

การออกแบบและพัฒนานวัตกรรมในการวิเคราะห์ข้อมูลเพื่อนำไปใช้ในการ  
ตรวจสอบคุณภาพการให้บริการโทรศัพท์เคลื่อนที่

DESIGNING AND DEVELOPING INNOVATIONS IN DATA ANALYSIS IN  
ORDER TO USE THE DATA TO CHECK THE QUALITY OF MOBILE PHONE  
SERVICE

โดย

นางสาวพรธิติ

ว่องไววิทย์

นายสิทธิศาสตร์

ไชยหาญ

โครงการนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2566



การออกแบบและพัฒนานวัตกรรมในการวิเคราะห์ข้อมูลเพื่อนำไปใช้ในการ  
ตรวจสอบคุณภาพการให้บริการโทรศัพท์เคลื่อนที่  
DESIGNING AND DEVELOPING INNOVATIONS IN DATA ANALYSIS IN  
ORDER TO USE THE DATA TO CHECK THE QUALITY OF MOBILE PHONE  
SERVICE

โดย

นางสาวพรธิติ	ว่องไววิทย์	63010638
นายสิทธิศาสตร์	ไชยหาญ	63010967

อาจารย์ที่ปรึกษา  
รศ.ดร. พิสิฐ บุญศรีเมือง

โครงการนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต  
ภาควิชาวิศวกรรมโทรคมนาคม  
คณะวิศวกรรมศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ปีการศึกษา 2566

โครงการปีการศึกษา 2566

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การออกแบบและพัฒนานวัตกรรมในการวิเคราะห์ข้อมูลเพื่อนำไปใช้ในการตรวจสอบ  
คุณภาพการให้บริการโทรศัพท์เคลื่อนที่ชื่อโครงการภาษาไทย

DESIGNING AND DEVELOPING INNOVATIONS IN DATA ANALYSIS IN ORDER  
TO USE THE DATA TO CHECK THE QUALITY OF MOBILE PHONE SERVICE

ผู้จัดทำ

- |                   |             |          |
|-------------------|-------------|----------|
| 1. นางสาวพรธิติ   | ว่องไววิทย์ | 63010638 |
| 2. นายสิทธิศาสตร์ | ไชยหาญ      | 63010967 |

..... อาจารย์ที่ปรึกษา  
( รศ.ดร. พิสิฐ บุญศรีเมือง )

## กิตติกรรมประกาศ

การดำเนินโครงการเรื่อง “การออกแบบและพัฒนานวัตกรรมการวิเคราะห์ข้อมูลเพื่อตรวจสอบคุณภาพการให้บริการโทรศัพท์เคลื่อนที่” จะไม่สามารถสำเร็จลุล่วงไปได้ด้วยดี หากไม่ได้รับความช่วยเหลือ และความอนุเคราะห์อย่างดียิ่งจากรศ.ดร. พิสิฐ บุญศรีเมือง อาจารย์ที่ปรึกษาโครงการ นายศักร์ภวิรัช แมนเมือง และนายพงษ์ศักดิ์ ดวงศรี เจ้าหน้าที่ของคณะกรรมการกิจการกระจายเสียง กิจการโทรทัศน์ และกิจการโทรคมนาคมแห่งชาติที่กรุณาให้คำแนะนำ คำปรึกษา และแนวทางการแก้ไขปัญหาที่เป็นประโยชน์ต่อการศึกษา ค้นคว้าวิจัยให้โครงการนี้สำเร็จสมบูรณ์ยิ่งขึ้น รวมถึงสนับสนุนสถานที่ เครื่องมือ และอุปกรณ์ต่างๆ ที่ใช้ระหว่างการจัดทำโครงการ

ขอขอบคุณคณาจารย์และเจ้าหน้าที่ประจำภาควิชาวิศวกรรมโทรคมนาคม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังทุกท่าน ที่ได้อบรมสั่งสอน ประสิทธิ์ประสาทวิชา ความรู้ และประสบการณ์ให้แก่ผู้จัดทำ

ขอกราบขอบพระคุณ บิดา มารดา และครอบครัว ที่ให้ความรัก ความห่วงใย และเป็นกำลังใจที่สำคัญเสมอมาและที่สำคัญคือสนับสนุนให้โอกาสทางด้านการศึกษามีค่ายิ่งแก่ผู้จัดทำ

นางสาวพรธิดี ว่องไววิทย์  
นายสิทธิศาสตร์ ไชยหาญ  
ผู้จัดทำ

การออกแบบและพัฒนานวัตกรรมการวิเคราะห์ข้อมูลเพื่อตรวจสอบคุณภาพ  
การให้บริการโทรศัพท์เคลื่อนที่  
DESIGNING AND DEVELOPING INNOVATIONS IN DATA ANALYSIS TO  
VERIFY THE QUALITY OF MOBILE PHONE SERVICE

โดย นางสาวพรธิติ ว่องไววิทย์ 63010638  
นายสิทธิศาสตร์ ไชยหาญ 63010967

อาจารย์ที่ปรึกษา รศ.ดร. พิสิฐ บุญศรีเมือง

### บทคัดย่อ

เนื่องจากสำนักงานดูแลกิจการโทรคมนาคม (ดท.) ของสำนักงานคณะกรรมการกิจการกระจายเสียง กิจการโทรทัศน์ และกิจการโทรคมนาคมแห่งชาติ (สำนักงาน กสทช.) มีหน้าที่กำกับดูแลและการควบคุมคุณภาพการให้บริการโทรคมนาคมตามมาตรฐานกฎหมาย จึงมีการตรวจสอบคุณภาพการให้บริการโทรคมนาคมผ่านโครงข่ายโทรคมนาคมเคลื่อนที่ส่งผลให้ต้องเก็บข้อมูลของคุณภาพการให้บริการโทรคมนาคมจำนวนมาก โดยโปรแกรม MICROSOFT EXCEL ซึ่งเป็นโปรแกรมหลักที่สำนักงานดูแลกิจการโทรคมนาคม (ดท.) ของสำนักงาน กสทช. ใช้ในการประมวลผลวิเคราะห์ข้อมูลในปัจจุบันนั้นประมวลผลได้ช้า ทางผู้พัฒนามีความเห็นว่าในปัจจุบันได้มีนวัตกรรมที่ดีกว่าในการใช้งานเพื่อวิเคราะห์ข้อมูล ผู้พัฒนาเลือกใช้ PYTHON PROGRAMING ในการประมวลผล วิเคราะห์ และจัดเตรียมข้อมูล จากนั้นนำไฟล์ที่ได้ไปทำการแสดงผลในรูปแบบของหน้าจอแสดงผล (DASHBOARD) ซึ่งประกอบไปด้วยส่วนของ กราฟ ตารางสรุปผล และแผนที่ในการแสดงผล ค่าพารามิเตอร์ต่างๆ โดยใช้โปรแกรม POWER BI ในการออกแบบส่วนของการแสดงผล ส่งผลให้หน้าจอแสดงผล (DASHBOARD) ที่ได้นั้นมีความสวยงาม ง่ายต่อการใช้งาน และที่สำคัญคือความเร็วในการแสดงผลที่ดีกว่า MICROSOFT EXCEL โดยโครงการได้ออกแบบเฉพาะเจาะจงเพื่อใช้งานในส่วน of สำนักงานดูแลกิจการโทรคมนาคม (ดท.) ของสำนักงาน กสทช. เท่านั้น

## ABSTRACT

The Telecommunication Enforcement Bureau, under the National Broadcasting and Telecommunications Commission (NBTC), is responsible for overseeing and regulating the quality of telecommunications services under laws and regulations. As part of their duties, they examine quality assessments of mobile telecommunications services through the mobile network infrastructure, collecting a massive amount of quality data. This data was traditionally processed using Microsoft Excel, the primary software used by the NBTC's Department of Telecommunication Affairs for data analysis. However, the current Excel-based processing is slow and inefficient.

The developers have chosen to use Python programming to recognize the need for more efficient data analysis. This approach allows for faster data processing, analysis, and data preparation. Subsequently, the processed data is used to create a dashboard comprising graphs, summary tables, and maps, utilizing the Power BI software for data visualization. This choice results in a visually appealing and user-friendly dashboard and significantly improves the speed of data presentation compared to Microsoft Excel.

This project has been specifically tailored to serve the needs of the NBTC's Telecommunication Enforcement Bureau, aiming to enhance data analysis and presentation for their telecommunications quality assessment activities.

## สารบัญ

	หน้า
กิตติกรรมประกาศ	I
บทคัดย่อ	II
สารบัญ	IV
สารบัญรูป	VIII
สารบัญตาราง	X
<b>บทที่ 1</b>	<b>บทนำ</b>
1.1	ความเป็นมาและความสำคัญของปัญหา 1
1.2	วัตถุประสงค์ 1
1.3	ขอบเขตของโครงการ 1
<b>บทที่ 2</b>	<b>ทฤษฎีและหลักการที่เกี่ยวข้อง</b>
2.1	มาตรฐานของคุณภาพการให้บริการโทรคมนาคม 2565 3
2.1.1	บริการโทรคมนาคมผ่านโครงข่ายโทรคมนาคมเคลื่อนที่ 3
2.1.1.1	พารามิเตอร์ค่าชี้วัดคุณภาพบริการประเภทเสียง 3
2.1.1.2	พารามิเตอร์ค่าชี้วัดคุณภาพบริการประเภทข้อมูล 4
2.2	การตรวจสอบคุณลักษณะทางเทคนิคของการแพร่กระจายคลื่นความถี่วิทยุสำหรับการให้บริการโทรศัพท์เคลื่อนที่ 9
2.2.1	เครื่องมือตรวจสอบคุณลักษณะทางเทคนิคของการแพร่กระจายคลื่นความถี่วิทยุสำหรับการให้บริการโทรศัพท์เคลื่อนที่ 9
2.3	รายละเอียดตำแหน่งของภาค และเขต ของสำนักงานกสทช. 11
2.3.1	สำนักงาน กสทช. ภาค 1 11
2.3.2	สำนักงาน กสทช. ภาค 2 12
2.3.3	สำนักงาน กสทช. ภาค 3 12
2.3.4	สำนักงาน กสทช. ภาค 4 13



	สารบัญ (ต่อ)	หน้า
	2.4 ศึกษาการทดสอบคุณภาพการให้บริการโทรศัพท์เคลื่อนที่ (QUALITY OF SERVICE AND COMPLAIN)	13
	2.4.1 แผนการทดสอบ QOS ตาม เขตและภูมิภาคต่างๆ ของ กสทช.	13
	2.4.2 บริการโทรคมนาคมผ่านโครงข่ายโทรคมนาคมเคลื่อนที่	17
	2.4.2.1 บริการประเภทเสียง	17
	2.4.2.2 บริการประเภทข้อมูล	18
	2.4.3 ค่าชี้วัดคุณภาพบริการประเภทข้อมูล และค่าเป้าหมาย ของ PING	18
	2.4.4 ค่าชี้วัดคุณภาพบริการประเภทข้อมูล และค่าเป้าหมาย ของ FTP	19
	2.4.5 ค่าชี้วัดคุณภาพบริการประเภทข้อมูล และค่าเป้าหมาย ของ HTTP	20
	2.4.6 การวิเคราะห์ข้อร้องเรียนปัญหาคุณภาพการให้บริการ โทรศัพท์เคลื่อนที่และการกำหนดเงื่อนไขการทดสอบ	20
<b>บทที่ 3</b>	<b>การออกแบบและการจัดทำโครงงาน</b>	
	3.1 การออกแบบ	25
	3.1.1 การออกแบบระบบวิเคราะห์ข้อมูลเพื่อนำไปใช้สำหรับ รายงานค่าชี้วัดคุณภาพการบริการโทรคมนาคมผ่านโครงข่ายโทรคมนาคมเคลื่อนที่	25
	3.1.2 การออกแบบระบบการแสดงผลข้อมูลเพื่อนำไปใช้ สำหรับรายงานค่าชี้วัดคุณภาพการบริการโทรคมนาคมผ่านโครงข่ายโทรคมนาคมเคลื่อนที่	38
	3.2 เครื่องมือที่ใช้ในการทดลอง	39
	3.2.1 เครื่องมือทดสอบต้นทาง 1 (ORIGINATE I)	39

## สารบัญ (ต่อ)

	หน้า
3.2.2 เครื่องมือทดสอบต้นทาง 2 (ORIGINATE II)	40
3.2.3 เครื่องมือทดสอบต้นทาง 3 (ORIGINATE III)	41
3.2.4 เครื่องมือทดสอบประเภทเสียงปลายทาง 1 (VOICE TERMINATE I)	41
3.2.5 เครื่องมือทดสอบประเภทเสียงปลายทาง 2 (VOICE TERMINATE II)	42
3.2.6 เครื่องมือทดสอบประเภทเสียงปลายทาง 3 (VOICE TERMINATE III)	43
3.2.7 เครื่องมือทดสอบประเภทข้อมูลปลายทาง 1 (DATA TERMINATE I)	43
3.3 การจัดเก็บผลการทดลอง	43
3.3.1 การทดสอบการวิเคราะห์ข้อมูลคุณภาพการให้บริการ ประเภทข้อมูล	43
3.3.2 การทดสอบการวิเคราะห์ข้อมูลคุณภาพการให้บริการ ประเภทเสียง	43
3.3.3 การทดสอบการแสดงผลในส่วนของการร่างสรุปผล คุณภาพการให้บริการประเภทข้อมูล	44
3.3.4 การทดสอบการแสดงผลในส่วนของการร่างสรุปผล คุณภาพการให้บริการประเภทเสียง	44
3.3.5 การทดสอบการแสดงผลในส่วนของการแสดง ประสิทธิภาพของคุณภาพสัญญาณ	44
 <b>บทที่ 4</b>	
<b>ผลการทดลอง</b>	
4.1 การทดสอบการวิเคราะห์ข้อมูลคุณภาพการให้บริการประเภทข้อมูล	45
4.1.1 การตรวจสอบความถูกต้องสำหรับการเปลี่ยน NETWORK TYPE ให้อยู่ในรูปแบบของ TECHNOLOGY GENERATION	45

## สารบัญ (ต่อ)

	หน้า
4.1.2 การตรวจสอบความถูกต้องสำหรับการตรวจสอบ โครงข่ายของผลการทดสอบว่าเป็นประเภท การใช้บริการโครงข่ายของ ตนเอง (ON-NET) หรือ การใช้โครงข่ายของผู้ร่วมให้บริการ (ROAMING)	46
4.1.3 การตรวจสอบความถูกต้องสำหรับ THROUGHPUT ของผลการทดสอบ	47
4.2 การทดสอบการวิเคราะห์ข้อมูลคุณภาพการให้บริการประเภทเสียง	47
4.3 การทดสอบการแสดงผลในส่วนของการสร้างสรุปผลคุณภาพการให้บริการ ประเภทข้อมูล	48
4. 4 การทดสอบการแสดงผลในส่วนของการสร้างสรุปผลคุณภาพการ ให้บริการประเภทเสียง	49
4.5 การทดสอบการแสดงผลในส่วนแผนที่แสดงประสิทธิภาพของคุณภาพ สัญญาณ	50
<b>บทที่ 5</b>	<b>สรุปผลและข้อเสนอแนะ</b>
5.1 สรุปผล	52
5.2 ข้อเสนอแนะ	52
<b>บรรณานุกรม</b>	53
<b>ภาคผนวก ก</b>	<b>ชุดรหัสคำสั่งการวิเคราะห์ข้อมูลดิบที่นำเข้าจากโปรแกรม SYBERIZ ด้วย ภาษา PYTHON ในโปรแกรม VISUAL STUDIO CODE</b>
<b>ภาคผนวก ข</b>	<b>ชุดรหัสคำสั่งการคำนวณและการแสดงผลข้อมูลที่ได้จาก VISUAL STUDIO CODE ในโปรแกรม POWER BI</b>

## สารบัญรูป

รูปที่	หน้า
2.1 หลักการทำงานของเครื่องมือตรวจสอบคุณลักษณะทางเทคนิคของการแพร่กระจายคลื่นความถี่วิทยุ	9
2.2 ระบบการแสดงผลการตรวจสอบส่วนกลาง (DASHBOARD)	11
3.1 แผนผังการดำเนินงานการวิเคราะห์ข้อมูลประเภทเสียง 1	29
3.2 แผนผังการดำเนินงานการวิเคราะห์ข้อมูลประเภทเสียง 2	29
3.3 แผนผังการดำเนินงานการวิเคราะห์ข้อมูลประเภทเสียง 3	30
3.4 แผนผังการดำเนินงานการวิเคราะห์ข้อมูลประเภทเสียง 4	30
3.5 แผนผังการดำเนินงานการวิเคราะห์ข้อมูลประเภทเสียง 5	31
3.6 แผนผังการดำเนินงานการวิเคราะห์ข้อมูลประเภทเสียง 6	31
3.7 แผนผังการดำเนินงานการวิเคราะห์ข้อมูลประเภทเสียง 7	32
3.8 แผนการดำเนินการทดสอบประเภทข้อมูลประจำไตรมาสที่ 1	32
3.9 แผนการดำเนินการทดสอบประเภทข้อมูลประจำไตรมาสที่ 2	33
3.10 แผนการดำเนินการทดสอบประเภทข้อมูลประจำไตรมาสที่ 3	33
3.11 แผนการดำเนินการทดสอบประเภทข้อมูลประจำไตรมาสที่ 4	34
3.12 แผนผังการดำเนินงานวิเคราะห์ข้อมูลในส่วนของ SIGNAL STRENGTH และ SIGNAL QUALITY	36
3.13 ข้อมูลในส่วนของ SIGNAL STRENGTH และ SIGNAL QUALITY	36
3.14 เครื่องมือทดสอบต้นทาง 1 (ORIGINATE I)	40
3.15 เครื่องมือทดสอบต้นทาง 2 (ORIGINATE II)	40
3.16 เครื่องมือทดสอบต้นทาง 3 (ORIGINATE III)	41
3.17 เครื่องมือทดสอบประเภทเสียงปลายทาง 1 (VOICE TERMINATE I)	42
3.18 เครื่องมือทดสอบประเภทเสียงปลายทาง 2 (VOICE TERMINATE II)	42
3.19 เครื่องมือทดสอบประเภทเสียงปลายทาง 3 (VOICE TERMINATE III)	43
3.20 เครื่องมือทดสอบประเภทข้อมูลปลายทาง 1 (DATA TERMINATE I)	43
4.1 ผลการทดสอบ TECHNOLOGY GENERATION กรณีของ 3G	45

## สารบัญรูป (ต่อ)

รูปที่		หน้า
4.2	การตรวจสอบเงื่อนไขการใช้โครงข่ายของตนเอง (ON-NET)	46
4.3	การตรวจสอบเงื่อนไขการใช้โครงข่ายแบบร่วมให้บริการ (ROAMING)	46
4.4	ตัวอย่างผลการทดสอบที่ผ่านมาตรฐาน 4G แบบ FTP DOWNLOAD	47
4.5	ตารางสรุปผลการทดสอบประเภทข้อมูล	48
4.6	ตารางสรุปผลการทดสอบประเภทเสียง	49
4.7	FILLED MAP ที่บ่งบอกถึงประสิทธิภาพของคุณภาพสัญญาณในแต่ละพื้นที่	51

## สารบัญตาราง

ตารางที่	หน้า
2.1 ตารางคุณสมบัติของเครื่องมือและโปรแกรมควบคุมการทำงานการตรวจสอบคุณลักษณะทางเทคนิคของการแพร่กระจายคลื่นความถี่วิทยุฯ	10
2.2 ตารางตำแหน่งของภาค 1 และเขต 11 ถึง 16 ของสำนักงาน กสทช.	11
2.3 ตารางตำแหน่งของภาค 2 และเขต 21 ถึง 25 ของสำนักงาน กสทช.	12
2.4 ตารางตำแหน่งของภาค 3 และเขต 31 ถึง 35 ของสำนักงาน กสทช.	12
2.5 ตารางตำแหน่งของภาค 4 และเขต 41 ถึง 45 ของสำนักงาน กสทช.	13
2.6 ตาราง QUALITY OF SERVICES (DATA SERVICES)	13
2.7 ตาราง QUALITY OF SERVICES (VOICE SERVICES)	15
2.8 ตารางบริการโทรคมนาคมผ่านโครงข่ายโทรคมนาคมเคลื่อนที่ประเภทเสียง	17
2.9 ตารางบริการโทรคมนาคมผ่านโครงข่ายโทรคมนาคมเคลื่อนที่ประเภทข้อมูล	18
2.10 ตารางค่าชี้วัดคุณภาพบริการประเภทข้อมูล และค่าเป้าหมายของ PING	18
2.11 ตารางค่าชี้วัดคุณภาพบริการประเภทข้อมูล และค่าเป้าหมายของ FTP	19
2.12 ตารางค่าชี้วัดคุณภาพบริการประเภทข้อมูล และค่าเป้าหมายของ HTTP	20
2.13 ตารางการวิเคราะห์พื้นที่ประสบปัญหา และลักษณะการกำหนดเงื่อนไขการทดสอบ	21
2.14 ตารางรายละเอียด SCRIPT TEST (VOICE)	23
2.15 ตารางรายละเอียด SCRIPT TEST (DATA)	24
3.1 ตารางขนาดเว็บไซต์และเวลาที่กำหนดสำหรับการทดสอบดาวน์โหลดเว็บไซต์	25
3.2 ตารางขนาดไฟล์และเวลาที่กำหนดสำหรับทดสอบ DOWNLOAD และ UPLOAD ผ่าน FTP	26

## สารบัญตาราง

ตารางที่	หน้า
3.3 ตารางขนาดและหมายเลข IP ADDRESS สำหรับการทดสอบแบบ PING	26
3.4 ตารางรายละเอียดวิธีการทดสอบคุณภาพการให้บริการประเภทข้อมูล	27
3.5 ตารางรายละเอียดวิธีการทดสอบคุณภาพการให้บริการประเภทเสียง	28
3.6 ตารางเงื่อนไข NETWORK TYPE และ TECHNOLOGY GENERATION	34
3.7 ตารางเงื่อนไขการใช้งาน MOBILE NETWORK CODE ประเภท ON-NET	35
3.8 ตารางเงื่อนไขการใช้งาน MOBILE NETWORK CODE ประเภท ROAMING	35
3.9 ตารางตำแหน่งละติจูดลองจิจูดอ้างอิงของสำนักงาน กสทช. ประจำ เขต/ ภูมิภาค	37
3.10 ตารางรายละเอียดพารามิเตอร์ของการทดสอบประเภทเสียง	38
3.11 ตารางรายละเอียดพารามิเตอร์ของการทดสอบประเภทข้อมูล	39
4.1 ตารางเงื่อนไขตัวแปรสำหรับ SIGNAL STRENGTH และ SIGNAL QUALITY ของแต่ละเทคโนโลยี	50
4.2 ตารางช่วงระยะของค่าพารามิเตอร์ในการบ่งบอกประสิทธิภาพของ คุณภาพสัญญาณ	50

# บทที่ 1

## บทนำ

### 1.1 ความเป็นมาและความสำคัญของปัญหา

เนื่องจากสำนักงานดูแลกิจการโทรคมนาคม (ดท.) ของสำนักงานคณะกรรมการกิจการกระจายเสียง กิจการโทรทัศน์ และกิจการโทรคมนาคมแห่งชาติ (สำนักงาน กสทช.) มีหน้าที่กำกับดูแลและการควบคุมคุณภาพการให้บริการโทรคมนาคมตามมาตรากฎหมาย จึงมีการตรวจสอบคุณภาพการให้บริการโทรคมนาคมผ่านโครงข่ายโทรคมนาคมเคลื่อนที่ส่งผลให้ต้องเก็บข้อมูลของคุณภาพการให้บริการโทรคมนาคมจำนวนมากโดยโปรแกรม Microsoft Excel ซึ่งเป็นโปรแกรมหลักที่สำนักงานดูแลกิจการโทรคมนาคม (ดท.) ของสำนักงาน กสทช. ใช้ในการประมวลผลวิเคราะห์ข้อมูลในปัจจุบันนั้นประมวลผลได้ช้า ทางผู้พัฒนามีความเห็นว่าในปัจจุบันได้มีนวัตกรรมที่ดีกว่าในการใช้งานเพื่อวิเคราะห์ข้อมูล ผู้พัฒนาเลือกใช้ Python Programming ในการประมวลผลวิเคราะห์ และจัดเตรียมข้อมูล จากนั้นนำไฟล์ที่ได้ไปทำการแสดงผลในรูปแบบของหน้าจอแสดงผล (Dashboard) ซึ่งประกอบไปด้วยส่วนของ กราฟ ตารางสรุปผล และแผนที่ในการแสดงผลค่าพารามิเตอร์ต่างๆ โดยใช้โปรแกรม Power BI ในการออกแบบส่วนของแสดงผล ส่งผลให้หน้าจอแสดงผล (Dashboard) ที่ได้นั้นมีความสวยงาม ง่ายต่อการใช้งาน และที่สำคัญคือความเร็วในการแสดงผลที่ดีกว่า Microsoft Excel โดยโครงงานได้ออกแบบเฉพาะเจาะจงเพื่อใช้งานในส่วน of สำนักงานดูแลกิจการโทรคมนาคม (ดท.) ของสำนักงาน กสทช. เท่านั้น

### 1.2 วัตถุประสงค์

- 1) เพื่อศึกษาวิธีการตรวจสอบคุณภาพการให้บริการตามประกาศนโยบายของสำนักงาน กสทช. เรื่อง มาตรฐานของคุณภาพการให้บริการโทรคมนาคม ฉบับ พ.ศ. 2565
- 2) เพื่อศึกษาการเขียนโค้ด Python ในการสร้างระบบประมวลผลและวิเคราะห์ข้อมูลเชิงเทคนิค
- 3) เพื่อศึกษาการใช้งาน Power BI ในการออกแบบ Dashboard ที่แสดงผลให้เป็นไปตามนโยบายและประกาศของสำนักงาน กสทช.
- 4) เพื่อศึกษาและพัฒนาระบบจัดเก็บข้อมูลต่าง ๆ

### 1.3 ขอบเขตของปริญญานิพนธ์

ปริญญานิพนธ์นี้ได้ทำการออกแบบและพัฒนานวัตกรรมในการวิเคราะห์ข้อมูลเพื่อนำไปใช้ในการตรวจสอบคุณภาพการให้บริการโทรศัพท์เคลื่อนที่ (Quality of Service) โดยระบบประกอบไปด้วย 3 ส่วนดังนี้



1) ส่วนที่ทำหน้าที่เป็นอินพุตของระบบอันได้แก่ชุดข้อมูลของผลการทดสอบที่ถูกแยกตามบริการประกอบไปด้วย HTTP Download , FTP Upload , FTP Download , Ping และ Voice ซึ่งมีลักษณะเป็นไฟล์ที่ถูกเก็บข้อมูลไว้แบบ CSV (Comma Separated Value) โดยการที่จะได้ไฟล์ข้อมูลเหล่านี้ออกมาจะต้องทำการนำออกมาจากซอฟต์แวร์ Syberiz ที่ทางสำนักงาน กสทช. ใช้ในการจัดเก็บข้อมูลเท่านั้น

2) ทำการนำเอาไฟล์ข้อมูลทุกบริการไปผ่านการประมวลผลทางคอมพิวเตอร์ จัดเรียงและทำการรวมข้อมูลผลการทดสอบของทุกบริการเข้าด้วยกันโดยใช้ Python จัดการข้อมูลทั้งหมด

3) นำไฟล์ผลลัพธ์ที่ได้มาสร้าง Dashboard และทำการแสดงผลบน Power BI โดยมีลักษณะการแสดงผลที่เป็นไปตามความต้องการของเจ้าหน้าที่และพนักงานของสำนักงาน กสทช.

## บทที่ 2

### ทฤษฎีและหลักการที่เกี่ยวข้อง

#### 2.1 มาตรฐานของคุณภาพการให้บริการโทรคมนาคม 2565 [1]

##### 2.1.1 บริการโทรคมนาคมผ่านโครงข่ายโทรคมนาคมเคลื่อนที่

###### 2.1.1.1 พารามิเตอร์ค่าชี้วัดคุณภาพบริการประเภทเสียง

- 1) อัตราส่วนการเรียกสำเร็จ (Successful call ratio) กรณีที่เป็นการโทรศัพท์ภายในโครงข่ายของผู้ประกอบการเดียวกัน
  - หมายถึง อัตราส่วนจำนวนการเรียกที่สำเร็จต่อจำนวนการเรียกทั้งหมด
  - ค่าเป้าหมายตามมาตรฐานของคุณภาพโทรคมนาคม 2565
    - ไม่น้อยกว่าร้อยละ 90
- 2) อัตราส่วนการเรียกสำเร็จ (Successful call ratio) กรณีที่เป็นการโทรข้ามโครงข่ายต่างผู้ประกอบการ
  - หมายถึง อัตราส่วนจำนวนการเรียกที่สำเร็จต่อจำนวนการเรียกทั้งหมด
  - ค่าเป้าหมายตามมาตรฐานของคุณภาพโทรคมนาคม 2565 ไม่น้อยกว่าร้อยละ 90
- 3) อัตราส่วนของกรณีที่สายหลุด (Drop Call Rate)
  - หมายถึง อัตราส่วนของจำนวนสายหลุดต่อจำนวนการเรียกใช้ทั้งหมดภายในระยะเวลาที่กำหนด
  - ค่าเป้าหมายตามมาตรฐานของคุณภาพโทรคมนาคม 2565
    - ไม่มากกว่าร้อยละ 2
- 4) คุณภาพของเสียง (Mean Opinion Score (MOS))
  - หมายถึง การวัดคุณภาพของเสียงจากโครงข่ายโทรศัพท์

- ค่าเป้าหมายตามมาตรฐานของคุณภาพโทรคมนาคม 2565
  - ไม่ต่ำกว่า 2.5 (ร้อยละ 90 ของข้อมูลที่วัดได้ต้องไม่ต่ำกว่าค่านี้)
- 5) อัตราส่วนของกรณีที่สายหลุด (Drop Call Rate) ในพื้นที่เฝ้าระวัง
  - หมายถึง อัตราส่วนของจำนวนสายหลุดต่อจำนวนการเรียกใช้ทั้งหมด ภายใน 24 ชั่วโมง ในพื้นที่เฝ้าระวัง
  - ค่าเป้าหมายตามมาตรฐานของคุณภาพโทรคมนาคม 2565
    - ไม่มากกว่าร้อยละ 2

#### 2.1.1.2 พารามิเตอร์ค่าชี้วัดคุณภาพบริการประเภทข้อมูล

- 1) อัตราส่วนจำนวนครั้งที่ค่า Round Trip Time (RTT) ต่ำกว่าค่าที่กำหนด
  - หมายถึง  $100(\%) \times \text{จำนวนครั้งที่ทำการวัดค่า RTT ได้ต่ำกว่าค่าที่กำหนด} / \text{จำนวนครั้งที่ทำการทดสอบวัดค่า RTT ทั้งหมด}$  โดยกำหนดให้ RTT เป็นระยะเวลาที่ใช้ส่งแพ็คเก็ตจากฝั่งส่งไปยังฝั่งรับจนกระทั่งกลับมายังฝั่งส่งอีกครั้งหนึ่ง และให้คำนวณจากผลต่างระหว่างเวลาที่แพ็คเก็ตกลับมายังฝั่งส่ง และ เวลาที่แพ็คเก็ตถูกส่งออกไปจากฝั่งส่ง
  - ค่าเป้าหมายตามมาตรฐานของคุณภาพโทรคมนาคม 2565
    - สำหรับเทคโนโลยี 2G ไม่เกิน 1,000 มิลลิวินาที
    - สำหรับเทคโนโลยี 3G ไม่เกิน 500 มิลลิวินาที
    - สำหรับเทคโนโลยี 4G, 5G (Non-standalone Mode) และ 5G (Standalone Mode) ที่ไม่ใช้คลื่นความถี่ย่าน 2,600 เมกะเฮิรตซ์ ไม่เกิน 150 มิลลิวินาที
    - สำหรับเทคโนโลยี 5G (Standalone Mode) ที่ใช้คลื่นความถี่ย่าน 2,600 เมกะเฮิรตซ์ ไม่เกิน 110 มิลลิวินาที
- 2) อัตราส่วนจำนวนครั้งที่ใช้ FTP ได้สำเร็จ (FTP success ratio)

- หมายถึง 100 (%) x จำนวนครั้งที่รับส่งข้อมูลแบบ FTP ได้สำเร็จ / จำนวนครั้งที่ทดสอบการรับส่ง FTP ทั้งหมด
  - ค่าเป้าหมายตามมาตรฐานของคุณภาพโทรคมนาคม 2565
    - กรณี Download ไม่ต่ำกว่าร้อยละ 80 สำหรับ 2G ขึ้นไป
    - กรณี Upload ไม่ต่ำกว่าร้อยละ 70 สำหรับ 2G ขึ้นไป
- 3) อัตราส่วนของการรับส่งข้อมูลแบบ FTP ตาม ความเร็วเฉลี่ยในการส่งข้อมูลที่กำหนด (FTP ratio subjected to specified data rate)
- หมายถึง 100 (%) x จำนวนครั้งที่รับส่งข้อมูลแบบ FTP ได้ไม่ต่ำกว่า อัตราบิตที่กำหนด / จำนวนครั้งที่ทดสอบการรับส่ง FTP สำเร็จทั้งหมด โดยกำหนดให้อัตราบิตเป็นจำนวนบิตที่รับส่งข้อมูลแบบ FTP ทหารด้วย ผลต่างระหว่างเวลาสิ้นสุดและเริ่มต้นการรับส่งข้อมูลแบบ FTP
  - ค่าเป้าหมายตามมาตรฐานของคุณภาพโทรคมนาคม 2565
 

