

โครงการ เลขที่ CXX /2568

การวิเคราะห์คุณภาพข้อมูลและการจัดทำแดชบอร์ด
สำหรับแสดงข้อมูลความไวของเชื้อต่อยาปฏิชีวนะ

Assessment of Data Quality and Development
of an Interactive Dashboard for Presenting Antibigram Results

จิตติพัฒน์ สนม 6504101414

วิทยาศาสตร์บัณฑิต
สาขาวิชาวิทยาการคอมพิวเตอร์

คณะวิทยาศาสตร์ มหาวิทยาลัยแม่โจ้
ปีการศึกษา 2568

การวิเคราะห์คุณภาพข้อมูลและการจัดทำแดชบอร์ด
สำหรับแสดงข้อมูลความไวของเชื้อต่อยาปฏิชีวนะ
Assessment of Data Quality and Development
of an Interactive Dashboard for Presenting Antibigram Results

จิตติพัฒน์ สนม 6504101414

โครงการนี้เสนอต่อสาขาวิชาวิทยาการคอมพิวเตอร์เพื่อเป็นส่วนหนึ่ง
ของการศึกษาตามหลักสูตรปริญญา
วิทยาศาสตรบัณฑิต
สาขาวิชาวิทยาการคอมพิวเตอร์

คณะวิทยาศาสตร์ มหาวิทยาลัยแม่โจ้
ปีการศึกษา 2568

การวิเคราะห์คุณภาพข้อมูลและการจัดทำแดชบอร์ด
สำหรับแสดงข้อมูลความไวของเชื้อต่อยาปฏิชีวนะ
Assessment of Data Quality and Development
of an Interactive Dashboard for Presenting Antibigram Results

จิตติพัฒน์ สนม 6504101414

โครงการนี้ได้รับการพิจารณาอนุมัติให้เป็นส่วนหนึ่งของการศึกษา
ตามหลักสูตรปริญญาวิทยาศาสตรบัณฑิต
สาขาวิชาวิทยาการคอมพิวเตอร์

คณะกรรมการสอบโครงการ

..... ประธานกรรมการ

ผศ.ดร.สมนึก สีนรุปวน

..... กรรมการ

ผศ.ดร.พาสน์ ปราโมกษ์ชน

..... กรรมการ

อ.อลงกต กองมณี

..... ที่ปรึกษาพิเศษ

อ.ดร.กิตติกร หาญตระกูล

ปีการศึกษา 2568

กิตติกรรมประกาศ

โครงการ เรื่อง “การวิเคราะห์คุณภาพข้อมูลและการจัดทำแดชบอร์ดสำหรับแสดงข้อมูลความไวของเชื้อต่อยาปฏิชีวนะ” เป็นโครงการที่มีวัตถุประสงค์เพื่อพัฒนาเครื่องมือเชิงวิเคราะห์ที่ช่วยจัดการข้อมูลในงานด้านสาธารณสุข โดยอาศัยกระบวนการประเมินคุณภาพข้อมูลและการนำเสนอข้อมูลซึ่งมีประโยชน์อย่างยิ่งต่อการตัดสินใจ

ข้าพเจ้าขอกราบขอบพระคุณเป็นอย่างสูงต่อ อาจารย์นายแพทย์ฉันทชนก อร่ามรัตน์ ผู้เป็นแรงบันดาลใจและผู้ให้แนวคิดสำคัญในการริเริ่มโครงการนี้ รวมถึงการกำหนดประเด็นศึกษาและคำปรึกษาอย่างต่อเนื่องในทุกกระบวนการของการดำเนินโครงการ

ขอขอบพระคุณ ศูนย์วิจัย Global Health Research Center มหาวิทยาลัยเชียงใหม่ ที่ให้การสนับสนุนข้อมูลเชิงลึกเครื่องมือวิชาการ

ข้าพเจ้าขอแสดงความขอบพระคุณอย่างยิ่งต่อ ผู้ช่วยศาสตราจารย์ ดร.สมนึก สินธุพาน ประธานกรรมการสอบโครงการ ที่ได้ให้คำแนะนำอย่างตรงประเด็นและช่วยตรวจสอบความถูกต้องของเนื้อหาและกระบวนการทางวิชาการรวมถึง ผู้ช่วยศาสตราจารย์ดร.พาสน์ ปราโมกษ์ชน และอาจารย์ลงกต กองมณี ในฐานะกรรมการสอบโครงการ ที่ได้ให้ข้อเสนอแนะอันเป็นประโยชน์อย่างยิ่งต่อการพัฒนาโครงการให้มีความสมบูรณ์ยิ่งขึ้น

ข้าพเจ้าขอขอบคุณอาจารย์ ดร.กิตติกร หาญตระกูล ที่ปรึกษาพิเศษ ซึ่งได้ให้คำปรึกษาอย่างใกล้ชิด ทั้งในด้านการวางแผน วิเคราะห์ เพื่อให้โครงการนี้ดำเนินไปอย่างมีประสิทธิภาพ

ในสุดท้ายนี้ ข้าพเจ้ารู้สึกซาบซึ้ง และขอขอบพระคุณของทุกท่านเป็นอย่างยิ่งที่มีส่วนสนับสนุนให้โครงการฉบับนี้สามารถดำเนินไปได้อย่างราบรื่นและสำเร็จลุล่วงตามวัตถุประสงค์ที่ตั้งไว้ ขอแสดงความขอบคุณมา ณ โอกาสนี้ด้วยความเคารพอย่างสูง

จิตติพัฒน์ สนม

หัวข้อโครงการ

การวิเคราะห์คุณภาพข้อมูลและการจัดทำแดชบอร์ด
สำหรับแสดงข้อมูลความไวของเชื้อต่อยาปฏิชีวนะ

ผู้เขียน

จิตติพัฒน์ สนม รหัสนักศึกษา 6504101414

ปริญญา

วิทยาศาสตร์บัณฑิต (วิทยาการคอมพิวเตอร์)

คณะกรรมการที่ปรึกษาโครงการ

ผศ.ดร.สมนึก สิ้นธูปวน ประธานกรรมการ

ผศ.ดร.พาสน์ ปราโมกษ์ชน กรรมการ

อ.อลงกต กองมณี กรรมการ

อ.ดร.กิตติกร หาญตระกูล ที่ปรึกษาพิเศษ

บทคัดย่อ

ในการบริหารจัดการระบบบริการสุขภาพข้อมูลถือเป็นทรัพยากรสำคัญที่มีผลต่อการตัดสินใจของบุคลากรทางการแพทย์และการกำหนดนโยบายด้านสาธารณสุขโดยเฉพาะข้อมูลที่เกี่ยวข้องกับการเพาะเชื้อและการใช้ยาปฏิชีวนะซึ่งเป็นข้อมูลพื้นฐานในการประเมินสถานการณ์เชื้อดื้อยา ปัญหาสำคัญที่มักพบคือข้อมูลที่มีคุณภาพต่ำ เช่น ข้อมูลซ้ำซ้อน ข้อมูลไม่ครบถ้วน หรือไม่สอดคล้องกันระหว่างแฟ้มข้อมูล ส่งผลให้การวิเคราะห์ข้อมูลและการสื่อสารผลต่อทีมรักษาไม่แม่นยำ โครงการนี้มีวัตถุประสงค์เพื่อพัฒนาเครื่องมือสำหรับประเมินคุณภาพของข้อมูลทางคลินิก โดยอิงจากเกณฑ์มาตรฐาน 7 ด้าน ได้แก่ ความถูกต้อง, ความเชื่อถือได้, ความสอดคล้อง, ความครบถ้วน, ความเกี่ยวข้อง, ความสามารถในการเข้าถึง และความทันเวลา ข้อมูลที่ผ่านการประเมินแล้วจะถูกจัดการผ่านกระบวนการ ETL (Extract, Transform, Load) เพื่อให้พร้อมต่อการวิเคราะห์และแสดงผลผ่านแดชบอร์ดแบบโต้ตอบ (Interactive Dashboard) ที่พัฒนาขึ้นสามารถแสดงข้อมูล antibiogram ซึ่งเป็นตารางสรุปเปอร์เซ็นต์ความไวของเชื้อแบคทีเรียแต่ละชนิดต่อยาปฏิชีวนะ โดยผู้ใช้งานสามารถเลือกดูข้อมูลเฉพาะกลุ่มได้ตามชนิดสิ่งส่งตรวจ อายุ เพศ หน่วยรักษา หรือเลือกเชื้อและยาเฉพาะ

เพื่อวิเคราะห์แนวโน้มการติดยาได้อย่างมีประสิทธิภาพ นอกจากนี้ ระบบยังสามารถตรวจสอบปัญหาคุณภาพข้อมูล เช่น ค่า Missing, ข้อมูลซ้ำซ้อน และข้อมูลผิดรูปแบบ พร้อมรายงานผลแยกตามแฟ้มและฟิลด์ข้อมูล ผลลัพธ์ของโครงการนี้คาดว่าจะประโยชน์ต่อโรงพยาบาลในการยกระดับคุณภาพของข้อมูลผู้ป่วย เพิ่มความแม่นยำในการวิเคราะห์การใช้ยาปฏิชีวนะ และสนับสนุนการตัดสินใจด้านบริการสุขภาพอย่างมีประสิทธิภาพ พร้อมทั้งช่วยให้นักศึกษาได้รับประสบการณ์ตรงในการพัฒนาเครื่องมือวิเคราะห์ข้อมูลที่สามารถนำไปประยุกต์ใช้ได้จริงในระบบสาธารณสุข

Project Name	Assessment of Data Quality and Development of an Interactive Dashboard for Presenting Antibigram Results
Author	Mr. Jittiphat Sanom, Student ID 6504101414
Degree	Bachelor of Science (Computer Science)
Project Advisory committee	Dr. Somnuek Sinthupuan, Chairperson Asst. Prof. Dr. Part Pramokchon, Committee Mr. Alongkot Gongmanee, Committee Dr. Kittikorn Hantrakul, Special Advisor

ABSTRACT

In managing the healthcare system, data is considered an important resource that impacts the decision-making of medical personnel and the formulation of public health policies, especially data related to pathogen cultivation and the use of antibiotics, which is fundamental information for assessing the situation of drug-resistant infections. An important issue that is often encountered is low-quality data, such as redundant data, incomplete data, or inconsistent data between datasets, which results in inaccurate data analysis and communication of results to the

maintenance team. This project aims to develop a tool for assessing the quality of clinical data based on seven standard criteria: accuracy, reliability, consistency, completeness, relevance, accessibility, and timeliness. Data that has been evaluated will be managed through the ETL (Extract, Transform, Load) process to prepare it for analysis and presentation through an Interactive Dashboard that has been developed to display antibiogram data, which is a summary table of the percentage sensitivity of each bacterial strain to antibiotics. Users can selectively view information by type of submitted materials, age, gender, treatment unit, or choose specific pathogens and medications to effectively analyze drug resistance trends. Additionally, the system can check for data quality issues such as missing values, duplicate data, and incorrectly formatted data, along with reporting results categorized by file and data field. The results of this project are expected to benefit hospitals in improving the quality of patient data, increasing the accuracy of antibiotic usage analysis, and supporting effective decision-making in health services, while also providing students with hands-on experience in developing data analysis tools that can be practically applied in the public health system.

สารบัญ

	หน้า
กิตติกรรมประกาศ	ค
บทคัดย่อภาษาไทย	ง
บทคัดย่อภาษาอังกฤษ	จ-ฉ
สารบัญตาราง	ญ
สารบัญภาพ	ฎ
 บทที่ 1 บทนำ	
1.1 ที่มาและความสำคัญของปัญหา	1
1.2 วัตถุประสงค์	3
1.3 ฮาร์ดแวร์ ซอฟต์แวร์ เครื่องมือและเทคโนโลยีที่ใช้ในการศึกษา	3-4
1.4 วิธีการศึกษา	4-5
1.5 ขอบเขตของโครงการ	5-6
1.6 ผลที่คาดว่าจะได้รับ	7
1.7 สถานที่ดำเนินงาน	7
1.8 ระยะเวลาการดำเนินงาน	8
 บทที่ 2 การศึกษาค้นคว้าและงานที่เกี่ยวข้อง	
2.1 ระบบข้อมูลผู้ป่วย (IPD/OPD)	9
2.2 การประเมินคุณภาพข้อมูล	9-10
2.3 กระบวนการ ETL (Extract, Transform, Load)	10-11

	หน้า
บทที่ 3 เทคโนโลยีที่ใช้ในการศึกษา	
3.1 Python	12-13
3.2 Pandas และ NumPy	13-15
3.3 Rapid Fuzz	15-16
3.4 OpenPyXL	17
3.5 Scikit-learn	18
3.6 Matplotlib, Seaborn	18
3.7 Regex (RegularExpression)	19-20
บทที่ 4 วิเคราะห์ระบบ	
4.1 การวิเคราะห์ผู้ใช้งาน (User Analysis)	21
4.2 ขั้นตอนการวิเคราะห์ระบบ	21-23
4.3 แผนภาพแสดงกรณีการใช้งาน (Use Case Diagram)	23-26
4.4 การออกแบบโครงสร้างฐานข้อมูล (Database Schema Design)	26-29
4.5 การออกแบบกระบวนการเข้าถึงและการจัดการข้อมูล (Database Access and Management Design)	30-32
4.6 การออกแบบการประมวลผลข้อมูล (Data Processing Design)	32-35
4.7 การออกแบบการส่งออกผลลัพธ์ (Export Data Design)	35-36
บทที่ 5 การทดสอบระบบ	
5.1 วัตถุประสงค์และขอบเขตการทดสอบ	37
5.2 นิยามคอลัมน์ที่ใช้ในการทดสอบ	37-38
5.3 วิธีการทดสอบตามกระบวนการทำงานจริง	38-40
5.4 เกณฑ์ยอมรับ (Acceptance Criteria)	40-41
5.5 การเทียบกับรายงานอ้างอิง	42
5.6 สรุปผลการทดสอบและข้อจำกัด	43

	หน้า
บทที่ 6 บทสรุปและข้อเสนอแนะ	
6.1 บทสรุปการพัฒนาโครงการ	44
6.2 ปัญหาและข้อจำกัด	44-45
6.3 ข้อเสนอแนะ	45
บรรณานุกรม	47-48
ภาคผนวก	49
ภาคผนวก ก คู่มือการติดตั้งโปรแกรม	50-52
ภาคผนวก ข คู่มือการใช้งานโปรแกรม	53
ประวัติผู้เขียน	54-55
สถานประกอบการ	
ข้อมูลสถานประกอบการ	56
งานที่ได้รับมอบหมายระหว่างปฏิบัติสหกิจศึกษา	56-57
เทคโนโลยีที่ใช้ในสถานประกอบการ	57-58

สารบัญตาราง

ตาราง	หน้า
1.1 ตารางแสดงระยะเวลาการดำเนินงาน	8
4.1 การใช้งาน (Use Case List)	24
4.2 แสดงดาต้าดิกชันนารีของตาราง: IPD_Culture	26
4.3 แสดงดาต้าดิกชันนารีของตาราง: IPD_Antibiotic	27
4.4 แสดงดาต้าดิกชันนารีของตาราง: IPD_PatientData	27
4.5 แสดงดาต้าดิกชันนารีของตาราง: IPD_Diagnosis	27-28
4.6 พจนานุกรมข้อมูลของชุดข้อมูลกลางที่ผ่านการตรวจคุณภาพ และการทำความสะอาด/มาตรฐาน แล้ว	29
4.7 พจนานุกรมข้อมูลของผลลัพธ์ (ซีตแอนติไบโอแกรม และซีต แยกตามสิ่งส่งตรวจ)	29
5.1 พจนานุกรมข้อมูลของผลลัพธ์ (ซีตแอนติไบโอแกรม และซีต แยกตามสิ่งส่งตรวจ)	27-38

สารบัญภาพ

รูป	หน้า
3.1 ตัวอย่างโค้ด อ่านไฟล์ IPD_Data, IPD_Cul, OPD_Cul	13
3.2 ตัวอย่างโค้ด การรวมข้อมูลผู้ป่วยกับผลเพาะเชื้อ โดยใช้ Pandas เป็นตัวกลาง	15
3.3 ตัวอย่างโค้ด from rapid fuzz import fuzz	16
3.4 ตัวอย่าง Excel Antibigram	17
3.5 confusion matrix เพื่อวิเคราะห์ผล antibiotic susceptibility	18
3.6 ใช้ Regex เพื่อลบ noise ที่เกิดจาก encoding, ลบข้อความในวงเล็บ, และจัดการช่องว่างที่ซ้ำซ้อน	19
3.7 ใช้ Regex เพื่อค้นหาข้อความที่อยู่หลังคำว่า “@Lab Comment:” และดึงออกมาเป็น specimen	19
3.8 ใช้ Regex เพื่อตัด prefix ที่ไม่จำเป็นและตรวจสอบว่าข้อความเป็นตัวเลขล้วนหรือไม่	19
3.9 ใช้ Regex เพื่อลบข้อความในวงเล็บและแท็กที่ไม่เกี่ยวข้อง เช่น ESBL, CRE, MRSA	20
3.10 ใช้ Regex ลบ resistance tags และข้อความในวงเล็บ	20
4.1 ผังการไหลของข้อมูล (Data Flow)	32
ก.1 เปิดแพลตฟอร์ม Google Colab	50
ก.2 เตรียมไฟล์ข้อมูล	50
ก.3 ตรวจสอบคุณภาพข้อมูลและสรุปเพื่อวางแผน	51
ก.4 ทำความสะอาดข้อมูล	51
ก.5 การส่งออกข้อมูลตาราง Antibigram	52
ก.6 การส่งออกข้อมูลตาราง Antibigram แยกตามสิ่งส่งตรวจตามที่ผู้ใช้ต้องการ	52

สารบัญภาพ(ต่อ)

รูป	หน้า
ข.1 คีย์ฟิลด์ที่ใช้ในการคำนวณ antibiogram และโดเมนของข้อมูล	53
ข.2 ใช้ตรวจ duplicate/ตรรกะวันเวลา และช่วยตามรอยปัญหาใน Data Quality Assessment	53
ข.3 ทดสอบ Data Quality Assessment → ETL/Cleaning → คำนวณ %S → Export Antibiogram	53

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญของปัญหา

การเพาะเชื้อเพื่อระบุชนิดของเชื้อโรคในผู้ป่วยมักจะใช้เวลาประมาณ 3-5 วัน เพื่อให้ได้ผลลัพธ์ที่ชัดเจนเกี่ยวกับชนิดของเชื้อที่เป็นสาเหตุของการติดเชื้อในร่างกายผู้ป่วยซึ่งในระหว่างที่ต้องรอผลการเพาะเชื้อเหล่านี้ ผู้ป่วยที่มีภาวะติดเชื้อจำเป็นต้องได้รับการรักษาด้วยยาปฏิชีวนะ (antibiotics) ทันทีเพื่อหยุดยั้งการแพร่กระจายของเชื้อและลดความเสี่ยงต่อภาวะแทรกซ้อนที่อาจเกิดขึ้น

อย่างไรก็ตาม การให้ยาปฏิชีวนะในช่วงที่ไม่ทราบชนิดของเชื้อเป็นการให้ยาแบบ "อาการเบื้องต้น" (empirical treatment) ซึ่งอาจไม่ได้ผลเต็มที่กับเชื้อที่เกิดขึ้น เนื่องจากยาปฏิชีวนะบางตัวอาจไม่สามารถฆ่าเชื้อบางชนิดได้ การเลือกใช้ยาฆ่าเชื้อที่มีประสิทธิภาพสูงสุดในระหว่างการรอผลการเพาะเชื้อจึงเป็นสิ่งสำคัญมาก เพราะจะช่วยให้การรักษาผู้ป่วยเป็นไปได้อย่างมีประสิทธิภาพและลดการใช้ยาผิดประเภทที่อาจทำให้เกิดผลข้างเคียงหรือการดื้อยาได้

การสร้าง แอนติไบโอแกรม (Antibiogram) เป็นเครื่องมือที่สำคัญในการช่วยให้การเลือกยาปฏิชีวนะในช่วงนี้มีความแม่นยำมากขึ้น โดย จะช่วยระบุข้อมูลจากการเพาะเชื้อในอดีตว่า เชื้อโรคแต่ละชนิดตอบสนองต่อยาปฏิชีวนะชนิดใดได้ดีที่สุด ซึ่งข้อมูลเหล่านี้จะช่วยให้แพทย์เลือกให้ยาปฏิชีวนะที่มีประสิทธิภาพในการฆ่าเชื้อได้รวดเร็วขึ้น ลดเวลาที่ผู้ป่วยต้องเสี่ยงกับการติดเชื้อที่ไม่สามารถควบคุมได้ นอกจากนี้ การรู้ข้อมูลเกี่ยวกับความไวของเชื้อต่อยาที่มีประสิทธิภาพในช่วงนี้ยังช่วยลดการใช้ยาฆ่าเชื้อที่ไม่เหมาะสม ซึ่งอาจส่งผลกระทบต่อผู้ป่วยในระยะยาว เช่น การดื้อยา หรือการเกิดเชื้อดื้อยา

ด้วยเหตุนี้ แอนติไบโอแกรม (Antibiogram) จึงเป็นเครื่องมือที่สำคัญในการช่วยเลือกให้ยาปฏิชีวนะในระหว่างการรอผลการเพาะเชื้อ และมีบทบาทในการพัฒนาการรักษาผู้ป่วยอย่างมีประสิทธิภาพ โดยไม่ต้องรอผลการเพาะเชื้อจนเสี่ยงต่อการเกิดภาวะแทรกซ้อนต่างๆ ที่อาจเกิดขึ้นจากการใช้ยาผิดประเภท

โครงการนี้จึงมีเป้าหมายเพื่อพัฒนาเครื่องมือสำหรับประเมินคุณภาพของข้อมูล การประเมินคุณภาพข้อมูล (DQA) ในโครงการนี้มี 7 โดเมนหลักที่ใช้ในการตรวจสอบคุณภาพข้อมูลทางคลินิก ได้แก่ Accuracy, Reliability, Consistency, Completeness, Relevance, Accessibility, และ Timeliness ซึ่งการประเมินคุณภาพข้อมูล (DQA) จะเน้นไปที่การประเมิน คุณภาพของข้อมูล ในบางมิติเท่านั้นโดยที่ในการประเมินข้อมูลในโครงการนี้จะใช้เฉพาะ 2 โดเมนหลัก ได้แก่ Consistency (ความสอดคล้องกันของข้อมูล) และ Completeness(ความครบถ้วนของข้อมูล) เพื่อให้มั่นใจว่าข้อมูลที่ใช้ในการสร้าง แอนติไบโอแกรม (Antibiogram) นั้นมีความสอดคล้องและครบถ้วน ซึ่งจะช่วยให้การประมวลผลข้อมูลและการตัดสินใจ เป็นไปอย่างแม่นยำและมีประสิทธิภาพ

