stroke	5,197	5,318	97.72	3,707	6,402	57.9	2,444	5,034	48.55
Brain_Injury	752	780	96.41	632	1,606	39.35	475	1,096	43.34
Spinal_Cord_Injury	36	37	97.3	22	52	42.31	14	25	56

ภาพที่ 25 ตารางแสดงจำนวนผู้ป่วยวิกฤตฉุกเฉิน แยกตามกลุ่มโรค

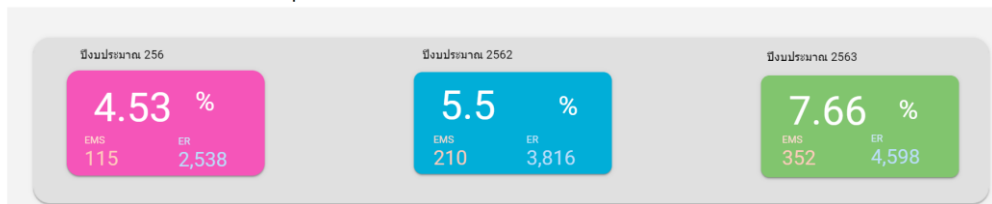
เมนูการให้บริการการแพทย์ฉุกเฉิน



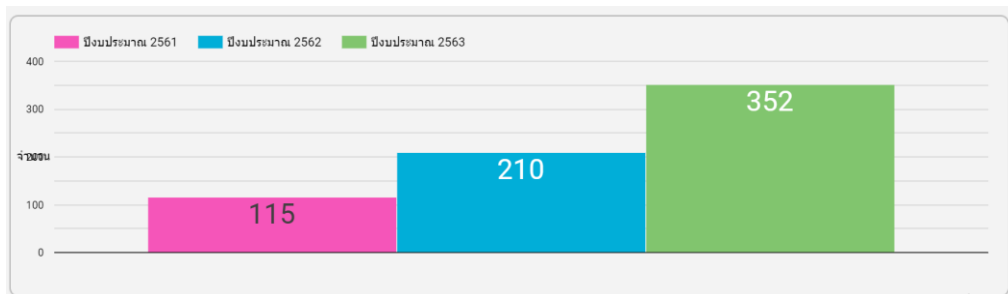
ภาพที่ 26 เมนูการให้บริการการแพทย์ฉุกเฉิน