กรณี Download

    - สำหรับเทคโนโลยี 2G ไม่ต่ำกว่า 48 กิโลบิตต่อวินาที
    - สำหรับเทคโนโลยี 3G ไม่ต่ำกว่า 750 กิโลบิตต่อวินาที
    - สำหรับเทคโนโลยี 4G, 5G (Non-standalone Mode) และ 5G (Standalone Mode) ที่ไม่ใช่คลื่นความถี่ย่าน 2,600 เมกะเฮิรตซ์ ไม่ต่ำกว่า 2.5 เมกะบิตต่อวินาที
    - สำหรับเทคโนโลยี 5G ที่ใช้คลื่นความถี่ย่าน 2,600 เมกะเฮิรตซ์ ไม่ต่ำกว่า 5 เมกะบิตต่อวินาที

สำหรับร้อยละ 75 ของการรับส่ง FTP ที่สำเร็จภายในเวลาที่กำหนด (Timeout)

กรณี Upload

    - สำหรับเทคโนโลยี 2G ไม่ต่ำกว่า 20 กิโลบิตต่อวินาที
    - สำหรับเทคโนโลยี 3G ไม่ต่ำกว่า 300 กิโลบิตต่อวินาที

- สำหรับเทคโนโลยี 4G, 5G (Non-standalone Mode) และ 5G (Standalone Mode) ที่ไม่ใช่คลื่นความถี่ย่าน 2,600 เมกะเฮิรตซ์ ไม่ต่ำกว่า 500 กิโลบิตต่อวินาที
- สำหรับเทคโนโลยี 5G ที่ใช้คลื่นความถี่ย่าน 2,600 เมกะเฮิรตซ์ ไม่ต่ำกว่า 1.25 เมกะบิตต่อวินาที

สำหรับร้อยละ 75 ของการรับส่ง FIP ที่สำเร็จภายในเวลาที่กำหนด (Timeout)

- 4) อัตราส่วนจำนวนครั้งที่ HTTP โหลดได้สำเร็จ (HTTP success ratio)
  - หมายถึง  $100 (\%) \times \text{จำนวนครั้งที่ HTTP โหลดได้ภายในเวลาที่กำหนด} / \text{จำนวนครั้งที่ทดสอบ HTTP ทั้งหมด}$
  - ค่าเป้าหมายตามมาตรฐานของคุณภาพโทรคมนาคม 2565
    - สำหรับ 2G ไม่ต่ำกว่าร้อยละ 80 โดยเวลาที่กำหนดไม่เกิน 10 นาที
    - สำหรับ 3G ไม่ต่ำกว่าร้อยละ 90 โดยเวลาที่กำหนดไม่เกิน 3 นาที
    - สำหรับ 4G, 5G (Non-standalone Mode) และ 5G (Standalone Mode) ที่ไม่ใช่คลื่นความถี่ย่าน 2,600 เมกะเฮิรตซ์ ไม่ต่ำกว่าร้อยละ 90 โดยเวลาที่กำหนดไม่เกิน 1 นาที
    - สำหรับ 5G (Standalone Mode) ที่ใช้คลื่นความถี่ย่าน 2,600 เมกะเฮิรตซ์ ไม่ต่ำกว่าร้อยละ 90 โดยเวลาที่กำหนดไม่เกิน 25 วินาที
- 5) อัตราส่วนจำนวนครั้งที่สามารถเข้าถึงบริการสตรีมมิง (Streaming service accessibility)
  - หมายถึง  $100 (\%) \times \text{จำนวนครั้งที่เข้าถึงบริการสตรีมมิงได้สำเร็จ} / \text{จำนวนครั้งในการทดสอบทั้งหมด}$
  - หมายเหตุ: ในกรณีที่ทำการวัดตามมาตรฐาน ETSI 5 102 250-2 v2.7.11 (2019-11) ข้อ 6.5.4 [3] สามารถคำนวณได้จาก  $(1 - \text{Streaming service non-accessibility}) \times 100 (\%)$
  - ค่าเป้าหมายตามมาตรฐานของคุณภาพโทรคมนาคม 2565

- สำหรับเทคโนโลยี 3G ไม่น้อยกว่าร้อยละ 80
  - สำหรับเทคโนโลยี 4G, 5G (Non-standalone Mode) และ 5G (Standalone Mode) ที่ไม่ใช่คลื่นความถี่ย่าน 2,600 เมกะเฮิรตซ์ ไม่ต่ำกว่าร้อยละ 85
  - สำหรับเทคโนโลยี 5G (Standalone Mode) ที่ใช้คลื่นความถี่ย่าน 2,600 เมกะเฮิรตซ์ ไม่ต่ำกว่าร้อยละ 90
- 6) อัตราส่วนจำนวนครั้งการแสดงผลวีดิทัศน์แบบสตรีมมิ่งได้อย่างสมบูรณ์ (Streaming reproduction success ratio)
- หมายถึง  $100 (\%) \times \text{จำนวนครั้งที่การแสดงผลวีดิทัศน์แบบสตรีมมิ่งได้อย่างสมบูรณ์} / \text{จำนวนครั้งในการทดสอบทั้งหมด}$   
 หมายเหตุ: ในกรณีที่ทำการวัดตามมาตรฐาน [ETSITS 102 250-2 v 2.7.11 (2019-11) ข้อ 6.5.6 สามารถคำนวณได้จาก  $(1 - \text{Streaming reproduction cut-off ratio}) \times 100 (\%)$
  - ค่าเป้าหมายตามมาตรฐานของคุณภาพโทรคมนาคม 2565
    - สำหรับเทคโนโลยี 3G ไม่น้อยกว่าร้อยละ 80
    - สำหรับเทคโนโลยี 4G, 5G (Non-standalone Mode) และ 5G (Standalone Mode) ที่ไม่ใช่คลื่นความถี่ย่าน 2,600 เมกะเฮิรตซ์ ไม่ต่ำกว่าร้อยละ 85
    - สำหรับเทคโนโลยี 5G (Standalone Mode) ที่ใช้คลื่นความถี่ย่าน 2,600 เมกะเฮิรตซ์ ไม่ต่ำกว่าร้อยละ 90
- 7) ระยะเวลาที่ใช้ในการส่ง SMS จากฝั่งส่งไปยังฝั่งรับ (SMS end-to-end delivery time)
- หมายถึง เวลาที่ใช้ในการส่ง Short Message จากอุปกรณ์ต้นทางส่งไปยังอุปกรณ์ปลายทางสำเร็จ หรือก็คือ จุดของเวลาที่อุปกรณ์ปลายทางได้รับ Short Message จากอุปกรณ์ต้นทาง - จุดของเวลาที่อุปกรณ์ต้นทางส่ง Short Message

- ค่าเป้าหมายตามมาตรฐานของคุณภาพโทรคมนาคม 2565
  - ไม่เกิน 90 วินาที สำหรับร้อยละ 80 ของ SMS ที่ฝั่งรับได้รับสำเร็จ ภายใน Timeout ที่ 175 วินาที

8) อัตราส่วนจำนวน SMS ที่ไปถึงฝั่งรับได้สำเร็จ (SMS completion success ratio)

- หมายถึง  $100 (\%) \times \text{จำนวน Short Message ที่อุปกรณ์ปลายทางได้รับ} / \text{จำนวน Short Message ที่ส่งในการทดสอบทั้งหมด}$

หมายเหตุ: ในกรณีที่ทำกรวัดตามมาตรฐาน [ETSI TS 102 250-2 v2.7.1] สามารถคำนวณได้จาก  $(1 - \text{SMS completion failure ratio}) \times 100 (\%)$

- ค่าเป้าหมายตามมาตรฐานของคุณภาพโทรคมนาคม 2565
  - ไม่ต่ำกว่าร้อยละ 90 ของการทดสอบทั้งหมด จะต้องได้รับสำเร็จที่ฝั่งรับ ภายใน Timeout ที่ 175 วินาที

9) ระยะเวลาที่ใช้ในการส่ง MMS จากฝั่งส่งไปยังฝั่งรับ (MMS end-to-end delivery time)

- หมายถึง เวลาที่ใช้ในการส่ง Multimedia Message จากอุปกรณ์ต้นทางส่งไปยังอุปกรณ์ปลายทางสำเร็จ หรือก็คือ จุดของเวลาที่อุปกรณ์ปลายทางได้รับ Multimedia Message จากอุปกรณ์ต้นทาง - จุดของเวลาที่อุปกรณ์ต้นทางส่ง Multimedia Message

หมายเหตุ: อุปกรณ์ปลายทางตั้งโหมดของการดาวน์โหลด Multimedia Message แบบอัตโนมัติ

- ค่าเป้าหมายตามมาตรฐานของคุณภาพโทรคมนาคม 2565
  - ภายใน 5 นาที สำหรับร้อยละ 70 ของ MMS ที่ฝั่งรับได้รับสำเร็จ ภายใน Timeout ที่ 13 นาที

10) อัตราส่วนจำนวน MMS ที่ไปถึงฝั่งรับได้สำเร็จ (MMS completion success ratio)

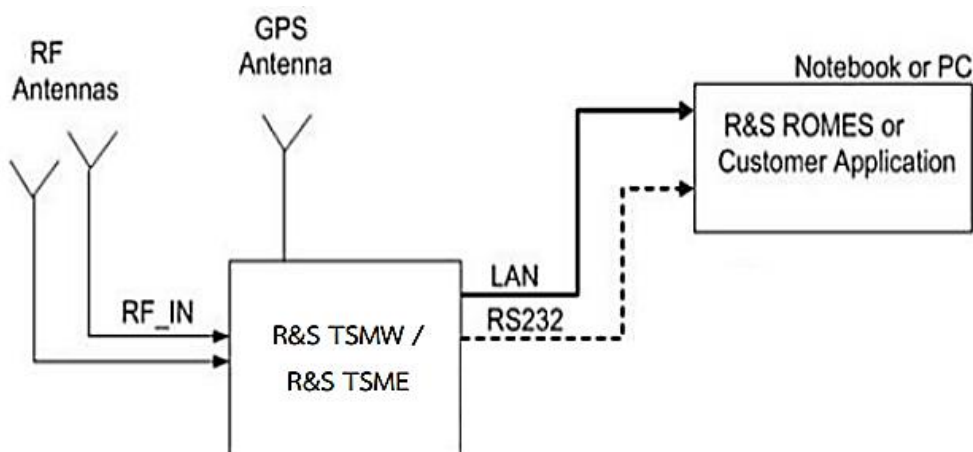
- หมายถึง จำนวน Multimedia Message ที่อุปกรณ์ปลายทางได้รับ ภายในเวลาที่กำหนด / จำนวน Multimedia Message ที่ส่งในการทดสอบทั้งหมด  $\times 100$  (%)

หมายเหตุ: ในกรณีที่ทำการวัดตามมาตรฐาน [ETSITS 102 250-2 v2.7.1] สามารถคำนวณได้จาก  $(1 - \text{MMS end-to-end failure ratio}) \times 100$  (%)

- ค่าเป้าหมายตามมาตรฐานของคุณภาพโทรคมนาคม 2565
  - ไม่ต่ำกว่าร้อยละ 80 ของการทดสอบทั้งหมดจะต้องได้รับสำเร็จ ที่ฝั่งรับภายใน Timeout ที่ 13 นาที

## 2.2 การตรวจสอบคุณลักษณะทางเทคนิคของการแพร่กระจายคลื่นความถี่วิทยุ สำหรับการให้บริการโทรศัพท์เคลื่อนที่

2.2.1 เครื่องมือตรวจสอบคุณลักษณะทางเทคนิคของการแพร่กระจายคลื่นความถี่วิทยุ สำหรับการให้บริการโทรศัพท์เคลื่อนที่แสดงดังรูปที่ 2.1



รูปที่ 2.1 หลักการทำงานของเครื่องมือตรวจสอบคุณลักษณะทางเทคนิคของการแพร่กระจายคลื่นความถี่วิทยุ

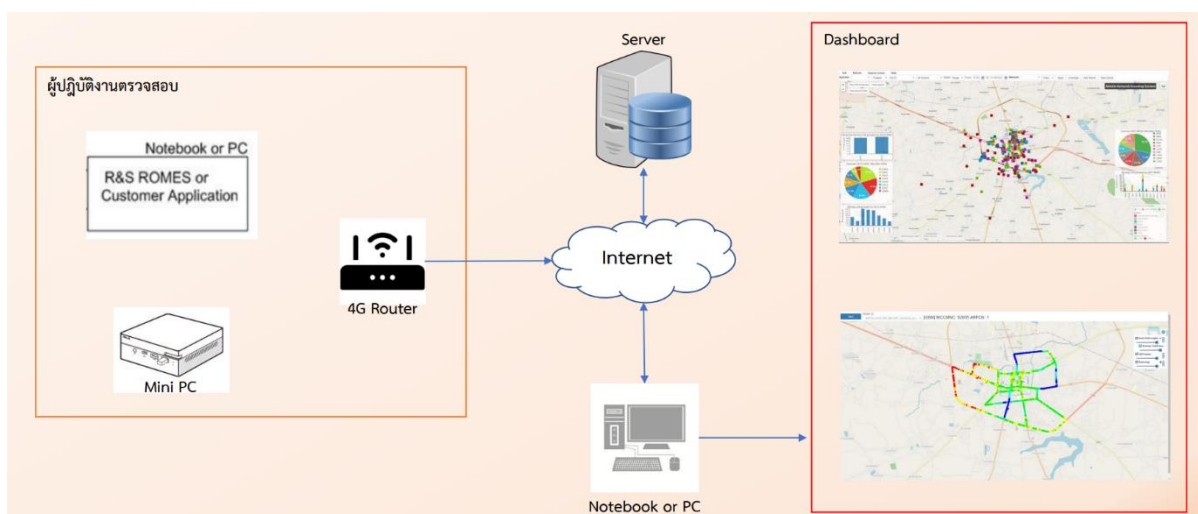


ตารางที่ 2.1 ตารางคุณสมบัติของเครื่องมือและโปรแกรมควบคุมการทำงานการตรวจสอบ  
คุณลักษณะทางเทคนิคของการแพร่กระจายคลื่นความถี่วิทยุ

ลำดับ	รายละเอียด
1	User-definable input frequency range from 30 MHz to 6 GHz
2	Two independent RF and signal processing paths, each with a bandwidth of up to 20 MHz
3	Parallel measurements in GSM, WCDMA and LTE (FDD and TDD)
4	Sensitivity for initial physical cell ID decoding: SYNC signal power -123 dB
5	ควบคุมและแสดงผลการตรวจสอบคลื่นความถี่วิทยุ ในรูปแบบกำหนดให้เครื่องมือทดสอบเลือกเทคโนโลยีและช่องความถี่ (Fix technology and Channel)
6	ควบคุมและแสดงผลการตรวจสอบคลื่นความถี่วิทยุ ในรูปแบบกำหนดให้เครื่องมือตรวจสอบการแพร่คลื่นความถี่วิทยุในบริเวณ นั้นๆ ในเทคโนโลยีและย่านความถี่ที่ต้องการทั้งหมด (Auto Channel Detection)
7	ตรวจสอบการใช้งานคลื่นความถี่วิทยุในรูปแบบของ RF Spectrum ได้
8	ถอดรหัสข้อมูลทางเทคนิคของสถานีฐาน เช่น MCC / MNC / TAC / Cell ID / EARFCN
9	ตรวจสอบระดับค่าความแรงและคุณภาพของสัญญาณโทรศัพท์เคลื่อนที่
10	แสดงพิกัดของสถานีฐานของโทรศัพท์เคลื่อนที่ โดยการประมาณตำแหน่งสถานีฐาน (Location Estimation) ได้

1. ควบคุมและแสดงผลการตรวจสอบคลื่นความถี่วิทยุ ในรูปแบบกำหนดให้เครื่องมือทดสอบเลือกเทคโนโลยีและช่องความถี่ (Fix technology and Channel)
  - โดยดำเนินการตรวจสอบการแพร่กระจายคลื่นความถี่วิทยุสำหรับการให้บริการโทรศัพท์เคลื่อนที่เฉพาะเทคโนโลยี 4G Band 1 (2100 MHz), Band 3 (1800 MHz) และ Band 41 (2600 MHz)
2. ควบคุมและแสดงผลการตรวจสอบคลื่นความถี่วิทยุ ในรูปแบบกำหนดให้เครื่องมือตรวจสอบการแพร่คลื่นความถี่วิทยุในบริเวณ นั้นๆ ในเทคโนโลยีและย่านความถี่ที่ต้องการทั้งหมด (Auto Channel Detection)

3. ตรวจสอบการใช้งานคลื่นความถี่วิทยุในรูปแบบของ RF Spectrum ได้
4. ถอดรหัสข้อมูลทางเทคนิคของสถานีฐาน เช่น MCC / MNC / TAC / Cell ID / EARFCN เป็นต้น
5. ตรวจสอบระดับค่าความแรงและคุณภาพของสัญญาณโทรศัพท์เคลื่อนที่
6. แสดงพิกัดของสถานีฐานของโทรศัพท์เคลื่อนที่ โดยการประมาณตำแหน่งสถานีฐาน (Location Estimation) ได้



รูปที่ 2.2 ระบบการแสดงผลการตรวจสอบส่วนกลาง (Dashboard)

## 2.3 รายละเอียดตำแหน่งของภาค และเขต ของสำนักงานกสทช.

### 2.3.1 สำนักงาน กสทช. ภาค 1 แสดงรายละเอียดได้ดังตารางที่ 2.2

ตารางที่ 2.2 ตารางตำแหน่งของภาค 1 และเขต 11 ถึง 16 ของสำนักงาน กสทช.

สำนักงาน กสทช.	จังหวัดที่ตั้งทำการ	ท้องที่ความรับผิดชอบ
สำนักงาน กสทช. ภาค 1	นนทบุรี	กรุงเทพมหานคร
สำนักงาน กสทช. เขต 11	สมุทรปราการ	สมุทรปราการ, นนทบุรี, ปทุมธานี
สำนักงาน กสทช. เขต 12	จันทบุรี	จันทบุรี, ตราด, ระยอง, ชลบุรี
สำนักงาน กสทช. เขต 13	สุพรรณบุรี	สุพรรณบุรี, กาญจนบุรี, ชัยนาท, สิงห์บุรี

ตารางที่ 2.2 ตารางตำแหน่งของภาค 1 และเขต 11 ถึง 16 ของสำนักงาน กสทช. (ต่อ)

สำนักงาน กสทช. เขต 14	ปราจีนบุรี	ปราจีนบุรี,นครนายก,ฉะเชิงเทรา,สระแก้ว
สำนักงาน กสทช. เขต 15	พระนครศรีอยุธยา	พระนครศรีอยุธยา,อ่างทอง,ลพบุรี,สระบุรี
สำนักงาน กสทช. เขต 16	ราชบุรี	ราชบุรี,เพชรบุรี,สมุทรสาคร,สมุทรสงคราม, นครปฐม

2.3.2 สำนักงาน กสทช. ภาค 2 แสดงรายละเอียดได้ดังตารางที่ 2.3

ตารางที่ 2.3 ตารางตำแหน่งของภาค 2 และเขต 21 ถึง 25 ของสำนักงาน กสทช.

สำนักงาน กสทช.	จังหวัดที่ตั้งที่ทำการ	ท้องที่ความรับผิดชอบ
สำนักงาน กสทช. ภาค 2	ขอนแก่น	ขอนแก่น,มหาสารคาม,กาฬสินธุ์
สำนักงาน กสทช. เขต 21	ร้อยเอ็ด	ร้อยเอ็ด,ยโสธร,อำนาจเจริญ
สำนักงาน กสทช. เขต 22	อุบลราชธานี	อุบลราชธานี,ศรีสะเกษ,สุรินทร์
สำนักงาน กสทช. เขต 23	นครราชสีมา	นครราชสีมา,บุรีรัมย์,ชัยภูมิ
สำนักงาน กสทช. เขต 24	อุดรธานี	อุดรธานี,หนองคาย,หนองบัวลำภู,เลย
สำนักงาน กสทช. เขต 25	นครพนม	นครพนม,สกลนคร,มุกดาหาร,บึงกาฬ

2.3.3 สำนักงาน กสทช. ภาค 3 แสดงรายละเอียดได้ดังตารางที่ 2.4

ตารางที่ 2.4 ตารางตำแหน่งของภาค 3 และเขต 31 ถึง 35 ของสำนักงาน กสทช.

สำนักงาน กสทช.	จังหวัดที่ตั้งที่ทำการ	ท้องที่ความรับผิดชอบ
สำนักงาน กสทช. ภาค 3	เชียงใหม่	เชียงใหม่,แม่ฮ่องสอน,ลำพูน
สำนักงาน กสทช. เขต 31	ลำปาง	ลำปาง,แพร่,ตาก
สำนักงาน กสทช. เขต 32	ลำพูน	ลำพูน,เชียงใหม่,แม่ฮ่องสอน
สำนักงาน กสทช. เขต 33	พิษณุโลก	พิษณุโลก,เพชรบูรณ์,สุโขทัย,อุตรดิตถ์
สำนักงาน กสทช. เขต 34	เชียงราย	เชียงราย,น่าน,พะเยา
สำนักงาน กสทช. เขต 35	นครสวรรค์	นครสวรรค์,กำแพงเพชร,อุทัยธานี,พิจิตร

### 2.3.4 สำนักงาน กสทช. ภาค 4 แสดงรายละเอียดได้ดังตารางที่ 2.5

ตารางที่ 2.5 ตารางตำแหน่งของภาค 4 และเขต 41 ถึง 45 ของสำนักงานกสทช.

สำนักงาน กสทช.	จังหวัดที่ตั้งที่ทำการ	ท้องที่ความรับผิดชอบ
สำนักงาน กสทช. ภาค 4	สงขลา	สงขลา,สตูล
สำนักงาน กสทช. เขต 41	ยะลา	ยะลา,นราธิวาส,ปัตตานี
สำนักงาน กสทช. เขต 42	ภูเก็ต	ภูเก็ต,พังงา,กระบี่
สำนักงาน กสทช. เขต 43	นครศรีธรรมราช	นครศรีธรรมราช,ตรัง,พัทลุง
สำนักงาน กสทช. เขต 44	สุราษฎร์ธานี	สุราษฎร์ธานี,ระนอง
สำนักงาน กสทช. เขต 45	ชุมพร	ชุมพร,ประจวบคีรีขันธ์

## 2.4 ศึกษาการทดสอบคุณภาพการให้บริการโทรศัพท์เคลื่อนที่ (Quality of Service And Complain)

2.4.1 แผนการทดสอบ QoS ตาม เขตและภูมิภาคต่างๆ ของ กสทช. แสดงได้ดังตารางที่

2.6 - 2.7 ตามลำดับ

ตารางที่ 2.6 ตาราง Quality of Services (Data Services)

เลขที่กล่อง	เขต/จังหวัด	เทคโนโลยี	ภูมิภาค
1	กสทช. ภูมิภาค 1 กรุงเทพมหานคร	4G	1
2	กสทช. เขต 11 สมุทรปราการ	3G	1
3	กสทช. เขต 12 จันทบุรี	3G	1
4	กสทช. เขต 13 สุพรรณบุรี	5G	1
5	กสทช. เขต 14 ปราจีนบุรี	4G	1
6	กสทช. เขต 15 ออยุธยา	3G	1
7	กสทช. เขต 16 ราชบุรี	4G	1
8	กสทช. ภูมิภาค 2 ขอนแก่น	4G	2
9	กสทช. เขต 21 ร้อยเอ็ด	4G	2
10	กสทช. เขต 22 อุบลราชธานี	4G	2
11	กสทช. เขต 23 นครราชสีมา	5G	2
12	กสทช. เขต 24 อุตรธานี	3G	2

ตารางที่ 2.6 ตาราง Quality of Services (Data Services) (ต่อ)

13	กสทช. เขต 25 นครพนม	4G	2
14	กสทช. ภูมิภาค 3 เชียงใหม่	4G	3
15	กสทช. เขต 31 ลำปาง	4G	3
16	กสทช. เขต 32 ลำพูน	4G	3
17	กสทช. เขต 33 พิชญ์โลก	4G	3
18	กสทช. เขต 34 เชียงราย	3G	3
19	กสทช. เขต 35 นครสวรรค์	5G	3
20	กสทช. ภูมิภาค 4 สงขลา	4G	4
21	กสทช. เขต 41 ยะลา	3G	4
22	กสทช. เขต 42 ภูเก็ต	4G	4
23	กสทช. เขต 43 นครศรีธรรมราช	4G	4
24	กสทช. เขต 44 สุราษฎร์ธานี	5G	4
25	กสทช. เขต 45 ชุมพร	3G	4
26	ทั่วประเทศ	TEAM A	-
27	ทั่วประเทศ	ANY TEAM	-
28	กสทช. เขต 41 ยะลา	3G	4
29	ทั่วประเทศ	TEAM B	-
30	กสทช. ภูมิภาค 1 กรุงเทพมหานคร	-	1
31	กสทช. ภูมิภาค 1 กรุงเทพมหานคร	2G	1
32	ทั่วประเทศ	TEAM A	-
33	ทั่วประเทศ	TEAM B	-
34	ทั่วประเทศ	5G	-

ตารางที่ 2.7 ตาราง Quality of Services (Voice Services)

เลขที่กล่อง	เขต/จังหวัด	Call Type	ภูมิภาค
1	กสทช. ภูมิภาค 1 กรุงเทพมหานคร	ON NET & OFF NET	1
2	กสทช. เขต 11 สมุทรปราการ	ON NET & OFF NET	1
3	กสทช. เขต 12 จันทบุรี	ON NET & OFF NET	1
4	กสทช. เขต 13 สุพรรณบุรี	ON NET & OFF NET	1
5	กสทช. เขต 14 ปราจีนบุรี	ON NET & OFF NET	1
6	กสทช. เขต 15 ออยุธยา	ON NET & OFF NET	1
7	กสทช. เขต 16 ราชบุรี	ON NET & OFF NET	1
8	กสทช. ภูมิภาค 2 ขอนแก่น	ON NET & OFF NET	2
9	กสทช. เขต 21 ร้อยเอ็ด	ON NET & OFF NET	2
10	กสทช. เขต 22 อุบลราชธานี	ON NET & OFF NET	2
11	กสทช. เขต 23 นครราชสีมา	ON NET & OFF NET	2
12	กสทช. เขต 24 อุดรธานี	ON NET & OFF NET	2
13	กสทช. เขต 25 นครพนม	ON NET & OFF NET	2
14	กสทช. ภูมิภาค 3 เชียงใหม่	ON NET & OFF NET	3
15	กสทช. เขต 31 ลำปาง	ON NET & OFF NET	3
16	กสทช. เขต 32 ลำพูน	ON NET & OFF NET	3
17	กสทช. เขต 33 พิชญโลก	ON NET & OFF NET	3
18	กสทช. เขต 34 เชียงราย	ON NET & OFF NET	3
19	กสทช. เขต 35 นครสวรรค์	ON NET & OFF NET	3
20	กสทช. ภูมิภาค 4 สงขลา	ON NET & OFF NET	4
21	กสทช. เขต 41 ยะลา	ON NET & OFF NET	4
22	กสทช. เขต 42 ภูเก็ต	ON NET & OFF NET	4
23	กสทช. เขต 43 นครศรีธรรมราช	ON NET & OFF NET	4
24	กสทช. เขต 44 สุราษฎร์ธานี	ON NET & OFF NET	4
25	กสทช. เขต 45 ชุมพร	ON NET & OFF NET	4
26	ทั่วภูมิภาค	ON NET & OFF NET	-

ตารางที่ 2.7 ตาราง Quality of Services (Voice Services) (ต่อ)

27	ทั่วประเทศ	ON NET & OFF NET	-
28	กสทช. เขต 41 ยะลา	ON NET & OFF NET	4
29	ทั่วประเทศ	ON NET & OFF NET	-
30	กสทช. ภูมิภาค 1 กรุงเทพมหานคร	ON NET & OFF NET	1
31	กสทช. ภูมิภาค 1 กรุงเทพมหานคร	ON NET & OFF NET	1
32	ทั่วประเทศ	ON NET & OFF NET	-
33	ทั่วประเทศ	ON NET & OFF NET	-
34	กสทช. ภูมิภาค 1 กรุงเทพมหานคร	ON NET & OFF NET	1

## 2.4.2 บริการโทรคมนาคมผ่านโครงข่ายโทรคมนาคมเคลื่อนที่

### 2.4.2.1 บริการประเภทเสียง แสดงรายละเอียดการทดสอบได้ดังตารางที่ 2.8

ตารางที่ 2.8 บริการโทรคมนาคมผ่านโครงข่ายโทรคมนาคมเคลื่อนที่ประเภทเสียง

บริการประเภทเสียง				
อัตราส่วนการ เรียกสำเร็จ (Successful Call Ratio) <u>ต้องไม่น้อยกว่าร้อยละ</u> <u>90</u> กรณีที่เป็นการโทร ภายในโครงข่าย ของผู้ประกอบการ เดียวกัน	อัตราส่วนการ เรียกสำเร็จ (Successful Call Ratio) <u>ต้องไม่น้อยกว่า</u> <u>ร้อยละ 90</u> กรณีที่เป็นการโทร ข้ามโครงข่าย ต่างผู้ประกอบการ	อัตราส่วนของ กรณีที่สายหลุด (Drop Call Rate) <u>ต้องไม่มากกว่า</u> <u>ร้อยละ 2</u>	คุณภาพของ เสียง(Mean Opinion Score (MOS)) <u>ต้องไม่น้อยกว่า</u> <u>ร้อยละ 2.5</u> (ร้อยละ 90 ของ ข้อมูลที่วัดได้ ต้องไม่ต่ำกว่าค่า นี้)	อัตราส่วนของกรณีที่ สายหลุด (Drop Call Rate) <u>ต้องไม่มากกว่าร้อยละ</u> <u>2</u> ในพื้นที่เฝ้าระวัง
แนวทางการการวัดและการรายงานผล วัดใน 2 ช่วงเวลาได้แก่ 1. ช่วงเวลา 10.00 – 13.00 น. เฉลี่ยทุก 3 เดือน 2. ช่วงเวลา 16.00 – 19.00 น. เฉลี่ยทุก 3 เดือน โดยแยกข้อมูลในแต่ละช่วงเวลา			แนวทางการการวัดและการรายงานผล วัดเฉลี่ยทุก 3 เดือน	



### 2.4.2.2 บริการประเภทข้อมูล แสดงได้ดังตารางที่ 2.9

ตารางที่ 2.9 บริการโทรคมนาคมผ่านโครงข่ายโทรคมนาคมเคลื่อนที่ประเภทข้อมูล

บริการประเภทข้อมูล							
ค่า RTT (Round Trip Time)	FTP Transfer (DL & UL)		HTTP success ratio	Streaming		SMS/MMS	
	FTP success ratio	FTP ratio subjected to specified data rate		Streaming service accessibility	Streaming reproduction success ratio	SMS/MMS end-to-end delivery time	SMS/MMS completion success ratio

### 2.4.3 ค่าชี้วัดคุณภาพบริการประเภทข้อมูล และค่าเป้าหมายของ Ping ดังตารางที่ 2.10

ตารางที่ 2.10 ตารางค่าชี้วัดคุณภาพบริการประเภทข้อมูล และค่าเป้าหมายของ Ping

ค่า RTT (Round Trip Time)			
บนเทคโนโลยี 2G	บนเทคโนโลยี 3G	บนเทคโนโลยี 4G, 5G (Not 2600 MHz)	บนเทคโนโลยี 5G (2600 MHz)
เป้าหมาย			
ค่าเฉลี่ย RTT ไม่เกิน 1000 มิลลิวินาที	ค่าเฉลี่ย RTT ไม่เกิน 500 มิลลิวินาที	ค่าเฉลี่ย RTT ไม่เกิน 150 มิลลิวินาที	ค่าเฉลี่ย RTT ไม่เกิน 110 มิลลิวินาที
ค่าเป้าหมาย $\geq 80$			
วิธีการทดสอบ			
ทดสอบด้วยคำสั่ง Ping แบบ ICMP (Internet Control Message Protocol) ขนาด 32 Byte จากอุปกรณ์ปลายทางไปยัง Server			