โดยการใช้ทั้งสองโดเมนนี้ในการตรวจสอบข้อมูลจะช่วยให้ได้ข้อมูลที่มีคุณภาพสำหรับการสร้างแอนติไบโอแกรม (Antibiogram) และสามารถนำไปใช้ในการตัดสินใจทางการแพทย์ได้อย่างถูกต้องและมีประสิทธิภาพ เมื่อข้อมูลผ่านกระบวนการตรวจสอบแล้ว จะถูกนำเข้าสู่กระบวนการจัดการข้อมูลแบบ ETL (Extract, Transform, Load) เพื่อแปลงให้อยู่ในรูปแบบที่พร้อมใช้งานและวิเคราะห์ต่อได้สะดวก โดยเฉพาะข้อมูลเกี่ยวกับการเพาะเชื้อและการใช้ยาปฏิชีวนะ

การออกแบบระบบทั้งหมดนี้มีจุดมุ่งหมายเพื่อให้แพทย์หรือหน่วยงานสาธารณสุขสามารถเข้าถึงข้อมูลที่มีคุณภาพได้อย่างสะดวก ช่วยเสริมสร้างประสิทธิภาพในการวิเคราะห์ข้อมูล ส่งเสริมการใช้ยาปฏิชีวนะอย่างเหมาะสม และลดความเสี่ยงของการเกิดเชื้อดื้อยาในระยะยาว

ผู้จัดทำเห็นว่า การพัฒนาเครื่องมือที่สามารถประเมินคุณภาพข้อมูล และสามารถนำเสนอผลได้อย่างเป็นระบบ จะช่วยเสริมประสิทธิภาพในการบริหารจัดการข้อมูลทางการแพทย์ โดยสนับสนุนการใช้ข้อมูลที่มีความถูกต้อง ครบถ้วน และนำไปสู่การตัดสินใจที่มีประสิทธิภาพ ช่วยเพิ่มประโยชน์สูงสุดทั้งในด้านการรักษาผู้ป่วยและการพัฒนาระบบสุขภาพโดยรวม

1.2 วัตถุประสงค์

โครงการนี้มีวัตถุประสงค์เพื่อพัฒนาเครื่องมือสำหรับวิเคราะห์และแสดงข้อมูลผู้ป่วยใน โดยมีเป้าหมายหลักดังนี้

1. เพื่อประเมินและตรวจสอบคุณภาพของข้อมูลผู้ป่วยในจากฐานข้อมูลโรงพยาบาล โดยอิงตามเกณฑ์มาตรฐาน 2 ด้าน ได้แก่ Consistency (ความสอดคล้องกันของข้อมูล) และ Completeness(ความครบถ้วนของข้อมูล)
2. เพื่อออกแบบและพัฒนากระบวนการ ETL (Extract, Transform, Load) สำหรับรวบรวม แปลง และจัดเตรียมข้อมูลจากหลายแหล่งให้พร้อมต่อการใช้งาน
3. เพื่อสร้าง ต้นแบบ (Prototype) สำหรับแสดงข้อมูลสำคัญเกี่ยวกับผู้ป่วย เช่น จำนวนผู้ป่วย ผลตรวจทางห้องปฏิบัติการ ผลการเพาะเชื้อ และผลการต้อยา
4. เพื่อให้ระบบที่พัฒนาขึ้นสามารถนำไปประยุกต์ใช้ในโรงพยาบาลได้จริง ทั้งในด้านการวิเคราะห์คุณภาพข้อมูล การติดตามแนวโน้มการรักษา และการสนับสนุนการตัดสินใจของบุคลากรทางการแพทย์

1.3 ฮาร์ดแวร์ ซอฟต์แวร์ เครื่องมือ และเทคโนโลยีที่ใช้ในการศึกษาและพัฒนาโครงการ

1. ฮาร์ดแวร์ เครื่องคอมพิวเตอร์ส่วนบุคคล จำนวน 1 เครื่อง สำหรับพัฒนาและประมวลผลข้อมูล
2. ซอฟต์แวร์และระบบปฏิบัติการ
 - 1) ระบบปฏิบัติการ Windows 11
 - 2) Google Colab (สำหรับพัฒนาและรันโค้ด Python)
3. ภาษาที่ใช้ในการพัฒนาและวิเคราะห์ข้อมูล Python (เป็นภาษาหลักในการพัฒนาและวิเคราะห์ข้อมูล)
4. ไบเบรารีและเฟรมเวิร์กที่ใช้ในโครงการ
 - 1) Pandas (สำหรับการจัดการข้อมูลตารางและกระบวนการ ETL)

- 2) NumPy (สำหรับการคำนวณเชิงตัวเลข)
 - 3) RapidFuzz (สำหรับการเปรียบเทียบข้อความและ normalize ชื่อเชื้อ/สิ่งส่งตรวจ)
 - 4) Regex (Regular Expression) – สำหรับการจัดการข้อความ เช่น ลบ noise, ดึงข้อมูลจาก comment
 - 5) OpenPyXL (สำหรับการจัดการและบันทึกข้อมูลในไฟล์ Excel พร้อมปรับสี/รูปแบบ)
 - 6) ipywidgets (Filter UI) (สำหรับสร้าง UI กรองข้อมูล (Sex, Age, Specimen, Organism, Year) และ export Excel)
 - 7) scikit-learn (Random Forest) (สำหรับการจำแนกผล S/I/R และสร้าง confusion matrix/classification report)
 - 8) Matplotlib (สำหรับการสร้างกราฟพื้นฐาน เช่น line plot, bar chart, scatter plot)
 - 9) Seaborn (สำหรับการสร้างกราฟเชิงสถิติ เช่น heatmap, histogram)
5. ฐานข้อมูล ไฟล์ข้อมูล .CSV (ใช้สำหรับการเก็บข้อมูลเบื้องต้นจากการเพาะเชื้อและการทดสอบยาปฏิชีวนะ)

1.4 วิธีการศึกษา

1. ศึกษาโครงสร้างข้อมูลและความสัมพันธ์ระหว่างแฟ้มต่าง ๆ ที่เกี่ยวข้องกับผู้ป่วยในและผู้ป่วยนอก (IPD และ OPD) โดยอ้างอิงจากเอกสาร Data Dictionary เพื่อทำความเข้าใจในรูปแบบข้อมูลและชนิดของข้อมูลที่จะใช้ในการวิเคราะห์วิเคราะห์ปัญหาด้านคุณภาพข้อมูลในแต่ละแฟ้ม เช่น ความไม่ครบถ้วน ความไม่ถูกต้อง ความไม่สอดคล้อง และข้อมูลซ้ำซ้อน
2. วิเคราะห์ปัญหาเกี่ยวกับคุณภาพของข้อมูลในแต่ละแฟ้ม เช่น ความไม่ครบถ้วนของข้อมูล ข้อมูลที่ผิดรูปแบบหรือไม่เป็นปัจจุบัน ความไม่สอดคล้องกันระหว่างแฟ้มข้อมูล และปัญหาการบันทึก

ข้อมูลซ้ำซ้อน เพื่อเตรียมความพร้อมก่อนเข้าสู่กระบวนการตรวจสอบคุณภาพออกแบบและพัฒนากระบวนการ ETL (Extract, Transform, Load) เพื่อรวบรวม แปลง และจัดเตรียมข้อมูลจากหลายแฟ้มให้พร้อมสำหรับการวิเคราะห์

3. พัฒนาโค้ดสำหรับตรวจสอบคุณภาพข้อมูลโดยใช้ภาษา Python และ SQL ตามเกณฑ์การประเมินคุณภาพข้อมูล 2 โดเมน ได้แก่

- 1) ความสอดคล้องกันของข้อมูล (Consistency)
- 2) ความครบถ้วนของข้อมูล (Completeness)

4. ออกแบบและพัฒนากระบวนการ ETL (Extract, Transform, Load) เพื่อรวมข้อมูลจากหลายแฟ้ม เช่น ข้อมูลผู้ป่วย ข้อมูลการวินิจฉัย ข้อมูลการเพาะเชื้อ และผลตรวจทางห้องปฏิบัติการ โดยทำการแปลงชนิดข้อมูลให้เหมาะสม และจัดการข้อมูลให้อยู่ในรูปแบบที่สามารถนำไปวิเคราะห์ต่อได้สรุปผลการดำเนินงาน จัดทำรายงานโครงการ และเตรียมเอกสารนำเสนอสำหรับส่งมหาวิทยาลัยและสถานประกอบการ

5. ทดสอบระบบที่พัฒนาขึ้นกับข้อมูลจริง โดยประเมินทั้งด้านประสิทธิภาพ ความถูกต้องของการแสดงผล และความเหมาะสมของรูปแบบการใช้งาน พร้อมรับฟังความคิดเห็นจากผู้ใช้หรือผู้ควบคุมงาน เพื่อนำไปปรับปรุงระบบให้ตอบโจทย์การใช้งานจริง

6. สรุปผลการดำเนินงาน วิเคราะห์ปัญหา อุปสรรคที่พบ และแนวทางในการพัฒนาต่อย่อดระบบในอนาคต

1.5 ขอบเขตของโครงการ

1. การนำเสนอผลผ่านไฟล์ Excel

โครงการนี้พัฒนาเครื่องมือที่สามารถคำนวณ แอนติไบโอแกรม (Antibiogram) และนำเสนอผลในรูปแบบ Excel โดยมีรายละเอียดดังนี้

1. ผลการคำนวณ แอนติไบโอแกรม (Antibiogram) จะถูกส่งออกในรูปแบบไฟล์ Excel โดยใช้ pandas และ Openpyxl สำหรับการจัดการและบันทึกข้อมูล
2. ข้อมูลในไฟล์ Excel จะประกอบด้วย:

- Organism (เชื้อเชื้อ)
- Antibiotics (ยาปฏิชีวนะ)
- Susceptibility Rate (เปอร์เซ็นต์ความไว)
- Number of Isolates (จำนวนตัวอย่างที่ทดสอบ)
- Specimen Type (ประเภทสิ่งส่งตรวจ เช่น ปัสสาวะ, เลือด, นหนอง)

3. ข้อมูลใน Excel จะถูกแยกตามประเภทของ Specimen เช่น Urine, Blood, Sputum, และ Pus ซึ่งสามารถกรองข้อมูลได้ตามความต้องการ
4. ระบบจะส่งออก Log ของการประเมินคุณภาพข้อมูล เพื่อการตรวจสอบย้อนหลัง

2. ระบบประเมินคุณภาพข้อมูล

ระบบจะทำการประเมินคุณภาพข้อมูลจากแหล่งข้อมูลที่น่าเชื่อถือในโครงการนี้ โดยจะตรวจสอบโดย 2 โดเมนความสอดคล้องกันของข้อมูล (Consistency) และความครบถ้วนของข้อมูล (Completeness)

3. ระบบแปรรูปข้อมูล (ETL)

โครงการนี้พัฒนาระบบที่สามารถแปรรูปข้อมูลจากแหล่งข้อมูลหลายประเภทได้ตามกระบวนการ ETL ดังนี้

1. รองรับการนำเข้าข้อมูลจากไฟล์ CSV/Excel
2. แปลงชนิดข้อมูล ให้ตรงกับ Data Dictionary ที่กำหนดไว้ เพื่อให้ข้อมูลมีความถูกต้องและสอดคล้อง
3. รวมข้อมูลจากหลายแฟ้ม เช่น ข้อมูลผู้ป่วย ข้อมูลแล็บ ข้อมูลการเพาะเชื้อ
4. ทำความสะอาดข้อมูล (Data Cleaning) ก่อนนำไปแสดงผลในไฟล์ Excel เพื่อให้ได้ข้อมูลที่มีความน่าเชื่อถือ

5. ส่งออกข้อมูลพร้อมใช้งาน ให้ Excel โดยอัตโนมัติ ซึ่งจะถูกนำไปใช้ในการตัดสินใจหรือการวิเคราะห์ต่อไป

1.6 ผลที่คาดว่าจะได้รับ

1. บุคลากรทางการแพทย์สามารถตรวจสอบและประเมินข้อมูลได้อย่างเป็นระบบซึ่งจะช่วยลดข้อผิดพลาดในการบันทึกข้อมูลและเพิ่มความเชื่อมั่นในการนำข้อมูลไปใช้งานต่อ
2. บุคลากรทางการแพทย์สามารถเข้าถึงข้อมูลสำคัญได้สะดวกและ เข้าใจง่ายผ่าน ไฟล์ Excel ที่ส่งออกจากระบบ โดยสามารถกรองข้อมูลได้จากการใช้ Excel (เช่น Filter/Sort) ซึ่งช่วยให้ผู้ใช้งานสามารถติดตามแนวโน้มการดื้อยา การกระจายตัวของเชื้อ และผลการเพาะเชื้อได้อย่างมีประสิทธิภาพ ช่วยเสริมสร้างการใช้ข้อมูลในการวางแผนและตัดสินใจด้านบริการสุขภาพ
3. ส่งเสริมการใช้ข้อมูลที่มีคุณภาพในการวางแผนและตัดสินใจด้านบริการสุขภาพโดยข้อมูลที่ผ่านการตรวจสอบและจัดการอย่างเหมาะสมจะสามารถนำมาใช้ควบคุมการใช้จ่ายปฏิชีวนะ

1.7 สถานที่ดำเนินงาน

1. สถานที่วิเคราะห์และออกแบบระบบ และทดลองใช้งาน

ศูนย์วิจัยสุขภาพ (Global Health Research Center)

คณะแพทยศาสตร์ มหาวิทยาลัยเชียงใหม่

อาคารราชนครินทร์ ชั้น 8 ถนนอินทวิโรด ตำบลศรีภูมิ อำเภอเมือง จังหวัดเชียงใหม่

2. สถานที่พัฒนาระบบ

ศูนย์วิจัยสุขภาพ (Global Health Research Center)

คณะแพทยศาสตร์ มหาวิทยาลัยเชียงใหม่

อาคารราชนครินทร์ ชั้น 8 ถนนอินทวิโรด ตำบลศรีภูมิ อำเภอเมือง จังหวัดเชียงใหม่

1.8 ระยะเวลาการดำเนินงาน

การพัฒนาโครงการ เรื่อง การวิเคราะห์คุณภาพข้อมูลและพัฒนาแดชบอร์ดสำหรับข้อมูลผู้ป่วยมีระยะเวลาดำเนินงานระหว่างวันที่ 23 มิถุนายน พ.ศ. 2568 ถึงวันที่ 31 ตุลาคม พ.ศ. 2568 โดยมีลำดับกิจกรรมดังนี้

ตารางที่ 1.1 ตารางแสดงระยะเวลาการดำเนินงาน

ลำดับ	กิจกรรม	พ.ศ. 2568				
		มิ.ย.	ก.ค.	ส.ค.	ก.ย.	ต.ค.
1	แนะนำเบื้องต้น เกี่ยวกับเทคโนโลยีต่าง ๆ สำหรับใช้ในโครงการ	←→				
2	ประชุมทำความเข้าใจรายละเอียดของโปรเจค และรับมอบหมายงาน		←→			
3	พัฒนาระบบตามส่วนที่ได้รับมอบหมาย			←→		
4	ทดสอบและแก้ไขเว็บ				←→	
5	จัดทำเอกสารที่ต้องใช้ในการประกอบการใช้งานให้แก่ทางบริษัทและทางมหาวิทยาลัย					←→

บทที่ 2

การศึกษาค้นคว้าและงานที่เกี่ยวข้อง

การพัฒนาโครงงานเรื่อง การวิเคราะห์คุณภาพข้อมูลและการจัดทำแดชบอร์ดสำหรับแสดงข้อมูลความไวของเชื้อต่อยาปฏิชีวนะ ได้ทำการศึกษาค้นคว้าองค์ความรู้ และงานที่เกี่ยวข้องกับการจัดการข้อมูลทางโรงพยาบาล และศึกษาแนวทาง เทคนิค เครื่องมือ และงานวิจัยที่เกี่ยวข้องกับการวิเคราะห์คุณภาพข้อมูล, กระบวนการ ETL(Extract, Transform, Load), ตาราง แอนติไบโอแกรม (Antibiogram) และ ศึกษาโครงสร้างระบบงานเดิมที่มีอยู่ก่อน เพื่อทำความเข้าใจระบบงานเดิม และสามารถพัฒนาระบบได้อย่างถูกต้องและมีประสิทธิภาพ

2.1 ระบบข้อมูลผู้ป่วย (IPD/OPD)

ระบบเดิมข้อมูลผู้ป่วยภายในโรงพยาบาลมีลักษณะการจัดเก็บข้อมูลหลายประเภท แยกเป็นแฟ้มต่าง ๆ โดยแบ่งเป็นสองกลุ่มใหญ่ ได้แก่: IPD (Inpatient Department) ข้อมูลผู้ป่วยใน เช่น อายุ เพศ สิ่งส่งตรวจ วันที่นอนโรงพยาบาล ผลตรวจทางห้องปฏิบัติการ การให้ยา และผลเพาะเชื้อ OPD (Outpatient Department) ข้อมูลผู้ป่วยนอก เช่น ประวัติการเข้ารับบริการ ผลการตรวจพื้นฐาน และบางกรณีมีการเพาะเชื้อร่วม ในปัจจุบัน ระบบข้อมูลเหล่านี้มักแยกเก็บโดยใช้ฐานข้อมูลหรือแฟ้มที่อยู่คนละระบบและมีโครงสร้างต่างกัน ทำให้การวิเคราะห์แบบรวมทั้งระบบทำได้ยากและเสี่ยงต่อความผิดพลาด

2.2 การประเมินคุณภาพข้อมูล

การประเมินคุณภาพของข้อมูล (Data Quality Assessment) เป็นขั้นตอนที่สำคัญในการรับประกันความถูกต้องของผลการวิเคราะห์และความน่าเชื่อถือของข้อมูล แอนติไบโอแกรม (Antibiogram) ซึ่งเป็นส่วนสำคัญของโครงงานนี้ ข้อมูลที่ใช้ในการวิเคราะห์ เช่น ข้อมูลผู้ป่วย ข้อมูลผลแล็บ และผลการเพาะเชื้อ เป็นข้อมูลที่ต้องมีความแม่นยำสูง หากเกิดข้อผิดพลาดแม้เพียงเล็กน้อย เช่น การบันทึกข้อมูลซ้ำซ้อน หรือค่าที่หายไป อาจนำไปสู่การวิเคราะห์ที่ผิดพลาด ซึ่งส่งผลโดยตรงต่อการตัดสินใจทางการแพทย์

การตรวจสอบคุณภาพข้อมูลในโครงงานนี้เน้นที่การประเมิน Consistency และ Completeness ของข้อมูล ซึ่งจะช่วยให้ได้ผลการทดสอบความไวของเชื้อจุลินทรีย์ต่อด้านยาปฏิชีวนะที่แม่นยำ และสามารถนำไปใช้ในการตัดสินใจทางการแพทย์ได้อย่างมีประสิทธิภาพ

2.2.1 การประเมินคุณภาพข้อมูล

ในโครงการนี้ การประเมินคุณภาพข้อมูลจะอิงตามกรอบแนวคิดจาก DAMA-UK (Data Management Association - UK Chapter) ซึ่งเป็นมาตรฐานสากลที่ได้รับการยอมรับในด้านการจัดการข้อมูล โดยโครงการนี้เลือกใช้เพียง 2 ด้านจากกรอบ DAMA-UK ดังนี้:

1. ความสอดคล้อง (Consistency) การตรวจสอบความเชื่อมโยงของข้อมูลข้ามแฟ้ม เช่น วันที่รับผู้ป่วยควรเกิดขึ้นก่อนวันที่เก็บสิ่งส่งตรวจ เพื่อให้ข้อมูลที่เชื่อมโยงกันมีความสอดคล้องและสามารถใช้งานได้
2. ความครบถ้วน (Completeness) การตรวจสอบว่าฟิลด์สำคัญ เช่น ชนิดเชื้อ, รหัสยา หรือผลการทดสอบ (S/I/R) มีการกรอกข้อมูลครบถ้วนหรือไม่ หากพบว่าข้อมูลหาย (Missing) หรือว่างเปล่า ระบบจะไม่สามารถนำข้อมูลนั้นมาคำนวณหรือแสดงผลได้

การนำแนวคิดการประเมินคุณภาพข้อมูลเหล่านี้มาประยุกต์ใช้กับระบบข้อมูลทางโรงพยาบาลจะช่วยยกระดับมาตรฐานการจัดการข้อมูลในโรงพยาบาลให้มีความน่าเชื่อถือสูงขึ้น และเตรียมความพร้อมสำหรับการวิเคราะห์ข้อมูลในระดับที่ซับซ้อนมากขึ้นในอนาคต

2.3 กระบวนการ ETL (Extract, Transform, Load)

กระบวนการ ETL เป็นขั้นตอนสำคัญในการจัดเตรียมข้อมูลให้พร้อมใช้งาน โดยเฉพาะการนำข้อมูลที่ได้จากหลายแหล่งมาใช้ในการคำนวณและสร้าง Antibioqram ในโครงการนี้ โดยกระบวนการ ETL แบ่งออกเป็น 3 ขั้นตอนหลัก ได้แก่

1. การนำเข้า (Extract)

ข้อมูลจะถูกดึงมาจากหลายแหล่ง เช่น

1. แฟ้มข้อมูลผู้ป่วย
2. ผลการตรวจแลบ
3. ผลการเพาะเชื้อ
4. รายการยาปฏิชีวนะ

ข้อมูลเหล่านี้จะถูกนำเข้ามาในระบบเพื่อใช้ในการประมวลผลและคำนวณค่าที่จำเป็นสำหรับการสร้าง แอนติไบโอแกรม (Antibiogram)

2. การแปลงข้อมูล (Transform)

ข้อมูลจะได้รับการจัดระเบียบและแปลงให้มีความสอดคล้องและเหมาะสม เช่น

1. การแก้ไขชื่อยาและชื่อเชื้อให้อยู่ในรูปแบบเดียวกัน
2. ตรวจสอบความครบถ้วนของข้อมูลสำคัญ เช่น รหัสผู้ป่วย ชนิดสิ่งส่งตรวจ ผลตรวจ และผลการเพาะเชื้อ
3. สร้างข้อมูลเพิ่มเติม เช่น การคำนวณ เปอร์เซ็นต์ความไวต่อยา (Susceptibility Rate) และการจับคู่ชื่อยาและเชื้อ เพื่อให้ข้อมูลพร้อมสำหรับการแสดงผลและการคำนวณ

3. การโหลดข้อมูล (Load)

ข้อมูลที่ผ่านการแปลงและเตรียมไว้จะถูกโหลดเข้าสู่ระบบ Excel หรือ ฐานข้อมูล และสามารถแสดงผลในรายงาน แบบอัตโนมัติ โดยจัดข้อมูลให้สามารถกรองและเลือกดูได้ตามความต้องการ เช่น