เมื่อคลิกเลือกแล้วจะแสดงดังต่อไปนี้

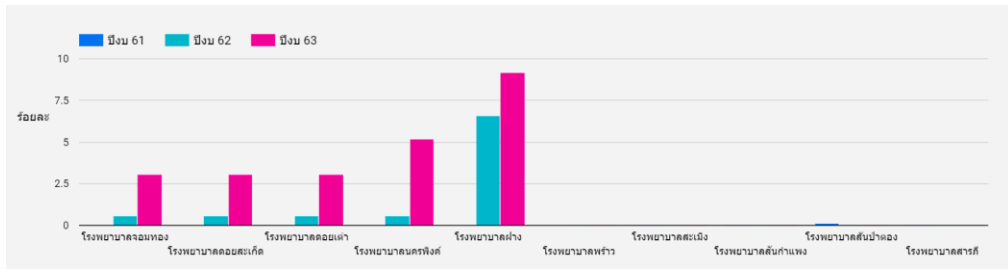
3. จำนวนการให้บริการการแพทย์ฉุกเฉิน



ภาพที่ 27 dashboard แสดงจำนวนและร้อยละการให้บริการการแพทย์ฉุกเฉิน

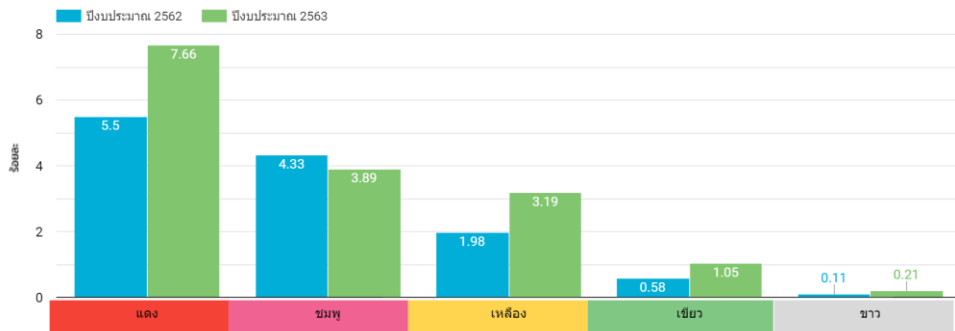


ภาพที่ 28 กราฟแสดงจำนวนการให้บริการการแพทย์ฉุกเฉิน แยกตามปีงบประมาณ



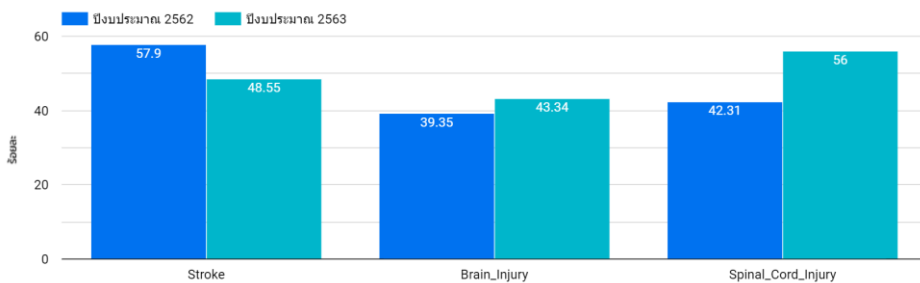
ภาพที่ 29 กราฟแสดงร้อยละการใช้บริการการแพทย์ฉุกเฉิน แยกตามหน่วยบริการและปีงบประมาณ

4. ร้อยละของประชากรที่เจ็บป่วยวิกฤตฉุกเฉิน(สีแดง) ได้รับการผ่านระบบการแพทย์ฉุกเฉิน



ภาพที่ 30 กราฟแสดงร้อยละผู้ป่วยวิกฤตฉุกเฉิน ได้รับการผ่านระบบการแพทย์ฉุกเฉิน แยกตามสีและปีงบประมาณ

5. อัตราผู้ป่วยภาวะฉุกเฉิน (Stroke, Traumatic Brani Injury และ Spinal Cord Injury) ได้รับการติดตามดูแลอย่างต่อเนื่อง



ภาพที่ 31 กราฟแสดงร้อยละผู้ป่วยภาวะฉุกเฉิน ได้รับการติดตามดูแลอย่างต่อเนื่อง แยกตามกลุ่มโรคและปีงบประมาณ

ตารางที่ 2 จำนวนและร้อยละผู้ป่วยภาวะฉุกเฉิน แยกตามกลุ่มโรคและปีงบประมาณ

โรค	ปีงบประมาณ 2561			ปีงบประมาณ 2562			ปีงบประมาณ 2563		
	EMS	ER	ร้อยละ	EMS	ER	ร้อยละ	EMS	ER	ร้อยละ
Stroke	5,197	5,318	97.72	3,707	6,402	57.9	2,444	5,034	48.55
Brain_Injury	752	780	96.41	632	1,606	39.35	475	1,096	43.34
Spinal_Cord_Injury	36	37	97.3	22	52	42.31	14	25	56

3. ผลความเป็นไปได้ของการใช้ระบบเชื่อมโยงโปรแกรมที่เกี่ยวข้องกับบริการเพื่อดูแลภาวะฉุกเฉินด้านการแพทย์และสาธารณสุข จังหวัดเชียงใหม่

จากการตอบแบบสอบถามผู้เกี่ยวข้องในการใช้โปรแกรมที่เกี่ยวข้องกับบริการเพื่อดูแลภาวะฉุกเฉิน ด้านการแพทย์และสาธารณสุข จำนวน 34 คน เพื่อประเมินความเป็นไปได้ในการใช้โปรแกรมพบว่า กลุ่มตัวอย่างส่วนใหญ่ทำงานตำแหน่งเจ้าพนักงาน/นักวิชาการคอมพิวเตอร์ ร้อยละ 55.0 รองลงมาคือ นักวิชาการ และพยาบาลร้อยละ 18.0 เท่ากัน และสังกัดในหน่วยงานยุทธศาสตร์และอื่น ๆ ร้อยละ 50.0 รองลงมาคือ งานเทคโนโลยีสารสนเทศและงานห้องฉุกเฉิน ร้อยละ 32.0 และ 18.0 ตามลำดับ ดังแสดงในตารางที่ 3