#### 2.4.5 ค่าชี้วัดคุณภาพบริการประเภทข้อมูล และค่าเป้าหมายของ HTTP

ตารางที่ 2.12 ตารางค่าชี้วัดคุณภาพบริการประเภทข้อมูล และค่าเป้าหมายของ HTTP

HTTP success ratio			
บนเทคโนโลยี 2G	บนเทคโนโลยี 3G	บนเทคโนโลยี 4G	บนเทคโนโลยี 5G
<u>ค่าเป้าหมาย</u> ไม่ต่ำกว่าร้อยละ 80	<u>ค่าเป้าหมาย</u> ไม่ต่ำกว่าร้อยละ 90		
<u>ขนาด Website</u> <u>สำหรับใช้ทดสอบ</u> 1 Mbyte	<u>ขนาด Website สำหรับใช้ทดสอบ</u> 3.5 Mbyte		
เวลาที่กำหนด 10 นาที	เวลาที่กำหนด 3 นาที	เวลาที่กำหนด 1 นาที	เวลาที่กำหนด 25 วินาที

2.4.6 การวิเคราะห์ข้อร้องเรียนปัญหาคุณภาพการให้บริการโทรศัพท์เคลื่อนที่และการกำหนดเงื่อนไขการทดสอบ

##### 1. การแยกประเด็นปัญหา

- ประเด็น ผู้ร้องเรียนประสบปัญหาการใช้บริการประเภทเสียง
  - ไม่สามารถโทรออกหรือรับสายได้ เสียงไม่ชัดหลุดบ่อย สัญญาณขาดหายระหว่างสนทนา ไม่สามารถใช้งานได้อย่างปกติ
- ประเด็น ผู้ร้องเรียนประสบปัญหาการใช้บริการประเภทข้อมูล
  - ไม่สามารถใช้อินเทอร์เน็ตผ่านโทรศัพท์เคลื่อนที่ได้ตามปกติ ไม่สามารถเข้าเว็บไซต์ต่างๆ หรือ ไม่สามารถใช้งานแอปพลิเคชันได้
  - คุณภาพสัญญาณอินเทอร์เน็ตผ่านโทรศัพท์เคลื่อนที่ไม่ได้ดี ความเร็วดาวน์โหลด อัปโหลดต่ำ ไม่สามารถใช้งานได้ตามปกติ
  - ไม่สามารถใช้บริการได้ตามรายการส่งเสริมการขาย
- ประเด็น ผู้ร้องเรียนประสบปัญหาไม่มีสัญญาณโทรศัพท์เคลื่อนที่
  - ไม่มีสัญญาณ ไม่มีบริการ ไม่สามารถติดต่อสื่อสารได้
- ประเด็น เทคโนโลยีที่ประสบปัญหา
  - ปัญหาการเข้าถึง เทคโนโลยีการให้บริการ เช่น 2G 3G และ 4G

- ปัญหาการยี่ห้อนุ่นของเครื่องโทรศัพท์เคลื่อนที่ รองรับการเข้าถึงบริการตามเทคโนโลยีหรือไม่
- ประเด็น สถานที่ หรือบริเวณประสบปัญหา
  - พบปัญหาเฉพาะ ณ บริเวณ สถานที่ตำแหน่ง ที่ชัดเจน พื้นที่อาคารสูง คอนโดฯ โรงงานอุตสาหกรรม(ภายใน หรือ ภายนอกอาคาร)
  - ระบุมุมพื้นที่โดยรวมทั้งหมดหมู่บ้าน โรงงานอุตสาหกรรม
  - ระบุมุมพื้นที่เป็นวงกว้างทั้งตำบล ทั้งอำเภอ ภาพรวมทั้งจังหวัด

## 2.การวิเคราะห์พื้นที่ประสบปัญหา และลักษณะการกำหนดเงื่อนไขการทดสอบ

ตารางที่ 2.13 ตารางการวิเคราะห์พื้นที่ประสบปัญหา และลักษณะการกำหนดเงื่อนไขการทดสอบ

Location	Stationary Test	Walk Test	Drive Test	Remark
บ้าน,ทาวโฮม				กรณีทดสอบบริเวณหน้าบ้าน หรือภายในบ้านผู้ร้อง
โรงแรม,คอนโด, อาคารสูง				กรณีที่ผู้ร้องอนุญาตให้ทดสอบในห้อง
				กรณีที่ผู้ร้องไม่อนุญาตให้ทดสอบในห้อง
หมู่บ้าน				กรณีที่เป็นพื้นที่หมู่บ้านเล็กๆ ควรที่จะวิ่งแบบ (Lock Technology)
ตำบล,อำเภอ, จังหวัด				กรณีที่เป็นพื้นที่กว้าง ควรที่จะวิ่ง (Voice + Data) ในรอบเดียวกัน

### 3.Script ทดสอบ Voice & Data

#### บริการประเภทเสียง (Voice)

- Voice Shot Call (Operator)
- Voice Long Call (Operator)
- Voice Long Call On-net and Off-net (Operator)
- Voice Shot Call On-net and Off-net (Operator)

#### บริการประเภทข้อมูล (RTT HTTP FTP-DL and FTP-UL)

- Data Auto mode
- Lock 5G (Standalone Mode) ที่ใช้คลื่นความถี่ย่าน 2,600 เมกะเฮิรตซ์
- Lock 4G, 5G (Non-standalone Mode), 5G (Standalone Mode) ที่ไม่ใช่คลื่นความถี่ย่าน 2,600 เมกะเฮิรตซ์
- Lock 3G
- Lock 2G

#### บริการประเภทเสียงและข้อมูล (Voice RTT FTP-DL FTP-UL and HTTP)

- Data & Voice (Operator)

## 4.รายละเอียดการทดสอบ Script Test (Voice) แสดงดังตารางที่ 2.14

ตารางที่ 2.14 ตารางรายละเอียด Script test (Voice)

Voice test	Voice Shot Call	<u>ทดสอบการใช้งานประเภทเสียง</u> <u>กรณีที่โทรภายในโครงข่ายและโทรข้ามโครงข่าย</u> เป็นการโทรระยะสั้น ระยะเวลา 50 วินาที ในการโทร (Duration time)
	Voice Long Call	<u>ทดสอบการใช้งานประเภทเสียงกรณีที่โทรภายในโครงข่าย</u> <u>และโทรข้ามโครงข่าย</u> เป็นการโทรระยะยาว ระยะเวลา 180 วินาที ในการโทร (Duration time)
	Voice Shot Call On-net Only	<u>ทดสอบการใช้งานประเภทเสียง</u> <u>กรณีที่โทรภายในโครงข่ายเท่านั้น</u> เป็นการโทรระยะสั้น ระยะเวลา 50 วินาที ในการโทร (Duration time)
	Voice Long Call On-net Only	<u>ทดสอบการใช้งานประเภทเสียง</u> <u>กรณีที่โทรภายในโครงข่ายเท่านั้น</u> เป็นการโทรระยะยาว ระยะเวลา 180 วินาที ในการโทร (Duration time)
	Voice Shot Call Off-net Only	<u>ทดสอบการใช้งานประเภทเสียง</u> <u>กรณีที่โทรข้ามโครงข่ายเท่านั้น</u> เป็นการโทรระยะสั้น ระยะเวลา 50 วินาที ในการโทร (Duration time)
	Voice Long Call Off-net Only	<u>ทดสอบการใช้งานประเภทเสียง</u> <u>กรณีที่โทรข้ามโครงข่ายเท่านั้น</u> เป็นการโทรระยะยาว ระยะเวลา 180 วินาที ในการโทร (Duration time)

## 5. รายละเอียดการทดสอบ Script Test (Data) แสดงดังตารางที่ 2.15

ตารางที่ 2.15 ตารางรายละเอียด Script test (Data)

Data test	Data Auto mode	ทดสอบการใช้งานประเภทข้อมูล กรณีที่ตั้งเทคโนโลยีในโหมดอัตโนมัติ (Preferred network type 5G SA)
	Lock 5G SA	ทดสอบการใช้งานประเภทข้อมูล กรณีที่ตั้งเทคโนโลยีบน 5G SA เท่านั้น (NR Only)
	Prefer 4G / 5G NSA	ทดสอบการใช้งานประเภทข้อมูล กรณีที่ตั้งเทคโนโลยีบน 4G และ 5G เท่านั้น (NR/LTE Only)
	Lock 4G	ทดสอบการใช้งานประเภทข้อมูล กรณีที่ตั้งเทคโนโลยีบน 4G เท่านั้น (LTE Only)
	Lock 3G	ทดสอบการใช้งานประเภทข้อมูล กรณีที่ตั้งเทคโนโลยีบน 3G เท่านั้น (WCDMA Only)
	Lock 2G	ทดสอบการใช้งานประเภทข้อมูล กรณีที่ตั้งเทคโนโลยีบน 2G เท่านั้น (GSM Only)

## บทที่ 3

### การออกแบบและการจัดทำโครงงาน

#### 3.1 การออกแบบ

##### 3.1.1 การออกแบบระบบวิเคราะห์ข้อมูลเพื่อนำไปใช้สำหรับรายงานค่าชี้วัดคุณภาพการบริการโทรคมนาคมผ่านโครงข่ายโทรคมนาคมเคลื่อนที่

3.1.1.1 เจาะลึกและวิธีการทดสอบคุณภาพการให้บริการโทรศัพท์เคลื่อนที่ประเภทข้อมูลตามข้อกำหนดมาตรฐานโดยการทดสอบในแต่ละรูปแบบชี้แจงรายละเอียดดังต่อไปนี้

3.1.1.1.1 การทดสอบคุณภาพการให้บริการโทรศัพท์เคลื่อนที่ประเภทข้อมูลในส่วนของการทดสอบดาวน์โหลดเว็บไซต์ (HTTP Download)

ดำเนินการทดสอบโดยอุปกรณ์ปลายทางใส่ URL (Uniform Resource Locator) ของเว็บไซต์ผ่าน Browser ที่ใช้สำหรับการทดสอบเพื่อทำการดาวน์โหลดเว็บไซต์จากเซิร์ฟเวอร์กลางที่จัดเตรียมไว้โดยสำนักงาน กสทช. ผ่านโปรโตคอล HTTP (Hypertext Transfer Protocol) โดยมีขนาดเว็บไซต์และเวลาที่กำหนด (Timeout) สำหรับค่ากำหนดต่างๆ เป็นไปตามตารางที่ 3.1 โดยขนาดเว็บไซต์ที่โหลดได้มาต้องเท่ากับขนาดของเว็บไซต์ที่ทดสอบจึงจะถือว่าการทำ HTTP ดาวน์โหลดนั้นสำเร็จ ชนิดของ Browser ที่ใช้ในการทดสอบอ้างอิงตามมาตรฐาน ETSI TS 102 250-5 v2.5.1(2019-11) ข้อ 4.3.1 [3]

ตารางที่ 3.1 ตารางขนาดเว็บไซต์และเวลาที่กำหนดสำหรับการทดสอบดาวน์โหลดเว็บไซต์ [1]

เทคโนโลยี	ขนาดเว็บไซต์สำหรับทดสอบ	เวลาที่กำหนด
2G	1 Mbyte	10 นาที
3G	3.5 Mbyte	3 นาที
4G และ 5G (NSA) และ 5G (SA) ที่ไม่ใช้คลื่นความถี่ย่าน 2600 เมกะเฮิร์ตซ์		1 นาที
5G (SA) ที่ใช้คลื่นความถี่ย่าน 2600 เมกะเฮิร์ตซ์		25 วินาที

3.1.1.1.2 การทดสอบคุณภาพการให้บริการโทรศัพท์เคลื่อนที่ประเภทข้อมูลในส่วนของการ FTP Upload และ FTP Download



ดำเนินการทดสอบโดยอุปกรณ์ปลายทางทำการเชื่อมต่อไปยัง IP Address ของเซิร์ฟเวอร์กลางที่จัดเตรียมไว้โดยสำนักงาน กสทช. โดยมีขนาดไฟล์และเวลาที่กำหนดสำหรับ Download หรือ Upload ผ่าน FTP เป็นไปตามที่กำหนดในตารางที่ 3.2 รูปแบบการทดสอบอ้างอิงตามมาตรฐาน ETSI TS 102 250-2 v2.7.1 (2019-11) ข้อ 6.1.7 [3] โดยใช้จุดเริ่มต้นวัดตาม Method B การกำหนด Idle time ระหว่างการทดสอบ FTP Download หรือ Upload นั้นสามารถกำหนดได้ตามความเหมาะสม

ตารางที่ 3.2 ตารางขนาดไฟล์และเวลาที่กำหนดสำหรับทดสอบ Download และ Upload ผ่าน FTP

ประเภท	เทคโนโลยี	ขนาดไฟล์	เวลาที่กำหนด
Download	2G	1 Mbyte	10 นาที
	3G	15 Mbyte	
	4G และ 5G (ที่ไม่ได้ใช้คลื่นความถี่ย่าน 2600 เมกะเฮิรตซ์)	30 Mbyte	
	5G (ที่ใช้คลื่นความถี่ย่าน 2600 เมกะเฮิรตซ์)	500 Mbyte	
Upload	2G	0.5 Mbyte	
	3G	2 Mbyte	
	4G และ 5G (ที่ไม่ได้ใช้คลื่นความถี่ย่าน 2600 เมกะเฮิรตซ์)	4 Mbyte	
	5G (ที่ใช้คลื่นความถี่ย่าน 2600 เมกะเฮิรตซ์)	100 Mbyte	

หมายเหตุ รูปแบบการทดสอบในตารางที่ 3.2 ใช้ Single Session เหมือนกันทั้งหมด

#### 3.1.1.1.3 การทดสอบคุณภาพการให้บริการโทรศัพท์เคลื่อนที่ประเภทข้อมูลในส่วนของ RTT (Round Trip Time)

ดำเนินการทดสอบด้วยการส่งคำสั่ง Ping แบบ ICMP (Internet Control Message Protocol) ขนาด 32 Bytes จากอุปกรณ์ปลายทางไปยังเครื่องเซิร์ฟเวอร์กลางที่จัดเตรียมไว้โดยสำนักงาน กสทช. จนกระทั่งแพ็คเก็ตกลับมายังอุปกรณ์ปลายทางอีกครั้งหนึ่ง ด้วยวิธีการระบุเป้าหมายของคำสั่งเป็นเลขหมาย IP หรือ IP Address เท่านั้น ค่ากำหนดต่างๆ แสดงดังตารางที่ 3.3 และรูปแบบการทดสอบอ้างอิงตามมาตรฐาน ETSI TS 102 250-2 v2.7.1 (2019-11) ข้อ 6.3 [3]

ตารางที่ 3.3 ตารางขนาดและหมายเลข IP Address สำหรับการทดสอบแบบ Ping [1]

คำสั่ง	รูปแบบการทดสอบ	ขนาด	IP Address
ping	Internet Control Message Protocol (ICMP)	32 bytes	202.25.84.49

3.1.1.1.4 รายละเอียดการตั้งค่าเพื่อเลือกเทคโนโลยีสำหรับการทดสอบคุณภาพการให้บริการโทรศัพท์เคลื่อนที่ประเภทข้อมูล

เนื่องจากการจัดทำรายงานค่าชี้วัดคุณภาพการให้บริการโทรคมนาคมในแต่ละไตรมาสทางสำนักดูแลกิจการโทรคมนาคม (ตท.) จะต้องมีการวางแผนเพื่อทำการทดสอบการให้บริการโทรคมนาคมให้สอดคล้องกับเทคโนโลยีที่มีการพัฒนาเปลี่ยนแปลงตามสถานการณ์ปัจจุบันมากที่สุด ดังนั้นในแต่ละพื้นที่จึงจะต้องมีการทดสอบของเทคโนโลยีที่แตกต่างกันตามความเหมาะสม โดยรูปแบบการตั้งค่าต่างๆ แสดงได้ดังตารางที่ 3.4

ตารางที่ 3.4 ตารางรายละเอียดวิธีการทดสอบคุณภาพการให้บริการประเภทข้อมูล

Data test	Data Auto mode	ทดสอบการใช้งานประเภทข้อมูล กรณีที่ตั้งเทคโนโลยีในโหมดอัตโนมัติ (Preferred network type 5G SA)
	Lock 5G SA	ทดสอบการใช้งานประเภทข้อมูล กรณีที่ตั้งเทคโนโลยีบน 5G SA เท่านั้น (NR Only)
	Prefer 4G / 5G NSA	ทดสอบการใช้งานประเภทข้อมูล กรณีที่ตั้งเทคโนโลยีบน 4G และ 5G เท่านั้น (NR/LTE Only)
	Lock 4G	ทดสอบการใช้งานประเภทข้อมูล กรณีที่ตั้งเทคโนโลยีบน 4G เท่านั้น (LTE Only)
	Lock 3G	ทดสอบการใช้งานประเภทข้อมูล กรณีที่ตั้งเทคโนโลยีบน 3G เท่านั้น (WCDMA Only)
	Lock 2G	ทดสอบการใช้งานประเภทข้อมูล กรณีที่ตั้งเทคโนโลยีบน 2G เท่านั้น (GSM Only)

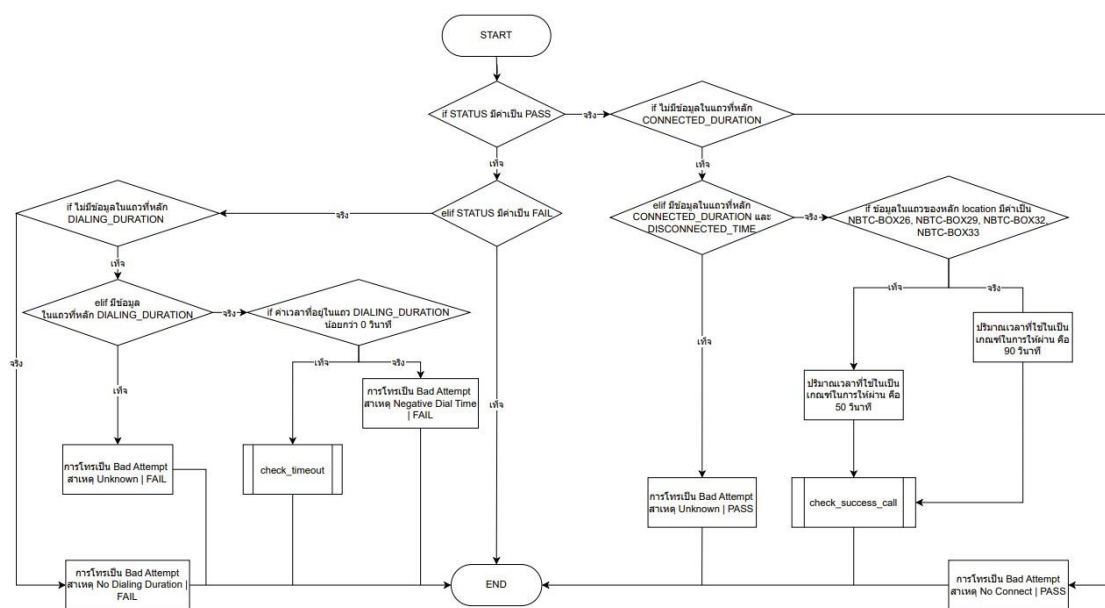
3.1.1.2 เงื่อนไขและวิธีการทดสอบคุณภาพการให้บริการโทรศัพท์เคลื่อนที่ประเภทเสียงตามข้อกำหนดมาตรฐาน

การตรวจสอบคุณภาพการให้บริการประเภทเสียงจะมีรูปแบบการทดสอบที่แตกต่างกันไปตามความเหมาะสม โดยรายละเอียดของการทดสอบแสดงได้ดังตารางที่ 3.5 และ

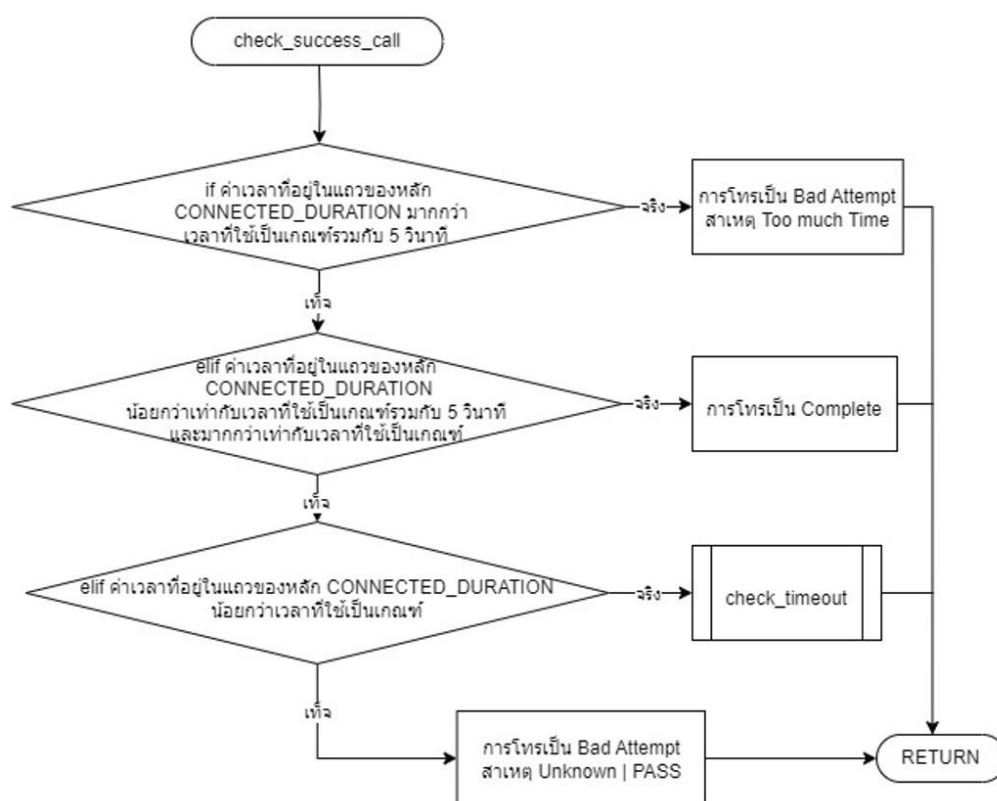
เงื่อนไขการให้เหตุผลในส่วนของ Success Call , Bad Attempt Call , Drop Call , Block Call  
เป็นไปตามแผนผังการดำเนินงานตามรูปที่ 3.1 – 3.7 ตามลำดับ

ตารางที่ 3.5 ตารางรายละเอียดวิธีการทดสอบคุณภาพการให้บริการประเภทเสียง

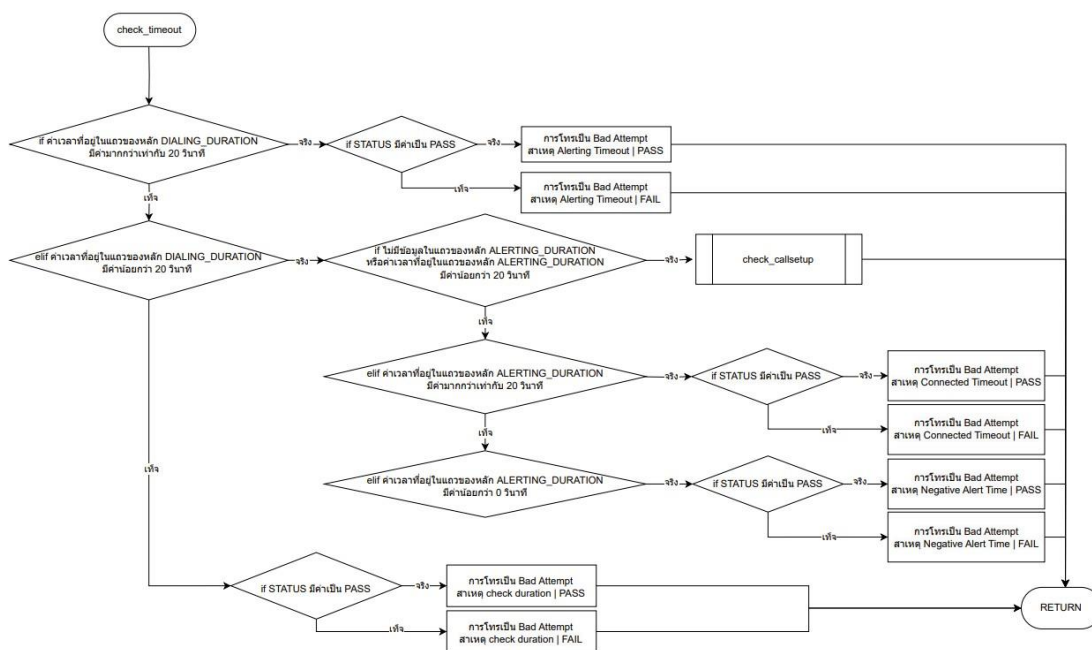
Voice test	Voice Shot Call	<u>ทดสอบการใช้งานประเภทเสียง</u> <u>กรณีที่โทรภายในโครงข่ายและโทรข้ามโครงข่าย</u> เป็นการโทรระยะสั้น ระยะเวลา 50 วินาที ในการโทร (Duration time)
	Voice Long Call	<u>ทดสอบการใช้งานประเภทเสียงกรณีที่โทรภายในโครงข่าย</u> <u>และโทรข้ามโครงข่าย</u> เป็นการโทรระยะยาว ระยะเวลา 180 วินาที ในการโทร (Duration time)
	Voice Shot Call On-net Only	<u>ทดสอบการใช้งานประเภทเสียง</u> <u>กรณีที่โทรภายในโครงข่ายเท่านั้น</u> เป็นการโทรระยะสั้น ระยะเวลา 50 วินาที ในการโทร (Duration time)
	Voice Long Call On-net Only	<u>ทดสอบการใช้งานประเภทเสียง</u> <u>กรณีที่โทรภายในโครงข่ายเท่านั้น</u> เป็นการโทรระยะยาว ระยะเวลา 180 วินาที ในการโทร (Duration time)
	Voice Shot Call Off-net Only	<u>ทดสอบการใช้งานประเภทเสียง</u> <u>กรณีที่โทรข้ามโครงข่ายเท่านั้น</u> เป็นการโทรระยะสั้น ระยะเวลา 50 วินาที ในการโทร (Duration time)
	Voice Long Call Off-net Only	<u>ทดสอบการใช้งานประเภทเสียง</u> <u>กรณีที่โทรข้ามโครงข่ายเท่านั้น</u> เป็นการโทรระยะยาว ระยะเวลา 180 วินาที ในการโทร (Duration time)



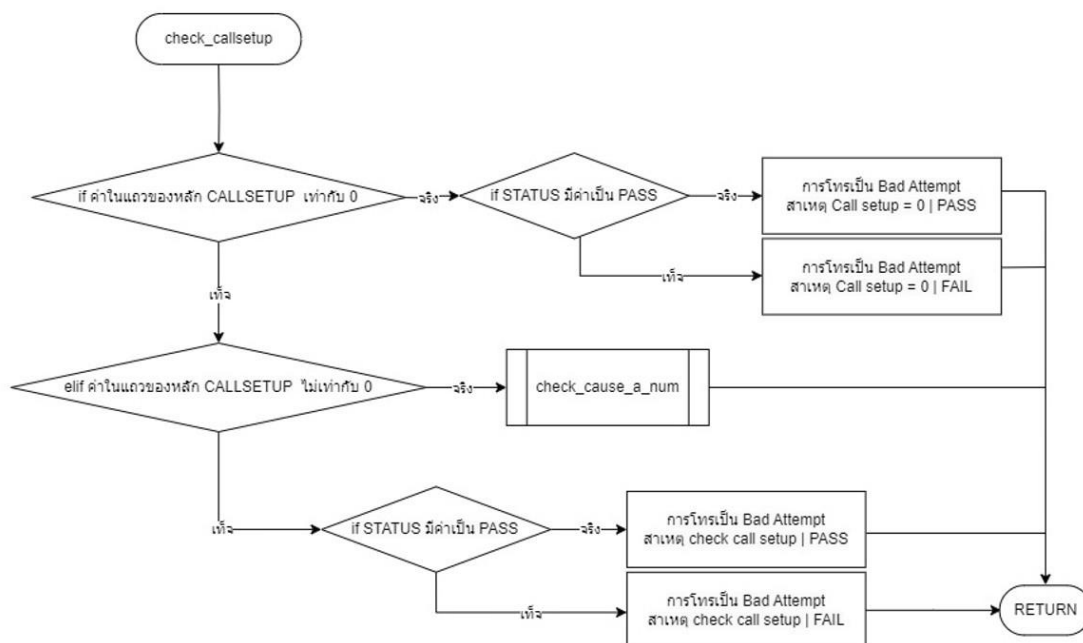
รูปที่ 3.1 แผนผังการดำเนินการวิเคราะห์ข้อมูลประเภทเสียง 1



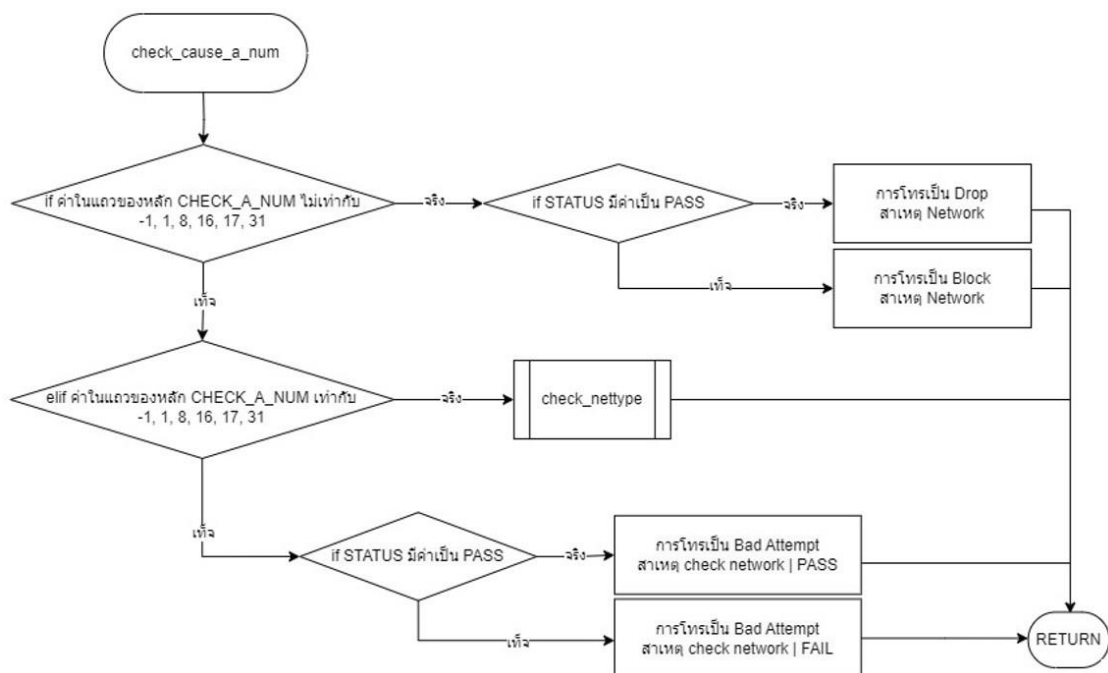
รูปที่ 3.2 แผนผังการดำเนินการวิเคราะห์ข้อมูลประเภทเสียง 2



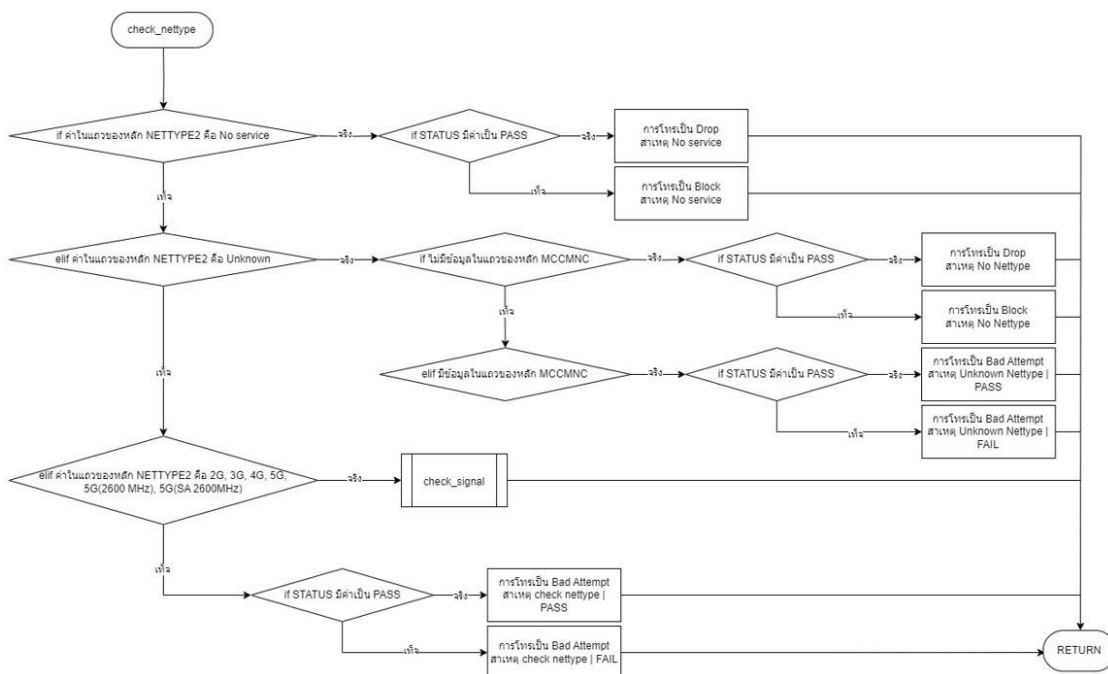
รูปที่ 3.3 แผนผังการดำเนินการวิเคราะห์ข้อมูลประเภทเสียง 3