1. รายงานตามช่วงเวลา (รายเดือน, รายไตรมาส)
2. รายงานแยกตามประเภทของสิ่งส่งตรวจ

บทที่ 3

เทคโนโลยีที่ใช้ในการศึกษา

ระบบที่พัฒนาขึ้นในโครงการนี้มีลักษณะเป็นระบบที่ใช้ในการ จัดการข้อมูลการเพาะเชื้อและการทดสอบยาปฏิชีวนะ โดยใช้ข้อมูลจากระบบสารสนเทศของโรงพยาบาล ซึ่งมีปริมาณข้อมูลจำนวนมากและความซับซ้อน ทั้งในแง่ของแหล่งข้อมูล ความหลากหลายของเชื้อและยา และข้อกำหนดด้านการใช้งานจริงในโรงพยาบาล

เป้าหมายผู้ใช้งานระบบ ได้แก่ แพทย์ และบุคลากรสาธารณสุขที่เกี่ยวข้อง โดยระบบช่วยให้สามารถเรียกดูแนวโน้มความไวของเชื้อในช่วงเวลาต่างๆ จำแนกตามเพศ อายุ ชนิดสิ่งส่งตรวจ หรือแผนกที่ส่งตรวจ และแสดงผลในรูปแบบ ไฟล์ Excel ที่มีการเน้นสีเพื่อสะท้อนระดับความไว (เช่น เชื้อไวสูง, แดง = ดื้อยา)

ระบบนี้จึงมีศักยภาพในการนำไปประยุกต์ใช้ในโรงพยาบาลเพื่อสนับสนุนการตัดสินใจเชิงนโยบายด้านการใช้ยาปฏิชีวนะอย่างมีประสิทธิภาพ

3.1 Python

เป็นภาษาหลักในการพัฒนาโค้ดในโครงการนี้ เนื่องจากมีไลบรารีมากมายที่ช่วยในการประมวลผลข้อมูลจากแหล่งต่างๆ และเหมาะสำหรับการจัดการข้อมูลทางคลินิก สามารถเชื่อมต่อกับเครื่องมืออื่นๆ ได้อย่างสะดวก เข้ามาช่วยเป็นเครื่องมือหลักที่ใช้สร้าง Data Pipeline สำหรับการวิเคราะห์คุณภาพข้อมูลและการจัดทำ Antibioqram โดย Python ถูกนำมาใช้ตั้งแต่ขั้นตอนการนำเข้าข้อมูลผู้ป่วยและผลเพาะเชื้อ การทำความสะอาดข้อมูลที่มีความซับซ้อน การตรวจสอบคุณภาพข้อมูล ไปจนถึงการคำนวณเปอร์เซ็นต์ความไวของเชื้อแบคทีเรียต่อยาปฏิชีวนะ และการนำเสนอผลลัพธ์ในรูปแบบที่แพทย์สามารถนำไปใช้ได้จริง

- ช่วยในขั้นตอนการทำความสะอาดข้อมูล Python ถูกใช้ร่วมกับไลบรารี pandas และ rapidfuzz เพื่อจัดการข้อมูลที่ซ้ำซ้อนและ normalize ชื่อเชื้อให้เป็นมาตรฐานเดียวกัน

- ใช้ร่วมกับ scikit-learn เพื่อสร้าง confusion matrix และ classification report ที่ช่วยตรวจสอบความถูกต้องของ pipeline

คุณลักษณะของ Python

- ความหลากหลายของไลบรารีที่รองรับงานด้านข้อมูล
- ทำงานได้หลายแพลตฟอร์ม
- สามารถนำไปใช้ได้ทั้งงานวิจัยทางการแพทย์ งานวิเคราะห์ข้อมูล

Step1.3 อ่านไฟล์ IPD_Data, IPD_Cul, OPD_Cul

- อ่าน IPD_Data.txt ด้วยตัวค้นแบบ (sep="\\t") และ encoding=ENC_IPD_DATA จากนั้น normalize ชื่อคอลัมน์ และคิดเฉพาะคอลัมน์สำคัญไว้สำหรับ enrich (เช่น เพศ อายุ วัคซีน/วันที่)
- อ่าน IPD_Cul02.txt (หรือชื่อไฟล์ที่ใช้งานจริง) ด้วย encoding=ENC_CULTURE, ปรับชื่อคอลัมน์ และคิดคอลัมน์ที่จำเป็นในการวิเคราะห์ (เช่น in/han/vn, org, anti/anti_name, anti_result, anti_mic, date_r, time_r, lst/rep เป็นต้น)
- อ่านไฟล์ OPD culture และปรับชื่อคอลัมน์ให้สอดคล้องกับ IPD (รวมถึง vn) เพื่อเตรียมรวมเป็นฐานเดียวกัน

```
def _read_ipd_data(path, enc):
    try:
        df = pd.read_csv(path, sep="\\t", encoding=enc, low_memory=False)
        if df.shape[1] == 1:
            df = pd.read_csv(path, sep="\\t", encoding=enc, low_memory=False)
        except Exception:
            df = pd.read_csv(path, sep="\\t", encoding=enc, low_memory=False)
        df = normalize_columns(df)

        # Cleaning ข้อมูล
        if 'sex' in df.columns:
            df['sex'] = df['sex'].str.strip().replace({'ชาย': 'Male', 'หญิง': 'Female'})
        if 'age' in df.columns:
            df['age'] = pd.to_numeric(df['age'], errors='coerce')
        return df

def _read_culture(path, enc):
    try:
        df = pd.read_csv(path, sep="\\t", encoding=enc, low_memory=False)
        if df.shape[1] == 1:
            df = pd.read_csv(path, sep="\\t", encoding=enc, low_memory=False)
        except Exception:
            df = pd.read_csv(path, sep="\\t", encoding=enc, low_memory=False)
        df = normalize_columns(df)

        # Cleaning ข้อมูล
        if 'anti_result' in df.columns:
            df['anti_result'] = (
                df['anti_result']
                .astype(str).str.strip().str.upper()
                .replace({
                    'NaN': np.nan, # ตัวว่าง
                    'nan': np.nan, # ตัวเล็ก
                    'NOME': np.nan, # ตัวใหญ่
                    '': np.nan, # ตัวว่าง
                })
            )

        if 'date_r' in df.columns:
            df['date_r'] = pd.to_datetime(df['date_r'], errors='coerce')
        if 'time_r' in df.columns:
            s = df['time_r'].astype(str).str.strip()
            df['time_r'] = pd.to_datetime(s, errors='coerce').dt.strftime('%H:%M:%S')
        return df

# อ่านและ clean data
df_ipd_data = _read_ipd_data(BASE_PATH/IPD_Data.txt", ENC_IPD_DATA)
df_ipd_cul = _read_culture(BASE_PATH/IPD_Cul02.txt", ENC_CULTURE)
df_opd_cul = _read_culture(BASE_PATH/OPD_Cul.txt", ENC_CULTURE)
```

รูปที่ 3.1 ตัวอย่างโค้ด อ่านไฟล์ IPD_Data, IPD_Cul, OPD_Cul

3.2 Pandas และ NumPy

เป็นไลบรารีหลักของภาษา Python เพื่อทำงานด้านการจัดการข้อมูลและการคำนวณเชิงตัวเลข โดย NumPy (Numerical Python) เป็นไลบรารีที่ออกแบบมาเพื่อการประมวลผลข้อมูลเชิงตัวเลขและการจัดการอาร์เรย์หลายมิติ (multi-dimensional arrays) ซึ่งมีประสิทธิภาพสูงและสามารถทำงานได้รวดเร็ว Pandas เป็นไลบรารีที่สร้างขึ้นบนพื้นฐานของ NumPy โดยเพิ่มความสามารถในการจัดการข้อมูลในรูปแบบตาราง (DataFrame) ทำให้สามารถอ่าน เขียน และประมวลผลข้อมูลได้อย่างสะดวกและมีโครงสร้างที่เข้าใจง่าย

การนำเข้าข้อมูล (Import Data) Pandas ถูกใช้ในการอ่านไฟล์ข้อมูลผู้ป่วย (IPD/OPD) และผลเพาะเชื้อจากหลายแหล่ง โดยสามารถรองรับไฟล์ที่มีรูปแบบแตกต่างกัน เช่น .txt, .csv หรือ .xlsx

การทำความสะอาดข้อมูล (Data Cleaning): Pandas ช่วยในการจัดการข้อมูลที่ซ้ำซ้อน การแทนค่าที่หายไป (missing values) และการปรับชื่อคอลัมน์ให้เป็นมาตรฐานเดียวกัน ขณะที่ NumPy ถูกใช้ในการตรวจสอบค่าตัวเลข เช่น อายุของผู้ป่วยที่อยู่นอกช่วงที่สมเหตุสมผล (เช่น <1 หรือ >120 ปี) และแทนค่าดังกล่าวด้วย NaN

การรวมข้อมูล (Data Integration) Pandas ถูกใช้ในการรวมข้อมูลจากหลายตาราง เช่น การ merge ข้อมูลผู้ป่วยกับผลเพาะเชื้อ เพื่อสร้างฐานข้อมูลกลางที่พร้อมสำหรับการวิเคราะห์

การคำนวณเชิงสถิติ (Statistical Calculation) NumPy ถูกใช้ในการคำนวณค่าทางสถิติ เช่น การหาค่าเปอร์เซ็นต์ความไว (%S) ของเชื้อแบคทีเรียต่อยาปฏิชีวนะ โดยสามารถจัดการข้อมูลขนาดใหญ่ได้อย่างมีประสิทธิภาพ

การตรวจสอบคุณภาพข้อมูล (Data Quality Assessment) Pandas และ NumPy ถูกใช้ร่วมกันในการสร้างรายงานตรวจสอบคุณภาพข้อมูล เช่น จำนวนข้อมูลที่ซ้ำซ้อน, จำนวนข้อมูลที่หายไป และการตรวจสอบความสอดคล้องของข้อมูล

คุณลักษณะของ Pandas และ NumPy

Pandas

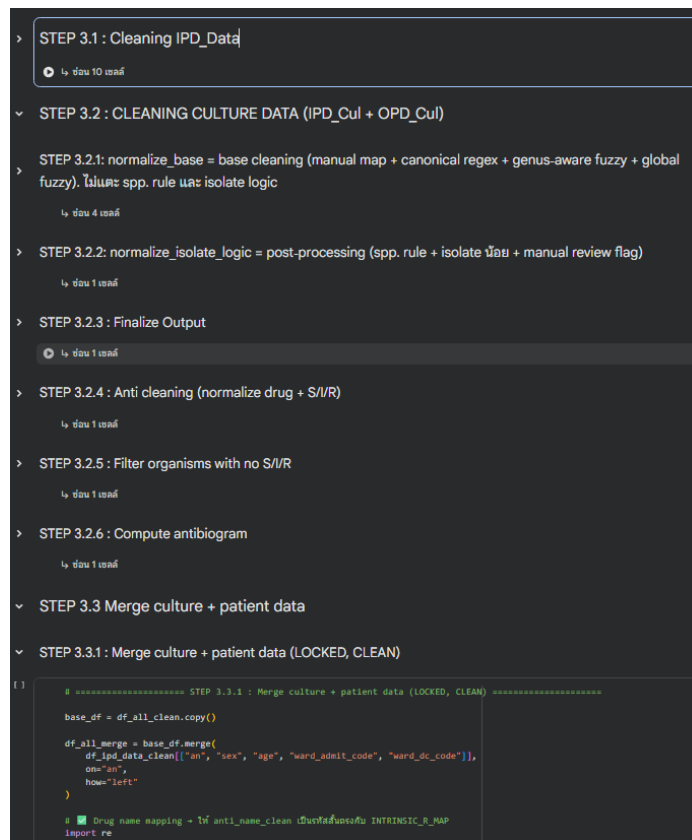
- มีโครงสร้างข้อมูลที่ชัดเจน เช่น Series และ DataFrame
- รองรับการอ่านและเขียนไฟล์ข้อมูลจากหลายรูปแบบ
- มีฟังก์ชันสำหรับการจัดการข้อมูล เช่น การกรอง (filter) และ การรวม (merge)
- เหมาะสำหรับการทำงานกับข้อมูลเชิงตารางที่ซับซ้อน

NumPy

- รองรับการทำงานกับอาร์เรย์หลายมิติที่มีประสิทธิภาพสูง
- มีฟังก์ชันทางคณิตศาสตร์และสถิติที่หลากหลาย
- ทำงานได้รวดเร็วและใช้หน่วยความจำน้อยเมื่อเทียบกับโครงสร้างข้อมูลพื้นฐานของ

Python

- เหมาะสำหรับการคำนวณเชิงตัวเลขและการวิเคราะห์ข้อมูลขนาดใหญ่



รูปที่ 3.2 ตัวอย่างโค้ด การรวมข้อมูลผู้ป่วยกับผลเพาะเชื้อ โดยใช้ Pandas เป็นตัวกลาง

3.3 Rapid Fuzz

เป็นไลบรารีของภาษา Python ที่ทำงานด้านการเปรียบเทียบข้อความ (string matching) โดยเฉพาะในกรณีที่ข้อมูลมีความคลาดเคลื่อน เช่น การสะกดผิด การใช้ตัวอักษรพิเศษ หรือการมีช่องว่างและสัญลักษณ์ที่ไม่สม่ำเสมอ Rapid Fuzz ออกแบบมาให้มีประสิทธิภาพสูงกว่าไลบรารีรุ่นก่อนหน้าอย่าง fuzzy Wuzzy ทั้งในด้านความเร็วและการใช้หน่วยความจำ ทำให้สามารถนำไปใช้กับข้อมูลขนาดใหญ่ได้อย่างมีประสิทธิภาพ และสามารถนำมาปรับใช้ในงาน ทำความสะอาดข้อมูลเชื้อ (organism) และสิ่งส่งตรวจ (specimen) โดยถูกใช้เพื่อ normalize ชื่อเชื้อและสิ่งส่งตรวจให้เป็นมาตรฐานเดียวกัน เช่น ข้อมูลที่ได้จากห้องปฏิบัติการอาจมีการบันทึกชื่อเชื้อที่แตกต่างกันไป เช่น “E. coli”, “Escherichia coli”, หรือ “Escherichia Coli+” ซึ่งหากไม่ทำการ normalize จะทำให้การวิเคราะห์ข้อมูลผิดพลาด Rapid Fuzz จึงถูกนำมาใช้ในการเปรียบเทียบข้อความและจับคู่กับชื่อมาตรฐานที่กำหนดไว้ใน MAP_TO_STANDARD เพื่อให้ข้อมูลทั้งหมดถูกจัดให้อยู่ในรูปแบบเดียวกัน อีกทั้งยังสามารถใช้ในการตรวจสอบและจัดการ

ข้อมูลที่มีการสะกดผิดหรือมี noise จากการแปลง encoding เช่น “acinetobacte”, “acinetobacter bau+++++++mannii” ซึ่งเป็นผลจากการอ่านไฟล์เก่าหรือไฟล์ที่มีปัญหาด้าน encoding Rapid Fuzz สามารถช่วยให้ข้อมูลเหล่านี้ถูกจัดให้อยู่ในรูปแบบที่ถูกต้อง และพร้อมสำหรับการวิเคราะห์อย่างมีประสิทธิภาพ

คุณลักษณะของ Rapid Fuzz

- มีประสิทธิภาพสูงในการทำงานกับข้อมูลขนาดใหญ่
- รองรับการเปรียบเทียบข้อความที่มีความคลาดเคลื่อน เช่น การสะกดผิดหรือการใช้สัญลักษณ์ที่ไม่สม่ำเสมอ

```
from rapidfuzz import fuzz, process

SPECIMEN_MAP = {}

# Blood sources
"femoral dlc": "Blood",
"lt femoral dlc": "Blood",
"rt femoral dlc": "Blood",
"jugular vein": "Blood",
"from jugular vein": "Blood",
"peripheral vein": "Blood",
"picc": "Blood",
"pv1": "Blood",
"pv2": "Blood",

# Urine sources
"rt pcn": "Urine",
"lt pcn": "Urine",
"pcn": "Urine",
"foley cath": "Urine",
"foley's cath": "Urine",
"folet": "Urine",
"nephrostomy": "Urine",
"nephrost": "Urine",
"pouch": "Urine", # ถ้าต้องการให้เป็น wound นอกได้

# Pus / wound / tissue
"pus": "Pus",
"pus from": "Pus",
"pus from rt.ear": "Pus",
"pus from peg stoma": "Pus",
"pressure sore": "Wound",
"bed sore": "Wound",
"pressure sore rt buttock": "Wound",
"tissue from pressure sore": "Tissue",
"rt toe": "Wound",
"lt drain": "Drain Fluid",
"drain": "Drain Fluid",
"rt drain": "Drain Fluid",
"submand area": "Pus",
"coloplast": "Wound",

# Eye
"from eye": "Eye Discharge",
"discharge rt eye": "Eye Discharge",

# Bile
"ptbd": "Bile",

# Genital
"penis": "Genital Swab",

# Default fallback
"stoma": "Wound",
```

รูปที่ 3.3 ตัวอย่างโค้ด from rapid fuzz import fuzz

3.4 Openpyxl

เป็นไลบรารีของภาษา Python ใช้สำหรับการสร้างและแก้ไขไฟล์ Excel (.xlsx) โดยตรง สามารถจัดการกับข้อมูลในรูปแบบตารางได้ การเขียนข้อมูล การปรับแต่งรูปแบบเซลล์ การกำหนดสีพื้นหลัง การจัดตำแหน่ง

- สร้างไฟล์ Excel หลายชีต (Multi-sheet Export)
- สร้างไฟล์ Excel ที่แบ่งตาม ชนิดสิ่งส่งตรวจ (Specimen)

คุณลักษณะของ OpenPyXL

- รองรับการสร้างและแก้ไขไฟล์ Excel ได้โดยตรง
- สามารถปรับแต่งรูปแบบเซลล์ได้ละเอียด เช่น สี, ฟอนต์, การจัดวางข้อความ
- รองรับการสร้างหลายชีตในไฟล์เดียว

Organism	TOTAL	/Clavulanic	mpicillin (AMftazidime (C/efazolin (CF2rofloxacin (Colistin (COL ftaroline (CFtrioxone (CFfotaxime (CTripenem (DC					
Achromobacter Spp.	69		69.6 (48/69)		4.3 (3/69)		1.5 (1/65)	
Acinetobacter Baumannii Comple	7992		23.1 (1848/7983)		24.2 (1923/7960)	99.5 (1013/1018)		
Acinetobacter Spp.	93		78.5 (73/93)		91.4 (85/93)	-		
Aeromonas Hydrophila	397	11.2 (43/385)	89.2 (297/333)	9.7 (6/62)	89.4 (353/395)		71.2 (37/52)	88.0 (293/333)
Bacteroides Spp.	1	-	-		-		-	-
Burkholderia Cepacia Complex	1069		94.7 (1012/1069)		-		-	-
Burkholderia Pseudomallei	65	-	100.0 (65/65)		-		-	-
Chryseobacterium Indologenes	9		-		-		-	-
Citrobacter Spp.	742	14.8 (86/580)	75.9 (488/643)	23.8 (15/63)	84.6 (602/712)		70.6 (48/68)	75.6 (487/644)
Cronobacter Dublinensis	2	-	-	-	-		-	-
Elizabethkingia Anophelis	22		-	-	-		-	-
Enterobacter Spp.	2851	0.6 (15/2556)	-	57.7 (1373/2379)	1.0 (2/204)	77.1 (2004/2600)	51.8 (115/222)	56.9 (1355/2381)
Enterococcus Faecalis	4727		99.0 (4677/4724)		42.1 (130/309)			90.0 (198/220)
Enterococcus Faecium	8143		5.0 (206/4132)		1.4 (5/364)			
Enterococcus Spp.	324		76.9 (249/324)		-			
Escherichia Coli	15399	69.9 (9966/14254)	14.1 (161/1141)	60.3 (7850/13013)	44.3 (497/1123)	42.6 (6060/14212)	49.7 (593/1193)	53.1 (6918/13023)
Gram Negative Bacilli	7	-	-		-			93.1 (1139/1224)
Haemophilus Influenzae	805	96.9 (775/800)	71.2 (571/802)		97.1 (776/799)		99.6 (799/802)	
Klebsiella Aerogenes	697	0.8 (5/660)		70.9 (450/635)	3.0 (1/33)	89.7 (600/669)	53.8 (21/39)	69.0 (438/635)
Klebsiella Oxytoca	309	94.1 (273/290)	R	93.7 (251/268)	-	91.7 (264/288)	-	88.6 (31/35)
Klebsiella Pneumoniae	12213	57.4 (5081/8847)	R	55.0 (4349/7908)	36.5 (302/827)	55.8 (4925/8821)	42.8 (382/893)	53.7 (4245/7912)
Leclercia Adecarboxylata	4	-	-	-	-	-		53.0 (670/1263)
Moraxella Catarrhalis	271	100.0 (35/35)		-		-	100.0 (35/35)	-
Moraznella Morganii	781	0.6 (5/773)	1.8 (1/56)	82.1 (1.8 (1/56)	75.6		67.8 (40/59)	79.6

รูปที่ 3.4 ตัวอย่าง Excel Antibigram

3.5 Scikit-learn

เป็นไลบรารีมาตรฐานของภาษา Python พัฒนาขึ้นเพื่อรองรับงานด้าน Machine Learning และการวิเคราะห์ข้อมูลเชิงสถิติ โดยมีฟังก์ชันสำเร็จรูปที่ครอบคลุมทั้งการจำแนกประเภท (classification), การถดถอย (regression), การจัดกลุ่ม (clustering), และการประเมินผล (evaluation metrics) โดย Scikit-learn ถูกใช้สำหรับ Random Forest Classification เพื่อจำแนกผล S/I/R โดยมีการทำ Label Encoding Feature Engineering และ สร้าง confusion matrix/classification report เพื่อประเมินผลการจำแนก

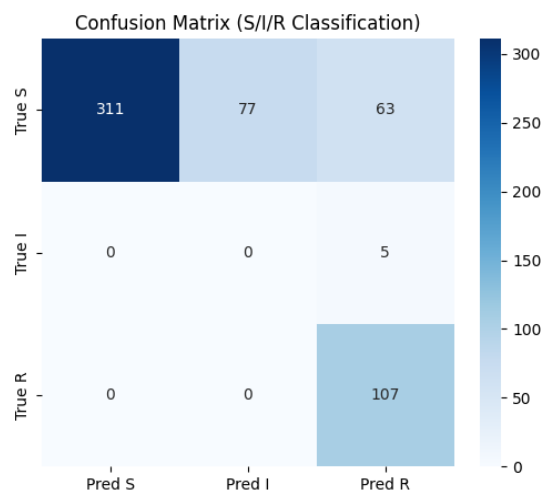
คุณลักษณะของ Scikit-learn

- มีฟังก์ชันสำหรับการสร้าง confusion matrix และ classification report
- สามารถเชื่อมต่อกับ Pandas Data Frame ได้โดยตรง
- รองรับการทำงานกับข้อมูลขนาดใหญ่และซับซ้อน

3.6 Matplotlib, Seaborn

เป็นไลบรารีของภาษา Python สำหรับการสร้างกราฟและการนำเสนอข้อมูลเชิงภาพ (Data Visualization)