ตารางที่ 3 จำนวนและร้อยละของข้อมูลทั่วไปของกลุ่มตัวอย่างในระยะดำเนินการ (n=34)

ข้อมูลทั่วไป	จำนวน	ร้อยละ
ตำแหน่ง		
เจ้าพนักงาน/นักวิชาการคอมพิวเตอร์	19	55.0
เจ้าพนักงานทันตสาธารณสุข	1	3.0
นักวิชาการสาธารณสุข	6	18.0
แพทย์	1	3.0
พยาบาล	6	18.0
เวชกิจฉุกเฉิน	1	3.0
งาน/กลุ่มงาน		
งานห้องฉุกเฉิน	6	18.0
งานเทคโนโลยีสารสนเทศ	11	32.0
ยุทธศาสตร์และอื่น ๆ	17	50.0

เมื่อประเมินความเป็นไปได้ในการใช้ระบบเชื่อมโยงโปรแกรมที่พัฒนาขึ้นไปใช้ พบว่า กลุ่มตัวอย่างมีความคิดเห็นว่ารระบบเชื่อมโยงโปรแกรมที่เกี่ยวข้องกับบริการเพื่อการดูแลภาวะฉุกเฉินด้านการแพทย์และสาธารณสุขมีความง่าย ปลอดภัย และมีประสิทธิภาพในการทำงานในระดับปานกลาง ดังแสดงในตารางที่ 4

ตารางที่ 4 ค่าเฉลี่ยและส่วนเบี่ยงเบนมาตรฐานของคะแนนความเป็นไปได้ในการใช้ระบบเชื่อมโยงโปรแกรมฯ (n = 34)

ความเป็นไปได้	ค่าเฉลี่ย	SD	การแปลผล
1 การใช้งานง่าย	2.49	0.15	ปานกลาง
2 ความปลอดภัยของข้อมูล	2.24	0.06	ปานกลาง
3 ประสิทธิภาพการทำงาน	2.42	0.07	ปานกลาง

เมื่อพิจารณาความเป็นไปได้ในด้านความง่ายในการนำระบบไปใช้ พบว่า กลุ่มตัวอย่างส่วนใหญ่มีความเห็นว่าการใช้สีของตัวอักษรและรูปแบบ และความเหมาะสมของตำแหน่งในการจัดวางส่วนต่าง ๆ บนหน้าจอดีความง่ายในการนำไปใช้ในระดับมาก คิดเป็นร้อยละ 70.59 และ 61.79 ตามลำดับ ในขณะที่ความง่ายต่อการใช้งาน การประมวลผลจากโปรแกรมฯ ได้ผลลัพธ์ถูกต้องตามความเป็นจริง และความถูกต้องในการเชื่อมโยงข้อมูลแต่ละเมนู มีความง่ายในระดับปานกลางร้อยละ 52.94 52.94 และ 67.65 ตามลำดับ ดังแสดงในตารางที่ 5

ตารางที่ 5 จำนวนและร้อยละของกลุ่มตัวอย่าง จำแนกตามความคิดเห็นที่มีต่อความเป็นไปได้ในด้านความง่ายในการนำไปใช้ (n = 34)

ความง่ายในการนำไปใช้	ระดับความคิดเห็น					
	มาก		ปานกลาง		น้อย	
	จำนวน	ร้อยละ	จำนวน	ร้อยละ	จำนวน	ร้อยละ
1. โปรแกรมฯ ง่ายต่อการใช้งาน	16	47.06	18	52.94	0	0.00
2. การประมวลผลจากโปรแกรมฯ ได้ผลลัพธ์ถูกต้องตามความเป็นจริง	15	44.12	18	52.94	1	2.94

ความง่ายในการนำไปใช้	ระดับความคิดเห็น					
	มาก		ปานกลาง		น้อย	
	จำนวน	ร้อยละ	จำนวน	ร้อยละ	จำนวน	ร้อยละ
3. การใช้สีของตัวอักษร และรูปแบบเหมาะสม	24	70.59	9	26.47	1	2.94
4. ความเหมาะสมของ ตำแหน่งในการจัดวางส่วน ต่าง ๆ บนหน้าจอ	21	61.76	13	38.24	0	0.00
5. ความถูกต้องในการ เชื่อมโยงข้อมูลแต่ละเมนู	10	29.41	23	67.65	1	2.94