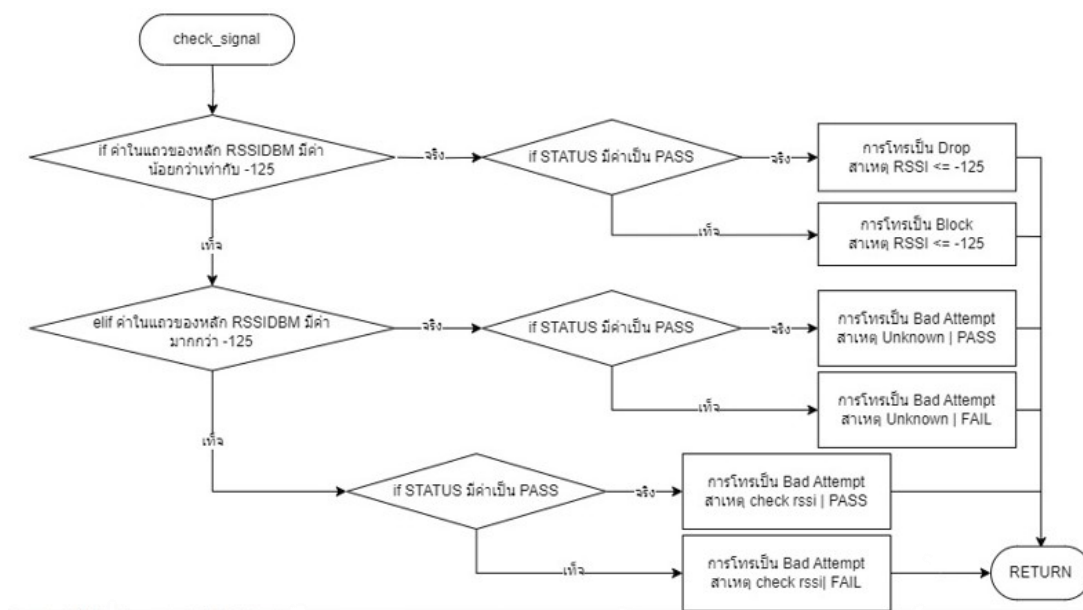
รูปที่ 3.4 แผนผังการดำเนินการวิเคราะห์ข้อมูลประเภทเสียง 4



รูปที่ 3.5 แผนผังการดำเนินการวิเคราะห์ข้อมูลประเภทเสียง 5

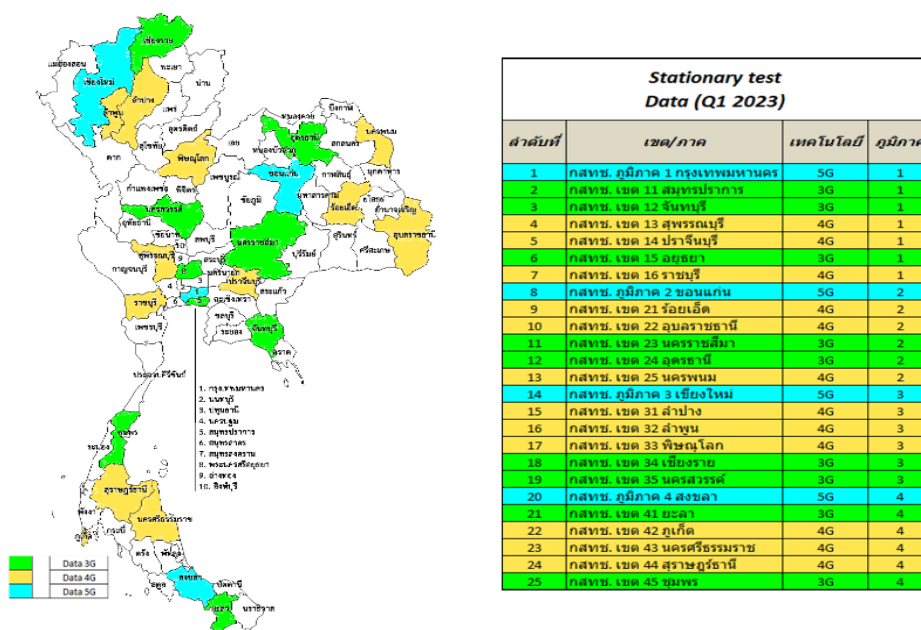


รูปที่ 3.6 แผนผังการดำเนินการวิเคราะห์ข้อมูลประเภทเสียง 6

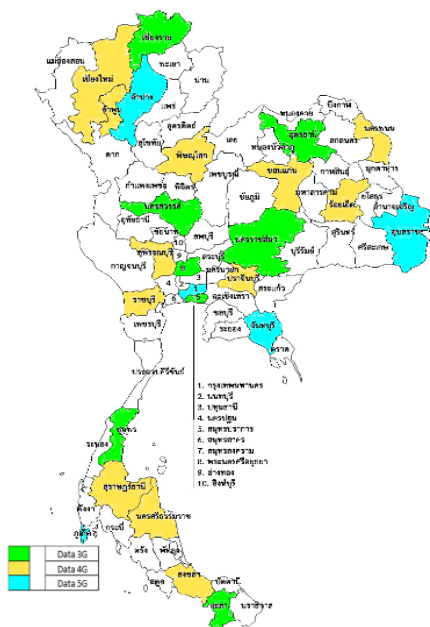


รูปที่ 3.7 แผนผังการดำเนินการวิเคราะห์ข้อมูลประเภทเสียง 7

3.1.1.3 แผนการทดสอบแบบแยกเทคโนโลยีในแต่ละพื้นที่การให้บริการของสำนักงาน กสทช. ประจำเขตและภาคอ้างอิงจากแผนการดำเนินงานของสำนักกำกับดูแลกิจการโทรคมนาคม (ตท.) โดยแสดงแผนการทดสอบประจำปี 2566 ได้ตามรูปที่ 3.8 – 3.11 ตามลำดับ

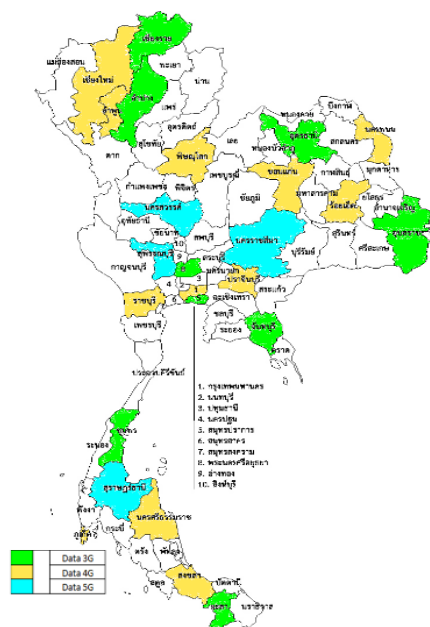


รูปที่ 3.8 แผนการดำเนินการทดสอบประเภทข้อมูลประจำไตรมาสที่ 1



ลำดับที่	เขต/ภาค	เทคโนโลยี	ภูมิภาค
1	กสทช. ภูมิภาค 1 กรุงเทพมหานคร	3G	1
2	กสทช. เขต 11 สมุทรปราการ	5G	1
3	กสทช. เขต 12 จันทบุรี	3G	1
4	กสทช. เขต 13 สุพรรณบุรี	4G	1
5	กสทช. เขต 14 ปราจีนบุรี	4G	1
6	กสทช. เขต 15 อุทัยธานี	3G	1
7	กสทช. เขต 16 ราชบุรี	4G	1
8	กสทช. ภูมิภาค 2 ขอนแก่น	4G	2
9	กสทช. เขต 21 ร้อยเอ็ด	4G	2
10	กสทช. เขต 22 อุบลราชธานี	5G	2
11	กสทช. เขต 23 นครราชสีมา	3G	2
12	กสทช. เขต 24 อุดรธานี	3G	2
13	กสทช. เขต 25 นครพนม	4G	2
14	กสทช. ภูมิภาค 3 เชียงใหม่	4G	3
15	กสทช. เขต 31 ลำปาง	5G	3
16	กสทช. เขต 32 ลำพูน	4G	3
17	กสทช. เขต 33 พะเยา	4G	3
18	กสทช. เขต 34 เชียงราย	3G	3
19	กสทช. เขต 35 นครสวรรค์	3G	3
20	กสทช. ภูมิภาค 4 สงขลา	4G	4
21	กสทช. เขต 41 ยะลา	3G	4
22	กสทช. เขต 42 ภูเก็ต	5G	4
23	กสทช. เขต 43 นครศรีธรรมราช	4G	4
24	กสทช. เขต 44 สุราษฎร์ธานี	4G	4
25	กสทช. เขต 45 นมพร	3G	4

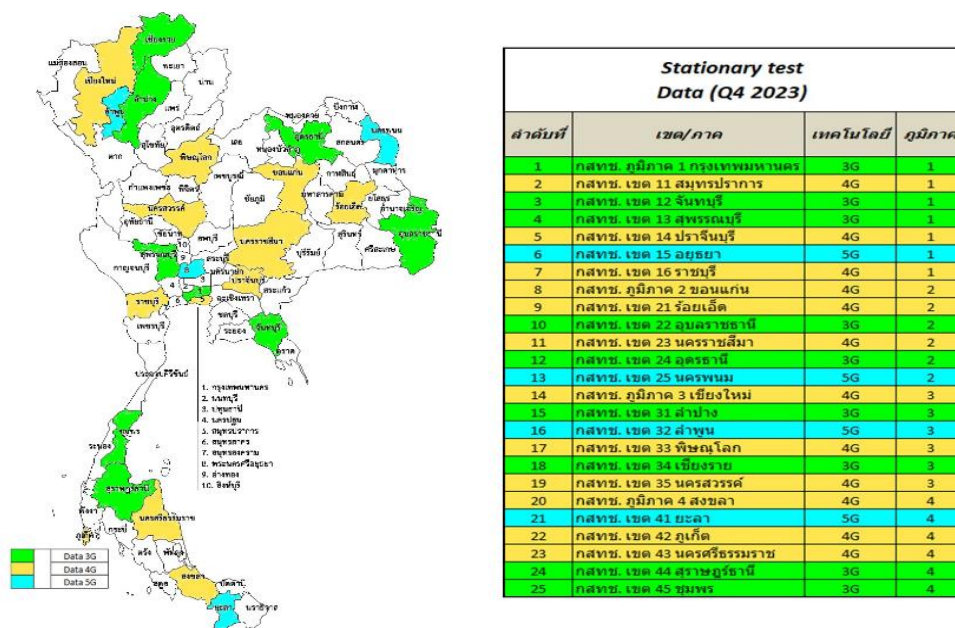
รูปที่ 3.9 แผนการดำเนินการทดสอบประสิทธิภาพข้อมูลประจำไตรมาสที่ 2



ลำดับที่	เขต/ภาค	เทคโนโลยี	ภูมิภาค
1	กสทช. ภูมิภาค 1 กรุงเทพมหานคร	4G	1
2	กสทช. เขต 11 สมุทรปราการ	3G	1
3	กสทช. เขต 12 จันทบุรี	3G	1
4	กสทช. เขต 13 สุพรรณบุรี	5G	1
5	กสทช. เขต 14 ปราจีนบุรี	4G	1
6	กสทช. เขต 15 อุทัยธานี	3G	1
7	กสทช. เขต 16 ราชบุรี	4G	1
8	กสทช. ภูมิภาค 2 ขอนแก่น	4G	2
9	กสทช. เขต 21 ร้อยเอ็ด	4G	2
10	กสทช. เขต 22 อุบลราชธานี	3G	2
11	กสทช. เขต 23 นครราชสีมา	5G	2
12	กสทช. เขต 24 อุดรธานี	3G	2
13	กสทช. เขต 25 นครพนม	4G	2
14	กสทช. ภูมิภาค 3 เชียงใหม่	4G	3
15	กสทช. เขต 31 ลำปาง	3G	3
16	กสทช. เขต 32 ลำพูน	4G	3
17	กสทช. เขต 33 พะเยา	4G	3
18	กสทช. เขต 34 เชียงราย	3G	3
19	กสทช. เขต 35 นครสวรรค์	5G	3
20	กสทช. ภูมิภาค 4 สงขลา	4G	4
21	กสทช. เขต 41 ยะลา	3G	4
22	กสทช. เขต 42 ภูเก็ต	4G	4
23	กสทช. เขต 43 นครศรีธรรมราช	4G	4
24	กสทช. เขต 44 สุราษฎร์ธานี	5G	4
25	กสทช. เขต 45 นมพร	3G	4

รูปที่ 3.10 แผนการดำเนินการทดสอบประสิทธิภาพข้อมูลประจำไตรมาสที่ 3





รูปที่ 3.11 แผนการดำเนินการทดสอบประเภทข้อมูลประจำไตรมาสที่ 4

3.1.1.4 เงื่อนไขการตรวจสอบ Network Type และแปลงให้อยู่ในรูปแบบของ Technology Generation อ้างอิงตามมาตรฐาน 3GPP แสดงเงื่อนไขได้ดังตารางที่ 3.6

ตารางที่ 3.6 ตารางเงื่อนไข Network Type และ Technology Generation

Network Type	Technology Generation
<ul style="list-style-type: none"> <li>- GSM</li> <li>- EDGE</li> <li>- GPRS</li> </ul>	2G
<ul style="list-style-type: none"> <li>- UMTS</li> <li>- HSPA</li> <li>- HSUPA</li> <li>- HSDPA</li> <li>- HSPAP</li> </ul>	3G
<ul style="list-style-type: none"> <li>- LTE</li> <li>- LTE Advance</li> <li>- LTE Advance Pro</li> </ul>	4G
<ul style="list-style-type: none"> <li>- NR NSA</li> <li>- NR SA</li> </ul>	5G

3.1.1.5 เงื่อนไขการตรวจสอบ Mobile Network Code (MNC) และแปลงให้อยู่ในรูปแบบสถานะว่าในผลการทดสอบนั้นๆ เป็นการใช้โครงข่ายของตนเอง หรือการใช้โครงข่ายของผู้ร่วมให้บริการ แสดงเงื่อนไขได้ดังตารางที่ 3.7 และ 3.8 ตามลำดับ

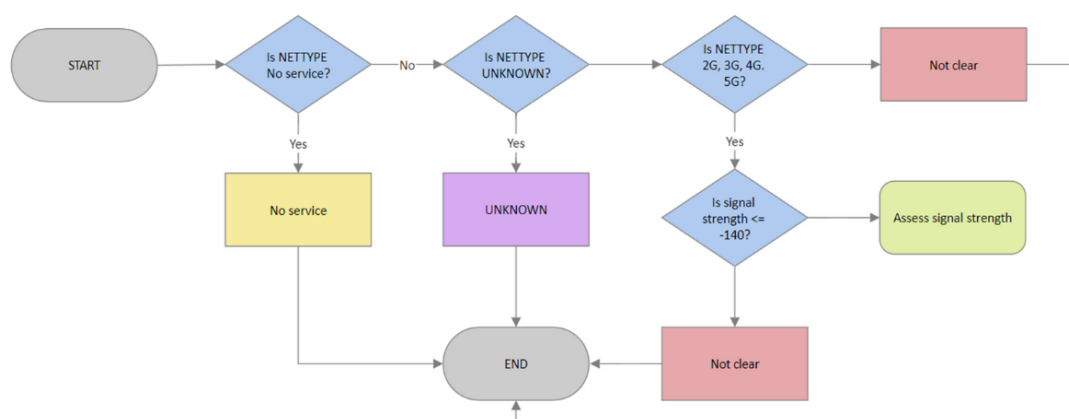
ตารางที่ 3.7 ตารางเงื่อนไขการใช้งาน Mobile Network Code ประเภท On-net

Operator	MNC Number	Status
AIS	01 , 03	โครงข่ายของตนเอง ( ON-NET )
DTAC	05 , 47	
TRUE-H	04 , 99	
NT-TOT	15 , 17	
NT-CAT	00 , 02	

ตารางที่ 3.8 ตารางเงื่อนไขการใช้งาน Mobile Network Code ประเภท Roaming

Operator	MNC Number	Status
AIS	15	โครงข่ายผู้ร่วมให้บริการ ( Roaming )
DTAC	99	
TRUE-H	00 , 18	
NT-TOT	02	
NT-CAT	15 , 04 , 99	

3.1.1.6 เงื่อนไขการตรวจสอบคุณภาพสัญญาณของผลการทดสอบอันประกอบไปด้วยพารามิเตอร์จำพวก Signal Strength และ Signal Quality โดยเกณฑ์การประเมินนี้อ้างอิงตามการดำเนินงานของสำนักกำกับดูแลกิจการโทรคมนาคม (ดท.) มีรูปแบบการทำงานดังแผนผังงานที่แสดงดังรูปที่ aa และเงื่อนไขเกณฑ์กำหนดแสดงได้ดังรูปที่ 3.12 และ 3.13 ตามลำดับ



รูปที่ 3.12 แผนผังการดำเนินงานวิเคราะห์ข้อมูลในส่วนของ Signal Strength และ Signal Quality

		2G	2G	3G	3G	4G	4G	5G	5G	5G SA 2600 MHz	5G SA 2600 MHz
Signal Strength	EXCELLENT	> -65	00	> -65	00	> -75	00	> -75	00	> -75	00
	VERY GOOD	> -75	≤ -65	> -75	≤ -65	> -85	≤ -75	> -85	≤ -75	> -85	≤ -75
	GOOD	> -85	≤ -75	> -85	≤ -75	> -95	≤ -85	> -95	≤ -85	> -95	≤ -85
	POOR	> -95	≤ -85	> -95	≤ -85	> -105	≤ -95	> -105	≤ -95	> -105	≤ -95
	WEAK	≥ -140	≤ -95	≥ -140	≤ -95	≥ -140	≤ -105	≥ -140	≤ -105	≥ -140	≤ -105
Signal Quality	EXCELLENT			> -6	00	> 25	00	> 25	00	> 25	00
	VERY GOOD			> -9	≤ -6	> 15	≤ 25	> 15	≤ 25	> 15	≤ 25
	GOOD			> -12	≤ -9	> 10	≤ 15	> 10	≤ 15	> 10	≤ 15
	POOR			> -14	≤ -12	> 0	≤ 10	> 0	≤ 10	> 0	≤ 10
	WEAK			= 00	≤ -14	= 00	≤ 0	= 00	≤ 0	= 00	≤ 0

รูปที่ 3.13 ข้อมูลในส่วนของ Signal Strength และ Signal Quality

3.1.1.7 นำค่า Latitude และ Longitude ของผลการทดสอบมาทำการวิเคราะห์ระยะห่างจากจุดอ้างอิงเนื่องจากอุปกรณ์ที่ใช้ในการทดสอบต้องวางอยู่ประจำที่สำนักงาน กสทช. ในแต่ละเขต/ภูมิภาค ดังนั้นจึงต้องมีการคัดเลือกผลการทดสอบที่มีค่า Latitude และ Longitude ไม่ตรงออกไปจากการทำรายงานค่าชี้วัด โดยทางผู้จัดทำได้เลือกใช้ Haversine equation ที่แสดงดัง

สมการที่ 3.1 ในการคำนวณหาระยะห่างของอุปกรณ์การทดสอบกับตำแหน่งอ้างอิงอันแสดงได้ดังตารางที่ 3.9 โดยเมื่อทำการใช้สมการที่ 3.1 แล้วพบว่ามียะห่างระหว่างจุดสองจุดน้อยกว่า 100 เมตร ให้แสดงผลลัพธ์เป็น “YES” หากระยะห่างมากกว่า 100 เมตรให้แสดงผลลัพธ์เป็น “NO” และหาก Latitude และ Longitude มีค่าเป็น 0 ทั้งหมดให้แสดงผลลัพธ์เป็น “NO GPS”

$$distance = 2R \arcsin \left( \sqrt{\sin^2 \left( \frac{\varphi_2 - \varphi_1}{2} \right) + \cos \varphi_1 \cdot \cos \varphi_2 \cdot \sin^2 \left( \frac{\lambda_2 - \lambda_1}{2} \right)} \right) \quad (3.1)$$

โดยกำหนดให้ R คือ รัศมีของโลกในการทดลองนี้เท่ากับ 6,371 กิโลเมตร

$\varphi_2, \varphi_1$  คือ ค่าละติจูดของจุดที่ 2 และ 1 ตามลำดับ

$\lambda_2, \lambda_1$  คือ ค่าลองจิจูดของจุดที่ 2 และ 1 ตามลำดับ

ตารางที่ 3.9 ตารางตำแหน่งละติจูดลองจิจูดอ้างอิงของสำนักงาน กสทช. ประจำ เขต/ภูมิภาค

หมายเลขกล่อง	สถานที่	ละติจูด	ลองจิจูด
NBTC-BOX01	กรุงเทพมหานคร	13.88537	100.57884
NBTC-BOX02	จังหวัดสมุทรปราการ	13.60213	100.59726
NBTC-BOX03	จังหวัดจันทบุรี	12.68109	102.20323
NBTC-BOX04	จังหวัดสุพรรณบุรี	14.45981	100.13090
NBTC-BOX05	จังหวัดปราจีนบุรี	14.05945	101.35821
NBTC-BOX06	จังหวัดพระนครศรีอยุธยา	14.33913	100.59762
NBTC-BOX07	จังหวัดราชบุรี	13.55077	99.81035
NBTC-BOX08	จังหวัดขอนแก่น	16.47918	102.80210
NBTC-BOX09	จังหวัดร้อยเอ็ด	16.05403	103.69435
NBTC-BOX10	จังหวัดอุบลราชธานี	15.27743	104.82552
NBTC-BOX11	จังหวัดนครราชสีมา	14.78416	102.04250
NBTC-BOX12	จังหวัดอุดรธานี	17.41506	102.79524
NBTC-BOX13	จังหวัดนครพนม	17.38494	104.79315
NBTC-BOX14	จังหวัดเชียงใหม่	18.77238	98.97542
NBTC-BOX15	จังหวัดลำปาง	18.30756	99.46983
NBTC-BOX16	จังหวัดลำพูน	18.55504	99.04079
NBTC-BOX17	จังหวัดพิษณุโลก	16.90281	100.27194
NBTC-BOX18	จังหวัดเชียงราย	19.92085	99.81401
NBTC-BOX19	จังหวัดนครสวรรค์	15.72519	100.04512

ตารางที่ 3.10 ตารางตำแหน่งละติจูดลองจิจูดอ้างอิงของสำนักงาน กสทช. ประจำ เขต/ภูมิภาค (ต่อ)

หมายเลขกล่อง	สถานที่	ละติจูด	ลองจิจูด
NBTC-BOX20	จังหวัดสงขลา	6.94137	100.38954
NBTC-BOX22	จังหวัดภูเก็ต	7.91861	98.34110
NBTC-BOX23	จังหวัดนครศรีธรรมราช	8.49320	99.92755
NBTC-BOX24	จังหวัดสุราษฎร์ธานี	9.11957	99.33781
NBTC-BOX25	จังหวัดชุมพร	10.42634	99.20051
NBTC-BOX28	จังหวัดยะลา	6.55586	101.28678
NBTC-BOX30	กรุงเทพมหานคร	13.78370	100.55073
NBTC-BOX31	กรุงเทพมหานคร	13.78370	100.55073
NBTC-BOX34	กรุงเทพมหานคร	13.78370	100.55073

### 3.1.2 การออกแบบระบบการแสดงผลข้อมูลเพื่อนำไปใช้สำหรับรายงานค่าชี้วัดคุณภาพการบริการโทรคมนาคมผ่านโครงข่ายโทรคมนาคมเคลื่อนที่

ในการแสดงผลข้อมูลการทดสอบทางคณะผู้จัดทำได้ทำการเลือกใช้อัปพลิเคชันที่ชื่อว่า Power BI Desktop ในการแสดงผลอันเนื่องมาจากแอปพลิเคชันนี้เป็นเครื่องมือในการนำเสนอข้อมูลแบบสำเร็จรูปที่สามารถเลือกฐานข้อมูลได้อย่างหลากหลาย รูปแบบในการแสดงผลที่สวยงามและมีความรวดเร็วในการประมวลผลอันเป็นที่นิยมในอุตสาหกรรมทางด้านข้อมูล ณ ปัจจุบันเป็นอย่างมาก

3.1.2.1 การออกแบบตารางสรุปผล อ้างอิงจากประกาศสำนักงานคณะกรรมการกิจการกระจายเสียง กิจการโทรทัศน์ และกิจการโทรคมนาคมแห่งชาติ เรื่อง แนวทางการวัดและการรายงานค่าชี้วัดคุณภาพบริการตามมาตรฐานของคุณภาพการให้บริการโทรคมนาคม เนื้อหารายละเอียดของผลการทดสอบประเภทเสียงเป็นไปดังตารางที่ 3.11 และผลการทดสอบประเภทข้อมูลเป็นไปดังตารางที่ 3.12

ตารางที่ 3.10 ตารางรายละเอียดพารามิเตอร์ของการทดสอบประเภทเสียง

Parameter	Description
Success Call	ผลการทดสอบที่มีสถานะเป็น “Complete”
Blocked Call	ผลการทดสอบที่มีสถานะเป็น “Block”
Dropped Call	ผลการทดสอบที่มีสถานะเป็น “Drop”
Bad Call	ผลการทดสอบที่มีสถานะเป็น “Bad”
Call Attempt	Success Call + Blocked Call + Dropped Call

ตารางที่ 3.11 รายละเอียดพารามิเตอร์ของการทดสอบประเภทข้อมูล

Parameter	Description
Number of Sample	จำนวนผลการทดสอบทั้งหมด
Complete Sample	จำนวนผลการทดสอบที่มีสถานะเป็น “PASS”
Fail Sample	จำนวนผลการทดสอบที่มีสถานะเป็น “FAIL”
Pass KPI Sample	จำนวนผลการทดสอบที่ผ่านมาตรฐาน
Complete Ratio	อัตราส่วนระหว่างผลการทดสอบที่มีสถานะเป็น “PASS” เทียบกับจำนวนผลการทดสอบทั้งหมด
Fail Ratio	อัตราส่วนระหว่างผลการทดสอบที่มีสถานะเป็น “FAIL” เทียบกับจำนวนผลการทดสอบทั้งหมด
Ratio subjected to specified	อัตราส่วนระหว่างผลการทดสอบที่ผ่านมาตรฐานเทียบกับจำนวนผลการทดสอบที่มีสถานะเป็น “PASS”
Average Data Throughput	ค่าเฉลี่ยค่าชี้วัดของผลการทดสอบที่มีสถานะเป็น “PASS”

## 3.2 เครื่องมือที่ใช้ในการทดลอง

ในโครงการนี้ มีอุปกรณ์และเครื่องมือที่ใช้ในการทดลอง ดังนี้

### 3.2.1 เครื่องมือทดสอบต้นทาง 1 (Originate I)

ยี่ห้อ : OnePlus

รุ่น : Nord N10

เทคโนโลยีการรับ/ส่งข้อมูล : 2G (EDGE/GPRS) , 3G (HSPA+) ,4G (LTE), 5G (NSA)

ระบบปฏิบัติการ: OxygenOS 10.5 based on Android 10.0

หน่วยประมวลผล Chip Set

- CPU : Qualcomm: Snapdragon 690 5G Octa Core
- ความเร็ว : 2.0 GHz
- GPU : Adreno 619L
- RAM : 6GB, ROM 128GB



รูปที่ 3.14 เครื่องมือทดสอบต้นทาง 1 (Originate I)

### 3.2.2 เครื่องมือทดสอบต้นทาง 2 (Originate II)

ยี่ห้อ : OnePlus

รุ่น : Nord CE 2

เทคโนโลยีการรับ/ส่งข้อมูล : 2G (EDGE/GPRS) , 3G (HSPA+) ,4G (LTE), 5G (NSA/SA)

ระบบปฏิบัติการ: OxygenOS 11 based on Android 11

หน่วยประมวลผล Chip Set

- CPU : Mediatek : Dimensity 900 Octa Core
- ความเร็ว : 2.4 GHz
- GPU : Mali-G68 MC4
- RAM 8GB, ROM 128GB : UFS 2.2, microSD สูงสุด 512 GB



รูปที่ 3.15 เครื่องมือทดสอบต้นทาง 2 (Originate II)

### 3.2.3 เครื่องมือทดสอบต้นทาง 3 (Originate III)

ยี่ห้อ : Xiaomi

รุ่น : POCO F4

เทคโนโลยีการรับ/ส่งข้อมูล : 2G (EDGE/GPRS) , 3G (HSPA+) , 4G (LTE), 5G (NSA/SA)

ระบบปฏิบัติการ: MIUI 13 for POCO based on Android 12

หน่วยประมวลผล Chip Set

- CPU : Qualcomm : Snapdragon 870 5G Octa Core
- ความเร็ว : 3.2 GHz
- GPU : Adreno 650
- RAM 8GB, ROM 128GB



รูปที่ 3.16 เครื่องมือทดสอบต้นทาง 3 (Originate III)

### 3.2.4 เครื่องมือทดสอบประเภทเสียงปลายทาง 1 (Voice Terminate I)

ยี่ห้อ : Samsung

รุ่น : Galaxy Ace Plus

เทคโนโลยีการรับ/ส่งข้อมูล : 2G (EDGE/GPRS) , 3G (HSPDPA) 7.2 Mbps

ระบบปฏิบัติการ: Android 2.3 Gingerbread

หน่วยประมวลผล Chip Set

- CPU : Qualcomm : Snapdragon S1
- ความเร็ว : 1.0 GHz
- GPU : Adreno 200
- RAM : 512MB





รูปที่ 3.17 เครื่องมือทดสอบประเภทเสียงปลายทาง 1 (Voice Terminate I)

### 3.2.5 เครื่องมือทดสอบประเภทเสียงปลายทาง 2 (Voice Terminate II)

ยี่ห้อ : Xiaomi

รุ่น : Redmi 9A

เทคโนโลยีการรับ/ส่งข้อมูล : 2G (EDGE/GPRS) , 3G (HSPA+) , 4G: (LTE)

ระบบปฏิบัติการ: MIUI 12 based on Android 10.0

หน่วยประมวลผล Chip Set

- CPU : Mediatek : Helio G25 Octa Core
- ความเร็ว : 2.0 GHz
- GPU : PowerVR GE8320
- RAM 2GB, ROM 32GB : UFS 0, microSD สูงสุด 512 GB



รูปที่ 3.18 เครื่องมือทดสอบประเภทเสียงปลายทาง 2 (Voice Terminate II)

### 3.2.6 เครื่องมือทดสอบประเภทเสียงปลายทาง 3 (Voice Terminate III)

- โครงข่ายโทรศัพท์ประจำที่ของบริษัท กสท. โทรคมนาคม จำนวน 30 คู่สาย



รูปที่ 3.19 เครื่องมือทดสอบประเภทเสียงปลายทาง 3 (Voice Terminate III)

### 3.2.7 เครื่องมือทดสอบประเภทข้อมูลปลายทาง 1 (Data Terminate I)

- เครื่องคอมพิวเตอร์แม่ข่าย จำนวน 2 เครื่อง
- ความเร็วในการเข้าถึง 250 Mbps
- สถานที่ติดตั้ง ณ ห้องปฏิบัติการคอมพิวเตอร์ สำนักงาน กสทช.