- Matplotlib เป็นไลบรารีพื้นฐานสำหรับการสร้างกราฟ 2 มิติ เช่น line plot, bar chart, scatter plot และ heatmap
- Seaborn ถูกสร้างขึ้นบน Matplotlib เพื่อเพิ่มความสามารถการสร้างกราฟเชิงสถิติ เช่น heatmap, boxplot และ histogram



รูปที่ 3.5 confusion matrix เพื่อวิเคราะห์ผล antibiotic susceptibility

3.7 Regex (Regular Expression)

Regular Expression หรือ Regex เป็นเครื่องมือของภาษา Python ที่ใช้สำหรับการจัดการข้อความ โดยสามารถกำหนดรูปแบบ (pattern) เพื่อค้นหา แทนที่ หรือตัดข้อความที่ไม่ต้องการออก จุดเด่นของ Regex คือสามารถจัดการกับข้อความที่มีความซับซ้อน เช่น การลบสัญลักษณ์พิเศษ การตรวจสอบรูปแบบตัวอักษร และการดึงข้อมูลบางส่วนออกมา โดยไม่ต้องเขียนโค้ดที่ยาวหรือซับซ้อน ถูกนำมาใช้ในหลายขั้นตอนของ Cleaning Pipeline เพื่อให้ข้อมูล specimen และ organism มีความสะอาดและพร้อมสำหรับการวิเคราะห์

- การทำความสะอาด specimen ใช้ Regex เพื่อลบ noise ที่เกิดจาก encoding ลบข้อความในวงเล็บ และจัดการช่องว่างที่ซ้ำซ้อน

```
s = re.sub(r"(. *?)", "", s)      # ลบข้อความในวงเล็บ
s = re.sub(r"\s+", " ", s)      # รวมช่องว่างหลายตัวให้เหลือเพียงหนึ่ง
```

รูปที่ 3.6 ใช้ Regex เพื่อลบ noise ที่เกิดจาก encoding, ลบข้อความในวงเล็บ, และจัดการช่องว่างที่ซ้ำซ้อน

- การดึงข้อมูล specimen จาก “@Lab Comment” ใช้ Regex เพื่อค้นหาข้อความที่อยู่หลังคำว่า “@Lab Comment:” และดึงออกมาเป็น specimen

```
match = re.search(r"@Lab Comment:\s*([A-Za-z0-9\ - ]+)", text)
```

รูปที่ 3.7 ใช้ Regex เพื่อค้นหาข้อความที่อยู่หลังคำว่า “@Lab Comment:” และดึงออกมาเป็น specimen

- การ normalize specimen ใช้ Regex เพื่อตัด prefix ที่ไม่จำเป็น และตรวจสอบว่าข้อความเป็นตัวเลขล้วนหรือไม่

```
s = re.sub(r"^@?\s*lab comment[:-]\s*", "", s).strip()
if re.match(r"^\d+(\.\d+)?$", s): return None
```

รูปที่ 3.8 ใช้ Regex เพื่อตัด prefix ที่ไม่จำเป็นและตรวจสอบว่าข้อความเป็นตัวเลขล้วนหรือไม่

- การทำความสะอาด organism ใช้ Regex เพื่อลบข้อความในวงเล็บและแท็กที่ไม่เกี่ยวข้อง เช่น EMBL, CRE, MRSA

```
s = re.sub(r"\(..*?\)", "", s)
s = re.sub(r"(ESBL|CRE|MRSA|VRE|MDR|XDR)", "", s, flags=re.I)
```

รูปที่ 3.9 ใช้ Regex เพื่อลบข้อความในวงเล็บและแท็กที่ไม่เกี่ยวข้อง เช่น ESBL, CRE, MRSA
คุณลักษณะของ Regex

- สามารถกำหนด pattern ที่ซับซ้อนเพื่อจัดการข้อความ
- ช่วยลดความผิดพลาดในการวิเคราะห์ข้อมูล

```
def clean_org_raw(x):
    if not isinstance(x, str):
        return np.nan
    s = x.strip()

    # remove parentheses content
    s = re.sub(r"\(..*?\)", "", s)

    # remove resistance tags
    s = re.sub(r"(ESBL|CRE|MRSA|VRE|MDR|XDR)", "", s, flags=re.I)

    # collapse spaces
    s = re.sub(r"\s+", " ", s)

    return s.strip()
```

รูปที่ 3.10 ใช้ Regex ลบ resistance tags และข้อความในวงเล็บ

บทที่ 4

การวิเคราะห์และออกแบบระบบ

การพัฒนาโครงการ เรื่อง การวิเคราะห์คุณภาพข้อมูลและการจัดทำแดชบอร์ดสำหรับแสดงข้อมูลความไวของเชื้อต่อยาปฏิชีวนะ มีเป้าหมายเพื่อสร้างระบบที่สามารถวิเคราะห์คุณภาพข้อมูลและนำเสนอผลลัพธ์ ให้สามารถตอบสนองต่อการใช้งานจริงของบุคลากรทางการแพทย์ โดยในระยະนี้ระบบสามารถส่งออกผลลัพธ์เป็นไฟล์ Excel ที่มีการจัดรูปแบบสีและตารางสรุป ซึ่งถือเป็นพื้นฐานสำคัญสำหรับการนำไปพัฒนาแดชบอร์ดต่อไป โดยมีขั้นตอนการวิเคราะห์ระบบ ดังนี้

4.1 การวิเคราะห์ผู้ใช้งาน (User Analysis)

ระบบการประเมินคุณภาพข้อมูลและการคำนวณ แอนติไบโอแกรม(Antibiogram) นี้ ออกแบบมาเพื่อรองรับการใช้งานของบุคลากรทางการแพทย์ในโรงพยาบาล โดยเฉพาะ แพทย์ และ บุคลากรสาธารณสุข ซึ่งมีฟังก์ชันหลักในการใช้ข้อมูลผลการเพาะเชื้อและการทดสอบยาปฏิชีวนะในการตัดสินใจทางการแพทย์

1. แพทย์ จะใช้ระบบในการเข้าถึงข้อมูล แอนติไบโอแกรม(Antibiogram) เพื่อตัดสินใจเลือกยาปฏิชีวนะที่เหมาะสมกับผู้ป่วย โดยสามารถกรองข้อมูลตามช่วงเวลา อายุ เพศ หรือชนิดของสิ่งส่งตรวจ เพื่อดูแนวโน้มความไวของเชื้อในกลุ่มผู้ป่วยหรือในแผนกเฉพาะทาง
2. บุคลากรสาธารณสุขจะใช้ระบบในการวิเคราะห์แนวโน้มของการดื้อยาปฏิชีวนะในโรงพยาบาลเพื่อติดตามการเปลี่ยนแปลงของเชื้อดื้อยา และสื่อสารร่วมกับทีมแพทย์ เพื่อให้คำแนะนำในการใช้ยา

4.2 ขั้นตอนการวิเคราะห์ระบบ

การวิเคราะห์ระบบในโครงการนี้มีขั้นตอนหลักที่มุ่งเน้นในการจัดการข้อมูลการเพาะเชื้อและการทดสอบยาปฏิชีวนะผ่านกระบวนการ DQA (Data Quality Assessment) และ ETL (Extract, Transform, Load) โดยมีขั้นตอนหลักดังนี้

4.2.1 การประเมินคุณภาพข้อมูล (DQA)

การประเมินคุณภาพข้อมูล (DQA) เป็นการตรวจสอบและยืนยันข้อมูลที่ได้จากกระบวนการ ETL เพื่อให้แน่ใจว่าข้อมูลที่ใช้งานมีความถูกต้องและน่าเชื่อถือ ซึ่งโครงการนี้จะมุ่งเน้นการตรวจสอบ 2 ด้านหลัก ได้แก่

1. Consistency (ความสอดคล้อง)

การตรวจสอบข้อมูลที่เชื่อมโยงระหว่างแฟ้ม เช่น วันที่รับผู้ป่วยต้องเกิดขึ้นก่อนวันที่เก็บสิ่งส่งตรวจ และการตรวจสอบข้อมูลที่เชื่อมโยงระหว่าง ผู้ป่วย กับ ผลการเพาะเชื้อ เพื่อให้แน่ใจว่าข้อมูลทั้งหมดมีความสัมพันธ์ที่ถูกต้อง

2. Completeness (ความครบถ้วน)

การตรวจสอบว่าแต่ละฟิลด์ที่สำคัญ เช่น ชนิดเชื้อ, รหัสยา, และ ผลการทดสอบ มีการกรอกข้อมูลครบถ้วนหรือไม่ โดยหากพบว่ามีข้อมูลหายหรือว่างเปล่า ระบบจะไม่สามารถนำข้อมูลนั้นไปคำนวณหรือแสดงผลได้

4.2.2 การออกแบบกระบวนการ ETL

กระบวนการ ETL เป็นกระบวนการสำคัญในการดึงข้อมูล (Extract) ทำการแปลงข้อมูล (Transform) และโหลดข้อมูล (Load) ไปยังรูปแบบที่พร้อมใช้งาน เช่น Excel ซึ่งประกอบไปด้วย 3 ขั้นตอนหลัก

- Extract ข้อมูลจะถูกดึงมาจากไฟล์ CSV/Excel ที่ได้จากระบบโรงพยาบาล เช่น ข้อมูลผลการเพาะเชื้อ ข้อมูลจากการทดสอบยาปฏิชีวนะ และข้อมูลผู้ป่วย ซึ่งข้อมูลที่นำเข้ามาจะรวมถึงข้อมูลผลการเพาะเชื้อ ข้อมูลจากการทดสอบยาปฏิชีวนะ และข้อมูลผู้ป่วย

- Transform ข้อมูลที่ดึงมาแล้วจะถูกแปลงให้อยู่ในรูปแบบที่เหมาะสม เช่น การทำ Data Cleaning โดยการใช้ Pandas เพื่อลบข้อมูลที่ขาดหายไปหรือไม่ตรงตามรูปแบบที่ต้องการ และการคำนวณ %S (Susceptibility Rate) สำหรับแต่ละเชื้อและยาปฏิชีวนะ

- Load ข้อมูลที่ผ่านการแปลงและทำความสะอาดแล้วจะถูกโหลดข้อมูลเข้าสู่ไฟล์ Excel โดยตรง (ผ่าน OpenPyXL) พร้อมการจัดสีตามเกณฑ์ %S, REVIEW flag และ Intrinsic Resistance นอกจากนี้ยังสามารถเลือกกรองข้อมูลด้วย Filter UI (ipywidgets) ตาม Sex, Age, Specimen, Organism และ Year

4.2.3 การจัดการข้อมูลและการคำนวณ

หลังจากที่ผ่านกระบวนการ DQA(Data Quality Assessment) และ ETL(Extract, Transform, Load) ข้อมูลที่ได้รับการแปลงและทำความสะอาดแล้วจะถูกคำนวณเพื่อสร้าง Antibioqram โดยการคำนวณ %S (Susceptibility Rate) ซึ่งแสดงถึงความไวของเชื้อโรค ต่อต้านยาปฏิชีวนะในแต่ละชนิด โดยการใช้สูตร

$$Susceptibility Rate(\%) = \left(\frac{S}{N} \right) \times 100$$

นอกจากนี้ยังมีการตรวจสอบ Intrinsic Resistance และ REVIEW flag (เช่น N < 30, MIC, No growth) เพื่อให้ผลลัพธ์มีความถูกต้องตามเกณฑ์ทางคลินิก

4.2.4 การแสดงผลข้อมูล

ผลลัพธ์ที่ได้จากการคำนวณจะถูกจัดระเบียบในรูปแบบไฟล์ Excel โดยมีการจัดสี เพื่อช่วยให้แพทย์ เข้าใจข้อมูลได้ง่ายขึ้น เช่น สีเขียวสำหรับ %S ≥ 90, สีเหลืองสำหรับ %S 76–89, สีส้มสำหรับ %S 61–75, สีแดงสำหรับ %S ≤ 60 และสีเทาสำหรับ REVIEW flag หรือ Intrinsic Resistance ผลลัพธ์สามารถส่งออกได้ทั้งแบบรวมทุก specimen และแบบ กรองตามเงื่อนไขที่ผู้ใช้เลือกผ่าน Filter UI

4.2.5 การตรวจสอบและบันทึกการประมวลผล

หลังจากที่ข้อมูลทั้งหมดถูกคำนวณและแสดงผลแล้ว ระบบจะทำการบันทึก Log File เพื่อเก็บประวัติการทำงานของระบบ เช่น การ Export ข้อมูล, การตรวจสอบข้อมูล, การโหลดข้อมูล และการบันทึก การประเมินคุณภาพข้อมูล (DQA) (Data Quality Assessment) ซึ่งช่วยในการตรวจสอบย้อนหลังและติดตามผลการทำงานของระบบ

4.3 แผนภาพแสดงกรณีการใช้งาน (Use Case Diagram)

การวิเคราะห์ระบบด้วยแผนภาพ Use Case Diagram เป็นขั้นตอนสำคัญที่ช่วยให้สามารถ มองเห็นความสัมพันธ์ระหว่างผู้ใช้งาน (Actors) กับหน้าที่การทำงานของระบบ (Use Cases) ได้อย่าง ชัดเจน ทำงานในลักษณะสคริปต์ประมวลผลเป็นชุด โดยมีลำดับ การประเมินคุณภาพข้อมูล (DQA) → กระบวนการ ETL (Extract, Transform, Load) → การคำนวณ Antibioqram → การ ส่งออกผลเป็นไฟล์ Excel เพื่อให้แพทย์และบุคลากรสาธารณสุขนำไปใช้ต่อ

4.3.2 การใช้งาน (Use Case List)

ตารางที่ 4.1 การใช้งาน (Use Case List)		
ลำดับ	Use Case	คำอธิบาย
01	นำเข้าข้อมูล (Extract)	อ่านข้อมูลจากไฟล์/แหล่งที่กำหนดเข้าสู่ DataFrame
02	ประเมินคุณภาพข้อมูล (DQA	ตรวจสอบ ความครบถ้วน (Completeness) และ ความสอดคล้อง (Consistency) ตามเกณฑ์ที่กำหนด
03	แปลงและทำความสะอาดข้อมูล (Transform)	กำหนดชื่อเชื้อ/ชื่อยา จัดรูปแบบคอลัมน์ และเตรียมตัวแปรสำหรับการคำนวณ
04	คำนวณ แอนติไบโอแกรม(Antibiogram)	คำนวณร้อยละความไวต่อยา (%S) และตัวชี้วัดที่ใช้งานจริง
05	ส่งออกผลเป็น Excel (Load/Export)	สร้างไฟล์ Excel รวมชีตสรุปและชีตแยกตามชนิดสิ่งส่งตรวจ พร้อมบันทึกปูมการประมวลผล (ถ้ามี)
06	ตรวจทานและเผยแพร่ไฟล์ผลลัพธ์	ผู้ใช้ตรวจสอบความครบถ้วน และส่งต่อให้หน่วยงานที่เกี่ยวข้อง

4.3.3 รายละเอียดกรณีการใช้งาน (Use Case Descriptions)

นำเข้าข้อมูล (Extract)

วัตถุประสงค์ ดึงข้อมูลดิบจากแหล่งที่ระบุเข้าสู่ระบบประมวลผล

เงื่อนไขก่อนหน้า กำหนดพาธไฟล์และรูปแบบคอลัมน์ขั้นต่ำถูกต้อง

ขั้นตอนหลัก ระบุแหล่งข้อมูล → อ่านไฟล์ (เช่น CSV/Excel) → ตรวจสอบ

จำนวนแถว/คอลัมน์เบื้องต้น

เงื่อนไขภายหลัง ได้ Data Frame สำหรับดำเนินการ DQA (Data Quality Assessment)

ข้อยกเว้น ไฟล์ไม่พบหรืออ่านไม่ได้ ให้บันทึกข้อผิดพลาดและยุติ

กระบวนการ

ประเมินคุณภาพข้อมูล DQA (Data Quality Assessment)

วัตถุประสงค์: รับรองคุณภาพข้อมูลก่อนการคำนวณผล

ขอบเขตการตรวจ:

- Completeness 필드สำคัญ เช่น สิ่งมีชีวิต (Organism), ยาปฏิชีวนะ (Antibiotic), ผล S/I/R ที่ผ่านการทำความสะอาด, ชนิดสิ่งส่งตรวจ (Specimen type) ต้องไม่ว่างตามเกณฑ์
- Consistency ความสัมพันธ์ของข้อมูลต้องสมเหตุสมผล (เช่น ผลทดสอบอยู่ในเซต {S, I, R};)

เงื่อนไขภายหลัง ได้รายงาน DQA (Data Quality Assessment) (รายการข้อมูลขาดหาย/ผิดรูปแบบ/ขัดแย้ง) และชุดข้อมูลผ่านเกณฑ์

แปลงและทำความสะอาดข้อมูล (Transform)

วัตถุประสงค์ มาตรฐานและเตรียมข้อมูลให้พร้อมสำหรับการคำนวณ
ขั้นตอนหลัก ทำ normalization ชื่อเชื้อ/ชื่อยา → จัดรูปแบบคอลัมน์ →
สร้างตัวแปรช่วยคำนวณ (เช่น จำนวนตัวอย่างที่ทดสอบ, จำนวนผล S)
เงื่อนไขภายหลัง ได้ตารางที่ผ่านการทำความสะอาดและพร้อมคำนวณ

คำนวณ แอนติไบโอแกรม(Antibiogram)

วัตถุประสงค์ สร้างตารางสรุปความไวต่อยาปฏิชีวนะที่ใช้งานจริง
สูตรที่ใช้ $\%S = (\text{จำนวนผล S} / \text{จำนวนที่ทดสอบทั้งหมด}) \times 100$
ขั้นตอนหลัก จัดกลุ่มตาม (เชื้อ, ยา [และชนิดสิ่งส่งตรวจหากกำหนด]) →
รวมยอด → คำนวณ %S และ n_tested → จัดรูปผลลัพธ์

ส่งออกผลเป็น Excel (Load/Export)

วัตถุประสงค์ จัดทำไฟล์ผลลัพธ์ที่พร้อมใช้งาน

โครงสร้างเอาต์พุตโดยสังเขป

1. ซีต Antibigram_All (รวมทุกชนิดสิ่งส่งตรวจ)
2. ซีตแยกตาม specimen_type เช่น Urine, Blood, Pus, Sputum (เฉพาะชนิดที่มีข้อมูลเพียงพอ)

เงื่อนไขภายหลัง ได้ไฟล์ Excel ล่าสุด สำหรับนำไปใช้ตรวจทานและเผยแพร่ไฟล์ผลลัพธ์

วัตถุประสงค์ ให้ผู้ใช้ปลายทางตรวจสอบความครบถ้วนและเผยแพร่ไฟล์
 ขั้นตอนหลัก เปิดไฟล์ → ตรวจสอบความครบถ้วนของซีต/คอลัมน์หลัก → ส่งต่อ
 ให้ทีมที่เกี่ยวข้อง (เช่น ทีมแพทย์/งานควบคุมการติดเชื้อ)

4.3.4 ขอบเขตและข้อจำกัด

- การประเมินคุณภาพข้อมูลในงานนี้ครอบคลุม 2 มิติหลัก คือ ความสอดคล้อง (Consistency) และ ความครบถ้วน (Completeness) ตามที่ใช้งานจริงในโค้ด
- ผลลัพธ์อยู่ในรูป ไฟล์ Excel เพื่อการใช้งานต่อโดยแพทย์และบุคลากรสาธารณสุข

4.4 การออกแบบโครงสร้างฐานข้อมูล (Database Schema Design)

โครงการนี้ใช้งานชุดข้อมูลจากไฟล์ CSV เป็นโครงสร้างข้อมูลแบบ Pandas DataFrame แทนฐานข้อมูล SQL จริง ดังนั้น “การออกแบบฐานข้อมูล” ในที่นี้จึงหมายถึง แบบจำลองเชิงตรรกะ (Logical Data Model) และ พจนานุกรมข้อมูล (Data Dictionary) ของคอลัมน์ที่ระบบใช้งานจริง โดยตารางที่แสดง (เช่น IPD_Culture, OPD_Culture, IPD_Data) เป็น โครงสร้างจำลองจากไฟล์ CSV ไม่ใช่ฐานข้อมูลจริง

ตารางที่ 4.2 แสดงดาต้าดิกชันนารีของตาราง: IPD_Culture

Field Name: IPD_Culture					
Description: ข้อมูลการเพาะเชื้อผู้ป่วยใน					
No.	Field Name	Type	Length	Format	Description / Constraint
1	Culture_ID	varchar	12	ตัวอักษร	รหัสกำกับข้อมูลการเพาะเชื้อ (Primary key)
2	HN	varchar	10	ตัวอักษร	หมายเลขโรงพยาบาลของผู้ป่วย (Foreign key)
3	Specimen	varchar	50	ตัวอักษร	ประเภทตัวอย่างที่ใช้ในการเพาะเชื้อ เช่น เลือด, ปัสสาวะ
4	Organism	varchar	100	ตัวอักษร	ชื่อเชื้อที่พบ
5	Culture_Date	date	-	วันที่	วันที่ส่งตรวจ
6	Ward	varchar	20	ตัวอักษร	หอผู้ป่วยที่เข้ารับการรักษ

ตารางที่ 4.3 แสดงดาต้าดิกชันนารีของตาราง: OPD_ Culture

Field Name: OPD_ Culture					
Description: ข้อมูลผลตอบสนองต่อยาปฏิชีวนะผู้ป่วยใน					
No.	Field Name	Type	Length	Format	Description / Constraint
1	Antibiotic_ID	varchar	12	ตัวอักษร	รหัสยา (Primary key)
2	Culture_ID	varchar	12	ตัวอักษร	อ้างอิงการเพาะเชื้อ (Foreign key)
3	Antibiotic_Name	varchar	100	ตัวอักษร	ชื่อยาปฏิชีวนะ
4	SIR_Result	varchar	1	ตัวอักษร	ผลการทดสอบความไว S / I / R

ตารางที่ 4.4 แสดงดาต้าดิกชันนารีของตาราง: IPD_ Data

Field Name: IPD_ Data					
Description: ข้อมูลพื้นฐานผู้ป่วยใน					
No.	Field Name	Type	Length	Format	Description / Constraint
1	HN	varchar	10	ตัวอักษร	หมายเลขโรงพยาบาลของผู้ป่วย (Primary key)
2	Gender	varchar	1	ตัวอักษร	เพศ
3	Age	int	3	ตัวเลข	อายุผู้ป่วย
4	Admit_Date	date	-	วันที่	วันที่เข้ารับการรักษา
5	Discharge_Date	date	-	วันที่	วันที่จำหน่ายผู้ป่วย

ตารางที่ 4.5 แสดงดาต้าดิกชันนารีของตาราง: IPD_Diagnosis

Field Name: IPD_Diagnosis						
Description: ข้อมูลการวินิจฉัยโรคผู้ป่วยใน						
No.	Field Name	Type	Length	Format	Description	Constraint
1	Diag_ID	varchar	12	ตัวอักษร	รหัสการวินิจฉัย	Primary key
2	HN	varchar	10	ตัวอักษร	รหัสผู้ป่วย	Foreign key
3	ICD10_Code	varchar	10	ตัวอักษร	รหัส ICD-10	Null
4	Diagnosis_Desc	varchar	255	ตัวอักษร	คำอธิบายโรค	Null