เมื่อพิจารณาความเป็นไปได้ในด้านความปลอดภัยของข้อมูล พบว่า กลุ่มตัวอย่างส่วนใหญ่มีความคิดเห็นว่าระบบแจ้งเตือนเมื่อเกิดข้อผิดพลาดในการทำงาน ระบบรักษาความปลอดภัย และระบบจำกัดการใช้งานตามระดับสิทธิของผู้ใช้งานมีความเป็นไปได้อยู่ในระดับปานกลางร้อยละ 73.53 67.65 และ 58.82 ตามลำดับ ดังแสดงในตารางที่ 6

ตารางที่ 6 จำนวนและร้อยละของกลุ่มตัวอย่าง จำแนกตามความคิดเห็นที่มีต่อความเป็นไปได้ในด้านความปลอดภัยของข้อมูล (n = 34)

ความปลอดภัยของข้อมูล	ระดับความคิดเห็น					
	มาก		ปานกลาง		น้อย	
	จำนวน	ร้อยละ	จำนวน	ร้อยละ	จำนวน	ร้อยละ
1. ระบบแจ้งเตือนเมื่อเกิด ข้อผิดพลาดในการทำงาน	7	20.59	25	73.53	2	5.88
2. ระบบรักษาความปลอดภัย ในการตรวจสอบรหัสผู้ใช้งาน และรหัสผ่านเมื่อเข้าใช้งาน โปรแกรมฯ	10	29.41	23	67.65	1	2.94
3. ระบบจำกัดการใช้งานตาม ระดับสิทธิของผู้ใช้งาน	12	35.29	20	58.82	2	5.88

เมื่อพิจารณาความเป็นไปได้ในด้านประสิทธิภาพการทำงาน พบว่า กลุ่มตัวอย่างส่วนใหญ่มีความคิดเห็นว่าการแสดงผลของข้อมูลและรายงานต่าง ๆ ให้อยู่ในระดับที่น้อยหรือระดับมากร้อยละ 55.88 สำหรับการทำงานในด้านอื่นมีประสิทธิภาพในการทำงานอยู่ในระดับปานกลาง ดังแสดงในตารางที่ 7

ตารางที่ 7 จำนวนและร้อยละของกลุ่มตัวอย่าง จำแนกตามความคิดเห็นที่มีต่อความเป็นไปในด้านประสิทธิภาพการทำงาน (n = 34)

ประสิทธิภาพการทำงาน	ระดับความคิดเห็น					
	มาก		ปานกลาง		น้อย	
	จำนวน	ร้อยละ	จำนวน	ร้อยละ	จำนวน	ร้อยละ
1. ความถูกต้องของผลลัพธ์ของข้อมูล	15	44.12	18	52.94	1	2.94
2. ความเร็วในการใช้งานโปรแกรมฯ	13	38.24	21	61.76	0	0.00
3. การแสดงผลของข้อมูลและรายงานต่าง ๆ อยู่ในรูปแบบที่สรุปและเข้าใจง่าย	19	55.88	15	44.12	0	0.00
4. การแสดงผลของข้อมูลและรายงานต่าง ๆ ให้อยู่ในระดับที่น้อยหรือระดับมากตามต้องการ	16	47.06	17	50.00	1	2.94
5. สามารถนำเสนอข้อมูลตามเงื่อนไขที่กำหนดได้ทั้งส่วนของการแสดงผลของข้อมูลและรายงานต่าง ๆ	15	44.12	19	55.88	0	0.00
6. การจัดลำดับในการทำงานของโปรแกรมมีความเหมาะสม	12	35.29	21	61.76	1	2.94
7. ขอบเขตในการทำงานของโปรแกรมฯ โดยรวมมีความเหมาะสม	15	44.12	18	52.94	1	2.94