รูปที่ 3.20 เครื่องมือทดสอบประเภทข้อมูลปลายทาง 1 (Data Terminate I)

## 3.3 การจัดเก็บผลการทดลอง

### 3.3.1 การทดสอบการวิเคราะห์ข้อมูลคุณภาพการให้บริการประเภทข้อมูล

ทำการทดสอบโดยการนำข้อมูลดิบจากเซิร์ฟเวอร์มาทำการวิเคราะห์ผ่านระบบวิเคราะห์ข้อมูลที่ออกแบบไว้ในส่วนของผลการทดสอบประเภทข้อมูลว่าเป็นไปตามที่ออกแบบไว้หรือไม่

### 3.3.2 การทดสอบการวิเคราะห์ข้อมูลคุณภาพการให้บริการประเภทเสียง

ทำการทดสอบโดยการนำข้อมูลดิบจากเซิร์ฟเวอร์มาทำการวิเคราะห์ผ่านระบบวิเคราะห์ข้อมูลที่ออกแบบไว้ในส่วนของผลการทดสอบประเภทเสียงว่าเป็นไปตามที่ออกแบบไว้หรือไม่

### 3.3.3 การทดสอบการแสดงผลในส่วนของการนำเสนอผลคุณภาพการให้บริการ ประเภทข้อมูล

การทดสอบในส่วนของการนำเสนอผลของคุณภาพการให้บริการประเภทข้อมูลนั้น  
ถูกต้องตามนิยามอันอ้างอิงจากประกาศสำนักงาน กสทช. หรือไม่

### 3.3.4 การทดสอบการแสดงผลในส่วนของการนำเสนอผลคุณภาพการให้บริการ ประเภทเสียง

การทดสอบในส่วนของการนำเสนอผลของคุณภาพการให้บริการประเภทเสียงนั้น  
ถูกต้องตามนิยามอันอ้างอิงจากประกาศสำนักงาน กสทช. หรือไม่

### 3.3.5 การทดสอบการแสดงผลในส่วนของการนำเสนอประสิทธิภาพของคุณภาพ สัญญาณ

การทดสอบส่วนของการนำเสนอประสิทธิภาพของคุณภาพสัญญาณไปทำการแสดงผลใน  
รูปแบบของแผนที่โดยสามารถวัดผลได้ละเอียดถึงระดับจังหวัดว่าสามารถใช้งานได้หรือไม่

## บทที่ 4

### ผลการทดลอง

ผู้จัดทำได้ทำการเก็บผลการทำงานของระบบ โดยแบ่งการทดลองและจัดเก็บผลการทดลองเป็นส่วนๆ ดังต่อไปนี้

#### 4.1 การทดสอบการวิเคราะห์ข้อมูลคุณภาพการให้บริการประเภทข้อมูล

ผลการทดสอบในส่วนของการวิเคราะห์ข้อมูลเบื้องต้นตามที่ออกแบบเพื่อตรวจสอบว่าเป็นไปตามนั้นหรือไม่ทางคณะผู้จัดทำได้ใช้การเปรียบเทียบเอาท์พุตและอินพุตของผลการทดสอบพบว่าผลการวิเคราะห์เป็นไปตามแนวทางที่ได้ออกแบบไว้อย่างเรียบร้อย

##### 4.1.1 การตรวจสอบความถูกต้องสำหรับการเปลี่ยน Network Type ให้อยู่ในรูปแบบของ Technology Generation

ทำการตรวจสอบผลการทดสอบโดยการใช้การกรองโดยตรงผ่านข้อมูลดิบโดยทำการเลือกที่คอลัมน์ 'NETTYPE2' (Technology Generation Output) ให้ครบและทำการเช็คค่าในคอลัมน์ 'NETTYPE' ว่าเป็นไปตามเงื่อนไขหรือไม่ ยกตัวอย่างสำหรับกรณีของ 3G หากถูกต้องค่าในคอลัมน์ 'NETTYPE' จะต้องประกอบไปด้วย UMTS , HSPA , HSUPA , HSDPA , HSPAP เท่านั้น  
แสดงผลดังรูปที่ 4.1

	A	B	C	D	E	F	G	H	I	J	K
1	START_TIME	FINISH_TIME	IMEI	location	location2	operator1	OPERATOR	NETTYPE	NETTYPE2	MCCMNC	MCCMNC2
18	09/01/23 0:03	09/01/23 0:03	8.61384E+14	NBTC-BOX06	3G	my			3G	52000	on-net
20	09/01/23 0:03	09/01/23 0:03	8.6936E+14	NBTC-BOX28	3G	TOT			3G	52015	on-net
26	09/01/23 0:03	09/01/23 0:04	8.6936E+14	NBTC-BOX28	3G	TH GSM			3G	52003	on-net
30	09/01/23 0:03	09/01/23 0:03	8.61384E+14	NBTC-BOX02	3G	DTAC			3G	52005	on-net
33	09/01/23 0:03	09/01/23 0:03	8.61384E+14	NBTC-BOX12	3G	DTAC			3G	52005	on-net
36	09/01/23 0:03	09/01/23 0:03	8.61384E+14	NBTC-BOX06	3G	DTAC			3G	52005	on-net
38	09/01/23 0:03	09/01/23 0:03	8.6936E+14	NBTC-BOX28	3G	DTAC			3G	52005	on-net
39	09/01/23 0:03	09/01/23 0:03	8.6936E+14	NBTC-BOX28	3G	my			3G	52000	on-net
40	09/01/23 0:03	09/01/23 0:03	8.61384E+14	NBTC-BOX12	3G	my			3G	52000	on-net
45	09/01/23 0:03	09/01/23 0:03	8.61384E+14	NBTC-BOX06	3G	TOT			3G	52015	on-net
50	09/01/23 0:03	09/01/23 0:04	8.61384E+14	NBTC-BOX02	3G	TRUE-H			3G	52004	on-net
51	09/01/23 0:03	09/01/23 0:03	8.61384E+14	NBTC-BOX02	3G	TOT			3G	52015	on-net
56	09/01/23 0:03	09/01/23 0:03	8.61384E+14	NBTC-BOX20	4G	my			3G	52000	on-net
58	09/01/23 0:03	09/01/23 0:03	8.61384E+14	NBTC-BOX12	3G	TH GSM			3G	52015	roaming
65	09/01/23 0:03	09/01/23 0:03	8.61384E+14	NBTC-BOX18	3G	my			3G	52000	on-net
68	09/01/23 0:03	09/01/23 0:03	8.61384E+14	NBTC-BOX12	3G	TOT			3G	52015	on-net
69	09/01/23 0:03	09/01/23 0:03	8.61384E+14	NBTC-BOX02	3G	TH GSM			3G	52015	roaming
70	09/01/23 0:03	09/01/23 0:04	8.61384E+14	NBTC-BOX18	3G	TH GSM			3G	52003	on-net
74	09/01/23 0:03	09/01/23 0:03	8.61384E+14	NBTC-BOX02	3G	my			3G	52000	on-net
76	09/01/23 0:03	09/01/23 0:03	8.61384E+14	NBTC-BOX18	3G	DTAC			3G	52005	on-net
78	09/01/23 0:03	09/01/23 0:04	8.61384E+14	NBTC-BOX06	3G	TRUE-H			3G	52000	roaming
79	09/01/23 0:03	09/01/23 0:04	8.6936E+14	NBTC-BOX28	3G	TRUE-H			3G	52000	roaming
80	09/01/23 0:03	09/01/23 0:03	8.61384E+14	NBTC-BOX18	3G	TOT			3G	52002	roaming
81	09/01/23 0:03	09/01/23 0:04	8.61384E+14	NBTC-BOX12	3G	TRUE-H			3G	52004	on-net
82	09/01/23 0:03	09/01/23 0:05	8.61384E+14	NBTC-BOX18	3G	TRUE-H			3G	52004	on-net

รูปที่ 4.1 ผลการทดสอบ Technology Generation กรณีของ 3G

#### 4.1.2 การตรวจสอบความถูกต้องสำหรับการตรวจสอบโครงข่ายของผลการทดสอบ ว่าเป็นประเภทการใช้บริการแบบใด

โดยการทดสอบนี้จะเช็คข้อมูลในส่วนของคอลัมน์ ‘MCCMNC2’ (MNC Status) เพื่อเทียบกับคอลัมน์ ‘MCCMNC’ (MNC Number) โดยทำการกรองโดยตรงผ่านข้อมูลดิบเพื่อเช็คว่าเป็นไปตามเงื่อนไขหรือไม่ ยกตัวอย่างสำหรับกรณีการใช้บริการโครงข่ายตนเอง (On-net ) เครือข่าย B ดังนั้นค่าที่ได้จากคอลัมน์ MCCMNC จะต้องประกอบไปด้วย 52004 , 52099 และกรณีการใช้โครงข่ายของผู้ให้บริการ (Roaming) MCCMNC จะต้องประกอบไปด้วย 52000 , 52018 เท่านั้น แสดงได้ดังรูปที่ 4.2 สำหรับกรณี On-net และรูปที่ 4.3 สำหรับกรณี Roaming

	A	B	C	D	E	F	G	H	I	J	K
1	START_TIME	FINISH_TIME	IMEI	location	location2	operator1	OPERATOR	NETTYPE	NETTYPE2	MCCMNC	MCCMNC2
5	09/01/23 0:00	nan	8.63364E+14	NBTC-BOX34	5G	TRUE-H	TRUE-H	NR NSA	Sort Smallest to Largest		on-net
7	09/01/23 0:00	nan	8.6936E+14	NBTC-BOX31	2G	TRUE-H	TRUE-H	EDGE	Sort Largest to Smallest		on-net
13	09/01/23 0:01	09/01/23 0:01	8.63364E+14	NBTC-BOX34	5G	TRUE-H	TRUE-H	NR NSA	Sort by Color		on-net
15	09/01/23 0:01	09/01/23 0:01	8.6936E+14	NBTC-BOX31	2G	TRUE-H	TRUE-H	EDGE	Sheet View		on-net
23	09/01/23 0:03	09/01/23 0:03	8.61384E+14	NBTC-BOX01	4G	TRUE-H	TRUE-H	LTE	Clear Filter From "MCCMNC"		on-net
34	09/01/23 0:03	09/01/23 0:03	8.61384E+14	NBTC-BOX13	4G	TRUE-H	TRUE-H	LTE	Filter by Color		on-net
47	09/01/23 0:03	09/01/23 0:03	8.61384E+14	NBTC-BOX16	4G	TRUE-H	TRUE-H	LTE	Number Filters		on-net
50	09/01/23 0:03	09/01/23 0:04	8.61384E+14	NBTC-BOX02	3G	TRUE-H	TRUE-H	HSPAP	Search		on-net
59	09/01/23 0:03	09/01/23 0:05	8.61384E+14	NBTC-BOX23	4G	TRUE-H	TRUE-H	LTE	<input checked="" type="checkbox"/> (Select All)		on-net
60	09/01/23 0:03	09/01/23 0:03	8.61384E+14	NBTC-BOX08	4G	TRUE-H	TRUE-H	LTE	<input checked="" type="checkbox"/> 52004		on-net
66	09/01/23 0:03	09/01/23 0:03	8.61384E+14	NBTC-BOX05	4G	TRUE-H	TRUE-H	LTE	<input checked="" type="checkbox"/> 52099		on-net
67	09/01/23 0:03	09/01/23 0:03	8.61384E+14	NBTC-BOX14	4G	TRUE-H	TRUE-H	LTE	OK		on-net
81	09/01/23 0:03	09/01/23 0:04	8.61384E+14	NBTC-BOX12	3G	TRUE-H	TRUE-H	HSPAP	Cancel		on-net
82	09/01/23 0:03	09/01/23 0:05	8.61384E+14	NBTC-BOX18	3G	TRUE-H	TRUE-H	HSPAP			on-net
86	09/01/23 0:05	09/01/23 0:05	8.63364E+14	NBTC-BOX34	5G	TRUE-H	TRUE-H	LTE			on-net
92	09/01/23 0:10	09/01/23 0:10	8.61384E+14	NBTC-BOX25	3G	TRUE-H	TRUE-H	HSPAP			on-net
99	09/01/23 0:10	09/01/23 0:10	8.61384E+14	NBTC-BOX07	4G	TRUE-H	TRUE-H	LTE			on-net
100	09/01/23 0:10	09/01/23 0:10	8.61384E+14	NBTC-BOX10	3G	TRUE-H	TRUE-H	HSPAP			on-net
105	09/01/23 0:10	09/01/23 0:10	8.61384E+14	NBTC-BOX22	4G	TRUE-H	TRUE-H	LTE			on-net
110	09/01/23 0:10	09/01/23 0:10	8.61384E+14	NBTC-BOX17	4G	TRUE-H	TRUE-H	LTE			on-net
121	09/01/23 0:10	09/01/23 0:10	8.61384E+14	NBTC-BOX19	5G	TRUE-H	TRUE-H	LTE			on-net
124	09/01/23 0:10	09/01/23 0:10	8.61384E+14	NBTC-BOX09	4G	TRUE-H	TRUE-H	LTE			on-net
127	09/01/23 0:10	09/01/23 0:10	8.61384E+14	NBTC-BOX24	5G	TRUE-H	TRUE-H	LTE			on-net
147	09/01/23 0:10	09/01/23 0:10	8.61384E+14	NBTC-BOX04	5G	TRUE-H	TRUE-H	LTE		4G	52004 on-net

รูปที่ 4.2 การตรวจสอบเงื่อนไขการใช้โครงข่ายของตนเอง (On-net)

	A	B	C	D	E	F	G	H	I	J	K
1	START_TIME	FINISH_TIME	IMEI	location	location2	operator1	OPERATOR	NETTYPE	NETTYPE2	MCCMNC	MCCMNC2
57	09/01/23 0:03	09/01/23 0:03	8.61384E+14	NBTC-BOX20	4G	TRUE-H	TRUE-H	LTE	Sort Smallest to Largest		roaming
78	09/01/23 0:03	09/01/23 0:04	8.61384E+14	NBTC-BOX06	3G	TRUE-H	TRUE-H	HSPAP	Sort Largest to Smallest		roaming
79	09/01/23 0:03	09/01/23 0:04	8.6936E+14	NBTC-BOX28	3G	TRUE-H	TRUE-H	HSPAP	Sort by Color		roaming
89	09/01/23 0:10	09/01/23 0:10	8.61384E+14	NBTC-BOX11	5G	TRUE-H	TRUE-H	LTE	Sheet View		roaming
116	09/01/23 0:10	09/01/23 0:10	8.61384E+14	NBTC-BOX03	3G	TRUE-H	TRUE-H	HSPAP	Clear Filter From "MCCMNC"		roaming
131	09/01/23 0:10	09/01/23 0:10	8.61384E+14	NBTC-BOX15	3G	TRUE-H	TRUE-H	HSPAP	Filter by Color		roaming
153	09/01/23 0:14	nan	8.61384E+14	NBTC-BOX03	3G	TRUE-H	TRUE-H	HSPAP	Number Filters		roaming
178	09/01/23 0:14	nan	8.61384E+14	NBTC-BOX11	5G	TRUE-H	TRUE-H	LTE	Search		roaming
212	09/01/23 0:14	09/01/23 0:14	8.6936E+14	NBTC-BOX28	3G	TRUE-H	TRUE-H	HSPAP	<input checked="" type="checkbox"/> (Select All)		roaming
216	09/01/23 0:14	09/01/23 0:14	8.61384E+14	NBTC-BOX18	3G	TRUE-H	TRUE-H	HSPAP	<input checked="" type="checkbox"/> 52000		roaming
242	09/01/23 0:14	09/01/23 0:14	8.61384E+14	NBTC-BOX06	3G	TRUE-H	TRUE-H	HSPAP	<input checked="" type="checkbox"/> 52018		roaming
243	09/01/23 0:14	09/01/23 0:14	8.61384E+14	NBTC-BOX20	4G	TRUE-H	TRUE-H	LTE	OK		roaming
251	09/01/23 0:14	nan	8.61384E+14	NBTC-BOX15	3G	TRUE-H	TRUE-H	HSPAP	Cancel		roaming
290	09/01/23 0:18	09/01/23 0:18	8.61384E+14	NBTC-BOX11	5G	TRUE-H	TRUE-H	LTE			roaming
328	09/01/23 0:18	09/01/23 0:19	8.61384E+14	NBTC-BOX03	3G	TRUE-H	TRUE-H	HSPAP			roaming
334	09/01/23 0:18	09/01/23 0:19	8.61384E+14	NBTC-BOX15	3G	TRUE-H	TRUE-H	HSPAP			roaming
348	09/01/23 0:25	09/01/23 0:25	8.61384E+14	NBTC-BOX18	3G	TRUE-H	TRUE-H	HSPAP			roaming
363	09/01/23 0:25	09/01/23 0:25	8.61384E+14	NBTC-BOX06	3G	TRUE-H	TRUE-H	HSPAP			roaming
401	09/01/23 0:25	09/01/23 0:25	8.61384E+14	NBTC-BOX20	4G	TRUE-H	TRUE-H	LTE			roaming
402	09/01/23 0:25	09/01/23 0:25	8.6936E+14	NBTC-BOX28	3G	TRUE-H	TRUE-H	HSPAP			roaming
438	09/01/23 0:29	nan	8.6936E+14	NBTC-BOX28	3G	TRUE-H	TRUE-H	HSPAP			roaming
449	09/01/23 0:29	09/01/23 0:29	8.61384E+14	NBTC-BOX15	3G	TRUE-H	TRUE-H	HSPAP			roaming
460	09/01/23 0:29	09/01/23 0:29	8.61384E+14	NBTC-BOX11	5G	TRUE-H	TRUE-H	LTE			roaming

รูปที่ 4.3 การตรวจสอบเงื่อนไขการใช้โครงข่ายแบบร่วมให้บริการ (Roaming)

### 4.1.3 การตรวจสอบความถูกต้องสำหรับ Throughput ของผลการทดสอบ

สำหรับการเช็คค่า Throughput ของข้อมูลในระบบที่ออกแบบไว้อ้างอิงมาจากประกาศของทางสำนักงาน กสทช. เรื่อง คุณภาพการให้บริการโทรคมนาคมผ่านโครงข่ายโทรคมนาคมเคลื่อนที่ ทางผู้จัดต้องทำการเช็คเอาต์พุตของขั้นตอนในคอลัมน์ชื่อ 'TP2' ที่มีค่าเป็น YES จัดว่าเป็นผลการทดสอบที่ผ่านมาตรฐานดังนั้นจึงทำการกรองข้อมูลดิบโดยตรงแล้วเลือกการทดสอบประเภท FTP Download และเทียบกับมาตรฐาน ยกตัวอย่างกรณีของ 4G ค่า FTP Download ที่ได้จะต้องไม่น้อยกว่า 2.5 Mbps ซึ่งแสดงผลได้ดังรูปที่ 4.4 ค่า Throughput ของผลการทดสอบแสดงที่คอลัมน์ TP มีหน่วยเป็น Kbps ดังนั้นผลการทดสอบที่กรองตามเงื่อนไขแล้วจะต้องมากกว่า 2500.00 ขึ้นไป

	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y
	MCCMNC	MCCMNC2	LAC	CID	RSSIDBM	RSCP	ECIO	RSRP	SINR	nr_ssrsp	nr_sssinr	NETTYPE2	TYPE	STATUS	TP	TP2
19	52047	on-net	31073	308507	-60			-97	7			4G				yes
21	52004	abnormal	1011	28950	-54			-81	5			4G				yes
22	52005	on-net	612	164465	-77			-109	0			4G				yes
23	52004	on-net	1033	155970	-59			-91	13			4G				yes
24	52015	on-net	23227	350059	-62			-88	18			4G				yes
25	52015	on-net	10376	410068	-62			-96	4			4G				yes
27	52015	on-net	12432	430894	-64			-94	8			4G				yes
28	52005	on-net	658	171030	-62			-90	8			4G				yes
29	52015	on-net	6576	590748	-46			-77	6			4G				yes
31	52005	on-net	538	105015	-49			-73	16			4G				yes
32	52004	abnormal	3701	977	-48			-76	22			4G				yes
34	52004	on-net	2708	197	-67			-98	-1			4G				yes
35	52004	abnormal	3505	111875	-44			-73	3			4G				yes
37	52003	on-net	20496	300016	-72			-105	4			4G				yes
41	52003	on-net	13064	151006	-65			-92	5			4G				yes
42	52005	on-net	194	54014	-55			-84	6			4G				yes
43	52017	on-net	36163	164148	-83			-115	4			4G				yes
44	52004	abnormal	8208	35022	-61			-94	4			4G				yes
47	52004	on-net	8208	19232	-56			-87	3			4G				yes
48	52003	on-net	947	630067	-62			-98	-7			4G				yes
49	52015	on-net	3003	630022	-61			-95	1			4G				yes
52	52003	on-net	10248	410068	-60			-85	16			4G				yes
53	52004	abnormal	3608	61286	-84			-115	-2			4G				yes
54	52015	on-net	12936	159435	-58			-84	24			4G				yes

รูปที่ 4.4 ตัวอย่างผลการทดสอบที่ผ่านมาตรฐาน 4G แบบ FTP Download

## 4.2 การทดสอบการวิเคราะห์ข้อมูลคุณภาพการให้บริการประเภทเสียง

ผลการทดสอบสำหรับการวิเคราะห์ข้อมูลของการทดสอบประเภทเสียงนั้นทางคณะผู้จัดทำได้ทำการเช็คจากการนำข้อมูลดิบมากรองโดยตรงผ่าน Microsoft Excel เพื่อทำการตรวจสอบผลลัพธ์ที่ได้พบว่าผลลัพธ์ที่ได้เป็นไปตามแนวทางที่ได้ออกแบบไว้อย่างเรียบร้อย

### 4.3 การทดสอบการแสดงผลในส่วนของการสรุปผลคุณภาพการให้บริการประเภทข้อมูล

ตารางสรุปผลในส่วนของการทดสอบคุณภาพการให้บริการประเภทข้อมูลนั้น ตามที่ได้ออกแบบไว้ผลลัพธ์พบว่าค่าในตารางถูกต้องเมื่อเทียบกับการกรองข้อมูลดิบโดยตรงผ่านโปรแกรม Microsoft Excel และตัวค่าในตารางสรุปผลของโครงการในครั้งนี้ถูกนำไปใช้งานจริงโดยทีมวิศวกรของทาง สำนักกำกับดูแลกิจการโทรคมนาคม (ดท.) สำหรับการทำรายงานค่าชี้วัดคุณภาพการให้บริการโทรคมนาคมผ่านโครงข่ายโทรคมนาคมเคลื่อนที่ ประจำไตรมาสที่ 3 ดังนั้นถือว่าผลลัพธ์เป็นไปตามแนวทางที่ได้ออกแบบไว้ หน้าตาของตารางสรุปผลเป็นไปดังรูปที่ 4.5

QoS Parameter	A	B	C	D	E
Ping					
Round Trip Time (< KPI) Sample	118,979	118,746	118,046	106,474	113,535
Average Round Trip Time (ms)	72	68	94	99	93
RTT success ratio subjected to specified criteria ( 80% )	99.06	98.84	98.62	92.34	98.91
Completed Sample	120,112	120,140	119,694	115,302	114,781
Number of Sample	120,478	120,257	120,116	115,749	114,996
Fail Sample	366	117	422	447	215
Fail Ratio ( % )	0.30	0.10	0.35	0.39	0.19
Completed RTT Success Ratio ( % )	99.70	99.90	99.65	99.61	99.81
HTTP Download					
Complete HTTP Download Sample	116,595	119,419	117,871	114,113	113,663
HTTP Download Success Ratio ( % )	97.68	99.79	98.99	99.09	99.26
HTTP Download Fail Sample	2,773	247	1,205	1,049	852
HTTP Download Fail Ratio ( % )	2.32	0.21	1.01	0.91	0.74
Page loading Time ( < KPI ) Sample	116,447	119,404	117,831	114,083	113,655
Average Page loading Time ( Seconds )	8.124	7.752	9.836	6.286	6.872
Number of Sample	119,368	119,666	119,076	115,162	114,535
Average Data Throughput ( Mbps )	10.796	10.042	9.428	11.309	7.804
FTP Upload					
Completed FTP Upload Sample	118,991	119,601	117,645	114,025	113,855
FTP Upload Success Ratio ( % )	99.34	99.83	98.55	98.91	99.32
FTP Upload Cut-Off Sample	786	203	1,735	1,259	783
FTP Upload Cut-Off Ratio ( % )	0.66	0.17	1.45	1.09	0.68
Data Throughput Upload ( > KPI ) Sample	113,522	118,700	115,142	108,772	112,154
Average Data Throughput ( Mbps )	6.781	5.574	5.806	4.764	6.198
FTP Upload ratio subjected to specified ( 80% )	95.40	99.25	97.87	95.39	98.49
Number of Sample	119,777	119,804	119,380	115,284	114,638
FTP Download					
Completed FTP Download Sample	115,365	118,113	115,799	112,984	111,593
FTP Download Success Ratio ( % )	97.64	99.70	98.14	99.04	98.31
FTP Download Cut-Off Sample	2,790	357	2,189	1,100	1,921
FTP Download Cut-Off Ratio ( % )	2.36	0.30	1.86	0.96	1.69
Data Throughput Download ( > KPI ) Sample	113,423	114,677	111,598	110,870	107,300
Average Data Throughput ( Mbps )	30.654	18.503	21.769	18.039	10.758
FTP Download ratio subjected to specified ( 80% )	98.32	97.09	96.37	98.13	96.15
Number of Sample	118,155	118,470	117,988	114,084	113,514

**DATE**  
7/1/2023 10/1/2023

**LOCATION2**  
All

**CHECK SCRIPT**  
All

**GPS**  
YES = Distance <= 100 m. NO = Distance > 100 m. NO GPS = Latitude and Longitude = 0  
All

**NETTYPE**  
All

**MCCMNC**  
yes = own network. yes2 = co-op netw...  
All

Clear all selection

**LOCATION**  
☐ Select all  
☐ NBTC-BOX01  
☐ NBTC-BOX02  
☐ NBTC-BOX03  
☐ NBTC-BOX04  
☐ NBTC-BOX05  
☐ NBTC-BOX06  
☐ NBTC-BOX07  
☐ NBTC-BOX08  
☐ NBTC-BOX09  
☐ NBTC-BOX10  
☐ NBTC-BOX11  
☐ NBTC-BOX12  
☐ NBTC-BOX13  
☐ NBTC-BOX14  
☐ NBTC-BOX15  
☐ NBTC-BOX16  
☐ NBTC-BOX17  
☐ NBTC-BOX18  
☐ NBTC-BOX19  
☐ NBTC-BOX20  
☐ NBTC-BOX21  
☐ NBTC-BOX22

รูปที่ 4.5 ตารางสรุปผลการทดสอบประเภทข้อมูล

#### 4.4 การทดสอบการแสดงผลในส่วนของการตรวจสอบคุณภาพการให้บริการประเภทเสียง

ตารางสรุปผลในส่วนของการทดสอบคุณภาพการให้บริการประเภทเสียงนั้น ตามที่ได้ ออกแบบไว้ผลลัพธ์พบว่าค่าในตารางถูกต้องเมื่อเทียบกับการกรองข้อมูลดิบโดยตรงผ่านโปรแกรม Microsoft Excel และตัวค่าในตารางสรุปผลของโครงการในครั้งนี้ถูกนำไปใช้งานจริงโดยทีมวิศวกร ของทาง สำนักกำกับดูแลกิจการโทรคมนาคม (ดท.) สำหรับการดำเนินงานค่าชี้วัดคุณภาพการ ให้บริการโทรคมนาคมผ่านโครงข่ายโทรคมนาคมเคลื่อนที่ ประจำไตรมาสที่ 3 ดังนั้นถือว่าผลลัพธ์ เป็นไปตามแนวทางที่ได้ออกแบบไว้ หน้าตาของตารางสรุปผลเป็นไปดังรูปที่ 4.6

QoS Parameter	A	B	C	D	E
Voice					
Call attempts (Sample)	11,669	14,753	10,085	14,746	14,429
Complete call (Sample)	11,663	14,750	10,085	14,744	14,425
Successful call ratio ( % )	99.95%	99.98%	100.00%	99.99%	99.98%
Blocked call (Sample)	6	3		2	3
Dropped call (Sample)					1
Call drop rate ( % )					0.01%
Call setup success (Sample)	11,663	14,750	10,085	14,744	14,426
Complete ratio ( % )	99.95%	99.98%	100.00%	99.99%	99.97%
Call bad (Sample)	46	118	64	76	218
Bad call rate ( % )	0.39%	0.80%	0.63%	0.52%	1.51%

TP

- Bad Attempt
- Block
- Complete
- Drop

DATE

7/1/2023 9/30/2023

GPS LAT/LNG

Multiple selecti...

TECHNOLOGY

All

Box on/off-net

Off-net

NBTC-BOX

All

MCC/MNC

on-net

TIME ZONE

Multiple selections

Clear all selection

รูปที่ 4.6 ตารางสรุปผลการทดสอบประเภทเสียง



#### 4.5 การทดสอบการแสดงผลในส่วนแผนที่แสดงประสิทธิภาพของคุณภาพสัญญาณ

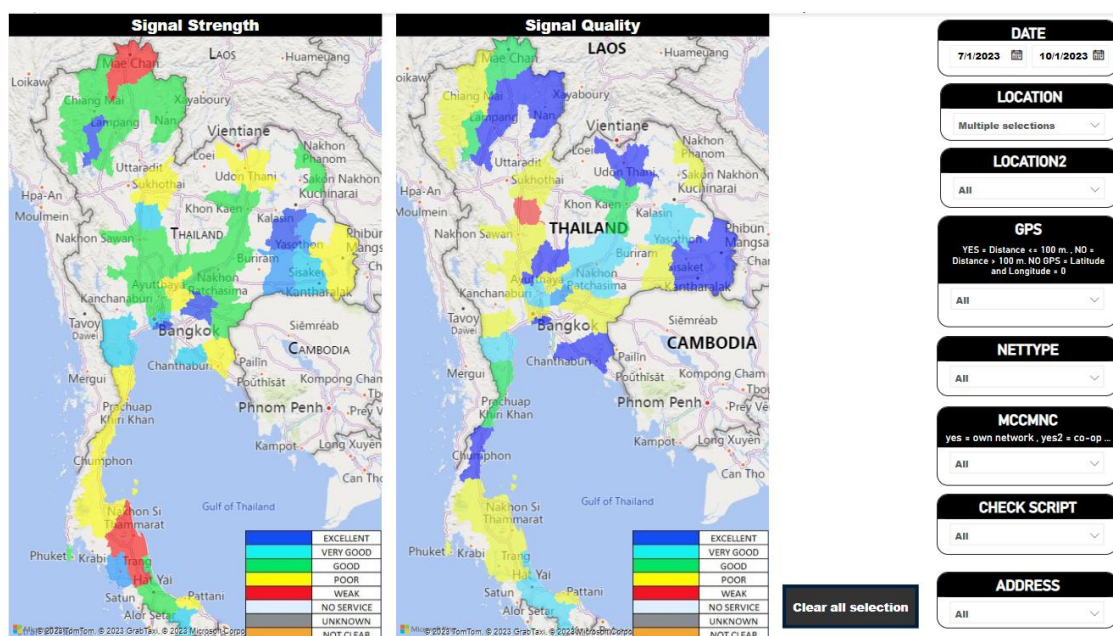
จากการนำเอาประสิทธิภาพของคุณภาพสัญญาณไปแสดงในลักษณะของแผนที่โดยทางผู้จัดทำได้เลือกใช้ฟังก์ชัน Filled Map ภายในโปรแกรม Power BI ในการแสดงผลและทำการเชื่อมต่อข้อมูลโดยใช้คอลัมน์ Address ในการแสดงผลและสร้างตัวแปรที่ใช้คำนวณหาอัตราส่วนระหว่างค่าคุณภาพสัญญาณต่างๆ และทำการเลือกค่ามาแสดงบนแผนที่โดยใช้ค่าฐานนิยมของข้อมูลประจำพื้นที่นั้นๆ และทำการแบ่งสีของแต่ละประสิทธิภาพออกเป็น 5 ระดับดังที่แสดงในรูปที่ 4.7 และเงื่อนไขการพิจารณาเป็นไปตามตารางที่ 4.1 และ 4.2 ตามลำดับ

ตารางที่ 4.1 เงื่อนไขตัวแปรสำหรับ Signal Strength และ Signal Quality ของแต่ละเทคโนโลยี

Technology	Parameter	
	Signal Strength	Signal Quality
2G	RSSIDBM	-
3G	RSCP	EC/IO
4G	RSRP	SINR
5G	NR_SSRSP	NR_SSSINR

ตารางที่ 4.2 ช่วงระยะของค่าพารามิเตอร์ในการบ่งบอกประสิทธิภาพของคุณภาพสัญญาณ

Type	Performance	Range			
		2G	3G	4G	5G
Signal Strength	EXCELLENT	(-65,0)		(-75,0)	
	VERY GOOD	(-75,-65)		(-85,-75)	
	GOOD	(-85,-75)		(-95,-85)	
	POOR	(-95,-85)		(-105,-95)	
	WEAK	(-140,-95)		(-140,-105)	
		2G	3G	4G	5G
Signal Quality	EXCELLENT	-	(-6,∞)	(25,∞)	
	VERY GOOD	-	(-9,-6)	(15,25)	
	GOOD	-	(-12,-9)	(10,15)	
	POOR	-	(-14,-12)	(0,10)	
	WEAK	-	(-∞,-14)	(-∞,0)	