4.4.1 นิยามชุดข้อมูลพร้อมประมวลผล

คือ “ชุดข้อมูลกลางที่ผ่านการตรวจคุณภาพ (DQA) และการทำความสะอาด/มาตรฐาน (Transform) แล้ว” ซึ่งระบบใช้เป็นอินพุตโดยตรงสำหรับการคำนวณ Antibioqram และการส่งออกผล โดยมีลักษณะสำคัญดังนี้

1. ที่มา: รวมและทำความสะอาดข้อมูลจากแหล่งอินพุตที่กำหนด(เช่นผลเพาะเชื้อ-ผลความไวต่อยา) ผ่านเกณฑ์ Completeness และ Consistency
2. ระดับความละเอียด (granularity): 1 แถว = 1 ผลการทดสอบความไวของ “เชื้อหนึ่งชนิด-ยาหนึ่งตัว-สิ่งส่งตรวจหนึ่งประเภท”
3. คอลัมน์ที่ต้องมี:
 - 1). ชื่อเชื้อที่ผ่านการทำมาตรฐานแล้ว
 - 2). ชื่อยาปฏิชีวนะที่ผ่านการทำมาตรฐานแล้ว
 - 3). ผลความไวอยู่ในเซต {S, I, R} เท่านั้น
 - 4). ค่ามาตรฐานของชนิดสิ่งส่งตรวจ (เช่น Urine, Blood, Pus, Sputum ตามโดเมนที่กำหนด)
4. ขอบเขตการใช้งาน: ใช้สำหรับจัดกลุ่ม (groupby) เพื่อคำนวณตัวชี้วัด เช่น จำนวนตัวอย่างที่ทดสอบ (n_tested), จำนวนผล S (n_S), และร้อยละความไวต่อยา (%S)

4.4.2 โครงสร้างเชิงตรรกะของข้อมูล (Logical Model)

ระบบใช้งานตารางเชิงตรรกะ 2 กลุ่มหลัก

1. อินพุตมาตรฐาน ชุดข้อมูลกลางที่ผ่านการตรวจคุณภาพ (DQA) และการทำความสะอาด/มาตรฐาน (Transform) แล้ว: ประกอบด้วยคอลัมน์ที่ต้องมีตามในข้อ 4.4.2

2. ผลลัพธ์สรุป แอนติไบโอแกรม (Antibiogram): ตารางที่เกิดจากการรวมกลุ่มตามคู่ ชื่อเชื้อ ที่ผ่านการทำมาตรฐานแล้ว และ ชื่อยาปฏิชีวนะที่ผ่านการทำมาตรฐานแล้ว และแยกตาม ค่ามาตรฐานของชนิดสิ่งส่งตรวจ เมื่อจำเป็น โดยมีคอลัมน์อย่างน้อย

- 1) ชื่อเชื้อที่ผ่านการทำมาตรฐานแล้ว
- 2) ชื่อยาปฏิชีวนะที่ผ่านการทำมาตรฐานแล้ว
- 3) จำนวนตัวอย่างที่ทดสอบ
- 4) จำนวนตัวอย่างที่ให้ผลเชื้อตอบสนองต่อยาปฏิชีวนะ
- 5) $\%S = (n_S / n_tested) \times 100$

4.4.3 พจนานุกรมข้อมูล (Data Dictionary)

ตารางที่ 4.6 พจนานุกรมข้อมูลของชุดข้อมูลกลางที่ผ่านการตรวจคุณภาพ และการทำความสะอาด/มาตรฐาน แล้ว

No.	Field Name	Type	Length	Format	Description / Constraint
1	org_clean	varchar	-	ตัวอักษร	ชื่อเชื้อมาตรฐานที่ผ่านการทำความสะอาด/แมปแล้ว (ต้องไม่ว่าง)
2	anti_name_clean	varchar	-	ตัวอักษร	ชื่อยาปฏิชีวนะมาตรฐานที่ผ่านการทำความสะอาด/แมปแล้ว (ต้องไม่ว่าง)
3	anti_result_clean	varchar	1	ตัวอักษร	ผลความไวต่อยา จำกัดโดเมน {S, I, R}
4	specimen_key	varchar	-	ตัวอักษร	ค่ามาตรฐานชนิดสิ่งส่งตรวจ (ต้องอยู่ในโดเมนที่กำหนด/ต้องไม่ว่าง)

ตารางที่ 4.7 พจนานุกรมข้อมูลของผลลัพธ์ (ซีตแอนติไบโอแกรม และซีตแยกตามสิ่งส่งตรวจ)

No.	Field Name	Type	Length	Format	Description / Constraint
1	organism	varchar	-	ตัวอักษร	เทียบค่าจาก ชื่อเชื้อที่ผ่านการทำ มาตรฐานแล้ว
2	antibiotic	varchar	-	ตัวอักษร	เทียบค่าจากชื่อยาปฏิชีวนะที่ผ่านการ ทำมาตรฐานแล้ว
3	n_tested	integer	-	ตัวเลข	จำนวนตัวอย่างทั้งหมดในกลุ่ม (เชื้อ ,ยา,สิ่งส่งตรวจ])
4	n_S	integer	-	ตัวเลข	จำนวนตัวอย่างที่ให้ผล ร้อยละความไว ต่อยา
5	%S	decimal	-	ตัวเลข	ร้อยละความไว = $(n_S / n_tested) \times 100$

4.5 การออกแบบกระบวนการเข้าถึงและการจัดการข้อมูล (Database Access and Management Design)

ระบบที่พัฒนาขึ้นทำงานในลักษณะสคริปต์แบบประมวลผลเป็นชุด (batch/script) บน Google Colab โดยมีวัตถุประสงค์เพื่อ

1. ตรวจสอบภาพข้อมูลเฉพาะมิติ Completeness และ Consistency,
2. ทำความสะอาด/ทำมาตรฐานข้อมูล
3. คำนวณตาราง Antibioqram
4. ส่งออกเป็นไฟล์ Excel ให้แพทย์และบุคลากรสาธารณสุขใช้งานต่อ

4.5.1 องค์ประกอบของระบบ (Modules)

1. Data Ingestion (Extract Module)

อ่านแฟ้มอินพุต (CSV/Excel) ของข้อมูลเพาะเชื้อและผลความไวต่อยาเข้าสู่ Data Frame มาที่กำหนดในข้อ 4.4.2

2. Data Quality Assessment (DQA Module)

ตรวจคุณภาพข้อมูลเฉพาะ 2 มิติที่ใช้งานจริง

- 1) Completeness: ชื่อเชื้อที่ทำความสะอาดแล้ว, ชื่อยาปฏิชีวนะที่ทำความสะอาดแล้ว, ผลความไวของเชื้อต่อยาปฏิชีวนะที่ทำความสะอาดแล้ว, ชนิดสิ่งส่งตรวจต้องไม่ว่าง
- 2) Consistency: ชื่อยาปฏิชีวนะที่ทำความสะอาดแล้ว $\in \{S, I, R\}$ และชนิดสิ่งส่งตรวจ ต้องอยู่ในโดเมนที่กำหนดบันทึกสถิติ/ระเบียบที่ไม่ผ่านเกณฑ์ลงปุม (DQA log) หากมีการจัดเก็บ

3. Transform & Standardization Module

ทำความสะอาดและทำมาตรฐาน ได้แก่

- 1) แมป/ทำมาตรฐาน ชื่อเชื้อ
- 2) แมป/ทำมาตรฐาน ชื่อยา
- 3) ทำความสะอาดผลทดสอบ
- 4) รวมมาตรฐานชนิดสิ่งส่งตรวจ
- 5) จัดการช่องว่าง, อักขระพิเศษ, ความไม่สม่ำเสมอของการสะกด; ตัดระเบียบซ้ำตามกฎเกณฑ์ใช้จริง

4. Aggregation & Calculation Module

รวมกลุ่มตามคู่ ชื่อยาปฏิชีวนะที่ทำความสะอาดแล้ว และ ผลความไวของเชื้อต่อยาปฏิชีวนะที่ทำความสะอาดแล้ว (และตาม ชนิดสิ่งส่งตรวจ เมื่อแยกตามสิ่งส่งตรวจ) เพื่อคำนวณตัวชี้วัด

- 1) $n_{\text{tested}} = \text{จำนวนระเบียบในกลุ่ม}$
- 2) $n_S = \text{จำนวนระเบียบที่ anti_result_clean} = 'S'$
- 3) $\%S = (n_S / n_{\text{tested}}) \times 100$

5. Export Module

ส่งออกผลเป็นไฟล์ Excel ชื่อ เช่น Antibigram_All พร้อมหลายชีต ได้แก่

- 1) Antibigram All (รวมทุกชนิดสิ่งส่งตรวจ)
- 2) ชีตแยกตาม ชนิดสิ่งส่งตรวจ(เช่น Urine, Blood, Pus, Sputum)

กฎสำคัญระหว่าง Export (ตามข้อกำหนดของโครงการ): ห้ามสร้าง/เขียนทับ ชื่อเชื้อที่ผ่านการทำความสะอาดแล้วจากคอลัมน์ดิบทุกกรณี และต้องตรวจให้มี ชื่อเชื้อที่ทำความสะอาดแล้ว, ผลความไวขอเชื้อต่อยาปฏิชีวนะที่ทำความสะอาดแล้ว, ผลความไวขอเชื้อต่อยาปฏิชีวนะที่ทำความสะอาดแล้ว, ชนิดสิ่งส่งตรวจ ครบก่อนส่งออก หากขาดให้หยุดกระบวนการและรายงานข้อผิดพลาด

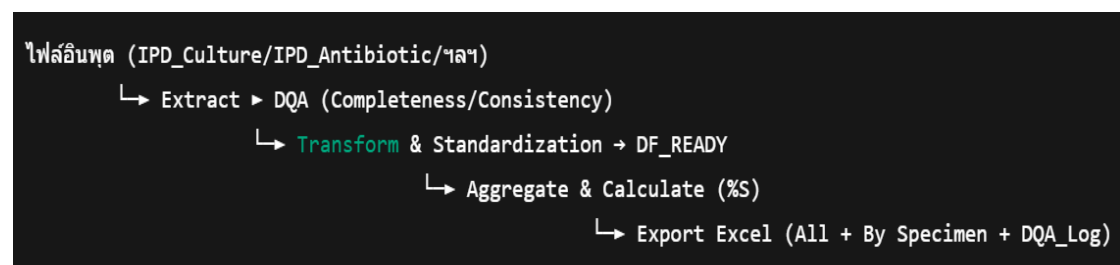
4.5.2 ข้อกำหนดอินพุต/เอาต์พุต (I/O Contracts)

- อินพุตขั้นต่ำ: ต้องสามารถสร้าง ชุดข้อมูลกลางที่ผ่านการตรวจคุณภาพ (DQA) และการทำความสะอาด/มาตรฐาน (Transform) แล้วได้โดยมีคอลัมน์ ชื่อเชื้อที่ทำความสะอาดแล้ว, ชื่อยาปฏิชีวนะที่ทำความสะอาดแล้ว, ผลความไวขอเชื้อต่อยาปฏิชีวนะที่ทำความสะอาดแล้ว, ชนิดสิ่งส่งตรวจ ครบ
- เอาต์พุต: ไฟล์ Excel 1 ไฟล์ประกอบด้วยชีตผลรวม, ชีตแยกตามสิ่งส่งตรวจ และปุม (ถ้ามี) โดยแต่ละชีตมีคอลัมน์ organism, antibiotic, n_tested, n_S, %S

4.5.3 ลำดับการประมวลผล (Processing Flow)

1. รับพาธไฟล์อินพุต → อ่านเป็น Data Frame
2. ตรวจ DQA (Completeness/Consistency) → ทำเครื่องหมาย/ตัดระเบียบวันที่ตกเกณฑ์ → สรุป DQA
3. Transform: ทำมาตรฐานชื่อเชื้อ/ชื่อยา/ผลทดสอบ/สิ่งส่งตรวจ → ได้ DF_READY
4. Aggregation: groupby และคำนวณ n_tested, n_S, %S (รวมและแยกตาม specimen_key)
5. Export: เขียนไฟล์ Excel ตามโครงสร้างที่กำหนด พร้อมตรวจเงื่อนไขบังคับก่อนบันทึก
6. ส่งมอบไฟล์ให้ผู้ปลายทางตรวจทานและนำไปใช้

4.5.4 ผังการไหลของข้อมูล (Data Flow)



รูปที่ 4.1 ผังการไหลของข้อมูล (Data Flow)

4.5.5 การจัดการข้อผิดพลาดและการบันทึก (Error Handling & Logging)

1. หากไฟล์อินพุตไม่พบ/อ่านไม่ได้: รายงานสาเหตุและยุติกระบวนการ

2. หากคอลัมน์ขึ้นต่ำหายไป: รายงานชื่อคอลัมน์ที่ขาด และไม่ดำเนินการต่อ
3. หากพบค่าที่อยู่นอกโดเมน (เช่น anti_result_clean ไม่ใช่ S/I/R): ตัดออกจากการคำนวณและบันทึกในปุม
4. บันทึกสรุปจำนวนระเบียบที่เข้า/ออกในแต่ละชั้น เพื่อการตรวจสอบย้อนหลัง

4.5.6 สมมติฐานและข้อจำกัดของระบบ

1. ระบบทำงานแบบสคริปต์ ไม่มี UI/แดชบอร์ด
2. ใช้มิติ DQA เพียง สองมิติ (Completeness, Consistency) ตามที่ใช้งานจริง
3. เอาต์พุตจัดทำในรูปไฟล์ Excel เพื่อให้ผู้เชี่ยวชาญปลายทางนำไปใช้/แปรผลต่อ
4. ไม่พิจารณาเกณฑ์ขึ้นต่ำของจำนวนตัวอย่าง (n_{tested}) เว้นแต่กำหนดเพิ่มเติมในภายหลัง

4.5.7 ความเป็นส่วนตัวและการใช้งานจริง

1. ไม่ส่งออกข้อมูลระบุตัวบุคคล (เช่น เลขประจำตัวผู้ป่วย (Hospital Number)) ในไฟล์ผลลัพธ์ที่เผยแพร่
2. ผู้ใช้งานเป้าหมายเฉพาะ แพทย์ และ บุคลากรสาธารณสุข ตามขอบเขตโครงการ

4.6 การออกแบบการประมวลผลข้อมูล (Data Processing Design)

ระบบประมวลผลทำงานแบบสคริปต์บน Google Colab โดยเรียงลำดับงานเป็น DQA → Transform/Standardize → Aggregate & Calculate → Export

4.6.1 Data Quality Assessment (DQA)

วัตถุประสงค์: คัดกรองความพร้อมของข้อมูลก่อนเข้าคำนวณ

อินพุต: DataFrame ที่อ่านมาจากไฟล์ต้นทาง

- 1). ชื่อเชื้อที่ผ่านการทำมาตรฐานแล้ว
- 2). ชื่อยาปฏิชีวนะที่ผ่านการทำมาตรฐานแล้ว
- 3). ผลความไวอยู่ในเซต {S, I, R} เท่านั้น
- 4). ค่ามาตรฐานของชนิดสิ่งส่งตรวจ (เช่น Urine, Blood, Pus, Sputum ตามโดเมนที่กำหนด)

เอาต์พุต: รายงานสถิติ DQA (สัดส่วน missing/invalid) และชุดข้อมูลที่ผ่านเกณฑ์

กฎ

1. Completeness

- คอลัมน์บังคับต้องไม่ว่างต้องมี ชื่อเชื้อที่ทำความสะอาดแล้ว, ชื่อยาปฏิชีวนะที่ทำความสะอาดแล้ว, ผลความไวขอเชื้อต่อยาปฏิชีวนะที่ทำความสะอาดแล้ว, ชนิดสิ่งส่งตรวจ
- แถวที่ขาดคอลัมน์บังคับจะถูกตัดออกจากการคำนวณ และ (ถ้ามี) บันทึกใน DQA Log

2. Consistency

- 1) ผลความไวขอเชื้อต่อยาปฏิชีวนะที่ทำความสะอาดแล้ว ต้องอยู่ในเซต {S, I, R}
- 2) ชนิดสิ่งส่งตรวจ ต้องอยู่ในโดเมนที่ระบบกำหนด (เช่น Urine, Blood, Pus, Sputum ฯลฯ)
- 3) ความซ้ำซ้อนไม่ให้มีระเบียบซ้ำในบริบทเดียวกันของ(ชนิดสิ่งส่งตรวจ, ชื่อเชื้อที่ทำความสะอาดแล้ว, ชื่อยาปฏิชีวนะที่ทำความสะอาดแล้ว หากพบ ให้คัดเลือก/คงไว้ 1 ระเบียบที่เหมาะสม และบันทึกรายการ

เงื่อนไขก่อน-หลัง

1. ก่อน: มี Data Frame จากขั้น Extract
2. หลัง: ได้ชุดข้อมูลที่ผ่าน Data Quality Assessment (DQA) เพื่อเข้าสู่การ Transform พร้อมสรุปผล Data Quality Assessment (DQA)

4.6.2 Transform / Standardize

วัตถุประสงค์ ทำมาตรฐานและเตรียมข้อมูลให้พร้อมคำนวณ

ขั้นตอนหลัก (อิงตามโค้ดที่ใช้งานจริง)

1. มาตรฐานชื่อเชื้อและชื่อยา
 - 1) แมปชื่อดิบให้เป็น ชื่อเชื้อที่ทำความสะอาดแล้ว, ชื่อยาปฏิชีวนะที่ทำความสะอาดแล้ว (แก้ ช่องว่าง, อักษรพิเศษ, ตัวพิมพ์, คำซ้ำ/คำเกิน ฯลฯ)
 - 2) หากแมปไม่สำเร็จ ให้จัดเป็นรายการตรวจสอบ (และไม่นำเข้าคำนวณ)
2. มาตรฐานผลความไว แปลงผลดิบให้เป็น ชื่อยาปฏิชีวนะที่ทำความสะอาดแล้ว $\in \{S, I, R\}$ เท่านั้น
3. มาตรฐานชนิดสิ่งส่งตรวจ แมปค่าไปยัง ชนิดสิ่งส่งตรวจที่ระบบกำหนด (โดเมนปิด)
4. จัดการข้อมูลซ้ำ ลบ/รวมระเบียบซ้ำตามที่กำหนดในข้อ 4.6.1

ผลลัพธ์: ได้ “ชุดข้อมูลพร้อมประมวลผล” ชุดข้อมูลกลางที่ผ่านการตรวจคุณภาพและการทำความสะอาด/มาตรฐาน แล้ว (DF_READY) ที่ประกอบด้วยคอลัมน์บังคับ 4 คอลัมน์และผ่านเกณฑ์ Data Quality Assessment (DQA) แล้ว

4.6.3 Aggregate & Calculate (คำนวณ Antibigram)

วัตถุประสงค์ สร้างตารางสรุปความไวของเชื้อแต่ละชนิดต่อยาปฏิชีวนะแต่ละตัว

อินพุต ชุดข้อมูลพร้อมประมวลผล”ชุดข้อมูลกลางที่ผ่านการตรวจคุณภาพ และการทำความสะอาด/มาตรฐานแล้ว(DF_READY)

วิธีคำนวณ (อิงตรรกะในโค้ด)

1. จัดกลุ่มตามคู่ (ชื่อเชื้อที่ทำความสะอาดแล้ว, ชื่อยาปฏิชีวนะที่ทำความสะอาดแล้ว) และเมื่อสร้างรายงานแยกตามชนิดสิ่งส่งตรวจ ให้เพิ่ม ชนิดสิ่งส่งตรวจ ในการจัดกลุ่ม
2. นิยามตัวชี้วัด:
 - 1) n_{tested} = จำนวนระยะเบี่ยงทั้งหมดในกลุ่ม
 - 2) n_S = จำนวนระยะเบี่ยงที่ `anti_result_clean = 'S'`
 - 3) $\%S = (n_S / n_{\text{tested}}) \times 100$
3. จัดรูปผลลัพธ์เป็นตารางมาตรฐาน เพื่อส่งต่อไปยังชั้น Export

เงื่อนไขก่อน-หลัง

1. ก่อน: `df_all` ต้องมีคอลัมน์ครบและผ่าน Data Quality Assessment (DQA)
2. หลัง: ได้ Data Frame ผลสรุป (รวมทุก specimen และแบบแยกตาม ชนิดสิ่งส่งตรวจ)

4.6.4 กรณีขอบและข้อกำหนดสำคัญ (Edge Cases & Guardrails)

1. ห้ามเขียนทับค่า ชื่อเชื้อที่ทำความสะอาดแล้ว (และ *_clean) ในทุกกรณีในช่วง Export
2. ต้องมี ชื่อยาปฏิชีวนะที่ทำความสะอาดแล้ว, ผลความไวของเชื้อต่อยาปฏิชีวนะที่ทำความสะอาดแล้ว, ชื่อเชื้อที่ทำความสะอาดแล้ว ครบ ก่อนคำนวณ/ส่งออก มิฉะนั้นให้หยุดและรายงานข้อผิดพลาด
3. หากมีค่าที่อยู่นอกโดเมน (เช่น ผลความไวของเชื้อต่อยาปฏิชีวนะที่ทำความสะอาดแล้ว ไม่ใช่ S/I/R) ให้ตัดออก
4. หากจำนวนข้อมูลในบางกลุ่มเท่ากับศูนย์หรือไม่เพียงพอ ให้ละเว้นการสร้างแถวผลลัพธ์สำหรับกลุ่มนั้น

4.7 การออกแบบการส่งออกผลลัพธ์ (Export Data Design)

วัตถุประสงค์ จัดทำไฟล์ผลลัพธ์ในรูปแบบ Excel เพื่อให้แพทย์และบุคลากรสาธารณสุขนำไปใช้งานได้ทันที

ชื่อไฟล์เอาต์พุต Antibigram_All.xlsx

4.7.1 โครงสร้างไฟล์และชีต

1. ชีต Antibioqram All – สรุปภาพรวมทุกชนิดสิ่งส่งตรวจ
2. ชีตแยกตามชนิดสิ่งส่งตรวจ – เช่น Urine, Blood, Pus, Sputum (สร้างเฉพาะชนิดที่มีข้อมูลเพียงพอ)
3. สรุปรายการที่ตกเกณฑ์หรือเหตุการณ์สำคัญระหว่างประมวลผล

4.7.2 โครงสร้างที่ใช้อธิบายและจัดระเบียบข้อมูลของแต่ละชีตผลลัพธ์

คอลัมน์มาตรฐานที่ต้องมีใน Antibioqram_All และชีตแยกตามชนิดสิ่งส่งตรวจ:

1. organism (อ้างอิงจาก ชื่อยาปฏิชีวนะที่ทำความสะอาดแล้ว)
2. antibiotic (อ้างอิงจาก ชื่อยาปฏิชีวนะที่ทำความสะอาดแล้ว)
3. n_tested
4. n_S
5. $%S = (n_S / n_tested) \times 100$

หมายเหตุ: ไม่ส่งออกตัวระบุตัวบุคคล (เช่น HN) ในไฟล์ผลลัพธ์

4.7.3 ลำดับและกติกากการบันทึก

1. ลำดับการเขียนชีต: Antibioqram_All ก่อน แล้วตามด้วยชีตแยกตาม ชนิดสิ่งส่งตรวจ โดยเรียงตามโดเมนที่ระบบกำหนด
2. ก่อนบันทึกไฟล์ ต้องตรวจว่า:
 1. ชุดผลสรุปมีคอลัมน์มาตรฐานครบทุกชีต
 2. ไม่มีค่าที่อยู่นอกโดเมนใน %S (เช่น หารด้วยศูนย์)
 3. ไม่มีการเขียนทับค่า *_clean ระหว่างการเตรียมข้อมูลเพื่อส่งออก

4.7.4 เส้นทางจัดเก็บและการตั้งชื่อ

1. เก็บเป็นไฟล์ Excel หนึ่งไฟล์ในไดเรกทอรีเอาต์พุต
2. ใช้ชื่อไฟล์คงที่ตามที่ระบุ

4.7.5 การตรวจรับผลลัพธ์หลังส่งออก (Post-Export Validation)

1. เปิดตรวจไฟล์ว่าแต่ละชีตมีคอลัมน์ครบ (organism, antibiotic, n_tested, n_S, %S)
2. ตรวจว่ามีเฉพาะชีตชนิดสิ่งส่งตรวจที่มีข้อมูลจริง
3. (ถ้ามี DQA Log) ตรวจสอบจำนวนระเบียบตกเกณฑ์สอดคล้องกับสรุป DQA

บทที่ 5

การทดสอบโปรแกรม

การทดสอบระบบเป็นขั้นตอนสำคัญในการยืนยันว่าเวิร์กโฟลว์ที่พัฒนาสามารถตรวจคุณภาพข้อมูลตาม 2 มิติ (Completeness และ Consistency), ทำความสะอาด/ทำมาตรฐาน (ETL), คำนวณตาราง Antibigram ได้ถูกต้อง, และส่งออกผลเป็นไฟล์ Excel ตามรูปแบบที่กำหนด โดยมีการเทียบค่า %S กับรายงานอ้างอิง Antibigram Thailand เขต 1 เพื่อประเมินความสอดคล้องเชิงแนวโน้ม ทั้งนี้ระบบยังรองรับการส่งออกผลลัพธ์แบบ Filtered Antibigram ผ่าน Filter UI (ipywidgets) ตามเงื่อนไขที่ผู้ใช้เลือก

5.1 วัตถุประสงค์และขอบเขตการทดสอบ

การทดสอบมุ่งยืนยันว่าเวิร์กโฟลว์ที่พัฒนาสามารถ

1. ตรวจคุณภาพข้อมูลตาม 2 มิติที่ใช้งานจริง คือ Completeness และ Consistency
2. ทำความสะอาด/ทำมาตรฐาน (ETL) ให้ได้ชุดข้อมูล “พร้อมคำนวณ”
3. คำนวณตารางแอนติไบโอแกรม(Antibigram)ได้ถูกต้อง
4. ส่งออกผลเป็นไฟล์Excelตามรูปแบบที่กำหนดโดยโค้ด
5. เทียบค่า %S กับรายงานอ้างอิง Antibigram Thailand เขต 1 (ภาคเหนือ) เพื่อประเมินความสอดคล้องการเทียบค่า %S กับรายงานอ้างอิง เป็นการตรวจสอบเชิงแนวโน้ม

5.2 นิยามคอลัมน์ที่ใช้ในการทดสอบ 5.2 นิยามคอลัมน์ที่ใช้ในการทดสอบ

ตารางที่ 5.1 พจนานุกรมข้อมูลของผลลัพธ์ (ชีตแอนติไบโอแกรม และชีตแยกตามสิ่งส่งตรวจ)

คอลัมน์	สถานะ	ความหมาย	บทบาทในเวิร์กโฟลว์
org	ดิบ	ชื่อเชื้อจากต้นทาง (สะกดหลากหลายแบบ)	ใช้ก่อนทำความสะอาด
org_clean	ทำความสะอาดแล้ว	ชื่อเชื้อแบบมาตรฐาน จัดกลุ่มสิ่งมีชีวิตที่คล้ายกันเข้าไว้ด้วยกัน (genus/species)	คีย์หลัก สำหรับจัดกลุ่มและส่งออก

ตารางที่ 5.1 พจนานุกรมข้อมูลของผลลัพธ์ (ซีตแอนติไบโอแกรม และซีตแยกตามสิ่งส่งตรวจ)(ต่อ)

คอลัมน์	สถานะ	ความหมาย	บทบาทในเวิร์กโฟลว์
anti_name_clean	ทำความสะอาดแล้ว	ชื่อยาปฏิชีวนะแบบมาตรฐาน (INN)	คีย์ร่วม กับ org_clean
anti_result_clean	ทำความสะอาดแล้ว	ผลความไวต่อยาแบบจำกัดโดเมน {S,I,R}	ใช้คำนวณ %S
specimen	ดิบ	ชนิดสิ่งส่งตรวจจากต้นทาง (สะกดหลากหลาย)	ใช้ก่อนทำความสะอาด
specimen_key	ทำความสะอาดแล้ว	ชนิดสิ่งส่งตรวจแบบมาตรฐาน (โดเมนปิด)	ใช้เทียบกับอ้างอิง

5.3 วิธีการทดสอบตามกระบวนการทำงานจริง

การทดสอบดำเนินการแบบเป็นลำดับขั้นเพื่อประเมินคุณภาพและความสอดคล้องของข้อมูลในทุกคอลัมน์ของทุกตารางที่นำเข้า ก่อนส่งต่อเข้าสู่การคำนวณค่าเปอร์เซ็นต์ความไวต่อยาปฏิชีวนะ (%S) และการส่งออกไฟล์ผลลัพธ์ โดยยึดกรอบการประเมินคุณภาพข้อมูล 2 มิติที่ใช้งานจริง ได้แก่ ความครบถ้วน (Completeness) และ ความสอดคล้อง (Consistency)

5.3.1 การเตรียมฐานข้อมูลสำหรับ DQA (Extract & Harmonize ขั้นต้น)

อ่านแฟ้มข้อมูล **CSV/Excel** เช่น IPD_data, IPD_diag, IPD_culture, OPD_culture และจัดรูปแบบหัวคอลัมน์ให้เป็นมาตรฐานเดียวกัน (ตัวพิมพ์, ช่องว่าง, ชนิดข้อมูลขั้นต่ำ) รวมผลเฉพาะเชื้อจาก IPD_culture + OPD_culture และเชื่อมกับ IPD_data เพื่อเติมบริบท (เช่น เพศ อายุ ช่วงเวลา) ที่จำเป็นต่อการตรวจคุณภาพและการคำนวณภายหลังอ้างอิงโครงสร้างคอลัมน์ของแต่ละตารางตาม Data Dictionary และ Mapping ที่กำหนด

5.3.2 สรุปรายการคอลัมน์และเกณฑ์ตรวจรายการ (ย่อจาก Data Dictionary)

ตารางต่อไปนี้เป็นระบุดัชนีสำคัญ“ที่ใช้ตรวจสอบจริง”ในมุมมองDQA2มิติ (Completeness/Consistency) หลังการรวมตาราง

1. กลุ่มตารางผู้ป่วย (บริบทประกอบการตรวจ)

IPD_data – คีย์อ้างอิงและบริบททางคลินิก AN, VN, HN, Sex, Age, Adm_date/Adm_time, Dc_date/Dc_time, Spc/Spc_code, Ward_admit_code/Ward_dc_code ฯลฯ

1). ตรวจ Completeness (เฉพาะคีย์ที่ใช้เชื่อม เช่น AN/VN) และตรวจ Consistency ของชนิดข้อมูลเวลา/อายุ (อายุช่วงสมเหตุสมผล, วันเวลาแปลงได้/ไม่เป็นวันอนาคต) IPD_diag – บันทึกการวินิจฉัย: AN, VN, icd10_code, diag, dxtype

2). ใช้ประกอบการวิเคราะห์เชิงบริบท (ถ้ามีในเวิร์กโฟลว์); ตรวจ Consistency ของ icd10_code ตามรูปแบบมาตรฐาน

2. กลุ่มตารางผลแล็บพื้นฐาน (ใช้ตรวจเวลา/รหัสตัวอย่าง)

IPD_Cr / OPD_Cr – คีย์Specimenและเวลา barcode date_r, time_r

3. กลุ่มตารางเพาะเชื้อและความไวต่อยา (แกนคำนวณ)

IPD_culture/OPD_culture–คีย์แกนคำนวณ VN/AN(เฉพาะ IPD_culture), barcode, specimen, mb_code, org, dia, anti, anti_name, anti_result (S/I/R), anti_mic, date_r, time_r, lst, rep

1). ตรวจ Completeness/Consistency โดยเน้นคอลัมน์ที่จะถูกแปลงเป็นค่ามาตรฐานสำหรับคำนวณ:

- 1) org → org_clean (ชื่อเชื้อมาตรฐาน)
- 2) anti_name → anti_name_clean (ชื่อยามาตรฐาน INN)

- 3) anti_result \rightarrow anti_result_clean $\in \{S,I,R\}$
- 4) specimen \rightarrow specimen_key (โดเมนปิด เช่น {Urine, Sputum, Blood, Pus, Stool})

2. ตรวจสอบระยะเวลา (date_r/time_r) และความซ้ำซ้อนของ barcode.

5.3.3 การประเมินคุณภาพข้อมูล (DQA) – ตรวจสอบ “ทุกคอลัมน์” ด้วย 2 มิติ

DQA ครอบคลุมเพียง 2 มิติ (Completeness, Consistency)

1. ความครบถ้วน (Completeness): นับ/รายงานค่าที่ว่างของทุกคอลัมน์ โดยเฉพาะคอลัมน์บังคับ (org_clean, anti_name_clean, anti_result_clean, specimen_key) และคีย์เชื่อม (AN/VN/barcode) ที่จำเป็นต่อการอ้างอิง
2. ความสอดคล้อง (Consistency):
 - 1) โดเมนค่า: anti_result_clean $\in \{S,I,R\}$; specimen_key อยู่ในโดเมนที่กำหนด
 - 2) ตรรกะภายในแถว: มี anti_result_clean ต้องมี anti_name_clean และ org_clean เสมอ
 - 3) ระยะเวลา: date_r/time_r แปลงได้และไม่เป็นวัน/เวลาดังนอกขอบเขต
 - 4) ความซ้ำซ้อน/กุญแจ: barcode ซ้ำ, AN/VN เชื่อมแล้วต้องไม่เกิดเรคอร์ดซ้ำ

5.3.4 การวางแผนแก้ไข (Remediation Plan) จากผล DQA

1. Missing/Null: นิยามกฎ “ตัดออก” สำหรับแถวที่ขาดคอลัมน์บังคับ พร้อมบันทึกเหตุผล
2. Out-of-domain/ผิดรูป: จัดทำ mapping/กฎแปลง (เช่น sen \rightarrow S, Urinee \rightarrow Urine, cipro \rightarrow Ciprofloxacin, E.coli \rightarrow Escherichia coli)

3. Duplicates/Key Integrity: กำหนดกฎคัดเลือก (keep one) และบันทึก
ระเบียบที่ถูกตัด

5.3.5 การทำความสะอาดและแปลงข้อมูล (ETL/Cleaning)

ดำเนินการตามแผน 5.3.4 เพื่อสร้างคอลัมน์มาตรฐาน: org_clean, anti_name_clean, anti_result_clean, specimen_key และกำหนด row_status='ready' สำหรับแถวที่ผ่านเกณฑ์ทั้งหมด

5.3.6 การตรวจยืนยันหลังทำความสะอาด (Post-ETL Check)

รัน DQA แบบย่ออีกครั้งเพื่อตรวจว่า

1. Missing ของคอลัมน์บังคับ = 0
 2. ค่า out-of-domain = 0
 3. ซ้ำซ้อน/กุญแจเชื่อมโยงถูกจัดการครบถ้วน
- พร้อมบันทึก before/after count เพื่อการตรวจย้อนกลับ

5.4 เกณฑ์ยอมรับ (Acceptance Criteria)

1. DQA (Completeness & Consistency)
แถวที่จะเข้าสู่การคำนวณต้องไม่ขาดคอลัมน์บังคับ ได้แก่ org_clean, anti_name_clean, anti_result_clean, และ specimen_key
2. ค่า anti_result_clean ต้องอยู่ในโดเมนที่กำหนด {S, I, R}; ค่า specimen_key ต้องอยู่ในโดเมนที่กำหนด เช่น {Urine, Sputum, Blood, Pus, Stool}
3. คีย์อ้างอิงที่ใช้จริง (เช่น barcode, วันที่/เวลา) ต้องสามารถแปลงชนิดข้อมูลได้และไม่ขัดกับตรรกะเวลา (ไม่เป็นวันอนาคตหรือรูปแบบผิด)

Plan (Remediation Plan)

ปัญหาที่พบจากการตรวจสอบคุณภาพข้อมูล (Data Quality Assessment: DQA) ต้องถูกลงทะเบียนใน Issue Register และระบุวิธีแก้ไขอย่างชัดเจน เช่น การทำ mapping, การตัดทิ้ง, การรวมทะเบียนซ้ำ, หรือการแปลงรูปแบบ ก่อนเข้าสู่กระบวนการ Extract, Transform, Load (ETL) และการทำความสะอาดข้อมูล (Cleaning)

ETL/Cleaning ต้องกำจัดหรือรวมทะเบียนซ้ำตามคีย์ที่กำหนด และกำหนดค่า row_status = 'ready' เฉพาะแถวที่ผ่านเกณฑ์ทั้งหมด

Calc (การคำนวณ) สูตรการคำนวณ %S = $(S / (S + I + R)) \times 100$ ต้องถูกต้อง และมีการตรวจสอบ Intrinsic Resistance รวมถึง REVIEW flag (เช่น N < 30, MIC, No growth) เพื่อให้ผลลัพธ์มีความถูกต้องตามเกณฑ์ทางคลินิก

Export (การส่งออกผลลัพธ์) ต้องได้ไฟล์ Antibigram และซีตตามชนิดสิ่งส่งตรวจที่มีข้อมูลจริงโดยมีการจัดรูปแบบสีตามเกณฑ์ %S และ REVIEW flag ด้วย OpenPyXL เพื่อให้ผลลัพธ์เข้าใจง่ายและพร้อมใช้งานจริง

5.5 การเทียบกับรายงานอ้างอิง

จุดประสงค์: ตรวจสอบความสอดคล้องเชิงแนวโน้มของ %S ระหว่างผลที่คำนวณจากชุดข้อมูลหลัง DQA/ETL กับรายงานอ้างอิง

1. การทำให้คำจำกัดความตรงกัน (Harmonization)

- 1) Organism: ให้ org_clean ตรงกับชื่อที่รายงานกลางใช้ (เช่น *Escherichia coli*, *Klebsiella pneumoniae*, *A. baumannii* complex, *P. aeruginosa*, *S. aureus*).
- 2) Antibiotic: ให้ anti_name_clean ใช้ชื่อสามัญสากล (INN) ที่ตรงกับคอลัมน์ของรายงาน (เช่น Ciprofloxacin, Ceftriaxone, Imipenem, Meropenem, Amikacin, Piperacillin/Tazobactam, Co-trimoxazole, Nitrofurantoin).
- 3) Specimen: ให้ specimen_key อยู่ในโดเมนเดียวกับไฟล์อ้างอิง {All, Blood, Urine, Pus, Sputum, Stool}.

- 4) นิยามผล: ใช้ $S/I/R$ เหมือนกัน; $\%S = (S/(S+I+R)) \times 100$.
- 5) เกณฑ์จำนวนตัวอย่าง: คู่ที่ $n < 30$ (ฝั่งเรา/อ้างอิง) ให้ตีความเหตุและไม่นำไปสรุปเชิงนโยบาย.

2. ขั้นตอนคำนวณเพื่อนำไปเทียบ

- 1) ฝั่งเรา: คำนวณ $\%S$ ด้วย `groupby(['org_clean', 'anti_name_clean'])` (โหมด All) หรือ `groupby(['specimen_key', 'org_clean', 'anti_name_clean'])` (โหมดแยกชนิดสิ่งส่งตรวจ).
- 2) ฝั่งอ้างอิง: สรุป $\%S_{ref}$ และ n_{ref} ต่อ organism-antibiotic-specimen จากเอกสารเขต 1 (ช่วงเวลาเดียวกัน).
- 3) จับคู่ตารางด้วยคีย์ (`specimen_key`, `org_clean`, `anti_name_clean`) แล้วคำนวณ $\Delta\%S = \%S_{ours} - \%S_{ref}$ และคอลัมน์ Note (เช่น $N < 30$, หมายเหตุ mapping/ข้อสังเกต).

5.6 สรุปผลการทดสอบและข้อจำกัด

สรุปผลการทดสอบ Data Quality Assessment (DQA) → วางแผนแก้ไข → Extract, Transform, Load (ETL)/Cleaning → Export :

1. แฉวที่เข้าสู่การคำนวณผ่านเกณฑ์ Completeness/Consistency
2. ค่ามาตรฐาน `org_clean` / `anti_name_clean` / `anti_result_clean` / `specimen_key` ถูกต้องและสม่ำเสมอ
3. สูตร $\%S$ ถูกต้องและกันกรณิหารด้วยศูนย์
4. ไฟล์ Excel ผลลัพธ์มีโครงสร้าง/รูปแบบ/สีตามข้อกำหนด

ข้อจำกัด Data Quality Assessment (DQA) ครอบคลุมเพียง 2 มิติ (Completeness, Consistency) ตามสิ่งที่ใช้งานจริง และระบบยังไม่ถึงขั้น Dashboard ผลลัพธ์อยู่ในรูปแบบไฟล์ Excel และ Filter UI เท่านั้น

บทที่ 6

บทสรุปและข้อเสนอแนะ

6.1 บทสรุปการพัฒนาโครงการ

โครงการนี้พัฒนาบนGoogleColab(Python/pandas/openpyxl/rapidfuzz/ipywidgets) เพื่อแปรรูปข้อมูลเพาะเชื้อและผลความไวต่อยาปฏิชีวนะจากหลายแหล่งให้เป็น ตาราง แอนติไบแกรม (Antibiogram) ที่พร้อมใช้งาน โดยดำเนินงานตามลำดับ: Data Quality Assessment (DQA) → วางแผนแก้ไข (Remediation Plan) → Extract, Transform, Load (ETL)/Cleaning → คำนวณ %S → ส่งออก และจัดทำแนวทาง เทียบกับรายงานอ้างอิง เพื่อ ประเมินความสอดคล้อง ปัญหาและอุปสรรคที่พบระหว่างการดำเนินงาน

ในระหว่างการพัฒนา ยังพบอุปสรรคที่ส่งผลต่อกระบวนการและระยะเวลาในการทำงาน ดังนี้

1. รูปแบบข้อมูลที่ไม่เป็นมาตรฐาน (Unstandardized Data Format):

ข้อมูลจากห้องปฏิบัติการมีลักษณะไม่สอดคล้องกัน เช่น การสะกดชื่อเชื้อจุลชีพหรือชื่อยาที่หลากหลาย ทำให้ต้องใช้เวลาในการสร้างระบบ Mapping และ Dictionary เพื่อจัดกลุ่มและแปลงค่าที่เหมาะสม

2. ข้อจำกัดด้านความปลอดภัยของข้อมูล (Data Privacy and Access):

ข้อมูลส่วนบุคคล ทำให้ต้องมีการปกปิดข้อมูลและกำหนดสิทธิ์อย่างรัดกุม ซึ่งอาจส่งผลต่อการนำข้อมูลบางส่วนไปใช้ทดสอบในระบบจริง

3. การออกแบบให้สอดคล้องกับลักษณะการใช้งานจริง (User-centered Design):

จำเป็นต้องเก็บรวบรวมข้อเสนอแนะจากแพทย์และบุคลากรสาธารณสุข เพื่อนำมาปรับปรุงให้เหมาะสมทั้งในด้านความชัดเจนของข้อมูล

6.2 ปัญหาและข้อจำกัด

1. ขอบเขต Data Quality Assessment (DQA) : ประเมินเพียง Completeness และ Consistency ตามที่ใช้งานจริงในโค้ด ยังไม่ครอบคลุมมิติอื่น เช่น Accuracy, Timeliness, Accessibility และ Reliability
2. ความคลุมเครือของ Mapping: บางคำอาจเทียบได้มากกว่าหนึ่งค่า (เช่น ชื่อย่อย/ตัวย่อหลายแบบ) ทำให้ต้องพึ่งข้อมูลเพิ่มเติม

6.3 ข้อเสนอแนะ

ควรมีการทดสอบเพิ่มเติมกับกลุ่มแพทย์และบุคลากรสาธารณสุขในหลากหลายแผนก เพื่อประเมินความเหมาะสมข้อมูลที่แสดง

บรรณานุกรม

ศูนย์เฝ้าระวังการดื้อยาต้านจุลชีพแห่งชาติ (NARST), กรมวิทยาศาสตร์การแพทย์. Antibigrams แยกตาม เขตบริการสุขภาพ[ระบบออนไลน์].

แหล่งที่มา https://narst.dmsc.moph.go.th/Antibiogram_service

(วันที่สืบค้น 10 พฤษภาคม 2568).

Aramrat C, Boonma P. Development of data processing and visualization for bacterial and antibiotic susceptibility profile. Chiang Mai University Digital Collections [ระบบออนไลน์].

แหล่งที่มา <https://cmudc.library.cmu.ac.th/frontend/Info/item/dc%3A169623>

(วันที่สืบค้น 12 กรกฎาคม 2568).

NARST. รายงานประจำปีศูนย์เฝ้าระวังเชื้อดื้อยาต้านจุลชีพ พ.ศ. 2567 [ระบบออนไลน์].