บทที่ 5

สรุปผลและข้อเสนอแนะ

การศึกษาครั้งนี้เป็นการศึกษาแบบวิจัยและพัฒนา (Research and development) มีวัตถุประสงค์เพื่อพัฒนาระบบเชื่อมโยงโปรแกรมที่เกี่ยวข้องกับบริการเพื่อการดูแลภาวะฉุกเฉินด้านการแพทย์และสาธารณสุขในจังหวัดเชียงใหม่ ตั้งแต่เดือน ตุลาคม พ.ศ. 2562 ถึง พฤศจิกายน พ.ศ. 2563 กระบวนการในการพัฒนาระบบแบ่งออกเป็น 3 ระยะ คือ 1) ระยะวิเคราะห์สถานการณ์ 2) ระยะดำเนินการ และ 3) ระยะประเมินผล กลุ่มตัวอย่างในระยะวิเคราะห์สถานการณ์ คือ บุคลากรในสำนักงานสาธารณสุขจังหวัดเชียงใหม่และบุคลากรที่ดูแลระบบสารสนเทศในโรงพยาบาลจำนวน 15 คน และกลุ่มตัวอย่างในระยะดำเนินการและระยะประเมินผล คือ บุคลากรในสำนักงานสาธารณสุขจังหวัดเชียงใหม่และบุคลากรที่ใช้ระบบสารสนเทศในโรงพยาบาลจำนวน 34 คน โดยทำการคัดเลือกกลุ่มตัวอย่างแบบเจาะจง (Purposive sampling) เครื่องมือที่ใช้ในการศึกษาค้นคว้าครั้งนี้ประกอบด้วย แบบสอบถามข้อมูลทั่วไป และแบบประเมินความเป็นไปได้ของการใช้ระบบเชื่อมโยงโปรแกรมที่เกี่ยวข้องกับบริการเพื่อการดูแลภาวะฉุกเฉินด้านการแพทย์และสาธารณสุขมีการวิเคราะห์ข้อมูลด้วยวิธีวิเคราะห์เชิงเนื้อหาและสถิติเชิงพรรณนา

ผลการศึกษาพบว่า ระบบเชื่อมโยงโปรแกรมที่เกี่ยวข้องกับบริการเพื่อการดูแลภาวะฉุกเฉินด้านการแพทย์และสาธารณสุขในจังหวัดเชียงใหม่ เป็นระบบที่เชื่อมโยงข้อมูลเกี่ยวกับ PM 2.5 และข้อมูลการแพทย์ฉุกเฉินจากแต่ละโรงพยาบาลในจังหวัดเชียงใหม่เข้าสู่ศูนย์ข้อมูล API data center เพื่อใช้แสดงผลข้อมูลโดยภาพรวมของจังหวัด ทั้งนี้ จากการศึกษาความเป็นไปได้ของการนำระบบไปใช้ในกลุ่มตัวอย่างจำนวน 34 คน พบว่า ระบบเชื่อมโยงโปรแกรมที่เกี่ยวข้องกับบริการเพื่อการดูแลภาวะฉุกเฉินด้านการแพทย์และสาธารณสุขที่พัฒนาขึ้นมีความง่าย ความปลอดภัย และประสิทธิภาพในการทำงานในระดับปานกลาง (Mean = 2.49 2.24 และ 2.42 ตามลำดับ) เมื่อพิจารณาความเป็นไปได้ในด้านความง่ายในการนำระบบไปใช้ พบว่า กลุ่มตัวอย่างส่วนใหญ่มีความเห็นว่าการใช้สีของตัวอักษรและรูปแบบ และความเหมาะสมของตำแหน่งในการจัดวางส่วนต่าง ๆ บนหน้าจอมีความง่ายในการนำไปใช้ในระดับมาก ร้อยละ 70.59 และ 61.79 ตามลำดับ สำหรับความปลอดภัยของข้อมูลพบว่า ระบบแจ้งเตือนเมื่อเกิดข้อผิดพลาดในการทำงาน ระบบรักษาความปลอดภัย และระบบจำกัดการใช้งานตามระดับสิทธิของผู้ใช้งานมีความเป็นไปได้อยู่ในระดับปานกลาง ร้อยละ 73.53 67.65 และ 58.82 ตามลำดับ และด้านประสิทธิภาพการทำงาน พบว่า การแสดงผลของข้อมูลและรายงานต่าง ๆ ให้รายละเอียดครบถ้วนตามต้องการมีประสิทธิผลในระดับมาก ร้อยละ 55.88