รูปที่ 4.7 Filled Map ที่บ่งบอกถึงประสิทธิภาพของคุณภาพสัญญาณในแต่ละพื้นที่

## บทที่ 5

### สรุปผลและข้อเสนอแนะ

#### 5.1 สรุปผล

ปริญญานิพนธ์นี้นำเสนอการออกแบบและพัฒนานวัตกรรมในการวิเคราะห์ข้อมูลเพื่อนำไปใช้ในการตรวจสอบคุณภาพการให้บริการโทรศัพท์เคลื่อนที่ (Quality of Service) ประกอบไปด้วย 3 ส่วน ได้แก่ ส่วนที่ 1 ทำหน้าที่เป็นอินพุตของระบบอันได้แก่ชุดข้อมูลของผลการทดสอบที่ถูกแยกตามบริการประกอบไปด้วย HTTP Download , FTP Upload , FTP Download , Ping และ Voice ซึ่งมีลักษณะเป็นไฟล์ที่ถูกเก็บข้อมูลไว้แบบ CSV (Comma Separated Value) โดยการที่จะได้ไฟล์ข้อมูลเหล่านี้ออกมาจะต้องทำการนำออกมาจากซอฟต์แวร์ Syberiz ที่ทางสำนักงาน กสทช. ใช้ในการจัดเก็บข้อมูลเท่านั้น ส่วนที่ 2 นำเอาไฟล์ข้อมูลทุกบริการไปผ่านการประมวลผลและวิเคราะห์ทางคอมพิวเตอร์ จัดเรียงและทำการรวมข้อมูลผลการทดสอบของทุกบริการเข้าด้วยกันโดยใช้ Python จัดการข้อมูลทั้งหมด ส่วนที่ 3 นำไฟล์ผลลัพธ์ที่ได้มาสร้าง Dashboard และทำการแสดงผลบน Power BI โดยมีลักษณะการแสดงผลที่เป็นไปตามความต้องการของเจ้าหน้าที่และพนักงานของสำนักงาน กสทช. ซึ่งมีผลลัพธ์ที่สำเร็จตามขอบเขตของงานอย่างสมบูรณ์ โดยการเกิดขึ้นของโครงการนี้ผู้จัดทำคาดหวังว่าจะส่งผลอันเป็นประโยชน์กับทางสำนักงาน กสทช. ไม่มากนักน้อยเนื่องจากการเลือกใช้ Python Programming ในพาร์ทของการเตรียมข้อมูลส่งผลให้การคำนวณและวิเคราะห์ผลต่างๆ เป็นไปได้ดีกว่าระบบรูปแบบเดิมของทางสำนักงานซึ่งคือการใช้ VBA Macro ในการวิเคราะห์และในส่วนของการแสดงผลข้อมูลเพื่อนำไปสรุปผลนั้นโครงการนี้เลือกใช้ Power BI Application อันเป็นซอฟต์แวร์ที่เหมาะสมกับงานประเภท Data Visualization มากกว่าการแสดงผลใน Microsoft Excel ที่ทางสำนักงานใช้ในเดิมที และทางผู้จัดทำคิดว่าโครงการในครั้งนี้เป็นการเตรียมความพร้อมสำหรับงานในส่วนของอุตสาหกรรมข้อมูล ณ ปัจจุบันอีกด้วย อีกทั้งโครงการในครั้งนี่ยังสามารถนำไปประยุกต์ใช้กับการ Monitor ผลการทดสอบประเภทอื่นๆ ซึ่งทำให้เห็นถึงข้อมูลเชิงลึกแบบรวดเร็วอันนำมาซึ่งการแก้ไขปัญหาที่เกิดขึ้นได้อย่างมีประสิทธิภาพ

#### 5.2 ปัญหาและข้อเสนอแนะ

ข้อเสนอแนะในการพัฒนาโครงการ มีดังนี้

1. พัฒนาการจัดเก็บชุดข้อมูลของผลการทดสอบและข้อมูลอื่น ๆ ให้อยู่ในรูปแบบที่เป็นสากลและเป็นระบบมากขึ้น
2. ปรับเปลี่ยนการแสดงผลบน Power BI ให้มีความสวยงามมากขึ้น

## บรรณานุกรม

- [1] คณะกรรมการกิจการกระจายเสียง กิจการโทรทัศน์ และกิจการโทรคมนาคมแห่งชาติ, **ประกาศคณะกรรมการกิจการกระจายเสียง กิจการโทรทัศน์ และกิจการโทรคมนาคมแห่งชาติ เรื่อง มาตรฐานของคุณภาพการให้บริการโทรคมนาคม**, เล่ม ๑๓๙ ตอนพิเศษ ๓๐๒. [ออนไลน์]. เข้าถึงได้จาก <https://shorturl.at/luDES>. (วันที่ค้นข้อมูล: 1 พฤศจิกายน 2556).
- [2] คณะกรรมการกิจการกระจายเสียง กิจการโทรทัศน์ และกิจการโทรคมนาคมแห่งชาติ, **ประกาศคณะกรรมการกิจการกระจายเสียง กิจการโทรทัศน์ และกิจการโทรคมนาคมแห่งชาติ เรื่อง แนวทางการวัดและการรายงานค่าชี้วัดคุณภาพบริการตามมาตรฐานของคุณภาพการให้บริการโทรคมนาคม**, เล่ม ๑๓๙ ตอนพิเศษ ๓๐๗. [ออนไลน์]. เข้าถึงได้จาก <https://shorturl.at/czJ23>. (วันที่ค้นข้อมูล: 1 พฤศจิกายน 2556).
- [3] European Telecommunications Standards Institute Technical Specification (ETSI TS), **Speech and multimedia Transmission Quality (STQ); QoS aspects for popular services in mobile networks; Part 2: Definition of Quality of Service parameters and their computation, ETSI TS 102 250-2 V2.7.1 (2019-11)**. [ออนไลน์]. เข้าถึงได้จาก [https://www.etsi.org/deliver/etsi\\_ts/102200\\_102299/10225002/02.07.01\\_60/ts\\_10225002v020701p.pdf](https://www.etsi.org/deliver/etsi_ts/102200_102299/10225002/02.07.01_60/ts_10225002v020701p.pdf) (วันที่ค้นข้อมูล: 1 พฤศจิกายน 2556).
- [4] TELECOMMUNICATION STANDARDIZATION SECTOR OF ITU, **Usage of cause and location in the Digital Subscriber Signalling System No. 1 and the Signalling System No. 7 ISDN user part, Q.850 (10/2018)**. [ออนไลน์]. เข้าถึงได้จาก <https://www.itu.int/rec/T-REC-Q.850-201810-I/en> (วันที่ค้นข้อมูล: 1 พฤศจิกายน 2556).

ภาคผนวก ก

ชุดรหัสคำสั่งการวิเคราะห์ข้อมูลดิบที่นำเข้าจากโปรแกรม Syberiz ด้วยภาษา  
Python ในโปรแกรม Visual Studio Code

```

import pandas as pd
import time
import math
import time

from geopy.geocoders import Nominatim
from geopy.exc import GeocoderTimedOut

df = pd.read_csv('qqq.csv')
df = df.dropna(subset=['location'])
pattern = r'NBTC-BOX0[1-9]|NBTC-BOX1[0-9]|NBTC-BOX2[0-5]|NBTC-BOX3[0,1,4]|NBTC-BOX28'
filtered_data = df[df['location'].str.contains(pattern)]
df = filtered_data
df = df[df['location'].notna()]
start = time.time()
mapping_data = pd.read_excel('mapping_Book.xlsx')
mapping_df = pd.read_excel('map_Voice.xlsx')
additional_columns = [f'x{i}' for i in range(1, 12)]
df[additional_columns] = None
source_columns = mapping_df['source_column']
target_columns = mapping_df['target_column']
df = df[[list(target_columns) + list(set(df.columns) - set(target_columns) - set(source_columns))]]
column_mapping = mapping_df.set_index('source_column')['target_column'].to_dict()

df['START_TIME'] = pd.to_datetime(df['START_TIME'])
df.insert(df.columns.get_loc('START_TIME') + 1, 'TIME_ZONE', [""] * len(df))
time_ranges = [
    (pd.to_datetime('10:00').time(), pd.to_datetime('13:00').time(), 'DAY'),
    (pd.to_datetime('16:00').time(), pd.to_datetime('19:00').time(), 'NIGHT')

```

```

    ]
    for start_range, end_range, value in time_ranges:
        check = (df['START_TIME'].dt.time >= start_range) &
(df['START_TIME'].dt.time < end_range)
        df.loc[check, 'TIME_ZONE'] = value
        df.loc[df['TIME_ZONE'] == '', 'TIME_ZONE'] = 'OTHERS'
        df.rename(columns={'band': 'BAND', 'TIME_ZONE': 'FINISH_TIME'},
inplace=True)

df.insert(df.columns.get_loc('location') + 1, 'location2', [""] * len(df))
df.insert(df.columns.get_loc('location') + 4, 'BOX_TYPE', [""] * len(df))
df.insert(df.columns.get_loc('NETTYPE') + 1, 'NETTYPE2', [""] * len(df))
df.insert(df.columns.get_loc('MCCMNC') + 1, 'MCCMNC2', df['MCCMNC'])
df.insert(df.columns.get_loc('geom') + 1, 'voc_status', '0')
for index, row in mapping_data.iterrows():
    df.loc[(df['location'] == row['location']), 'BOX_TYPE'] = row['BNUMBER3']
    df.loc[(df['BOX_TYPE'] == row['box_type']) & (df['operator1'] ==
row['operator1(voc)']) & (df['BNUMBER'] == row['bnumber']), 'location2'] =
row['voc_onoff']
    df.loc[(df['operator1'] == row['operator1(CP)']) & (df['BNUMBER'] ==
row['bnumber(CP)']), 'location2'] = row['voc(CP)']
    df.loc[(df['BNUMBER'] == row['fixed_line']), 'location2'] = row['voc(CP)']
    df.loc[df['NETTYPE'] == row['NETTYPE'], 'NETTYPE2'] = row['NETTYPE2']
    df.loc[(df['operator1'] == row['operator1']) & (df['MCCMNC'] ==
row['MCCMNC']), 'MCCMNC2'] = row['MCCMNC2']
    df.loc[(df['BOX_TYPE'] == row['box_type']) & (df['operator1'] ==
row['operator1(voc)']) & (df['BNUMBER'] == row['bnumber']), 'voc_status'] =
row['voc_status']
    for index, row in mapping_data.iterrows():
        if not pd.isna(row['nr_band']) :

```

```

df.loc[(df['NETTYPE'] == row['NETTYPE']) & (df['nr_band'] ==
row['nr_band']) , 'NETTYPE2'] = row['NETTYPE2']
df.loc[df['location2'] == "", 'location2'] = 'ANY'

geolocator = Nominatim(user_agent="reverse_geocoder")
cache = {}
def reverse_geocode_with_cache(lat, lng):
    if (lat, lng) in cache:
        return cache[(lat, lng)]
    try:
        location = geolocator.reverse((lat, lng), exactly_one=True)
        address = location.address if location else None
        district = None
        if address:
            address_components = address.split(',')
            for component in address_components:
                if 'จังหวัด' in component.lower() or 'กรุงเทพมหานคร' in
component.lower():
                    district = component.strip()
                    break
            cache[(lat, lng)] = district
        return district
    except (AttributeError, GeocoderTimedOut):
        return None

df['CONNECTED_TIME'] = pd.to_datetime(df['CONNECTED_TIME'])
df['ALERTING_TIME'] = pd.to_datetime(df['ALERTING_TIME'])
df['DIALING_TIME'] = pd.to_datetime(df['DIALING_TIME'])
df['DISCONNECTED_TIME'] = pd.to_datetime(df['DISCONNECTED_TIME'])
df.insert(df.columns.get_loc('DISCONNECTED_TIME') + 1,
'DIALING_DURATION', None)

```



```

df.insert(df.columns.get_loc('DISCONNECTED_TIME') + 2,
'ALERTING_DURATION', None)
df.insert(df.columns.get_loc('DISCONNECTED_TIME') + 3,
'CONNECTED_DURATION', None)

```

```

def one_time(time1,time2):
    if time1 == " or time2 == ":
        return "
    else:
        time = time2 - time1
        hours, remainder = divmod(time.total_seconds(), 3600)
        minutes, seconds = divmod(remainder, 60)
        return f"{int(hours):02}:{int(minutes):02}:{int(seconds):02}"

```

```

for index, row in df.iterrows():
    disconnected_time = row['DISCONNECTED_TIME']
    connected_time = row['CONNECTED_TIME']
    alerting_time = row['ALERTING_TIME']
    dialing_time = row['DIALING_TIME']

    if pd.isna(dialing_time) or all(time == " for time in [disconnected_time,
connected_time, alerting_time]):
        df.at[index, 'DIALING_DURATION'] = None
        df.at[index, 'ALERTING_DURATION'] = None
        df.at[index, 'CONNECTED_DURATION'] = None

    elif pd.notna(dialing_time):
        if pd.isna(disconnected_time):
            if pd.isna(connected_time) and pd.notna(alerting_time):
                df.at[index, 'DIALING_DURATION'] = one_time(dialing_time,
alerting_time)

```

```

elif pd.notna(connected_time) and pd.isna(alerting_time):
    df.at[index, 'DIALING_DURATION'] = one_time(dialing_time,
connected_time)

elif pd.notna(connected_time) and pd.notna(alerting_time):
    df.at[index, 'DIALING_DURATION'] = one_time(dialing_time,
alerting_time)

    df.at[index, 'ALERTING_DURATION'] = one_time(alerting_time,
connected_time)

elif pd.isna(connected_time):
    if pd.notna(disconnected_time) and pd.isna(alerting_time):
        df.at[index, 'DIALING_DURATION'] = one_time(dialing_time,
disconnected_time)

    elif pd.notna(disconnected_time) and pd.notna(alerting_time):
        df.at[index, 'DIALING_DURATION'] = one_time(dialing_time,
alerting_time)

        df.at[index, 'ALERTING_DURATION'] = one_time(alerting_time,
disconnected_time)

elif pd.isna(alerting_time):
    if pd.notna(disconnected_time) and pd.notna(connected_time):
        df.at[index, 'DIALING_DURATION'] = one_time(dialing_time,
connected_time)

        df.at[index, 'CONNECTED_DURATION'] =
one_time(connected_time, disconnected_time)

    elif all(pd.notna(time) for time in [disconnected_time,
connected_time, alerting_time, dialing_time]):
        df.at[index, 'DIALING_DURATION'] =
one_time(dialing_time, alerting_time)

```

```

df.at[index, 'ALERTING_DURATION'] =
one_time(alerting_time,connected_time)
df.at[index, 'CONNECTED_DURATION'] =
one_time(connected_time,disconnected_time)

```

```

df.insert(df.columns.get_loc('LNG') + 1, 'LATLNG2', [""] * len(df))
for index, row in df.iterrows():
    mapping_row = mapping_data[mapping_data['location'] ==
row['location']]
    if not mapping_row.empty:
        mapping_row = mapping_row.iloc[0]
        if row['LAT'] != 0 and row['LNG'] != 0:
            radius = 6371
            dlat = math.radians(mapping_row['LAT2'] - row['LAT'])
            dlon = math.radians(mapping_row['LNG2'] - row['LNG'])
            a = math.sin(dlat / 2) ** 2 + math.cos(math.radians(row['LAT'])) *
math.cos(math.radians(mapping_row['LAT2'])) * math.sin(dlon / 2) ** 2
            c = 2 * math.atan2(math.sqrt(a), math.sqrt(1 - a))
            distance = radius * c * 1000
            if row['LAT'] == 0 and row['LNG'] == 0:
                distance = 0
            if distance >= 100:
                output = 'NO'
            if distance < 100:
                output = 'YES'
            if distance == 0:
                output = 'NO GPS'
        else:
            output = 'TEAM DRIVE TEST'
        df.at[index, 'LATLNG2'] = output
for index, row in mapping_data.iterrows():

```

```

df.loc[(df['LATLNG2'] == row['LATLNG2']) & (df['location'] ==
row['location'])], 'Address'] = row['Address']

for index, row in mapping_data.iterrows():
    df.loc[(df['Address'].isna() ) & (df['location'] == row['location'])], 'Address']
= row['Address']

print(df['Address'])

```

```

def check_duration(duration):
    duration = duration.split(':')
    hours, minutes, seconds = map(int, duration)
    duration_seconds = hours * 3600 + minutes * 60 + seconds
    return duration_seconds

```

```

def check_signal(row):
    if rssi <= -125:
        return ('Drop', 'RSSI <= -125') if status == 'PASS' else ('Block', 'RSSI <= -
125')

    elif rssi > -125:
        return ('Bad', 'Unknown | PASS') if status == 'PASS' else ('Bad', 'Unknown
| FAIL')

    else:
        return ('Bad', 'check rssi | PASS') if status == 'PASS' else ('Bad', 'check
rssi | FAIL')

```

```

def check_nettype(row):
    if nettype2 == 'No service':
        return ('Drop', 'No service') if status == 'PASS' else ('Block', 'No service')
    elif nettype2 == 'UNKNOWN':
        if pd.isna(mccmnc):
            return ('Drop', 'No Nettype') if status == 'PASS' else ('Block', 'No
Nettype')

```

```

elif pd.isna(mccmnc):
    return ('Bad', 'Unknown Nettype | PASS') if status == 'PASS' else
('Bad', 'Unknown Nettype | FAIL')
elif nettype2 in ['2G', '3G', '4G', '5G', '5G (2600 MHz)', '5G (SA 2600 MHz)':
    a, b = check_signal(row)
    return a, b
else:
    return ('Bad', 'check nettype | PASS') if status == 'PASS' else ('Bad',
'check nettype | FAIL')

```

```

def check_cause_a_num(row):
    cause_a_num = row['CAUSE_A_NUM']
    if cause_a_num not in [-1, 1, 8, 16, 17, 31]:
        return ('Drop', 'Network') if status == 'PASS' else ('Block', 'Network')
    elif cause_a_num in [-1, 1, 8, 16, 17, 31]:
        a, b = check_nettype(row)
        return a, b
    else:
        return ('Bad', 'check network | PASS') if status == 'PASS' else ('Bad',
'check network | FAIL')

```

```

def check_callsetup(row):
    if call_setup == 0:
        return ('Bad', 'Call setup = 0 | PASS') if status == 'PASS' else ('Bad', 'Call
setup = 0 | FAIL')
    elif call_setup != 0:
        a, b = check_cause_a_num(row)
        return a, b
    else:
        return ('Bad', 'check call setup | PASS') if status == 'PASS' else ('Bad',
'check call setup | FAIL')

```

```

def check_timeout(row):
    alerting_duration = row['ALERTING_DURATION']
    if check_duration(dialing_duration) >= 20:
        return ('Bad', 'Alerting Timeout | PASS') if status == 'PASS' else ('Bad',
'Alerting Timeout | FAIL')
    elif check_duration(dialing_duration) < 20:
        if pd.isna(alerting_duration) or check_duration(alerting_duration) < 20:
            a, b = check_callsetup(row)
            return a, b
        elif check_duration(alerting_duration) >= 20:
            return ('Bad', 'Connected Timeout | PASS') if status == 'PASS' else
('Bad', 'Connected Timeout | FAIL')
        elif check_duration(alerting_duration) < 0:
            return ('Bad', 'Negative Alert Time | PASS') if status == 'PASS' else
('Bad', 'Negative Alert Time | FAIL')
    else:
        return ('Bad', 'check duration | PASS') if status == 'PASS' else ('Bad',
'check duration | FAIL')

def check_success_call(connected_duration, time):
    if check_duration(connected_duration) > time+5:
        return 'Bad', 'Too much Time | PASS'
    elif time <= check_duration(connected_duration) <= (time + 5):
        return 'Complete', '-'
    elif check_duration(connected_duration) < time:
        return check_timeout(row)
    else:
        return 'Bad', 'Unknown | PASS'

for index, row in df.iterrows():

```

```

status = row['STATUS']
disconnected_time = row['DISCONNECTED_TIME']
connected_duration = row['CONNECTED_DURATION']
alerting_duration = row['ALERTING_DURATION']
dialing_duration = row['DIALING_DURATION']
cause_a_num = row['CAUSE_A_NUM']
nettype2 = row['NETTYPE2']
mccmnc = row['MCCMNC']
rssi = row['RSSIDBM']
call_setup = row['CALLSETUP']
location = row['location']

if status == 'PASS':
    if pd.isna(connected_duration):
        df.at[index, 'CAUSE_A_TXT'] = 'Bad'
        df.at[index, 'CAUSE_B_TXT'] = 'No Connect | PASS'
    elif pd.notna(connected_duration) and pd.notna(disconnected_time):
        if location in ['NBTC-BOX26', 'NBTC-BOX29', 'NBTC-BOX32' , 'NBTC-BOX33']:
            df.at[index, 'CAUSE_A_TXT'], df.at[index, 'CAUSE_B_TXT'] =
check_success_call(connected_duration, 90)
        else:
            df.at[index, 'CAUSE_A_TXT'], df.at[index, 'CAUSE_B_TXT'] =
check_success_call(connected_duration, 50)
        else:
            df.at[index, 'CAUSE_A_TXT'], df.at[index, 'CAUSE_B_TXT'] = ('Bad',
'Unknown | PASS')

elif status == 'FAIL':
    if pd.isna(dialing_duration):

```

```

        df.at[index, 'CAUSE_A_TXT'], df.at[index, 'CAUSE_B_TXT'] = ('No
Time', 'No Dialing Duration | FAIL')
    elif pd.notna(dialing_duration):
        if check_duration(dialing_duration) < 0 :
            df.at[index, 'CAUSE_A_TXT'], df.at[index, 'CAUSE_B_TXT'] = ('Bad',
'Negative Dial Time | FAIL')
        else:
            df.at[index, 'CAUSE_A_TXT'], df.at[index, 'CAUSE_B_TXT'] =
check_timeout(row)
        else:
            df.at[index, 'CAUSE_A_TXT'], df.at[index, 'CAUSE_B_TXT'] = ('Bad',
'Unknown | FAIL')

```

```

df['DIALING_TIME'] = df['DIALING_TIME'].dt.strftime('%H:%M:%S')
df['ALERTING_TIME'] = df['ALERTING_TIME'].dt.strftime('%H:%M:%S')
df['CONNECTED_TIME'] = df['CONNECTED_TIME'].dt.strftime('%H:%M:%S')
df['DISCONNECTED_TIME'] =
df['DISCONNECTED_TIME'].dt.strftime('%H:%M:%S')
df.rename(columns={'CAUSE_A_TXT':
'TP','BNUMBER':'FILE_PATH','BOX_TYPE':'FILE_PATH_STATUS','CAUSE_B_TXT':'TP2'},
inplace=True)

```

```

df.insert(df.columns.get_loc('nr_band') + 1, 'Signal_Strength', [""] * len(df))
df.loc[(df['NETTYPE2'] == '2G') & ((df['RSSIDBM'].isna()) | (df['RSSIDBM'] == 0)),
'Signal_Strength'] = 'not clear'
df.loc[(df['NETTYPE2'] == '3G') & ((df['RSCP'].isna()) | (df['RSCP'] ==
0)), 'Signal_Strength'] = 'not clear'
df.loc[(df['NETTYPE2'] == '4G') & ((df['RSRP'].isna()) | (df['RSRP'] ==
0)), 'Signal_Strength'] = 'not clear'
df.loc[(df['NETTYPE2'].str.contains('5G')) & ((df['nr_ssrsp'].isna()) |
(df['nr_ssrsp'] == 0)), 'Signal_Strength'] = 'not clear'

```



```

df.loc[df['NETTYPE2'] == 'No service', 'Signal_Strength'] = 'No service'
df.loc[df['NETTYPE2'] == 'UNKNOWN', 'Signal_Strength'] = 'UNKNOWN'
df.loc[df['Signal_Strength'] == '', 'Signal_Strength'] = 'not clear'

for index, row in mapping_data.iterrows():
    if row['NETTYPE2(SS)'] == '2G':
        df.loc[(df['NETTYPE2'] == row['NETTYPE2(SS)']) & (df['RSSIDBM'] >=
row['SS_legend2'] ) & (df['RSSIDBM'] < row['SS_legend1']) , 'Signal_Strength'] =
row['SS_level']
        elif row['NETTYPE2(SS)'] == '3G' :
            df.loc[(df['NETTYPE2'] == row['NETTYPE2(SS)']) & (df['RSCP'] >=
row['SS_legend2'] ) & (df['RSCP'] < row['SS_legend1']) , 'Signal_Strength'] = row['SS_level']
            elif row['NETTYPE2(SS)'] == '4G' :
                df.loc[(df['NETTYPE2'] == row['NETTYPE2(SS)']) & (df['RSRP'] >=
row['SS_legend2'] ) & (df['RSRP'] < row['SS_legend1']) , 'Signal_Strength'] = row['SS_level']
            else :
                df.loc[(df['NETTYPE2'] == row['NETTYPE2(SS)']) & (df['nr_ssrsp'] >=
row['SS_legend2'] ) & (df['nr_ssrsp'] < row['SS_legend1']) , 'Signal_Strength'] =
row['SS_level']

df.insert(df.columns.get_loc('nr_band') + 2 , 'Signal_Quality', [''] * len(df))
df.loc[(df['NETTYPE2'] == '3G') & ((df['ECIO'].isna()) | (df['ECIO'] ==
0)), 'Signal_Quality'] = 'not clear'
df.loc[(df['NETTYPE2'] == '4G') & ((df['SINR'].isna()) | (df['SINR'] ==
0)), 'Signal_Quality'] = 'not clear'
df.loc[(df['NETTYPE2'].str.contains('5G')) & ((df['nr_sssinr'].isna()) |
(df['nr_sssinr'] == 0)), 'Signal_Quality'] = 'not clear'
df.loc[df['NETTYPE2'] == 'No service', 'Signal_Quality'] = 'No service'
df.loc[df['NETTYPE2'] == 'UNKNOWN', 'Signal_Quality'] = 'UNKNOWN'
df.loc[df['Signal_Quality'] == '', 'Signal_Quality'] = 'not clear'

```

```

for index, row in mapping_data.iterrows():
    if row['NETTYPE2(SS)'] == '3G' :
        df.loc[(df['NETTYPE2'] == row['NETTYPE2(SS)']) & (df['ECIO'] >=
row['SQ_legend2']) & (df['ECIO'] < row['SQ_legend1']) , 'Signal_Quality'] = row['SS_level']
    elif row['NETTYPE2(SS)'] == '4G' :
        df.loc[(df['NETTYPE2'] == row['NETTYPE2(SS)']) & (df['SINR'] >=
row['SQ_legend2']) & (df['SINR'] < row['SQ_legend1']) , 'Signal_Quality'] = row['SS_level']
    else :
        df.loc[(df['NETTYPE2'] == row['NETTYPE2(SS)']) & (df['nr_sssinr'] >=
row['SQ_legend2']) & (df['nr_sssinr'] < row['SQ_legend1']) , 'Signal_Quality'] =
row['SS_level']

df.insert(df.columns.get_loc('x5') + 1 , 'x13', [""] * len(df))
df['x2'] = df['location']
df['x3'] = df['location2']
df['x4'] = df['operator1']
df['x5'] = df['TP']
df['x13'] = df['TP2']

columns_to_drop = ['location', 'location2' , 'operator1' , 'TP', 'TP2']
df.drop(columns=columns_to_drop, inplace=True, errors='ignore')
df.rename(columns={'x2': 'location', 'x3': 'location2', 'x4': 'operator1', 'x5': 'TP'
, 'x6': 'ERROR2' , 'x7': 'ERROR3', 'x13': 'TP2'}, inplace=True)
columns_to_delete = ['x1', 'x8', 'x9', 'x10', 'x11']
df.drop(columns=columns_to_delete, inplace=True)

df.drop('LNG2', axis=1, inplace=True, errors='ignore')
df.drop('LAT2', axis=1, inplace=True, errors='ignore')
df.to_csv('QQQ.csv', index=False , encoding = 'utf-8-sig')

```

ภาคผนวก ข

ชุดรหัสคำสั่งการคำนวณและการแสดงผลข้อมูลที่ได้จาก Visual Studio Code ใน  
โปรแกรม Power BI

### ส่วนของการคำนวณการบริการประเภทข้อมูล

AWN =

VAR NoS\_Ping = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] = "Ping",'Database'[operator1] = "TH GSM")

VAR NoS\_HTTP = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] = "HTTP Download",'Database'[operator1] = "TH GSM")

VAR NoS\_FTPUL = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] = "FTP Upload",'Database'[operator1] = "TH GSM")

VAR NoS\_FTPDL = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] = "FTP Download",'Database'[operator1] = "TH GSM")

VAR PASS\_ping = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] = "Ping",'Database'[operator1] = "TH GSM",'Database'[STATUS] = "PASS")

VAR PASS\_http = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] = "HTTP Download",'Database'[operator1] = "TH GSM",'Database'[STATUS] = "PASS")

VAR PASS\_ftpul = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] = "FTP Upload",'Database'[operator1] = "TH GSM",'Database'[STATUS] = "PASS")

VAR PASS\_ftpdl = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] = "FTP Download",'Database'[operator1] = "TH GSM",'Database'[STATUS] = "PASS")

VAR FAIL\_ping = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] = "Ping",'Database'[operator1] = "TH GSM",'Database'[STATUS] = "FAIL")

VAR FAIL\_http = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] = "HTTP Download",'Database'[operator1] = "TH GSM",'Database'[STATUS] = "FAIL")

VAR FAIL\_ftpul = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] = "FTP Upload",'Database'[operator1] = "TH GSM",'Database'[STATUS] = "FAIL")

VAR FAIL\_ftpdl = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] = "FTP Download",'Database'[operator1] = "TH GSM",'Database'[STATUS] = "FAIL")

VAR TP\_PASS\_ping = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] = "Ping",'Database'[operator1] = "TH GSM",'Database'[STATUS] = "PASS",'Database'[TP2] = "yes")

```

VAR TP_PASS_http = CALCULATE(COUNTROWS('Database'),'Database'[TYPE]
= "HTTP Download",'Database'[operator1] = "TH GSM",'Database'[STATUS] =
"PASS",'Database'[TP2] = "yes")

```

```

VAR TP_PASS_ftpul = CALCULATE(COUNTROWS('Database'),'Database'[TYPE]
= "FTP Upload",'Database'[operator1] = "TH GSM",'Database'[STATUS] =
"PASS",'Database'[TP2] = "yes")

```

```

VAR TP_PASS_ftpdl = CALCULATE(COUNTROWS('Database'),'Database'[TYPE]
= "FTP Download",'Database'[operator1] = "TH GSM",'Database'[STATUS] =
"PASS",'Database'[TP2] = "yes")

```

```

VAR Avg_TP_Ping =
CALCULATE(AVERAGE('Database'[TP_CAL]),'Database'[TYPE]
= "Ping",'Database'[operator1] = "TH GSM",'Database'[STATUS] = "PASS")

```

```

VAR Avg_TP_HTTP =
CALCULATE(AVERAGE('Database'[USED_TIME]),'Database'[TYPE]
= "HTTP Download",'Database'[operator1] = "TH GSM",'Database'[STATUS] = "PASS")

```

```

VAR Avg_TP_ftpul =
CALCULATE(AVERAGE('Database'[TP_CAL]),'Database'[TYPE]
= "FTP Upload",'Database'[operator1] = "TH GSM",'Database'[STATUS] = "PASS")

```

```

VAR Avg_TP_ftpdl =
CALCULATE(AVERAGE('Database'[TP_CAL]),'Database'[TYPE]
= "FTP Download",'Database'[operator1] = "TH GSM",'Database'[STATUS] = "PASS")

```

```

VAR Avg_TP_web =
CALCULATE(AVERAGE('Database'[TP_CAL]),'Database'[TYPE]
= "HTTP Download",'Database'[operator1] = "TH GSM",'Database'[STATUS] = "PASS")

```

```

RETURN

```

```

SWITCH(

```

```

    SELECTEDVALUE('QoS_Parameter'[Index]),

```

```

    6, FORMAT(NoS_Ping, "###0"),

```

```

    17, FORMAT(NoS_HTTP, "###0"),

```

```

    27, FORMAT(NoS_FTPUL, "###0"),

```

```

    36, FORMAT(NoS_FTPDL, "###0"),

```

```

5, FORMAT(PASS_ping, "###0"),
11, FORMAT(PASS_http, "###0"),
20, FORMAT(PASS_ftpul, "###0"),
29, FORMAT(PASS_ftpdL, "###0"),
13, FORMAT(FAIL_http, "###0"),
22, FORMAT(FAIL_ftpul, "###0"),
31, FORMAT(FAIL_ftpdL, "###0"),
2, FORMAT(TP_PASS_ping, "###0"),
15, FORMAT(TP_PASS_http, "###0"),
24, FORMAT(TP_PASS_ftpul, "###0"),
33, FORMAT(TP_PASS_ftpdL, "###0"),
12,          IF(ISBLANK(NoS_HTTP), "N/A", FORMAT(100
DIVIDE(PASS_http, NoS_HTTP, "##0.00")),
21,          IF(ISBLANK(NoS_FTPUL), "N/A", FORMAT(100
DIVIDE(PASS_ftpul, NoS_FTPUL, "##0.00")),
30,          IF(ISBLANK(NoS_FTPDL), "N/A", FORMAT(100*
DIVIDE(PASS_ftpdL, NoS_FTPDL, "##0.00")),
14,          IF(ISBLANK(NoS_HTTP), "N/A", FORMAT(100*
DIVIDE(FAIL_http, NoS_HTTP, "##0.00")),
23,          IF(ISBLANK(NoS_FTPUL), "N/A", FORMAT(100*
DIVIDE(FAIL_ftpul, NoS_FTPUL, "##0.00")),
32,          IF(ISBLANK(NoS_FTPDL), "N/A", FORMAT(100*
DIVIDE(FAIL_ftpdL, NoS_FTPDL, "##0.00")),
4,          IF(ISBLANK(PASS_ping), "N/A", FORMAT(100*
DIVIDE(TP_PASS_ping, PASS_ping, "##0.00")),
26,          IF(ISBLANK(PASS_ftpul), "N/A", FORMAT(100*
DIVIDE(TP_PASS_ftpul, PASS_ftpul, "##0.00")),
35,          IF(ISBLANK(PASS_ftpdL), "N/A", FORMAT(100*
DIVIDE(TP_PASS_ftpdL, PASS_ftpdL, "##0.00")),
3, FORMAT(Avg_TP_Ping, "###0"),
16, FORMAT(Avg_TP_HTTP / 1000, "###0.000"),