แหล่งที่มา narst.dmsc.moph.go.th

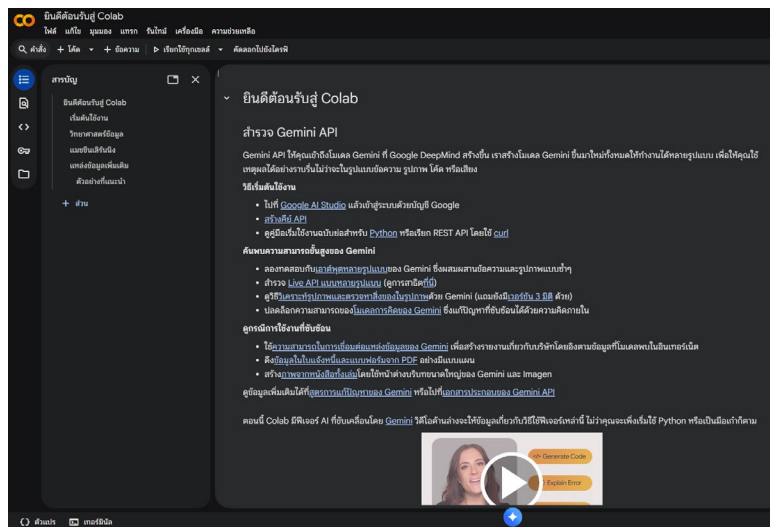
(วันที่สืบค้น 1 ตุลาคม 2568)

ภาคผนวก

ภาคผนวก ก

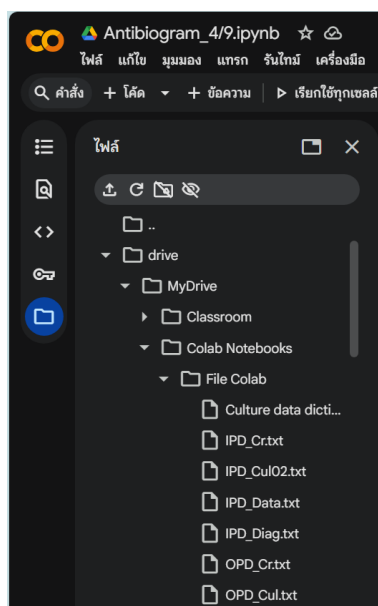
คู่มือการใช้งานสคริปต์และการส่งออก Antibioqram (Google Colab)

1 เปิดแพลตฟอร์ม Google Colab



รูปที่ ก.1 เปิดแพลตฟอร์ม Google Colab

2 เตรียมข้อมูล



รูปที่ ก.2 เตรียมไฟล์ข้อมูล

3 ตรวจสอบคุณภาพข้อมูลและสรุปเพื่อวางแผน “ก่อนทำความสะอาด”

IPD_Data table

ตาราง Consistency

Column	Column% Consistency หมายถึง % ของข้อมูลที่ไม่สอดคล้องกับ เกณฑ์	Out of Pattern (Consistency) ค่าผิดที่พบ	เกณฑ์ในการวัด Consistency
an (Admission Number)	-	-	an ไม่สามารถตรวจ Consistency ได้จากตัวเอง เพราะเป็นเพียงรหัสระบุตัวคนหรือการเข้ารับบริการ การตรวจ Consistency จะทำได้ก็ต่อเมื่อข้อมูล เหล่านี้ต้องไปสัมพันธ์กับแฟ้มอื่น
vn (Visit Number)	-	-	vn ไม่สามารถตรวจ Consistency ได้จากตัวเอง เพราะเป็นเพียงรหัสระบุตัวคนหรือการเข้ารับบริการ การตรวจ Consistency จะทำได้ก็ต่อเมื่อข้อมูล เหล่านี้ต้องไปสัมพันธ์กับแฟ้มอื่น
hn (Hospital Number)	-	-	hn ไม่สามารถตรวจ Consistency ได้จากตัวเอง เพราะเป็นเพียงรหัสระบุตัวคนหรือการเข้ารับบริการ การตรวจ Consistency จะทำได้ก็ต่อเมื่อข้อมูล เหล่านี้ต้องไปสัมพันธ์กับแฟ้มอื่น
sex (เพศ)	0.00% ของข้อมูลทั้งหมดทุกค่าเป็นค่าที่ สอดคล้องกับเกณฑ์	ค่าผิดที่พบ: ไม่มีข้อมูลใดที่ผิด	ต้องเป็น "ชาย" หรือ "หญิง" เท่านั้น
age (อายุ)	2.65% มีค่าที่ ไม่สอดคล้องกับเกณฑ์	ค่าผิดที่พบ: [np.int64(0)]	ต้องเป็นจำนวนเต็ม และอยู่ในช่วง 1-120
nat_code (รหัส ประเทศ)	มีค่าที่ ไม่สอดคล้องกับเกณฑ์ 0.39% ของข้อมูลทั้งหมด	ค่าผิดที่พบ: [เป็นค่าว่าง]	ต้องอยู่ในชุดรหัสที่โรงพยาบาลใช้(ข้อมูลต้องถูกต้อง ไม่พิจารณาการว่าง) เช่น '99.0' สำหรับไทย, '88.0' สำหรับพม่า ฯลฯ
nat (ชื่อประเทศ)	มีค่าที่ ไม่สอดคล้องกับเกณฑ์ 4.6% ของข้อมูลทั้งหมด	ค่าผิดที่พบ: [nan]	ต้องสัมพันธ์กับ nat_code (รหัสประเทศ) ตาม mapping ที่โรงพยาบาลกำหนด
ward_admit_code (รหัสหอผู้ป่วยตอน รับเข้า)	มีค่าที่ ไม่สอดคล้องกับเกณฑ์ 0.00% ของข้อมูลทั้งหมดทุกค่าเป็นค่าที่ สอดคล้องกับเกณฑ์	ค่าผิดที่พบ: ไม่มีข้อมูลใดที่ผิด	ต้องอยู่ในชุดรหัส ward Admit (หอที่รับผู้ป่วยเข้า) ที่ รพ. ใช้ รหัสที่ใช้แผนหอผู้ป่วย (ward) ภายในโรงพยาบาล แต่ละแห่ง เพื่อระบุว่าผู้ป่วยถูก Admit หรือ Discharge ที่ไหน

รูปที่ ก.3 ตรวจสอบคุณภาพข้อมูลและสรุปเพื่อวางแผน

4 Extract, Transform, Load /Cleaning

```
# --- STEP 3.1 Specimen Cleaning ---
df['specimen_key'] = df['specimen'].map(SPECIMEN_MAP).fillna('Unknown')

# --- STEP 3.2 Organism Cleaning ---
df['org_clean'] = df['organism'].map(MAP_TO_STANDARD_ORG)
# ใช้ regex/fuzzy matching สำหรับชื่อเชื้อที่ไม่ตรง dict

# --- STEP 3.2 Antibiotic Cleaning ---
df['anti_name_clean'] = df['antibiotic'].map(MAP_TO_STANDARD_ANTIBIOTIC)

# --- STEP 3.3 Result Cleaning ---
df['anti_result_clean'] = df['anti_result_raw'].replace({
    'S': 'S', 'I': 'I', 'R': 'R'
}).dropna()

# --- REVIEW flag ---
df['sir_flag'] = np.where(df['n_tested'] < 30, 'REVIEW', 'OK')
```

รูปที่ ก.4 ทำความสะอาดข้อมูล

5 การส่งออกข้อมูลตาราง Antibioqram (Excel)

Organism	TOTAL	n/Clavulanic #	Ampicillin (AMP)	Ceftazidime (CAZ)	Cefazolin (CFZ)	Ciprofloxacin (CIP)	Colistin (COL)	Farloxline (CF)riaxone (CFotaxime (CTripenem (D)Gmbutol (E)thromycin (R)apenem (E)tefepime (F)Efomycin (F)C				
Achromobacter Spp.	69			69.6 (48/69)	4.3 (3/69)			1.5 (1/65)				10.8 (7/65)
Acinetobacter Baumannii Comple	7992			23.1 (1848/7983)		24.2 (1923/7960)	99.5 (1013/1018)					
Acinetobacter Spp.	93			78.5 (73/93)		91.4 (85/93)	-					
Aeromonas Hydrophila	397	11.2 (43/385)		89.2 (297/333)	9.7 (6/62)	89.4 (353/395)			71.2 (37/52)	88.0 (293/333)	39.3 (22/56)	77.1 (290/376)
Bacteroides Spp.	1	-		-	-	-						-
Burkholderia Cepacia Complex	1069			94.7 (1012/1069)		-			-			-
Burkholderia Pseudomallei	65	-		100.0 (65/65)		-			-			-
Chryseobacterium Indologenes	9			-	-	-			-			-
Citrobacter Spp.	742	14.8 (86/580)		75.9 (488/643)	23.8 (15/63)	84.6 (602/712)			70.6 (48/68)	75.6 (487/644)	89.6 (60/67)	94.5 (683/723)
Cronobacter Dublinensis	2	-		-	-	-			-			-
Elizabethkingia Anophelis	22	-		-	-	-			-			-
Enterobacter Spp.	2851	0.6 (15/2556)	-	57.7 (1373/2379)	1.0 (2/204)	77.1 (2004/2600)			51.8 (115/222)	56.9 (1355/2381)	90.0 (198/220)	87.3 (2289/2621)
Enterococcus Faecalis	4727		99.0 (4677/4724)			42.1 (130/309)						83.7 (175/209)
Enterococcus Faecium	8143		5.0 (206/4132)			1.4 (5/364)						-
Enterococcus Spp.	324		76.9 (249/324)			-						-
Escherichia Coli	15399	69.9 (5966/14254)	14.1 (161/1141)	60.3 (7850/13013)	44.3 (497/1123)	42.6 (6060/14212)			49.7 (593/1193)	53.1 (6918/13023)	93.1 (1139/1224)	95.8 (13757/14360)
Gram Negative Bacilli	7	-		-	-	-						-
Haemophilus Influenzae	805	96.9 (775/800)	71.2 (571/802)			97.1 (776/799)			99.6 (799/802)			-
Klebsiella Aerogenes	697	0.8 (5/660)		70.9 (450/635)	3.0 (1/33)	89.7 (600/665)			53.8 (21/39)	69.0 (438/635)	88.6 (31/35)	96.0 (645/672)
Klebsiella Oxytoca	309	94.1 (273/290)	R	93.7 (251/268)	-	91.7 (264/288)			92.3 (267/268)			98.3 (286/291)
Klebsiella Pneumoniae	12213	57.4 (5081/8847)	R	55.0 (4349/7908)	36.5 (302/827)	55.8 (4925/8821)			42.8 (382/893)	53.7 (4245/7912)	53.0 (670/1263)	76.1 (7627/10022)
Leclercia Adecarbonylata	4	-	-	-	-	-			-	-	-	-
Moraxella Catarrhalis	271	100.0 (35/35)							100.0 (35/35)			99.6 (233/234)
Moraxella Moraxii	781	0.6 (15/773)	1.8 (11/56)	82.1 (584/711)	1.8 (11/56)	76.6 (582/770)			67.8 (40/96)	79.6		96.5 (55/57)

รูปที่ ก.5 การส่งออกข้อมูลตาราง Antibioqram

Organism	TOTAL	/Clavulanic .npicillin (AMftazidime (Cefazolin (CFzrofloxacn (Colistin (COLtrioxone (CFotaxime (CTripenem (DCthromycin (Etapenem (Etefepime (FEfomycin (FC									
Acinetobacter Baumannii Comple	57		5.3 (3/57)	8.9 (5/56)	-						
Acinetobacter Spp.	2		-	-							
Burkholderia Cepacia Complex	2		-	-							
Chryseobacterium Indologenes	1		-	-							
Citrobacter Spp.	16	-		-	-						
Enterobacter Spp.	32	0.0 (0/30)	-	-	43.3 (13/30)					59.4 (19/32)	-
Enterococcus Faecalis	121		98.3 (119/121)		-						
Enterococcus Faecium	73		3.8 (2/52)		-						
Enterococcus Spp.	7		-		-						
Escherichia Coli	166	59.4 (95/160)	16.2 (6/37)	25.0 (30/120)	37.8 (14/37)	18.2 (29/159)	42.5 (17/40)	17.5 (21/120)	94.9 (37/39)	96.2 (153/159)	73.0 (27/37)
Klebsiella Aerogenes	1	-		-	-	-					
Klebsiella Oxytoca	3			-	-	-					
Klebsiella Pneumoniae	62	65.0 (26/40)	R	48.4 (15/31)	-	46.2 (18/39)	-	48.4 (15/31)	-	69.8 (37/53)	-
Morganella Morganii	6	-		-	-	-					
Proteus Mirabilis	9	-	-	-	-	-					
Proteus Spp.	5	-	-	-	-	-					
Providencia Rettgeri	1	-		-	-	-					
Pseudomonas Aeruginosa	65			53.8 (35/65)	52.3 (34/65)	100.0 (34/34)			51.7 (31/60)		
Pseudomonas Spp.	1			-	-	-					
Salmonella Spp.	1	-		-	-	-					
Serratia Marcescens	6	R		-	-	-					
Staphylococcus Aureus	6			-	-	-					
Stenotrophomonas Maltophilia	35			-	-	-					
Streptococcus Analaetiae	2		-	-	-	-					

รูปที่ ก.6 การส่งออกข้อมูลตาราง Antibioqram แยกตามสิ่งส่งตรวจตามที่ใช้ต้องการ

ภาคผนวก ข

โครงสร้างข้อมูลหลักและการทดสอบกระบวนการ

(Key Data Fields & Process Testing)

1 คีย์ฟิลด์ที่ระบบใช้คำนวณ

Field	Meaning	Domain	Required?	Example
org_clean	ชื่อเชื้อที่ผ่านการทำความสะอาด/มาตรฐาน	(ชื่อเชื้อมาตรฐาน)	Yes	Escherichia coli
anti_name_clean	ชื่อยาปฏิชีวนะที่ผ่านการทำความสะอาด/มาตรฐาน	(INN/ATC/local standard)	Yes	Ciprofloxacin
anti_result_clean	ผลความไวต่อยา (มาตรฐาน)	(S, I, R)	Yes	S
specimen_key	กลุ่มชนิดสิ่งส่งตรวจ (มาตรฐาน)	(Urine, Sputum, Blood, Pus, Stool)	Yes	Urine

รูปที่ ข.1 คีย์ฟิลด์ที่ใช้ในการคำนวณ Antibigram และโดเมนของข้อมูล

2 ฟิลด์ช่วยตรวจ/เชื่อม

Field	Meaning	Rule/Domain	Required?	Example	Note
barcode	รหัสตัวอย่าง/สิ่งส่งตรวจ	unique in context	No	23-UR-000123	ตรวจ duplicate/traceback
date_r	วันที่เก็บตัวอย่าง	YYYY-MM-DD; ≤ today	No	21/9/2024	ตรวจตรวจเวลา/ความครบถ้วน
time_r	เวลาเก็บตัวอย่าง	HH:MM (24h)	No	13:45	ตรวจตรวจเวลา/ความครบถ้วน

รูปที่ ข.2 ใช้ตรวจ duplicate/ตรวจวันเวลา และช่วยตามรอยปัญหาใน Data Quality Assessment

3 ทดสอบ Data Quality Assessment → ETL/Cleaning → คำนวณ %S → Export Antibigram

หมวด	วัตถุประสงค์	เงื่อนไข/อินพุต	ผลคาดหวัง	หลักฐาน
DQA-Completeness	คัดกรองข้อมูลที่มีทั้งส่วนหัว	org_clean / anti_name_clean / anti_result_clean / specimen_key ว่าง	แถวถูกคัดออกจากการคำนวณ	logs/invalid_completeness_all.txt
DQA-Consistency	บังคับโดเมน	anti_result_clean ∈ {S,I,R}, specimen_key ∈ {urine,sputum,blood,pus,stool}	แถวหลุดโดเมนถูกกรองงาน	logs/invalid_consistency_all.txt
ETL/Cleaning	แมปค่าดิบให้เป็นค่า clean	ตัวอย่าง: E coli → Escherichia coli, cipro → Ciprofloxacin, urine → urine, sen → S	ได้ค่า clean มาตรวจครบ 4 คลินิค	/mapping/map_*.csv
CALC	ตรวจสอบสูตรคำนวณ	$\%S = (S/(S+I+R)) \times 100$, n_tested = S+I+R	ไม่มีหารศูนย์, แสดง xx.x (n)	สกรีนช็อต/โน้ตบุ๊ก
EXPORT	ตรวจสอบแบบไฟล์ผลลัพธ์	มี All + มีตาม specimen ที่มีข้อมูลจริง	Antibiogram_All_and_BySpecimen.xlsx มีสีตามเกณฑ์	/outputs/*.xlsx

รูปที่ ข.3 ทดสอบ Data Quality Assessment → ETL/Cleaning → คำนวณ %S → Export Antibigram

ภาคผนวก ค

-ร่าง-

บทความฉบับสมบูรณ์

เพื่อเข้าร่วม

งานประชุมวิชาการระดับชาติ วิทยาศาสตร์ เทคโนโลยีและนวัตกรรม ครั้งที่ 7

<https://sciencebase.mju.ac.th/CSTI2026>

การวิเคราะห์คุณภาพข้อมูลและการจัดทำแดชบอร์ดสำหรับแสดงข้อมูลความไวของเชื้อต่อยาปฏิชีวนะ

Assessment of Data Quality and Development of an Interactive Dashboard for Presenting Antibigram Results

สมนึก สินธุปวน, พาสน์ ปราโมกษ์ชน*, อลงกต กองมณี และ กิตติกร หาญตระกูล

Somnuek Sinthupuan, Part Pramokchon*, Alongkot Gongmanee and Kittikorn Hantrakul

สาขาวิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยแม่โจ้ ประเทศไทย 50290

* ผู้นิพนธ์ประสานงาน: ผศ.ดร.สมนึก สินธุปวน อีเมล: somnuk@mju.ac.th

บทคัดย่อ

ในการบริหารจัดการระบบบริการสุขภาพข้อมูลถือเป็นทรัพยากรสำคัญที่มีผลต่อการตัดสินใจของบุคลากรทางการแพทย์และการกำหนดนโยบายด้านสาธารณสุขโดยเฉพาะข้อมูลที่เกี่ยวข้องกับการเพาะเชื้อและการใช้ยาปฏิชีวนะซึ่งเป็นข้อมูลพื้นฐานในการประเมินสถานการณ์เชื้อดื้อยา ปัญหาสำคัญที่มักพบคือข้อมูลที่มีคุณภาพต่ำ เช่น ข้อมูลซ้ำซ้อน ข้อมูลไม่ครบถ้วน หรือไม่สอดคล้องกันระหว่างแฟ้มข้อมูล ส่งผลให้การวิเคราะห์ข้อมูลและการสื่อสารผลต่อทีมรักษาไม่แม่นยำ โครงการนี้มีวัตถุประสงค์เพื่อพัฒนาเครื่องมือสำหรับประเมินคุณภาพของข้อมูลทางคลินิก โดยอิงจากเกณฑ์มาตรฐาน 7 ด้าน ได้แก่ ความถูกต้อง, ความเชื่อถือได้, ความสอดคล้อง, ความครบถ้วน, ความเกี่ยวข้อง, ความสามารถในการเข้าถึง และความทันเวลา ข้อมูลที่ผ่านการประเมินแล้วจะถูกจัดการผ่านกระบวนการ ETL (Extract, Transform, Load) เพื่อให้พร้อมต่อการวิเคราะห์และแสดงผลผ่านแดชบอร์ดแบบโต้ตอบ (Interactive Dashboard) ที่พัฒนาขึ้นสามารถแสดงข้อมูล antibiogram ซึ่งเป็นตารางสรุปเปอร์เซ็นต์ความไวของเชื้อแบคทีเรียแต่ละชนิดต่อยาปฏิชีวนะ โดยผู้ใช้งานสามารถเลือกดูข้อมูลเฉพาะกลุ่มได้ตามชนิดสิ่งส่งตรวจ อายุ เพศ หน่วยรักษา หรือเลือกเชื้อและยาเฉพาะเพื่อวิเคราะห์แนวโน้มการดื้อยาได้อย่างมีประสิทธิภาพ นอกจากนี้ ระบบยังสามารถตรวจสอบปัญหา

คุณภาพข้อมูล เช่น ค่า Missing, ข้อมูลซ้ำซ้อน และข้อมูลผิดรูปแบบ พร้อมรายงานผลแยกตามแฟ้ม และฟิลด์ข้อมูล ผลลัพธ์ของโครงการนี้คาดว่าจะจะเป็นประโยชน์ต่อโรงพยาบาลในการยกระดับคุณภาพของข้อมูลผู้ป่วย เพิ่มความแม่นยำในการวิเคราะห์การใช้จ่ายปฏิชีวนะ และสนับสนุนการตัดสินใจด้านบริการสุขภาพอย่างมีประสิทธิภาพ พร้อมทั้งช่วยให้นักศึกษาได้รับประสบการณ์ตรงในการพัฒนาเครื่องมือวิเคราะห์ข้อมูลที่สามารถนำไปประยุกต์ใช้ได้จริงในระบบสาธารณสุข

ผลลัพธ์จากการพัฒนาระบบงาน พบว่าระบบสามารถตรวจสอบคุณภาพข้อมูลทางคลินิกได้อย่างมีประสิทธิภาพ โดยใช้เกณฑ์การประเมินคุณภาพข้อมูล ซึ่งช่วยให้สามารถระบุปัญหาที่เกิดขึ้นจริง และ กระบวนการ ETL ที่พัฒนาขึ้นสามารถทำความสะอาดและแปลงข้อมูลให้อยู่ในรูปแบบมาตรฐาน ผลลัพธ์ที่ได้ถูกส่งออกเป็นไฟล์ Excel ที่มีการจัดรูปแบบสีตามเกณฑ์ %S และ REVIEW flag รวมถึงสามารถกรองข้อมูลตามเงื่อนไขที่ผู้ใช้เลือกผ่าน Filter UI ทำให้ผู้ใช้งานสามารถเข้าถึงข้อมูล antibiogram ได้อย่างสะดวกและแม่นยำมากขึ้น ทั้งนี้ระบบยังช่วยเพิ่มความน่าเชื่อถือของข้อมูล ลดความผิดพลาดในการวิเคราะห์ และสนับสนุนการตัดสินใจด้านการใช้จ่ายปฏิชีวนะในโรงพยาบาลได้อย่างมีประสิทธิภาพ

คำสำคัญ: Data Quality Assessment, ETL, Antibiogram, Filter UI, Python, การใช้จ่ายปฏิชีวนะ, เชื้อดื้อยา

Abstract

In managing the healthcare system, data is considered an important resource that impacts the decision-making of medical personnel and the formulation of public health policies, especially data related to pathogen cultivation and the use of antibiotics, which is fundamental information for assessing the situation of drug-resistant infections. An important issue that is often encountered is low-quality data, such as redundant data, incomplete data, or inconsistent data between datasets, which results in inaccurate data analysis and communication of results to the maintenance team. This project aims to develop a tool for assessing the quality of clinical data based on seven standard criteria: accuracy, reliability, consistency, completeness, relevance, accessibility, and timeliness. Data that has been evaluated will be managed through the ETL (Extract, Transform, Load) process to prepare it for analysis and presentation

through an Interactive Dashboard that has been developed to display antibiogram data, which is a summary table of the percentage sensitivity of each bacterial strain to antibiotics. Users can selectively view information by type of submitted materials, age, gender, treatment unit, or choose specific pathogens and medications to effectively analyze drug resistance trends. Additionally, the system can check for data quality issues such as missing values, duplicate data, and incorrectly formatted data, along with reporting results categorized by file and data field. The results of this project are expected to benefit hospitals in improving the quality of patient data, increasing the accuracy of antibiotic usage analysis, and supporting effective decision-making in health services, while also providing students with hands-on experience in developing data analysis tools that can be practically applied in the public health system.