1. ข้อจำกัดในการดำเนินการวิจัย

1. เนื่องจากโครงสร้างข้อมูลและตัวแปรที่เก็บรวมถึงซอฟต์แวร์ของแต่ละโรงพยาบาลมีความแตกต่างกัน ทำให้ต้องใช้ระยะเวลาในการทำความเข้าใจและความต้องการข้อมูลจากผู้ดูแลระบบข้อมูลของแต่ละ โรงพยาบาลนาน
2. สัญญาณการเชื่อมต่อ API ของแต่ละโรงพยาบาลขัดข้องบ่อยเนื่องจากการวาง API CMDS ร่วมกับเครื่องคอมพิวเตอร์ที่ใช้งานร่วมกับงานอื่น ๆ
3. แต่ละโรงพยาบาลยังต้องมีการอัปเดตชุดคำสั่งในการเชื่อมโยงข้อมูล เมื่อมีความต้องการเพิ่มขึ้น หรือมีการปรับปรุงชุดคำสั่ง
4. ข้อมูล HIS ของแต่ละโรงพยาบาลไม่มีความสมบูรณ์และถูกต้องตามมาตรฐาน data structure ที่สร้างขึ้นภายใต้โครงการ ทำให้หลังจากการเชื่อมต่อยังมีข้อมูลจำนวนน้อยมาก
5. ทางผู้วิจัยมีการทำระบบเชื่อม API อัตโนมัติรายสัปดาห์และรายเดือน แต่ข้อมูลที่โรงพยาบาลภายใต้เครือข่ายไม่มีการนำเข้าสู่ข้อมูล ทำให้ข้อมูลภาพรวมยัง under report ถ้าเทียบกับจำนวนจริง ดังนั้นอาจเนื่องด้วยปัจจัยหลายอย่าง เช่น ทรัพยากรมีจำนวนจำกัด การเพิ่ม work load จากงานประจำ และเป็นงานที่ไม่ได้ถูกกำหนดจากผู้บริหาร

2. ข้อเสนอแนะในการนำไปใช้

1. ควรแยกเครื่องคอมพิวเตอร์สำหรับเชื่อมโยงข้อมูลด้วย CMDS API จากงานประจำอื่น ๆ เพื่อให้การเชื่อมต่อข้อมูลเป็นไปอย่างราบรื่นและรวดเร็วมากขึ้น
2. ควรเชื่อมต่อสัญญาณอินเทอร์เน็ตด้วยระบบสาย LAN เพื่อให้การเชื่อมโยงข้อมูลเร็วขึ้น
3. ควรจัดอบรม Project manager ของแต่ละกลุ่มงานของโรงพยาบาล สำนักงานสาธารณสุข อำเภอและจังหวัด ในการสร้างรายงานหรือ Dashboard ด้วย Google data studio
4. ควรเพิ่มระบบความปลอดภัยของข้อมูล
5. ควรสร้างความตระหนักให้ผู้บริหารแต่ละโรงพยาบาลเห็นถึงความสำคัญในการเชื่อมโยงข้อมูลในระดับจังหวัด เพื่อที่จะสามารถนำระบบที่พัฒนาขึ้นไปประยุกต์ใช้ในทุกโรงพยาบาลให้ครอบคลุมทั่วเชียงใหม่

เอกสารอ้างอิง

- กองยุทธศาสตร์และแผนงาน สำนักงานปลัดกระทรวงสาธารณสุข. (2562). ระบบการแพทย์ฉุกเฉิน. [ข้อมูลอิเล็กทรอนิกส์]. สืบค้นจาก https://bps.moph.go.th/new_bps/
- การสาธารณสุขไทย. (2554-58). ระบบการแพทย์ฉุกเฉินเพื่อรองรับภาวะปกติและภาวะภัยพิบัติในประเทศไทย. [ข้อมูลอิเล็กทรอนิกส์]. สืบค้นจาก http://wops.moph.go.th/ops/thp/thp/userfiles/14_lesson13.pdf
- คณะกรรมการการสาธารณสุข วุฒิสภา. (2554). ระบบสารสนเทศด้านการสาธารณสุขไทย. กรุงเทพฯ: สำนักงานกรรมการ 3 สำนักงานเลขาธิการวุฒิสภา.
- คณะทำงานพัฒนาการจัดการความรู้ หมวด 4 สำนักงานปลัดกระทรวงสาธารณสุข. (2553). การพัฒนาระบบข้อมูลข่าวสารและสารสนเทศสุขภาพ. [ข้อมูลอิเล็กทรอนิกส์]. สืบค้นจาก <http://203.157.10.11/pmqa53/?cat=7>
- บริษัทโค้ดบี จำกัด. (2014). Application Programming Interface (API). [ข้อมูลอิเล็กทรอนิกส์]. สืบค้นจาก <https://www.codebee.co.th/>
- สถาบันการแพทย์ฉุกเฉินแห่งชาติ. (2553). รายงานประจำปี 2553 สถาบันการแพทย์ฉุกเฉินแห่งชาติ. สืบค้นจาก <https://www.niems.go.th/1/Ebook/Detail/230?group=23>
- สถาบันการแพทย์ฉุกเฉินแห่งชาติ. (2553). ระบบสารสนเทศการแพทย์ฉุกเฉิน. สืบค้นจาก <https://www.niems.go.th/1/Ebook/Detail/230?group=23>
- Bangkok Medical Software. (2021). โปรแกรม BMS HOSxP. [ข้อมูลอิเล็กทรอนิกส์]. สืบค้นจาก <https://hosxp.net/>
- Mindphp. (2542). ภาษา PHP. [ข้อมูลอิเล็กทรอนิกส์]. สืบค้นจาก <https://www.mindphp.com/>
- Networks365 Co., Ltd. (2015). โปรแกรมระบบจัดการฐานข้อมูลด้วย MySQL. [ข้อมูลอิเล็กทรอนิกส์]. สืบค้นจาก <http://www.networks365.net/>
- Opensource technology. (2014). ระบบ HospitalOS. สืบค้นจาก <http://www.opensource-technology.com/th/>