```

```

25, FORMAT(Avg_TP_ftpul / 1000,"#,##0.000"),
34, FORMAT(Avg_TP_ftpdl / 1000 ,"#,##0.000"),
7, FORMAT(FAIL_ping,"#,##0"),
8,          IF(ISBLANK(NoS_Ping),"N/A",FORMAT(100          *
DIVIDE(FAIL_ping,NoS_Ping),"##0.00")),
9,          IF(ISBLANK(NoS_Ping),"N/A",FORMAT(100          *
DIVIDE(PASS_ping,NoS_Ping),"##0.00")),
18, FORMAT(Avg_TP_web / 1000 ,"#,##0.000"),
BLANK()
)

```

DTN =

```

VAR NoS_Ping = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"Ping",'Database'[operator1] = "DTAC")

```

```

VAR NoS_HTTP = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"HTTP Download",'Database'[operator1] = "DTAC")

```

```

VAR NoS_FTPUL = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"FTP Upload",'Database'[operator1] = "DTAC")

```

```

VAR NoS_FTPDL = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"FTP Download",'Database'[operator1] = "DTAC")

```

```

VAR PASS_ping = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"Ping",'Database'[operator1] = "DTAC",'Database'[STATUS] = "PASS")

```

```

VAR PASS_http = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"HTTP Download",'Database'[operator1] = "DTAC",'Database'[STATUS] = "PASS")

```

```

VAR PASS_ftpul = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"FTP Upload",'Database'[operator1] = "DTAC",'Database'[STATUS] = "PASS")

```

```

VAR PASS_ftpdl = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"FTP Download",'Database'[operator1] = "DTAC",'Database'[STATUS] = "PASS")

```

```

VAR FAIL_ping = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"Ping",'Database'[operator1] = "DTAC",'Database'[STATUS] = "FAIL")

```

```
VAR FAIL_http = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"HTTP Download",'Database'[operator1] = "DTAC",'Database'[STATUS] = "FAIL")
```

```
VAR FAIL_ftpul = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"FTP Upload",'Database'[operator1] = "DTAC",'Database'[STATUS] = "FAIL")
```

```
VAR FAIL_ftpdl = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"FTP Download",'Database'[operator1] = "DTAC",'Database'[STATUS] = "FAIL")
```

```
VAR TP_PASS_ping = CALCULATE(COUNTROWS('Database'),'Database'[TYPE]
= "Ping",'Database'[operator1] = "DTAC",'Database'[STATUS] = "PASS",'Database'[TP2] =
"yes")
```

```
VAR TP_PASS_http = CALCULATE(COUNTROWS('Database'),'Database'[TYPE]
= "HTTP Download",'Database'[operator1] = "DTAC",'Database'[STATUS] =
"PASS",'Database'[TP2] = "yes")
```

```
VAR TP_PASS_ftpul = CALCULATE(COUNTROWS('Database'),'Database'[TYPE]
= "FTP Upload",'Database'[operator1] = "DTAC",'Database'[STATUS] =
"PASS",'Database'[TP2] = "yes")
```

```
VAR TP_PASS_ftpdl = CALCULATE(COUNTROWS('Database'),'Database'[TYPE]
= "FTP Download",'Database'[operator1] = "DTAC",'Database'[STATUS] =
"PASS",'Database'[TP2] = "yes")
```

```
VAR Avg_TP_Ping =
CALCULATE(AVERAGE('Database'[TP_CAL]),'Database'[TYPE]
= "Ping",'Database'[operator1] = "DTAC",'Database'[STATUS] = "PASS")
```

```
VAR Avg_TP_HTTP =
CALCULATE(AVERAGE('Database'[USED_TIME]),'Database'[TYPE] =
"HTTP Download",'Database'[operator1] = "DTAC",'Database'[STATUS] = "PASS")
```

```
VAR Avg_TP_ftpul =
CALCULATE(AVERAGE('Database'[TP_CAL]),'Database'[TYPE] =
"FTP Upload",'Database'[operator1] = "DTAC",'Database'[STATUS] = "PASS")
```

```
VAR Avg_TP_ftpdl =
CALCULATE(AVERAGE('Database'[TP_CAL]),'Database'[TYPE] =
"FTP Download",'Database'[operator1] = "DTAC",'Database'[STATUS] = "PASS")
```



```

VAR                                Avg_TP_web                                =
CALCULATE(AVERAGE('Database'[TP_CAL]),'Database'[TYPE]                    =      "HTTP
Download",'Database'[operator1] = "DTAC",'Database'[STATUS] = "PASS")
RETURN
SWITCH(
    SELECTEDVALUE('QoS_Parameter'[Index]),
    6, FORMAT(NoS_Ping, "#,##0"),
    17, FORMAT(NoS_HTTP, "#,##0"),
    27, FORMAT(NoS_FTPUL, "#,##0"),
    36, FORMAT(NoS_FTPDL, "#,##0"),
    5, FORMAT(PASS_ping, "#,##0"),
    11, FORMAT(PASS_http, "#,##0"),
    20, FORMAT(PASS_ftpul, "#,##0"),
    29, FORMAT(PASS_ftpdl, "#,##0"),
    13, FORMAT(FAIL_http, "#,##0"),
    22, FORMAT(FAIL_ftpul, "#,##0"),
    31, FORMAT(FAIL_ftpdl, "#,##0"),
    2, FORMAT(TP_PASS_ping,"#,##0"),
    15, FORMAT(TP_PASS_http, "#,##0"),
    24, FORMAT(TP_PASS_ftpul,"#,##0"),
    33, FORMAT(TP_PASS_ftpdl,"#,##0"),
    12,                                IF(ISBLANK(NoS_HTTP),"N/A",FORMAT(100      *
DIVIDE(PASS_http,NoS_HTTP),"##0.00")),
    21,                                IF(ISBLANK(NoS_FTPUL),"N/A",FORMAT(100      *
DIVIDE(PASS_ftpul,NoS_FTPUL),"##0.00")),
    30,                                IF(ISBLANK(NoS_FTPDL),"N/A",FORMAT(100*
DIVIDE(PASS_ftpdl,NoS_FTPDL),"##0.00")),
    14,                                IF(ISBLANK(NoS_HTTP),"N/A",FORMAT(100*
DIVIDE(FAIL_http,NoS_HTTP),"##0.00")),
    23,                                IF(ISBLANK(NoS_FTPUL),"N/A",FORMAT(100*
DIVIDE(FAIL_ftpul,NoS_FTPUL),"##0.00")),

```

```

32, IF(ISBLANK(NoS_FTPDL),"N/A",FORMAT(100*
DIVIDE(FAIL_ftpdL,NoS_FTPDL),"##0.00")),
4, IF(ISBLANK(PASS_ping),"N/A",FORMAT(100*
DIVIDE(TP_PASS_ping,PASS_ping),"##0.00")),
26, IF(ISBLANK(PASS_ftpul),"N/A",FORMAT(100*
DIVIDE(TP_PASS_ftpul,PASS_ftpul),"##0.00")),
35, IF(ISBLANK(PASS_ftpdL),"N/A",FORMAT(100*
DIVIDE(TP_PASS_ftpdL,PASS_ftpdL),"##0.00")),
3, FORMAT(Avg_TP_Ping,"##0"),
16, FORMAT(Avg_TP_HTTP / 1000,"#,##0.000"),
25, FORMAT(Avg_TP_ftpul / 1000,"#,##0.000"),
34, FORMAT(Avg_TP_ftpdL / 1000,"#,##0.000"),
7, FORMAT(FAIL_ping,"#,##0"),
8, IF(ISBLANK(NoS_Ping),"N/A",FORMAT(100 *
DIVIDE(FAIL_ping,NoS_Ping),"##0.00")),
9, IF(ISBLANK(NoS_Ping),"N/A",FORMAT(100 *
DIVIDE(PASS_ping,NoS_Ping),"##0.00")),
18, FORMAT(Avg_TP_web / 1000,"#,##0.000"),
BLANK()
)

```

NT-CAT =

```

VAR NoS_Ping = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"Ping",'Database'[operator1] = "my")

```

```

VAR NoS_HTTP = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"HTTP Download",'Database'[operator1] = "my")

```

```

VAR NoS_FTPUL = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"FTP Upload",'Database'[operator1] = "my")

```

```

VAR NoS_FTPDL = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"FTP Download",'Database'[operator1] = "my")

```

```

VAR PASS_ping = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"Ping",'Database'[operator1] = "my",'Database'[STATUS] = "PASS")

VAR PASS_http = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"HTTP Download",'Database'[operator1] = "my",'Database'[STATUS] = "PASS")

VAR PASS_ftpul = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"FTP Upload",'Database'[operator1] = "my",'Database'[STATUS] = "PASS")

VAR PASS_ftpdl = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"FTP Download",'Database'[operator1] = "my",'Database'[STATUS] = "PASS")

VAR FAIL_ping = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"Ping",'Database'[operator1] = "my",'Database'[STATUS] = "FAIL")

VAR FAIL_http = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"HTTP Download",'Database'[operator1] = "my",'Database'[STATUS] = "FAIL")

VAR FAIL_ftpul = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"FTP Upload",'Database'[operator1] = "my",'Database'[STATUS] = "FAIL")

VAR FAIL_ftpdl = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"FTP Download",'Database'[operator1] = "my",'Database'[STATUS] = "FAIL")

VAR TP_PASS_ping = CALCULATE(COUNTROWS('Database'),'Database'[TYPE]
= "Ping",'Database'[operator1] = "my",'Database'[STATUS] = "PASS",'Database'[TP2] =
"yes")

VAR TP_PASS_http = CALCULATE(COUNTROWS('Database'),'Database'[TYPE]
= "HTTP Download",'Database'[operator1] = "my",'Database'[STATUS] =
"PASS",'Database'[TP2] = "yes")

VAR TP_PASS_ftpul = CALCULATE(COUNTROWS('Database'),'Database'[TYPE]
= "FTP Upload",'Database'[operator1] = "my",'Database'[STATUS] =
"PASS",'Database'[TP2] = "yes")

VAR TP_PASS_ftpdl = CALCULATE(COUNTROWS('Database'),'Database'[TYPE]
= "FTP Download",'Database'[operator1] = "my",'Database'[STATUS] =
"PASS",'Database'[TP2] = "yes")

VAR Avg_TP_Ping =
CALCULATE(AVERAGE('Database'[TP_CAL]),'Database'[TYPE]
= "Ping",'Database'[operator1] = "my",'Database'[STATUS] = "PASS")

```

```

VAR                                Avg_TP_HTTP                                =
CALCULATE(AVERAGE('Database'[USED_TIME]),'Database'[TYPE]                =      "HTTP
Download",'Database'[operator1] = "my",'Database'[STATUS] = "PASS")

VAR                                Avg_TP_ftpul                                =
CALCULATE(AVERAGE('Database'[TP_CAL]),'Database'[TYPE]                    =      "FTP
Upload",'Database'[operator1] = "my",'Database'[STATUS] = "PASS")

VAR                                Avg_TP_ftpdl                                =
CALCULATE(AVERAGE('Database'[TP_CAL]),'Database'[TYPE]                    =      "FTP
Download",'Database'[operator1] = "my",'Database'[STATUS] = "PASS")

VAR                                Avg_TP_web                                =
CALCULATE(AVERAGE('Database'[TP_CAL]),'Database'[TYPE]                    =      "HTTP
Download",'Database'[operator1] = "my",'Database'[STATUS] = "PASS")

RETURN

SWITCH(
    SELECTEDVALUE('QoS_Parameter'[Index]),
    6, FORMAT(NoS_Ping, "#,##0"),
    17, FORMAT(NoS_HTTP, "#,##0"),
    27, FORMAT(NoS_FTPUL, "#,##0"),
    36, FORMAT(NoS_FTPDL, "#,##0"),
    5, FORMAT(PASS_ping, "#,##0"),
    11, FORMAT(PASS_http, "#,##0"),
    20, FORMAT(PASS_ftpul, "#,##0"),
    29, FORMAT(PASS_ftpdl, "#,##0"),
    13, FORMAT(FAIL_http, "#,##0"),
    22, FORMAT(FAIL_ftpul, "#,##0"),
    31, FORMAT(FAIL_ftpdl, "#,##0"),
    2, FORMAT(TP_PASS_ping, "#,##0"),
    15, FORMAT(TP_PASS_http, "#,##0"),
    24, FORMAT(TP_PASS_ftpul, "#,##0"),
    33, FORMAT(TP_PASS_ftpdl, "#,##0"),

```

```

12,          IF(ISBLANK(NoS_HTTP),"N/A",FORMAT(100          *
DIVIDE(PASS_http,NoS_HTTP),"##0.00")),
21,          IF(ISBLANK(NoS_FTPUL),"N/A",FORMAT(100          *
DIVIDE(PASS_ftpul,NoS_FTPUL),"##0.00")),
30,          IF(ISBLANK(NoS_FTPDL),"N/A",FORMAT(100*
DIVIDE(PASS_ftpdl,NoS_FTPDL),"##0.00")),
14,          IF(ISBLANK(NoS_HTTP),"N/A",FORMAT(100*
DIVIDE(FAIL_http,NoS_HTTP),"##0.00")),
23,          IF(ISBLANK(NoS_FTPUL),"N/A",FORMAT(100*
DIVIDE(FAIL_ftpul,NoS_FTPUL),"##0.00")),
32,          IF(ISBLANK(NoS_FTPDL),"N/A",FORMAT(100*
DIVIDE(FAIL_ftpdl,NoS_FTPDL),"##0.00")),
4,          IF(ISBLANK(PASS_ping),"N/A",FORMAT(100*
DIVIDE(TP_PASS_ping,PASS_ping),"##0.00")),
26,          IF(ISBLANK(PASS_ftpul),"N/A",FORMAT(100*
DIVIDE(TP_PASS_ftpul,PASS_ftpul),"##0.00")),
35,          IF(ISBLANK(PASS_ftpdl),"N/A",FORMAT(100*
DIVIDE(TP_PASS_ftpdl,PASS_ftpdl),"##0.00")),
3, FORMAT(Avg_TP_Ping,"##0"),
16, FORMAT(Avg_TP_HTTP / 1000,"#,##0.000"),
25, FORMAT(Avg_TP_ftpul / 1000,"#,##0.000"),
34, FORMAT(Avg_TP_ftpdl / 1000 ,"#,##0.000"),
7, FORMAT(FAIL_ping,"#,##0"),
8,          IF(ISBLANK(NoS_Ping),"N/A",FORMAT(100          *
DIVIDE(FAIL_ping,NoS_Ping),"##0.00")),
9,          IF(ISBLANK(NoS_Ping),"N/A",FORMAT(100          *
DIVIDE(PASS_ping,NoS_Ping),"##0.00")),
18, FORMAT(Avg_TP_web / 1000 ,"#,##0.000"),
BLANK()
)

```

```

NT-TOT =
VAR NoS_Ping = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"Ping",'Database'[operator1] = "TOT")
VAR NoS_HTTP = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"HTTP Download",'Database'[operator1] = "TOT")
VAR NoS_FTPUL = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"FTP Upload",'Database'[operator1] = "TOT")
VAR NoS_FTPDL = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"FTP Download",'Database'[operator1] = "TOT")
VAR PASS_ping = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"Ping",'Database'[operator1] = "TOT",'Database'[STATUS] = "PASS")
VAR PASS_http = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"HTTP Download",'Database'[operator1] = "TOT",'Database'[STATUS] = "PASS")
VAR PASS_ftpul = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"FTP Upload",'Database'[operator1] = "TOT",'Database'[STATUS] = "PASS")
VAR PASS_ftpdl = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"FTP Download",'Database'[operator1] = "TOT",'Database'[STATUS] = "PASS")
VAR FAIL_ping = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"Ping",'Database'[operator1] = "TOT",'Database'[STATUS] = "FAIL")
VAR FAIL_http = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"HTTP Download",'Database'[operator1] = "TOT",'Database'[STATUS] = "FAIL")
VAR FAIL_ftpul = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"FTP Upload",'Database'[operator1] = "TOT",'Database'[STATUS] = "FAIL")
VAR FAIL_ftpdl = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"FTP Download",'Database'[operator1] = "TOT",'Database'[STATUS] = "FAIL")
VAR TP_PASS_ping = CALCULATE(COUNTROWS('Database'),'Database'[TYPE]
= "Ping",'Database'[operator1] = "TOT",'Database'[STATUS] = "PASS",'Database'[TP2] =
"yes")
VAR TP_PASS_http = CALCULATE(COUNTROWS('Database'),'Database'[TYPE]
= "HTTP Download",'Database'[operator1] = "TOT",'Database'[STATUS] =
"PASS",'Database'[TP2] = "yes")

```

```

VAR TP_PASS_ftpul = CALCULATE(COUNTROWS('Database'),'Database'[TYPE]
= "FTP Upload",'Database'[operator1] = "TOT",'Database'[STATUS] =
"PASS",'Database'[TP2] = "yes")
VAR TP_PASS_ftpdl = CALCULATE(COUNTROWS('Database'),'Database'[TYPE]
= "FTP Download",'Database'[operator1] = "TOT",'Database'[STATUS] =
"PASS",'Database'[TP2] = "yes")
VAR Avg_TP_Ping =
CALCULATE(AVERAGE('Database'[TP_CAL]),'Database'[TYPE]
= "Ping",'Database'[operator1] = "TOT",'Database'[STATUS] = "PASS")
VAR Avg_TP_HTTP =
CALCULATE(AVERAGE('Database'[USED_TIME]),'Database'[TYPE] = "HTTP
Download",'Database'[operator1] = "TOT",'Database'[STATUS] = "PASS")
VAR Avg_TP_ftpul =
CALCULATE(AVERAGE('Database'[TP_CAL]),'Database'[TYPE] = "FTP
Upload",'Database'[operator1] = "TOT",'Database'[STATUS] = "PASS")
VAR Avg_TP_ftpdl =
CALCULATE(AVERAGE('Database'[TP_CAL]),'Database'[TYPE] = "FTP
Download",'Database'[operator1] = "TOT",'Database'[STATUS] = "PASS")
VAR Avg_TP_web =
CALCULATE(AVERAGE('Database'[TP_CAL]),'Database'[TYPE] = "HTTP
Download",'Database'[operator1] = "TOT",'Database'[STATUS] = "PASS")
RETURN
SWITCH(
SELECTEDVALUE('QoS_Parameter'[Index]),
6, FORMAT(NoS_Ping, "#,##0"),
17, FORMAT(NoS_HTTP, "#,##0"),
27, FORMAT(NoS_FTPUL, "#,##0"),
36, FORMAT(NoS_FTPDL, "#,##0"),
5, FORMAT(PASS_ping, "#,##0"),
11, FORMAT(PASS_http, "#,##0"),
20, FORMAT(PASS_ftpul, "#,##0"),

```

```

29, FORMAT(PASS_ftpdl, "###0"),
13, FORMAT(FAIL_http, "###0"),
22, FORMAT(FAIL_ftpul, "###0"),
31, FORMAT(FAIL_ftpdl, "###0"),
2, FORMAT(TP_PASS_ping, "###0"),
15, FORMAT(TP_PASS_http, "###0"),
24, FORMAT(TP_PASS_ftpul, "###0"),
33, FORMAT(TP_PASS_ftpdl, "###0"),
12,          IF(ISBLANK(NoS_HTTP), "N/A", FORMAT(100          *
DIVIDE(PASS_http, NoS_HTTP, "##0.00")),
21,          IF(ISBLANK(NoS_FTPUL), "N/A", FORMAT(100          *
DIVIDE(PASS_ftpul, NoS_FTPUL, "##0.00")),
30,          IF(ISBLANK(NoS_FTPDL), "N/A", FORMAT(100*
DIVIDE(PASS_ftpdl, NoS_FTPDL, "##0.00")),
14,          IF(ISBLANK(NoS_HTTP), "N/A", FORMAT(100*
DIVIDE(FAIL_http, NoS_HTTP, "##0.00")),
23,          IF(ISBLANK(NoS_FTPUL), "N/A", FORMAT(100*
DIVIDE(FAIL_ftpul, NoS_FTPUL, "##0.00")),
32,          IF(ISBLANK(NoS_FTPDL), "N/A", FORMAT(100*
DIVIDE(FAIL_ftpdl, NoS_FTPDL, "##0.00")),
4,          IF(ISBLANK(PASS_ping), "N/A", FORMAT(100*
DIVIDE(TP_PASS_ping, PASS_ping, "##0.00")),
26,          IF(ISBLANK(PASS_ftpul), "N/A", FORMAT(100*
DIVIDE(TP_PASS_ftpul, PASS_ftpul, "##0.00")),
35,          IF(ISBLANK(PASS_ftpdl), "N/A", FORMAT(100*
DIVIDE(TP_PASS_ftpdl, PASS_ftpdl, "##0.00")),
3, FORMAT(Avg_TP_Ping, "##0"),
16, FORMAT(Avg_TP_HTTP / 1000, "###0.000"),
25, FORMAT(Avg_TP_ftpul / 1000, "###0.000"),
34, FORMAT(Avg_TP_ftpdl / 1000, "###0.000"),
7, FORMAT(FAIL_ping, "###0"),

```



```

8,          IF(ISBLANK(NoS_Ping),"N/A",FORMAT(100          *
DIVIDE(FAIL_ping,NoS_Ping),##0.00)),
9,          IF(ISBLANK(NoS_Ping),"N/A",FORMAT(100          *
DIVIDE(PASS_ping,NoS_Ping),##0.00)),
18, FORMAT(Avg_TP_web / 1000 ,"#,##0.000"),
BLANK()
)

```

```

TRUE-H =
VAR NoS_Ping = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"Ping",'Database'[operator1] = "TRUE-H")
VAR NoS_HTTP = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"HTTP Download",'Database'[operator1] = "TRUE-H")
VAR NoS_FTPUL = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"FTP Upload",'Database'[operator1] = "TRUE-H")
VAR NoS_FTPDL = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"FTP Download",'Database'[operator1] = "TRUE-H")
VAR PASS_ping = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"Ping",'Database'[operator1] = "TRUE-H",'Database'[STATUS] = "PASS")
VAR PASS_http = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"HTTP Download",'Database'[operator1] = "TRUE-H",'Database'[STATUS] = "PASS")
VAR PASS_ftpul = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"FTP Upload",'Database'[operator1] = "TRUE-H",'Database'[STATUS] = "PASS")
VAR PASS_ftpdl = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"FTP Download",'Database'[operator1] = "TRUE-H",'Database'[STATUS] = "PASS")
VAR FAIL_ping = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"Ping",'Database'[operator1] = "TRUE-H",'Database'[STATUS] = "FAIL")
VAR FAIL_http = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"HTTP Download",'Database'[operator1] = "TRUE-H",'Database'[STATUS] = "FAIL")
VAR FAIL_ftpul = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"FTP Upload",'Database'[operator1] = "TRUE-H",'Database'[STATUS] = "FAIL")

```

```

VAR FAIL_ftpdL = CALCULATE(COUNTROWS('Database'),'Database'[TYPE] =
"FTP Download",'Database'[operator1] = "TRUE-H",'Database'[STATUS] = "FAIL")

VAR TP_PASS_ping = CALCULATE(COUNTROWS('Database'),'Database'[TYPE]
= "Ping",'Database'[operator1] = "TRUE-H",'Database'[STATUS] = "PASS",'Database'[TP2] =
"yes")

VAR TP_PASS_http = CALCULATE(COUNTROWS('Database'),'Database'[TYPE]
= "HTTP Download",'Database'[operator1] = "TRUE-H",'Database'[STATUS] =
"PASS",'Database'[TP2] = "yes")

VAR TP_PASS_ftpul = CALCULATE(COUNTROWS('Database'),'Database'[TYPE]
= "FTP Upload",'Database'[operator1] = "TRUE-H",'Database'[STATUS] =
"PASS",'Database'[TP2] = "yes")

VAR TP_PASS_ftpdL = CALCULATE(COUNTROWS('Database'),'Database'[TYPE]
= "FTP Download",'Database'[operator1] = "TRUE-H",'Database'[STATUS] =
"PASS",'Database'[TP2] = "yes")

VAR Avg_TP_Ping =
CALCULATE(AVERAGE('Database'[TP_CAL]),'Database'[TYPE]
= "Ping",'Database'[operator1] = "TRUE-H",'Database'[STATUS] = "PASS")

VAR Avg_TP_HTTP =
CALCULATE(AVERAGE('Database'[USED_TIME]),'Database'[TYPE] = "HTTP
Download",'Database'[operator1] = "TRUE-H",'Database'[STATUS] = "PASS")

VAR Avg_TP_ftpul =
CALCULATE(AVERAGE('Database'[TP_CAL]),'Database'[TYPE] = "FTP
Upload",'Database'[operator1] = "TRUE-H",'Database'[STATUS] = "PASS")

VAR Avg_TP_ftpdL =
CALCULATE(AVERAGE('Database'[TP_CAL]),'Database'[TYPE] = "FTP
Download",'Database'[operator1] = "TRUE-H",'Database'[STATUS] = "PASS")

VAR Avg_TP_web =
CALCULATE(AVERAGE('Database'[TP_CAL]),'Database'[TYPE] = "HTTP
Download",'Database'[operator1] = "TRUE-H",'Database'[STATUS] = "PASS")

RETURN
SWITCH(

```

```

SELECTEDVALUE('QoS_Parameter'[Index]),
6, FORMAT(NoS_Ping, "###0"),
17, FORMAT(NoS_HTTP, "###0"),
27, FORMAT(NoS_FTPUL, "###0"),
36, FORMAT(NoS_FTPDL, "###0"),
5, FORMAT(PASS_ping, "###0"),
11, FORMAT(PASS_http, "###0"),
20, FORMAT(PASS_ftpul, "###0"),
29, FORMAT(PASS_ftpdl, "###0"),
13, FORMAT(FAIL_http, "###0"),
22, FORMAT(FAIL_ftpul, "###0"),
31, FORMAT(FAIL_ftpdl, "###0"),
2, FORMAT(TP_PASS_ping, "###0"),
15, FORMAT(TP_PASS_http, "###0"),
24, FORMAT(TP_PASS_ftpul, "###0"),
33, FORMAT(TP_PASS_ftpdl, "###0"),
12, IF(ISBLANK(NoS_HTTP), "N/A", FORMAT(100 *
DIVIDE(PASS_http, NoS_HTTP, "##0.00")),
21, IF(ISBLANK(NoS_FTPUL), "N/A", FORMAT(100 *
DIVIDE(PASS_ftpul, NoS_FTPUL, "##0.00")),
30, IF(ISBLANK(NoS_FTPDL), "N/A", FORMAT(100 *
DIVIDE(PASS_ftpdl, NoS_FTPDL, "##0.00")),
14, IF(ISBLANK(NoS_HTTP), "N/A", FORMAT(100 *
DIVIDE(FAIL_http, NoS_HTTP, "##0.00")),
23, IF(ISBLANK(NoS_FTPUL), "N/A", FORMAT(100 *
DIVIDE(FAIL_ftpul, NoS_FTPUL, "##0.00")),
32, IF(ISBLANK(NoS_FTPDL), "N/A", FORMAT(100 *
DIVIDE(FAIL_ftpdl, NoS_FTPDL, "##0.00")),
4, IF(ISBLANK(PASS_ping), "N/A", FORMAT(100 *
DIVIDE(TP_PASS_ping, PASS_ping, "##0.00")),

```

```

26,                                IF(ISBLANK(PASS_ftpul),"N/A",FORMAT(100*
DIVIDE(TP_PASS_ftpul,PASS_ftpul,"##0.00")),
35,                                IF(ISBLANK(PASS_ftpdl),"N/A",FORMAT(100*
DIVIDE(TP_PASS_ftpdl,PASS_ftpdl,"##0.00")),
3, FORMAT(Avg_TP_Ping,"##0"),
16, FORMAT(Avg_TP_HTTP / 1000,"#,##0.000"),
25, FORMAT(Avg_TP_ftpul / 1000,"#,##0.000"),
34, FORMAT(Avg_TP_ftpdl / 1000,"#,##0.000"),
7, FORMAT(FAIL_ping,"#,##0"),
8,                                IF(ISBLANK(NoS_Ping),"N/A",FORMAT(100
DIVIDE(FAIL_ping,NoS_Ping,"##0.00")),
9,                                IF(ISBLANK(NoS_Ping),"N/A",FORMAT(100
DIVIDE(PASS_ping,NoS_Ping,"##0.00")),
18, FORMAT(Avg_TP_web / 1000,"#,##0.000"),
BLANK()
)

```

### ส่วนของการแสดงผลของการบริการประเภทข้อมูล

```

CF_AWN =
// ping
VAR PASS_ping = CALCULATE(COUNTROWS('Database'), 'Database'[TYPE] =
"Ping", 'Database'[STATUS] = "PASS", 'Database'[operator1] = "TH GSM")
VAR TP_PASS_ping = CALCULATE(COUNTROWS('Database'), 'Database'[TYPE]
= "Ping", 'Database'[STATUS] = "PASS", 'Database'[operator1] = "TH GSM",'Database'[TP2]
= "yes")
// UL
VAR      NoS_FTPUL      =      CALCULATE(COUNTROWS('Database'),
'Database'[operator1] = "TH GSM",'Database'[TYPE] = "FTP Upload")
VAR      PASS_ftpul      =      CALCULATE(COUNTROWS('Database'),
'Database'[operator1] = "TH GSM",'Database'[TYPE] = "FTP Upload",'Database'[STATUS] =
"PASS")

```

```

        VAR      TP_PASS_ftpul      =      CALCULATE(COUNTROWS('Database'),
'Database'[operator1] = "TH GSM",'Database'[TYPE] = "FTP Upload",'Database'[STATUS] =
"PASS",'Database'[TP2] = "yes")

        // DL

        VAR      NoS_FTPDL      =      CALCULATE(COUNTROWS('Database'),
'Database'[operator1] = "TH GSM",'Database'[TYPE] = "FTP Download")

        VAR      PASS_ftpdl      =      CALCULATE(COUNTROWS('Database'),
'Database'[operator1]      =      "TH      GSM",'Database'[TYPE]      =      "FTP
Download",'Database'[STATUS] = "PASS")

        VAR      TP_PASS_ftpdl      =      CALCULATE(COUNTROWS('Database'),
'Database'[operator1]      =      "TH      GSM",'Database'[TYPE]      =      "FTP
Download",'Database'[STATUS] = "PASS",'Database'[TP2] = "yes")

        RETURN

        SWITCH(
            SELECTEDVALUE('QoS_Parameter'[Index]),
            // PING
            4,      IF(DIVIDE(TP_PASS_ping,PASS_ping)      >=      0.8      ,"#C1E1C1",
IF(DIVIDE(TP_PASS_ping,PASS_ping) < 0.8 ,"#FFB6B6")),
            // UL
            21,      IF(DIVIDE(PASS_ftpul,NoS_FTPUL)      >=      0.8      ,"#C1E1C1",
IF(DIVIDE(PASS_ftpul,NoS_FTPUL) < 0.8 ,"#FFB6B6")),
            26,      IF(DIVIDE(TP_PASS_ftpul,PASS_ftpul)      >=      0.8      ,"#C1E1C1",
IF(DIVIDE(TP_PASS_ftpul,PASS_ftpul) < 0.8 ,"#FFB6B6")),
            // DL
            30,      IF(DIVIDE(PASS_ftpdl,NoS_FTPDL)      >=      0.08      ,"#C1E1C1",
IF(DIVIDE(PASS_ftpdl,NoS_FTPDL) < 0.8 ,"#FFB6B6")),
            35,      IF(DIVIDE(TP_PASS_ftpdl,PASS_ftpdl)      >=      0.08      ,"#C1E1C1",
IF(DIVIDE(TP_PASS_ftpdl,PASS_ftpdl) < 0.8 ,"#FFB6B6"))
        )

        CF_CAT =

```

```

// ping
VAR PASS_ping = CALCULATE(COUNTROWS('Database'), 'Database'[TYPE] =
"Ping", 'Database'[STATUS] = "PASS", 'Database'[operator1] = "my")
VAR TP_PASS_ping = CALCULATE(COUNTROWS('Database'), 'Database'[TYPE]
= "Ping", 'Database'[STATUS] = "PASS", 'Database'[operator1] = "my", 'Database'[TP2] =
"yes")

// UL
VAR      NoS_FTPUL      =      CALCULATE(COUNTROWS('Database'),
'Database'[operator1] = "my", 'Database'[TYPE] = "FTP Upload")
VAR      PASS_ftpul      =      CALCULATE(COUNTROWS('Database'),
'Database'[operator1] = "my", 'Database'[TYPE] = "FTP Upload", 'Database'[STATUS] =
"PASS")
VAR      TP_PASS_ftpul      =      CALCULATE(COUNTROWS('Database'),
'Database'[operator1] = "my", 'Database'[TYPE] = "FTP Upload", 'Database'[STATUS] =
"PASS", 'Database'[TP2] = "yes")

// DL
VAR      NoS_FTPDL      =      CALCULATE(COUNTROWS('Database'),
'Database'[operator1] = "my", 'Database'[TYPE] = "FTP Download")
VAR      PASS_ftpdl      =      CALCULATE(COUNTROWS('Database'),
'Database'[operator1] = "my", 'Database'[TYPE] = "FTP Download", 'Database'[STATUS] =
"PASS")
VAR      TP_PASS_ftpdl      =      CALCULATE(COUNTROWS('Database'),
'Database'[operator1] = "my", 'Database'[TYPE] = "FTP Download", 'Database'[STATUS] =
"PASS", 'Database'[TP2] = "yes")

RETURN
SWITCH(
    SELECTEDVALUE('QoS_Parameter'[Index]),
    // PING
    4, IF(DIVIDE(TP_PASS_ping, PASS_ping) >= 0.8 , "#C1E1C1",
IF(DIVIDE(TP_PASS_ping, PASS_ping) < 0.8 , "#FFB6B6")),
    // UL