Results from system development. The results show that the system can effectively monitor the quality of clinical data and identify practical problems through data quality evaluation standards. The ETL process developed can clean up the data and convert it into standard format, output the results as excel files in color format based on% s standard and REVIEW mark, and filter the data according to the conditions selected by users through the filter UI, so that users can access the antibacterial chart data more conveniently and accurately, and at the same time, the reliability of the data is improved, the analysis errors are reduced, and the support is effective.

Keywords: Web Data Quality Assessment, ETL, Antibiogram, Filter UI, Python, Antibiotic Usage, Drug-resistant Bacteria

ที่มาและปัญหา

การเพาะเชื้อเพื่อระบุชนิดของเชื้อโรคในผู้ป่วยมักจะใช้เวลาประมาณ 3-5 วัน เพื่อให้ได้ผลลัพธ์ที่ชัดเจนเกี่ยวกับชนิดของเชื้อที่เป็นสาเหตุของการติดเชื้อในร่างกายผู้ป่วยซึ่งในระหว่างที่ต้องรอผลการเพาะเชื้อเหล่านี้ ผู้ป่วยที่มีภาวะติดเชื้อจำเป็นต้องได้รับการรักษาด้วยยาปฏิชีวนะ

(antibiotics) ทันทีเพื่อหยุดยั้งการแพร่กระจายของเชื้อและลดความเสี่ยงต่อภาวะแทรกซ้อนที่อาจเกิดขึ้น

อย่างไรก็ตาม การให้ยาปฏิชีวนะในช่วงที่ไม่ทราบชนิดของเชื้อเป็นการให้ยาแบบ "อาการเบื้องต้น"(empirical treatment) ซึ่งอาจไม่ได้ผลเต็มที่กับเชื้อที่เกิดขึ้น เนื่องจากยาปฏิชีวนะบางตัวอาจไม่สามารถฆ่าเชื้อบางชนิดได้ การเลือกใช้ยาฆ่าเชื้อที่มีประสิทธิภาพสูงสุดในระหว่างการรอผลเพาะเชื้อจึงเป็นสิ่งสำคัญมาก เพราะจะช่วยให้การรักษาผู้ป่วยเป็นไปได้อย่างมีประสิทธิภาพและลดการใช้ยาผิดประเภทที่อาจทำให้เกิดผลข้างเคียงหรือการดื้อยาได้

การสร้าง แอนติไบโอแกรม (Antibiogram) เป็นเครื่องมือที่สำคัญในการช่วยให้การเลือกยาปฏิชีวนะในช่วงนี้มีความแม่นยำมากขึ้น โดย จะช่วยระบุข้อมูลจากการเพาะเชื้อในอดีตว่า เชื้อโรคแต่ละชนิดตอบสนองต่อยาปฏิชีวนะชนิดใดได้ดีที่สุด ซึ่งข้อมูลเหล่านี้จะช่วยให้แพทย์เลือกให้ยาปฏิชีวนะที่มีประสิทธิภาพในการฆ่าเชื้อได้รวดเร็วขึ้น ลดเวลาที่ผู้ป่วยต้องเสี่ยงกับการติดเชื้อที่ไม่สามารถควบคุมได้ นอกจากนี้ การรู้ข้อมูลเกี่ยวกับความไวของเชื้อต่อยาที่มีประสิทธิภาพในช่วงนี้ยังช่วยลดการใช้ยาฆ่าเชื้อที่ไม่เหมาะสม ซึ่งอาจส่งผลกระทบต่อผู้ป่วยในระยะยาว เช่น การดื้อยา หรือการเกิดเชื้อดื้อยา

ด้วยเหตุนี้ แอนติไบโอแกรม (Antibiogram) จึงเป็นเครื่องมือที่สำคัญในการช่วยเลือกให้ยาปฏิชีวนะในระหว่างการรอผลเพาะเชื้อ และมีบทบาทในการพัฒนาการรักษาผู้ป่วยอย่างมีประสิทธิภาพ โดยไม่ต้องรอผลการเพาะเชื้อจนเสี่ยงต่อการเกิดภาวะแทรกซ้อนต่างๆ ที่อาจเกิดขึ้นจากการใช้ยาผิดประเภท

โครงการนี้จึงมีเป้าหมายเพื่อพัฒนาเครื่องมือสำหรับประเมินคุณภาพของข้อมูล การประเมินคุณภาพข้อมูล (DQA) ในโครงการนี้มี 7 โดเมนหลักที่ใช้ในการตรวจสอบคุณภาพข้อมูลทางคลินิก ได้แก่ Accuracy, Reliability, Consistency, Completeness, Relevance, Accessibility, และ Timeliness ซึ่งการประเมินคุณภาพข้อมูล (DQA) จะเน้นไปที่การประเมิน คุณภาพของข้อมูล ในบางมิติเท่านั้นโดยที่ในการประเมินข้อมูลในโครงการนี้จะใช้เฉพาะ 2 โดเมนหลัก ได้แก่ Consistency (ความสอดคล้องกันของข้อมูล) และ Completeness(ความครบถ้วนของข้อมูล) เพื่อให้มั่นใจว่าข้อมูลที่ใช้ในการสร้าง แอนติไบโอแกรม (Antibiogram) นั้นมีความสอดคล้องและครบถ้วน ซึ่งจะช่วยให้การประมวลผลข้อมูลและการตัดสินใจ เป็นไปอย่างแม่นยำและมีประสิทธิภาพ

โดยการใช้ทั้งสองโดเมนนี้ในการตรวจสอบข้อมูลจะช่วยให้ได้ข้อมูลที่มีคุณภาพสำหรับการสร้างแอนติไบโอแกรม (Antibiogram) และสามารถนำไปใช้ในการตัดสินใจทางการแพทย์ได้อย่าง

ถูกต้องและมีประสิทธิภาพ เมื่อข้อมูลผ่านกระบวนการตรวจสอบแล้ว จะถูกนำเข้าสู่กระบวนการจัดการข้อมูลแบบ ETL (Extract, Transform, Load) เพื่อแปลงให้อยู่ในรูปแบบที่พร้อมใช้งานและวิเคราะห์ต่อได้สะดวก โดยเฉพาะข้อมูลเกี่ยวกับการเพาะเชื้อและการใช้ยาปฏิชีวนะ

การออกแบบระบบทั้งหมดนี้มีจุดมุ่งหมายเพื่อให้แพทย์หรือหน่วยงานสาธารณสุขสามารถเข้าถึงข้อมูลที่มีคุณภาพได้อย่างสะดวก ช่วยเสริมสร้างประสิทธิภาพในการวิเคราะห์ข้อมูล ส่งเสริมการใช้ยาปฏิชีวนะอย่างเหมาะสม และลดความเสี่ยงของการเกิดเชื้อดื้อยาในระยะยาว

ผู้จัดทำเห็นว่า การพัฒนาเครื่องมือที่สามารถประเมินคุณภาพข้อมูล และสามารถนำเสนอผลได้อย่างเป็นระบบ จะช่วยเสริมประสิทธิภาพในการบริหารจัดการข้อมูลทางการแพทย์ โดยสนับสนุนการใช้ข้อมูลที่มีความถูกต้อง ครบถ้วน และนำไปสู่การตัดสินใจที่มีประสิทธิภาพ ช่วยเพิ่มประโยชน์สูงสุดทั้งในด้านการรักษาผู้ป่วยและการพัฒนาระบบสุขภาพโดยรวม

วัตถุประสงค์

โครงการนี้มีวัตถุประสงค์เพื่อพัฒนาเครื่องมือสำหรับวิเคราะห์และแสดงข้อมูลผู้ป่วยใน โดยมีเป้าหมายหลักดังนี้

1. เพื่อประเมินและตรวจสอบคุณภาพของข้อมูลผู้ป่วยในจากฐานข้อมูลโรงพยาบาล โดยอิงตามเกณฑ์มาตรฐาน 2 ด้าน ได้แก่ Consistency (ความสอดคล้องกันของข้อมูล) และ Completeness (ความครบถ้วนของข้อมูล)
2. เพื่อออกแบบและพัฒนาระบบการ ETL (Extract, Transform, Load) สำหรับรวบรวม แปลง และจัดเตรียมข้อมูลจากหลายแหล่งให้พร้อมต่อการใช้งาน
3. เพื่อสร้าง ต้นแบบ (Prototype) สำหรับแสดงข้อมูลสำคัญเกี่ยวกับผู้ป่วย เช่น จำนวนผู้ป่วย ผลตรวจทางห้องปฏิบัติการ ผลการเพาะเชื้อ และผลการดื้อยา
4. เพื่อให้ระบบที่พัฒนาขึ้นสามารถนำไปประยุกต์ใช้ในโรงพยาบาลได้จริง ทั้งในด้านการวิเคราะห์คุณภาพข้อมูล การติดตามแนวโน้มการรักษา และการสนับสนุนการตัดสินใจของบุคลากรทางการแพทย์

การวิเคราะห์และออกแบบระบบ:

โครงการนี้มุ่งเน้นการจัดการข้อมูลการเพาะเชื้อและผลการทดสอบยาปฏิชีวนะ โดยใช้กระบวนการ Data Quality Assessment (DQA) และ Extract, Transform, Load (ETL) เพื่อให้

ข้อมูลอยู่ในรูปแบบมาตรฐาน พร้อมทั้งรองรับการคำนวณ Antibigram และการส่งออกผลลัพธ์เป็นไฟล์ Excel รวมถึงการกรองข้อมูลผ่าน Filter UI

Use Case หลักของระบบ:

1. นำเข้าข้อมูล (Extract)
2. ประเมินคุณภาพข้อมูล (DQA)
3. ทำความสะอาดและแปลงข้อมูล (Transform)
4. คำนวณ Antibigram (%S, Intrinsic Resistance, REVIEW flag)
5. ส่งออกผลลัพธ์เป็น Excel
6. กรองข้อมูลด้วย Filter UI (Sex, Age, Specimen, Organism, Year)

ตารางข้อมูลหลัก:

1. IPD_Data: ข้อมูลผู้ป่วย (HN, Gender, Age, Admit_Date, Discharge_Date)
2. IPD_Culture / OPD_Culture: ข้อมูลการเพาะเชื้อและผลความไวต่อยา (Specimen, Organism, Antibiotic, S/I/R)
3. Mapping Tables: org_clean, anti_name_clean, specimen_key

ขั้นตอน ETL:

1. Extract: อ่านข้อมูลจากไฟล์ CSV/Excel
2. Transform: Cleaning → normalize organism/specimen, ตรวจสอบ Intrinsic Resistance, REVIEW flag
3. Load: ส่งออกเป็นไฟล์ Excel พร้อมสีตามเกณฑ์ %S และ REVIEW flag

ผลการทดสอบระบบ

จากผลการทดสอบยืนยันว่าระบบที่พัฒนาขึ้นสามารถตอบสนองต่อความต้องการใช้งานจริงของบุคลากรทางการแพทย์ได้อย่างมีประสิทธิภาพ ช่วยเพิ่มความน่าเชื่อถือของข้อมูล ลดความผิดพลาดในการวิเคราะห์ และสนับสนุนการตัดสินใจด้านการใช้ยาปฏิชีวนะในโรงพยาบาล

วิจารณ์และสรุปผล

ระบบที่พัฒนาขึ้นถือเป็นเครื่องมือที่สามารถสนับสนุนการตัดสินใจด้านการใช้ยาปฏิชีวนะในโรงพยาบาลได้อย่างมีประสิทธิภาพ ผลการทดสอบยืนยันว่าระบบสามารถลดความผิดพลาดในการวิเคราะห์ เพิ่มความน่าเชื่อถือของข้อมูล และช่วยให้บุคลากรทางการแพทย์เข้าถึงข้อมูลที่มีความถูกต้องและสอดคล้องกับมาตรฐาน ทั้งนี้ หากมีการพัฒนาต่อยอดในอนาคตเพื่อประเมินคุณภาพข้อมูลที่หลากหลายมาก ระบบนี้จะเป็นประโยชน์ต่อการจัดการข้อมูลด้านสาธารณสุขมากยิ่งขึ้น

กิตติกรรมประกาศ

โครงการงานวิเคราะห์คุณภาพข้อมูลและการจัดทำแดชบอร์ดสำหรับแสดงข้อมูลความไวของเชื้อต่อยาปฏิชีวนะ สำเร็จลุล่วงตามวัตถุประสงค์ ด้วยความกรุณาจาก ผู้ช่วยศาสตราจารย์ ดร.สมนึก สีนธูปวน ประธานที่ปรึกษาโครงการงานสหกิจ ผู้ช่วยศาสตราจารย์ ดร.พาสน์ ปราโมกษ์ชน และ อาจารย์อลงกต กองมณี กรรมการที่ปรึกษาโครงการงาน อาจารย์นายแพทย์ ฉันทชนก อร่ามรัตน์ ผู้ให้แนวคิด คำปรึกษาในทุกกระบวนการของการดำเนินโครงการ การปรับปรุงแก้ไขข้อบกพร่องต่างๆ ในครั้งนี้จนทำให้โครงการสมบูรณ์ ด้วยความเอาใจใส่อย่างยิ่ง ผู้จัดทำขอขอบพระคุณเป็นอย่างสูงไว้ ณ ที่นี้ด้วย

เอกสารอ้างอิง

- [1] ศูนย์เฝ้าระวังการดื้อยาต้านจุลชีพแห่งชาติ (NARST), กรมวิทยาศาสตร์การแพทย์. Antibigrams แยกตาม เขตบริการสุขภาพ[ระบบออนไลน์].
แหล่งที่มา https://narst.dmsc.moph.go.th/Antibiogram_service
(วันที่สืบค้น 10 พฤษภาคม 2568)
- [2] ข้อมูลบัญชีสหกรณ์การเกษตร [ระบบออนไลน์].
แหล่งที่มา https://www.cad.go.th/main.php?filename=accounting_system
(วันที่สืบค้น 16 กรกฎาคม 2566)

ประวัติผู้เขียน



1. ชื่อ นายจิตติพัฒน์ สนม
2. วัน เดือน ปีเกิด วันที่ 1 เดือน เมษายน พ.ศ. 2546
3. ศาสนา พุทธ สัญชาติ ไทย เชื้อชาติ ไทย
4. ภูมิลำเนา 18 หมู่ 6 ตำบลจิม อำเภอปง จังหวัดพะเยา
5. ที่อยู่ปัจจุบัน 196/36 ถนน มหิตล ตำบลป่าแดด อำเภอเมืองเชียงใหม่ จังหวัดเชียงใหม่
6. หมายเลขโทรศัพท์ 093-5809356
7. อีเมล Jittiphatwork@gmail.com
8. ประวัติการศึกษา
 - ปีการศึกษา 2561 สำเร็จการศึกษาระดับมัธยมศึกษาตอนต้น
โรงเรียนเฉลิมพระเกียรติฯ พะเยา ต.ห้วยแก้ว อ.ภูกามยาว จ.พะเยา
 - ปีการศึกษา 2564 สำเร็จการศึกษาระดับมัธยมศึกษาตอนปลาย
โรงเรียนเฉลิมพระเกียรติฯ พะเยา ต.ห้วยแก้ว อ.ภูกามยาว จ.พะเยา
 - ปีการศึกษา 2568 สำเร็จการศึกษาระดับปริญญาตรี สาขาวิชาวิทยาการคอมพิวเตอร์
คณะวิทยาศาสตร์ มหาวิทยาลัยแม่โจ้ อำเภอสันทราย จังหวัดเชียงใหม่
9. ประสบการณ์/ประวัติการฝึกอบรม/การฝึกงาน
 - ปี พ.ศ.2565 เข้าร่วมกิจกรรมค่ายยุวชนคอมพิวเตอร์ ครั้งที่ 20 มหาวิทยาลัยแม่โจ้
 - ปี พ.ศ. 2568 ฝึกงานที่ศูนย์วิจัย Global Health Research Center CMU ประเพณีสหกิจ

10. ผลงานทางวิชาการ

-

11. ทักษะและความสามารถทางคอมพิวเตอร์

- มีความรู้พื้นฐานทางด้านการเขียนโปรแกรมในภาษา HTML, Python, SQL, HTML, CSS, Vue.js
- สามารถใช้โปรแกรมพื้นฐานทางคอมพิวเตอร์ต่าง ๆ ได้ เช่น PowerPoint, Word, Excel, Adobe Photoshop และ Adobe Premiere Pro

สถานประกอบการ

1. ข้อมูลสถานประกอบการ

1.1 ข้อมูลของศูนย์วิจัย Global Health Research Center CMU

ศูนย์วิจัย Global Health Research Center CMU ก่อตั้งขึ้นในปี พ.ศ. 2565 มีพันธกิจเพื่อส่งเสริมและดำเนินงานวิจัยด้านสุขภาพในระดับระดับโลก โดยเฉพาะ

1. มลพิษทางสิ่งแวดล้อม (เช่น มลพิษทางอากาศ)
2. โรคติดเชื้อและเชื้อดื้อยา
3. ปัจจัยทางสังคมที่ส่งผลต่อสุขภาพประชากร
4. ระบบบริการสุขภาพ และสุขภาพในระดับชุมชน

ศูนย์ทำงานวิจัยแบบบูรณาการ ร่วมมือกับมหาวิทยาลัยและองค์กรวิจัยนานาชาติ

1.2 สถานที่ตั้งสถานประกอบการ

ศูนย์วิจัยสุขภาพโลก (GHRC) คณะแพทยศาสตร์ มหาวิทยาลัยเชียงใหม่ อาคารราชนครินทร์ (The 50th Anniversary Building) ชั้น 8 เลขที่ 110 ถนนอินทวิโรด ตำบลศรีภูมิ อำเภอเมืองเชียงใหม่ จังหวัดเชียงใหม่ 50200

2. งานที่ได้รับมอบหมายระหว่างปฏิบัติสหกิจศึกษา

ในการปฏิบัติงานสหกิจศึกษาที่ศูนย์วิจัย Global Health Research Center CMU ผู้ปฏิบัติงานสหกิจศึกษาได้ปฏิบัติงานในโครงการที่ได้รับมอบหมาย คือ การวิเคราะห์คุณภาพข้อมูลและการจัดทำข้อมูลความไวของเชื้อต่อยาปฏิชีวนะสำหรับการแสดงข้อมูลความไวของเชื้อต่อยาปฏิชีวนะ โดยจะแสดงเป็นตาราง แอนติไบโอแกรม(Antibiogram) ในระหว่างวันที่23 มิถุนายน พ.ศ 2568 ถึงวันที่ 31 ตุลาคม พ.ศ. 2568 ซึ่งผู้ปฏิบัติงานสหกิจศึกษาได้รับมอบหมายในการทำงานศึกษา และพัฒนาระบบ ดังนี้

สัปดาห์ที่ 1 ได้ศึกษา Data Quality Assessment(DQA) เพื่อนำไปใช้พัฒนาการวิเคราะห์คุณภาพข้อมูลและการจัดทำแดชบอร์ดสำหรับแสดงข้อมูลความไวของเชื้อต่อยาปฏิชีวนะ

วิธีการศึกษา

1. ศึกษาด้วยตนเองในเบื้องต้น และนำมาสรุปให้พี่เลี้ยงเพื่อเรียนรู้ร่วมกัน
2. พี่เลี้ยงมอบหมายงานให้ปฏิบัติโดยให้ใช้เทคโนโลยีที่ได้ศึกษา
3. ระหว่างการศึกษาหากมีปัญหา หรือมีข้อสงสัย พี่เลี้ยงจะเข้ามาช่วยอธิบายเพิ่มเติม

สัปดาห์ที่ 2-4 ได้เริ่มศึกษาดารางข้อมูลความไวเชื้อต่อยาปฏิชีวนะ เพื่อศึกษารูปแบบการแสดงผลของตารางข้อมูลความไวเชื้อต่อยาปฏิชีวนะ ศึกษาข้อมูลส่วนต่างๆ และศึกษาปัญหา วางแผนและหาวิธีแก้ไขและปรับปรุง

สัปดาห์ที่ 5 จัดทำตารางสรุปค่าผิดปกติของแต่ละไฟล์ข้อมูล เช่น
IPD_Data(AN,VN,HN,SEX,AGE),IPD_Culture,OPD_Culture(VN,AN,Specimen,Result)

สัปดาห์ที่ 6 หาวิธีทำความสะอาดข้อมูลที่ผิด หรือมีหลายหลายแบบแต่มีความหมายเดียวกัน ให้อยู่ในรูปแบบเดียวปรับปรุงข้อมูลในคอลัมน์(Column) ต่างๆในแต่ละชุดข้อมูล

สัปดาห์ที่ 7 สร้างตารางแสดงข้อมูลความไวเชื้อต่อยาปฏิชีวนะและหาวิธีตรวจสอบ

สัปดาห์ที่ 8-19 พัฒนาและทดสอบในส่วนอื่น ๆ ให้มีความสมบูรณ์ ทดสอบระบบร่วมกับพี่เลี้ยงเพื่อให้ระบบมีความถูกต้องและใช้งานได้

3. เทคโนโลยีที่ใช้ในสถานประกอบการ

3.1 ซอฟต์แวร์ที่ใช้ในการพัฒนา

1. แพลตฟอร์มพัฒนา Google Colab (Jupyter Notebook บนคลาวด์) ใช้สำหรับรันโค้ด Python ผ่านเว็บเบราว์เซอร์โดยไม่ต้องติดตั้งโปรแกรมเพิ่มเติม สามารถและเชื่อมกับ Google Drive เพื่ออ่าน-บันทึกไฟล์ข้อมูลได้

2. ภาษาโปรแกรม Python ภาษาหลักในการพัฒนาโค้ด มีไลบรารีด้านข้อมูลครบถ้วน เหมาะสำหรับงาน Data Cleaning/ETL และการสร้างตารางสรุปผล โดยใช้ทั้งโมดูลมาตรฐานและไลบรารีภายนอก
3. ไลบรารีจัดการข้อมูล pandas ใช้สำหรับอ่านไฟล์ตาราง (CSV/Excel), รวมตาราง (merge/join), จัดกลุ่ม (groupby), แปลงรูป (pivot), จัดการวันที่/สตริง
4. การประมวลผลข้อความ/ทำความสะอาดชื่อ re (Regular Expression) ใช้สำหรับมาตรฐานชื่อ/ชื่อยา/ผลทดสอบ เช่น การตัดคำเกิน แทนที่รูปแบบที่หลากหลาย และตรวจสอบรูปแบบสตริงให้เป็นมาตรฐานเดียวกัน
6. การส่งออกไฟล์ Excel และจัดรูปแบบ ใช้สร้างไฟล์สรุปผลหลายชีต จัดรูปแบบเซลล์ (ตัวหนา จัดกึ่งกลาง การเติมสีตามเกณฑ์ %S ฯลฯ) และเขียนข้อมูลจาก pandas ลง Excel อย่างมีประสิทธิภาพ