ภาคผนวก

แบบสอบถามความพึงพอใจของผู้ใช้ “ระบบเชื่อมโยงโปรแกรมที่เกี่ยวข้องกับบริการ เพื่อการดูแลภาวะฉุกเฉินด้านการแพทย์และสาธารณสุข จังหวัดเชียงใหม่”

คำชี้แจง : โปรดพิจารณาข้อความหรือข้อความอย่างละเอียด และทำเครื่องหมาย (√) ในช่อง
ที่ตรงกับความคิดเห็นของท่านมากที่สุด ตามเกณฑ์การพิจารณาตอบ ดังนี้

มีความเหมาะสมอยู่ในระดับมาก	3	คะแนน
มีความเหมาะสมอยู่ในระดับปานกลาง	2	คะแนน
มีความเหมาะสมอยู่ในระดับน้อย	1	คะแนน

หน่วยงาน..... ตำแหน่ง.....งาน/กลุ่มงาน.....

ลำดับที่	รายการประเมิน	ระดับความเหมาะสม		
		มาก	ปานกลาง	น้อย
ด้านการใช้งาน				
1	โปรแกรมฯ ง่ายต่อการใช้งาน			
2	การประมวลผลจากโปรแกรมฯ ได้ผลลัพธ์ถูกต้องตามความเป็นจริง			
3	การใช้สีของตัวอักษรและรูปแบบเหมาะสม			
4	ความเหมาะสมของตำแหน่งในการจัดวางส่วนต่าง ๆ บนหน้าจอ			
5	ความถูกต้องในการเชื่อมโยงข้อมูลแต่ละเมนู			
ด้านความปลอดภัยของข้อมูล				
6	ระบบแจ้งเตือนเมื่อเกิดข้อผิดพลาดในการทำงาน			
7	ระบบรักษาความปลอดภัยในการตรวจสอบรหัสผู้ใช้งานและรหัสผ่านเมื่อเข้าใช้งานโปรแกรมฯ			
8	ระบบจำกัดการใช้งานตามระดับสิทธิของผู้ใช้งาน			
ด้านประสิทธิภาพการทำงาน				
9	ความถูกต้องของผลลัพธ์ของข้อมูล			
10	ความเร็วในการใช้งานโปรแกรมฯ			
11	การแสดงผลของข้อมูลและรายงานต่าง ๆ อยู่ในรูปแบบที่สรุปและเข้าใจง่าย			

ลำดับที่	รายการประเมิน	ระดับความเหมาะสม		
		มาก	ปานกลาง	น้อย
12	การแสดงผลของข้อมูลและรายงานต่าง ๆ ให้รายละเอียดครบถ้วนตามต้องการ			
13	สามารถนำเสนอข้อมูลตามเงื่อนไขที่กำหนดได้ทั้งส่วนของการแสดงผลของข้อมูลและรายงานต่าง ๆ			
14	การจัดลำดับในการทำงานของโปรแกรมมีความเหมาะสม			
15	ขอบเขตในการทำงานของโปรแกรมฯ โดยรวมมีความเหมาะสม			