```

```

21,    IF(DIVIDE(PASS_ftpul,NoS_FTPUL)    >=    0.8    ,"#C1E1C1",
IF(DIVIDE(PASS_ftpul,NoS_FTPUL) < 0.8 ,"#FFB6B6")),
26,    IF(DIVIDE(TP_PASS_ftpul,PASS_ftpul)    >=    0.8    ,"#C1E1C1",
IF(DIVIDE(TP_PASS_ftpul,PASS_ftpul) < 0.8 ,"#FFB6B6")),
// DL
30,    IF(DIVIDE(PASS_ftpdL,NoS_FTPDL)    >=    0.08    ,"#C1E1C1",
IF(DIVIDE(PASS_ftpdL,NoS_FTPDL) < 0.8 ,"#FFB6B6")),
35,    IF(DIVIDE(TP_PASS_ftpdL,PASS_ftpdL)    >=    0.08    ,"#C1E1C1",
IF(DIVIDE(TP_PASS_ftpdL,PASS_ftpdL) < 0.8 ,"#FFB6B6"))
)

CF_DTN =
// ping
VAR PASS_ping = CALCULATE(COUNTROWS('Database'), 'Database'[TYPE] =
"Ping", 'Database'[STATUS] = "PASS", 'Database'[operator1] = "DTAC")
VAR TP_PASS_ping = CALCULATE(COUNTROWS('Database'), 'Database'[TYPE]
= "Ping", 'Database'[STATUS] = "PASS", 'Database'[operator1] = "DTAC",'Database'[TP2] =
"yes")

// UL
VAR    NoS_FTPUL    =    CALCULATE(COUNTROWS('Database'),
'Database'[operator1] = "DTAC",'Database'[TYPE] = "FTP Upload")
VAR    PASS_ftpul    =    CALCULATE(COUNTROWS('Database'),
'Database'[operator1] = "DTAC",'Database'[TYPE] = "FTP Upload",'Database'[STATUS] =
"PASS")
VAR    TP_PASS_ftpul    =    CALCULATE(COUNTROWS('Database'),
'Database'[operator1] = "DTAC",'Database'[TYPE] = "FTP Upload",'Database'[STATUS] =
"PASS",'Database'[TP2] = "yes")

// DL
VAR    NoS_FTPDL    =    CALCULATE(COUNTROWS('Database'),
'Database'[operator1] = "DTAC",'Database'[TYPE] = "FTP Download")

```

```

        VAR      PASS_ftpdl      =      CALCULATE(COUNTROWS('Database'),
'Database'[operator1] = "DTAC",'Database'[TYPE] = "FTP Download",'Database'[STATUS] =
"PASS")

        VAR      TP_PASS_ftpdl      =      CALCULATE(COUNTROWS('Database'),
'Database'[operator1] = "DTAC",'Database'[TYPE] = "FTP Download",'Database'[STATUS] =
"PASS",'Database'[TP2] = "yes")

        RETURN

        SWITCH(
            SELECTEDVALUE('QoS_Parameter'[Index]),
            // PING
            4,      IF(DIVIDE(TP_PASS_ping,PASS_ping)      >=      0.8      ,"#C1E1C1",
IF(DIVIDE(TP_PASS_ping,PASS_ping) < 0.8 ,"#FFB6B6")),
            // UL
            21,      IF(DIVIDE(PASS_ftpul,NoS_FTPUL)      >=      0.8      ,"#C1E1C1",
IF(DIVIDE(PASS_ftpul,NoS_FTPUL) < 0.8 ,"#FFB6B6")),
            26,      IF(DIVIDE(TP_PASS_ftpul,PASS_ftpul)      >=      0.8      ,"#C1E1C1",
IF(DIVIDE(TP_PASS_ftpul,PASS_ftpul) < 0.8 ,"#FFB6B6")),
            // DL
            30,      IF(DIVIDE(PASS_ftpdl,NoS_FTPDL)      >=      0.08      ,"#C1E1C1",
IF(DIVIDE(PASS_ftpdl,NoS_FTPDL) < 0.8 ,"#FFB6B6")),
            35,      IF(DIVIDE(TP_PASS_ftpdl,PASS_ftpdl)      >=      0.08      ,"#C1E1C1",
IF(DIVIDE(TP_PASS_ftpdl,PASS_ftpdl) < 0.8 ,"#FFB6B6"))
        )

        CF_TOT =

        // ping
        VAR PASS_ping = CALCULATE(COUNTROWS('Database'), 'Database'[TYPE] =
"Ping", 'Database'[STATUS] = "PASS", 'Database'[operator1] = "TOT")

        VAR TP_PASS_ping = CALCULATE(COUNTROWS('Database'), 'Database'[TYPE]
= "Ping", 'Database'[STATUS] = "PASS", 'Database'[operator1] = "TOT",'Database'[TP2] =
"yes")

```



```

// UL
VAR      NoS_FTPUL      =      CALCULATE(COUNTROWS('Database'),
'Database'[operator1] = "TOT",'Database'[TYPE] = "FTP Upload")
VAR      PASS_ftpul     =      CALCULATE(COUNTROWS('Database'),
'Database'[operator1] = "TOT",'Database'[TYPE] = "FTP Upload",'Database'[STATUS] =
"PASS")
VAR      TP_PASS_ftpul  =      CALCULATE(COUNTROWS('Database'),
'Database'[operator1] = "TOT",'Database'[TYPE] = "FTP Upload",'Database'[STATUS] =
"PASS",'Database'[TP2] = "yes")
// DL
VAR      NoS_FTPDL      =      CALCULATE(COUNTROWS('Database'),
'Database'[operator1] = "TOT",'Database'[TYPE] = "FTP Download")
VAR      PASS_ftpdl     =      CALCULATE(COUNTROWS('Database'),
'Database'[operator1] = "TOT",'Database'[TYPE] = "FTP Download",'Database'[STATUS] =
"PASS")
VAR      TP_PASS_ftpdl  =      CALCULATE(COUNTROWS('Database'),
'Database'[operator1] = "TOT",'Database'[TYPE] = "FTP Download",'Database'[STATUS] =
"PASS",'Database'[TP2] = "yes")
RETURN
SWITCH(
    SELECTEDVALUE('QoS_Parameter'[Index]),
    // PING
    4,    IF(DIVIDE(TP_PASS_ping,PASS_ping)    >=    0.8    ,"#C1E1C1",
IF(DIVIDE(TP_PASS_ping,PASS_ping) < 0.8 ,"#FFB6B6")),
    // UL
    21,   IF(DIVIDE(PASS_ftpul,NoS_FTPUL)    >=    0.8    ,"#C1E1C1",
IF(DIVIDE(PASS_ftpul,NoS_FTPUL) < 0.8 ,"#FFB6B6")),
    26,   IF(DIVIDE(TP_PASS_ftpul,PASS_ftpul)    >=    0.8    ,"#C1E1C1",
IF(DIVIDE(TP_PASS_ftpul,PASS_ftpul) < 0.8 ,"#FFB6B6")),
    // DL

```

```

30,    IF(DIVIDE(PASS_ftpdL,NoS_FTPDL)    >=    0.08    ,"#C1E1C1",
IF(DIVIDE(PASS_ftpdL,NoS_FTPDL) < 0.8 ,"#FFB6B6")),
35,    IF(DIVIDE(TP_PASS_ftpdL,PASS_ftpdL)    >=    0.08    ,"#C1E1C1",
IF(DIVIDE(TP_PASS_ftpdL,PASS_ftpdL) < 0.8 ,"#FFB6B6"))
)

```

```

CF_TRUE =
// ping
VAR PASS_ping = CALCULATE(COUNTROWS('Database'), 'Database'[TYPE] =
"Ping", 'Database'[STATUS] = "PASS", 'Database'[operator1] = "TRUE-H")
VAR TP_PASS_ping = CALCULATE(COUNTROWS('Database'), 'Database'[TYPE]
= "Ping", 'Database'[STATUS] = "PASS", 'Database'[operator1] = "TRUE-H",'Database'[TP2]
= "yes")
// UL
VAR    NoS_FTPUL    =    CALCULATE(COUNTROWS('Database'),
'Database'[operator1] = "TRUE-H",'Database'[TYPE] = "FTP Upload")
VAR    PASS_ftpul    =    CALCULATE(COUNTROWS('Database'),
'Database'[operator1] = "TRUE-H",'Database'[TYPE] = "FTP Upload",'Database'[STATUS] =
"PASS")
VAR    TP_PASS_ftpul    =    CALCULATE(COUNTROWS('Database'),
'Database'[operator1] = "TRUE-H",'Database'[TYPE] = "FTP Upload",'Database'[STATUS] =
"PASS",'Database'[TP2] = "yes")
// DL
VAR    NoS_FTPDL    =    CALCULATE(COUNTROWS('Database'),
'Database'[operator1] = "TRUE-H",'Database'[TYPE] = "FTP Download")
VAR    PASS_ftpdL    =    CALCULATE(COUNTROWS('Database'),
'Database'[operator1] = "TRUE-H",'Database'[TYPE] = "FTP Download",'Database'[STATUS]
= "PASS")
VAR    TP_PASS_ftpdL    =    CALCULATE(COUNTROWS('Database'),
'Database'[operator1] = "TRUE-H",'Database'[TYPE] = "FTP Download",'Database'[STATUS]
= "PASS",'Database'[TP2] = "yes")

```

```

RETURN
SWITCH(
    SELECTEDVALUE('QoS_Parameter'[Index]),
    // PING
    4,    IF(DIVIDE(TP_PASS_ping,PASS_ping)    >=    0.8    ,"#C1E1C1",
IF(DIVIDE(TP_PASS_ping,PASS_ping) < 0.8 ,"#FFB6B6")),
    // UL
    21,    IF(DIVIDE(PASS_ftpul,NoS_FTPUL)    >=    0.8    ,"#C1E1C1",
IF(DIVIDE(PASS_ftpul,NoS_FTPUL) < 0.8 ,"#FFB6B6")),
    26,    IF(DIVIDE(TP_PASS_ftpul,PASS_ftpul)    >=    0.8    ,"#C1E1C1",
IF(DIVIDE(TP_PASS_ftpul,PASS_ftpul) < 0.8 ,"#FFB6B6")),
    // DL
    30,    IF(DIVIDE(PASS_ftpdl,NoS_FTPDL)    >=    0.08    ,"#C1E1C1",
IF(DIVIDE(PASS_ftpdl,NoS_FTPDL) < 0.8 ,"#FFB6B6")),
    35,    IF(DIVIDE(TP_PASS_ftpdl,PASS_ftpdl)    >=    0.08    ,"#C1E1C1",
IF(DIVIDE(TP_PASS_ftpdl,PASS_ftpdl) < 0.8 ,"#FFB6B6"))
)

```

### ส่วนของการคำนวณของการบริการประเภทเสียง

```

AWN_voc =
VAR VOICE_SAMPLE = CALCULATE(COUNTAX(FILTER('voice database','voice
database'[TYPE] = "Voice Successful" && 'voice database'[operator1] = "TH GSM" &&
'voice database'[TP] in {"Complete","Drop","Block"}),'voice database'[IMEI]))
VAR    VOICE_COMPLETE    =    CALCULATE(COUNTAX(FILTER('voice
database','voice database'[TYPE] = "Voice Successful" && 'voice database'[operator1] =
"TH GSM" && 'voice database'[TP] = "Complete"),'voice database'[IMEI]))
VAR VOICE_BLOCK = CALCULATE(COUNTAX(FILTER('voice database','voice
database'[TYPE] = "Voice Successful" && 'voice database'[operator1] = "TH GSM" &&
'voice database'[TP] = "Block"),'voice database'[IMEI]))

```

```

VAR VOICE_DROP = CALCULATE(COUNTAX(FILTER('voice database','voice
database'[TYPE] = "Voice Successful" && 'voice database'[operator1] = "TH GSM" &&
'voice database'[TP] = "Drop"),'voice database'[IMEI]))

```

```

VAR VOICE_BAD = CALCULATE(COUNTAX(FILTER('voice database','voice
database'[TYPE] = "Voice Successful" && 'voice database'[operator1] = "TH GSM" &&
'voice database'[TP] = "Bad Attempt"),'voice database'[IMEI]))

```

```

RETURN

```

```

SWITCH(

```

```

    SELECTEDVALUE('QoS_Parameter(Voice)'[Index]),

```

```

    1, BLANK(),

```

```

    2, IF(ISBLANK(VOICE_SAMPLE), BLANK(), FORMAT(VOICE_SAMPLE,
"#,##0")),

```

```

    3, IF(ISBLANK(VOICE_COMPLETE), BLANK(), FORMAT(VOICE_COMPLETE,
"#,##0")),

```

```

    4, IF(ISBLANK(VOICE_SAMPLE) || ISBLANK(VOICE_COMPLETE +
VOICE_DROP), "0.00%", FORMAT(DIVIDE(VOICE_COMPLETE + VOICE_DROP,
VOICE_SAMPLE), "0.00%")),

```

```

    5, IF(ISBLANK(VOICE_BLOCK), BLANK(), FORMAT(VOICE_BLOCK, "#,##0")),

```

```

    6, IF(ISBLANK(VOICE_DROP), BLANK(), FORMAT(VOICE_DROP, "#,##0")),

```

```

    7, IF(ISBLANK(VOICE_DROP), BLANK(), FORMAT(DIVIDE(VOICE_DROP,
VOICE_SAMPLE - VOICE_BLOCK), "0.00%")),

```

```

    8, IF(ISBLANK(VOICE_COMPLETE + VOICE_DROP), BLANK(),
FORMAT(VOICE_COMPLETE + VOICE_DROP, "#,##0")),

```

```

    9, IF(ISBLANK(VOICE_SAMPLE) || ISBLANK(VOICE_COMPLETE), "0.00%",
FORMAT(DIVIDE(VOICE_COMPLETE, VOICE_SAMPLE), "0.00%")),

```

```

    10, IF(ISBLANK(VOICE_BAD), BLANK(), FORMAT(VOICE_BAD, "#,##0")),

```

```

    11, IF(ISBLANK(VOICE_SAMPLE) || ISBLANK(VOICE_BAD), "0.00%",
FORMAT(DIVIDE(VOICE_BAD, VOICE_SAMPLE), "0.00%")),

```

```

    0

```

```

)

```

```

CAT_voc =
VAR VOICE_SAMPLE = CALCULATE(COUNTAX(FILTER('voice database','voice
database'[TYPE] = "Voice Successful" && 'voice database'[operator1] = "my" && 'voice
database'[TP] in {"Complete","Drop","Block"}),'voice database'[IMEI]))
VAR VOICE_COMPLETE = CALCULATE(COUNTAX(FILTER('voice
database','voice database'[TYPE] = "Voice Successful" && 'voice database'[operator1] =
"my" && 'voice database'[TP] = "Complete"),'voice database'[IMEI]))
VAR VOICE_BLOCK = CALCULATE(COUNTAX(FILTER('voice database','voice
database'[TYPE] = "Voice Successful" && 'voice database'[operator1] = "my" && 'voice
database'[TP] = "Block"),'voice database'[IMEI]))
VAR VOICE_DROP = CALCULATE(COUNTAX(FILTER('voice database','voice
database'[TYPE] = "Voice Successful" && 'voice database'[operator1] = "my" && 'voice
database'[TP] = "Drop"),'voice database'[IMEI]))
VAR VOICE_BAD = CALCULATE(COUNTAX(FILTER('voice database','voice
database'[TYPE] = "Voice Successful" && 'voice database'[operator1] = "my" && 'voice
database'[TP] = "Bad Attempt"),'voice database'[IMEI]))
RETURN
SWITCH(
    SELECTEDVALUE('QoS_Parameter(Voice)'[Index]),
    1, BLANK(),
    2, IF(ISBLANK(VOICE_SAMPLE), BLANK(), FORMAT(VOICE_SAMPLE,
"#,##0")),
    3, IF(ISBLANK(VOICE_COMPLETE), BLANK(), FORMAT(VOICE_COMPLETE,
"#,##0")),
    4, IF(ISBLANK(VOICE_SAMPLE) || ISBLANK(VOICE_COMPLETE +
VOICE_DROP), "0.00%", FORMAT(DIVIDE(VOICE_COMPLETE + VOICE_DROP,
VOICE_SAMPLE), "0.00%")),
    5, IF(ISBLANK(VOICE_BLOCK), BLANK(), FORMAT(VOICE_BLOCK, "#,##0")),
    6, IF(ISBLANK(VOICE_DROP), BLANK(), FORMAT(VOICE_DROP, "#,##0")),
    7, IF(ISBLANK(VOICE_DROP), BLANK(), FORMAT(DIVIDE(VOICE_DROP,
VOICE_SAMPLE - VOICE_BLOCK), "0.00%")),

```

```

8,    IF(ISBLANK(VOICE_COMPLETE + VOICE_DROP),    BLANK(),
FORMAT(VOICE_COMPLETE + VOICE_DROP, "#,##0")),
9, IF(ISBLANK(VOICE_SAMPLE) || ISBLANK(VOICE_COMPLETE), "0.00%",
FORMAT(DIVIDE(VOICE_COMPLETE, VOICE_SAMPLE), "0.00%")),
10, IF(ISBLANK(VOICE_BAD), BLANK(), FORMAT(VOICE_BAD, "#,##0")),
11, IF(ISBLANK(VOICE_SAMPLE) || ISBLANK(VOICE_BAD), "0.00%",
FORMAT(DIVIDE(VOICE_BAD, VOICE_SAMPLE), "0.00%")),
0
)

```

```

DTN_voc =
VAR VOICE_SAMPLE = CALCULATE(COUNTAX(FILTER('voice database','voice
database'[TYPE] = "Voice Successful" && 'voice database'[operator1] = "DTAC" && 'voice
database'[TP] in {"Complete","Drop","Block"}),'voice database'[IMEI]))
VAR    VOICE_COMPLETE    =    CALCULATE(COUNTAX(FILTER('voice
database','voice database'[TYPE] = "Voice Successful" && 'voice database'[operator1] =
"DTAC" && 'voice database'[TP] = "Complete"),'voice database'[IMEI]))
VAR VOICE_BLOCK = CALCULATE(COUNTAX(FILTER('voice database','voice
database'[TYPE] = "Voice Successful" && 'voice database'[operator1] = "DTAC" && 'voice
database'[TP] = "Block"),'voice database'[IMEI]))
VAR VOICE_DROP = CALCULATE(COUNTAX(FILTER('voice database','voice
database'[TYPE] = "Voice Successful" && 'voice database'[operator1] = "DTAC" && 'voice
database'[TP] = "Drop"),'voice database'[IMEI]))
VAR VOICE_BAD = CALCULATE(COUNTAX(FILTER('voice database','voice
database'[TYPE] = "Voice Successful" && 'voice database'[operator1] = "DTAC" && 'voice
database'[TP] = "Bad Attempt"),'voice database'[IMEI]))
RETURN
SWITCH(
    SELECTEDVALUE('QoS_Parameter(Voice)'[Index]),
    1, BLANK(),

```

```

2, IF(ISBLANK(VOICE_SAMPLE), BLANK(), FORMAT(VOICE_SAMPLE,
"#,##0")),
3, IF(ISBLANK(VOICE_COMPLETE), BLANK(), FORMAT(VOICE_COMPLETE,
"#,##0")),
4, IF(ISBLANK(VOICE_SAMPLE) || ISBLANK(VOICE_COMPLETE +
VOICE_DROP), "0.00%", FORMAT(DIVIDE(VOICE_COMPLETE + VOICE_DROP,
VOICE_SAMPLE), "0.00%")),
5, IF(ISBLANK(VOICE_BLOCK), BLANK(), FORMAT(VOICE_BLOCK, "#,##0")),
6, IF(ISBLANK(VOICE_DROP), BLANK(), FORMAT(VOICE_DROP, "#,##0")),
7, IF(ISBLANK(VOICE_DROP), BLANK(), FORMAT(DIVIDE(VOICE_DROP,
VOICE_SAMPLE - VOICE_BLOCK), "0.00%")),
8, IF(ISBLANK(VOICE_COMPLETE + VOICE_DROP), BLANK(),
FORMAT(VOICE_COMPLETE + VOICE_DROP, "#,##0")),
9, IF(ISBLANK(VOICE_SAMPLE) || ISBLANK(VOICE_COMPLETE), "0.00%",
FORMAT(DIVIDE(VOICE_COMPLETE, VOICE_SAMPLE), "0.00%")),
10, IF(ISBLANK(VOICE_BAD), BLANK(), FORMAT(VOICE_BAD, "#,##0")),
11, IF(ISBLANK(VOICE_SAMPLE) || ISBLANK(VOICE_BAD), "0.00%",
FORMAT(DIVIDE(VOICE_BAD, VOICE_SAMPLE), "0.00%")),
0
)

```

TOT\_voc =

```

VAR VOICE_SAMPLE = CALCULATE(COUNTAX(FILTER('voice database','voice
database'[TYPE] = "Voice Successful" && 'voice database'[operator1] = "TOT" && 'voice
database'[TP] in {"Complete","Drop","Block"}),'voice database'[IMEI]))

```

```

VAR VOICE_COMPLETE = CALCULATE(COUNTAX(FILTER('voice
database','voice database'[TYPE] = "Voice Successful" && 'voice database'[operator1] =
"TOT" && 'voice database'[TP] = "Complete"),'voice database'[IMEI]))

```

```

VAR VOICE_BLOCK = CALCULATE(COUNTAX(FILTER('voice database','voice
database'[TYPE] = "Voice Successful" && 'voice database'[operator1] = "TOT" && 'voice
database'[TP] = "Block"),'voice database'[IMEI]))

```

```

VAR VOICE_DROP = CALCULATE(COUNTAX(FILTER('voice database','voice
database'[TYPE] = "Voice Successful" && 'voice database'[operator1] = "TOT" && 'voice
database'[TP] = "Drop"),'voice database'[IMEI]))

VAR VOICE_BAD = CALCULATE(COUNTAX(FILTER('voice database','voice
database'[TYPE] = "Voice Successful" && 'voice database'[operator1] = "TOT" && 'voice
database'[TP] = "Bad Attempt"),'voice database'[IMEI]))

RETURN

SWITCH(
    SELECTEDVALUE('QoS_Parameter(Voice)'[Index]),
    1, BLANK(),
    2, IF(ISBLANK(VOICE_SAMPLE), BLANK(), FORMAT(VOICE_SAMPLE,
"#,##0")),
    3, IF(ISBLANK(VOICE_COMPLETE), BLANK(), FORMAT(VOICE_COMPLETE,
"#,##0")),
    4, IF(ISBLANK(VOICE_SAMPLE) || ISBLANK(VOICE_COMPLETE +
VOICE_DROP), "0.00%", FORMAT(DIVIDE(VOICE_COMPLETE + VOICE_DROP,
VOICE_SAMPLE), "0.00%")),
    5, IF(ISBLANK(VOICE_BLOCK), BLANK(), FORMAT(VOICE_BLOCK, "#,##0")),
    6, IF(ISBLANK(VOICE_DROP), BLANK(), FORMAT(VOICE_DROP, "#,##0")),
    7, IF(ISBLANK(VOICE_DROP), BLANK(), FORMAT(DIVIDE(VOICE_DROP,
VOICE_SAMPLE - VOICE_BLOCK), "0.00%")),
    8, IF(ISBLANK(VOICE_COMPLETE + VOICE_DROP), BLANK(),
FORMAT(VOICE_COMPLETE + VOICE_DROP, "#,##0")),
    9, IF(ISBLANK(VOICE_SAMPLE) || ISBLANK(VOICE_COMPLETE), "0.00%",
FORMAT(DIVIDE(VOICE_COMPLETE, VOICE_SAMPLE), "0.00%")),
    10, IF(ISBLANK(VOICE_BAD), BLANK(), FORMAT(VOICE_BAD, "#,##0")),
    11, IF(ISBLANK(VOICE_SAMPLE) || ISBLANK(VOICE_BAD), "0.00%",
FORMAT(DIVIDE(VOICE_BAD, VOICE_SAMPLE), "0.00%")),
    0
)

```



```

TUC_voc =
VAR VOICE_SAMPLE = CALCULATE(COUNTAX(FILTER('voice database','voice
database'[TYPE] = "Voice Successful" && 'voice database'[operator1] = "TRUE-H" &&
'voice database'[TP] in {"Complete","Drop","Block"}),'voice database'[IMEI]))
VAR VOICE_COMPLETE = CALCULATE(COUNTAX(FILTER('voice
database','voice database'[TYPE] = "Voice Successful" && 'voice database'[operator1] =
"TRUE-H" && 'voice database'[TP] = "Complete"),'voice database'[IMEI]))
VAR VOICE_BLOCK = CALCULATE(COUNTAX(FILTER('voice database','voice
database'[TYPE] = "Voice Successful" && 'voice database'[operator1] = "TRUE-H" &&
'voice database'[TP] = "Block"),'voice database'[IMEI]))
VAR VOICE_DROP = CALCULATE(COUNTAX(FILTER('voice database','voice
database'[TYPE] = "Voice Successful" && 'voice database'[operator1] = "TRUE-H" &&
'voice database'[TP] = "Drop"),'voice database'[IMEI]))
VAR VOICE_BAD = CALCULATE(COUNTAX(FILTER('voice database','voice
database'[TYPE] = "Voice Successful" && 'voice database'[operator1] = "TRUE-H" &&
'voice database'[TP] = "Bad Attempt"),'voice database'[IMEI]))
RETURN
SWITCH(
    SELECTEDVALUE('QoS_Parameter(Voice)'[Index]),
    1, BLANK(),
    2, IF(ISBLANK(VOICE_SAMPLE), BLANK(), FORMAT(VOICE_SAMPLE,
"#,##0")),
    3, IF(ISBLANK(VOICE_COMPLETE), BLANK(), FORMAT(VOICE_COMPLETE,
"#,##0")),
    4, IF(ISBLANK(VOICE_SAMPLE) || ISBLANK(VOICE_COMPLETE +
VOICE_DROP), "0.00%", FORMAT(DIVIDE(VOICE_COMPLETE + VOICE_DROP,
VOICE_SAMPLE), "0.00%")),
    5, IF(ISBLANK(VOICE_BLOCK), BLANK(), FORMAT(VOICE_BLOCK, "#,##0")),
    6, IF(ISBLANK(VOICE_DROP), BLANK(), FORMAT(VOICE_DROP, "#,##0")),
    7, IF(ISBLANK(VOICE_DROP), BLANK(), FORMAT(DIVIDE(VOICE_DROP,
VOICE_SAMPLE - VOICE_BLOCK), "0.00%")),

```

```

8, IF(ISBLANK(VOICE_COMPLETE + VOICE_DROP), BLANK(),
FORMAT(VOICE_COMPLETE + VOICE_DROP, "#,##0")),
9, IF(ISBLANK(VOICE_SAMPLE) || ISBLANK(VOICE_COMPLETE), "0.00%",
FORMAT(DIVIDE(VOICE_COMPLETE, VOICE_SAMPLE), "0.00%")),
10, IF(ISBLANK(VOICE_BAD), BLANK(), FORMAT(VOICE_BAD, "#,##0")),
11, IF(ISBLANK(VOICE_SAMPLE) || ISBLANK(VOICE_BAD), "0.00%",
FORMAT(DIVIDE(VOICE_BAD, VOICE_SAMPLE), "0.00%")),
0
)

```

### ส่วนของการแสดงผลของการบริการประเภทเสียง

```

CF_AWN_voc =
VAR VOICE_SAMPLE = CALCULATE(COUNTROWS('Database'),
'Database'[TYPE] = "Voice Successful", 'Database'[operator1] = "TH GSM")
VAR VOICE_COMPLETE = CALCULATE(COUNTROWS('Database'),
'Database'[TYPE] = "Voice Successful", 'Database'[operator1] = "TH GSM", 'Database'[TP]
= "Complete")
VAR VOICE_BLOCK = CALCULATE(COUNTROWS('Database'), 'Database'[TYPE]
= "Voice Successful", 'Database'[operator1] = "TH GSM", 'Database'[TP] = "Block")
VAR VOICE_DROP = CALCULATE(COUNTROWS('Database'), 'Database'[TYPE]
= "Voice Successful", 'Database'[operator1] = "TH GSM", 'Database'[TP] = "Drop")
VAR VOICE_BAD = CALCULATE(COUNTROWS('Database'), 'Database'[TYPE] =
"Voice Successful", 'Database'[operator1] = "TH GSM", 'Database'[TP] = "Bad")
RETURN
SWITCH(
SELECTEDVALUE('QoS_Parameter(Voice)'[Index]),
// ให้ success call setup >= 90%
4, IF(DIVIDE(VOICE_COMPLETE + VOICE_DROP, VOICE_SAMPLE) >= 0.9 &&
DIVIDE(VOICE_COMPLETE + VOICE_DROP, VOICE_SAMPLE) <= 1.0, "#C1E1C1",
IF(DIVIDE(VOICE_COMPLETE + VOICE_DROP, VOICE_SAMPLE) < 0.9,
"#FFB6B6", BLANK()))),

```

```

// drop call rate <= 2%
7, IF(DIVIDE(VOICE_DROP, VOICE_SAMPLE - VOICE_BLOCK) >= 0.9 &&
DIVIDE(VOICE_DROP, VOICE_SAMPLE - VOICE_BLOCK) <= 1.0, "#FFB6B6",
IF(DIVIDE(VOICE_DROP, VOICE_SAMPLE - VOICE_BLOCK) < 0.9, "#C1E1C1",
BLANK())),
// successful call ratio >= 90%
9, IF(DIVIDE(VOICE_COMPLETE, VOICE_SAMPLE) >= 0.9 &&
DIVIDE(VOICE_COMPLETE, VOICE_SAMPLE) <= 1.0, "#C1E1C1",
IF(DIVIDE(VOICE_COMPLETE, VOICE_SAMPLE) < 0.9, "#FFB6B6", BLANK())),
// ให้ bad call rate <= 2%
11, IF(DIVIDE(VOICE_BAD, VOICE_SAMPLE) >= 0.9 && DIVIDE(VOICE_BAD,
VOICE_SAMPLE) <= 1.0, "#FFB6B6",
IF(DIVIDE(VOICE_BAD, VOICE_SAMPLE) < 0.9, "#C1E1C1", BLANK()))
)

```

```

CF_CAT_voc =
VAR VOICE_SAMPLE = CALCULATE(COUNTROWS('Database'),
'Database'[TYPE] = "Voice Successful", 'Database'[operator1] = "my")
VAR VOICE_COMPLETE = CALCULATE(COUNTROWS('Database'),
'Database'[TYPE] = "Voice Successful", 'Database'[operator1] = "my", 'Database'[TP] =
"Complete")
VAR VOICE_BLOCK = CALCULATE(COUNTROWS('Database'), 'Database'[TYPE]
= "Voice Successful", 'Database'[operator1] = "my", 'Database'[TP] = "Block")
VAR VOICE_DROP = CALCULATE(COUNTROWS('Database'), 'Database'[TYPE]
= "Voice Successful", 'Database'[operator1] = "my", 'Database'[TP] = "Drop")
VAR VOICE_BAD = CALCULATE(COUNTROWS('Database'), 'Database'[TYPE] =
"Voice Successful", 'Database'[operator1] = "my", 'Database'[TP] = "Bad")
RETURN
SWITCH(
SELECTEDVALUE('QoS_Parameter(Voice)')[Index]),
// ให้ success call setup >= 90%

```

```

4, IF(DIVIDE(VOICE_COMPLETE + VOICE_DROP, VOICE_SAMPLE) >= 0.9 &&
DIVIDE(VOICE_COMPLETE + VOICE_DROP, VOICE_SAMPLE) <= 1.0, "#C1E1C1",
IF(DIVIDE(VOICE_COMPLETE + VOICE_DROP, VOICE_SAMPLE) < 0.9,
"#FFB6B6", BLANK()))),
// drop call rate <= 2%
7, IF(DIVIDE(VOICE_DROP, VOICE_SAMPLE - VOICE_BLOCK) >= 0.9 &&
DIVIDE(VOICE_DROP, VOICE_SAMPLE - VOICE_BLOCK) <= 1.0, "#FFB6B6",
IF(DIVIDE(VOICE_DROP, VOICE_SAMPLE - VOICE_BLOCK) < 0.9, "#C1E1C1",
BLANK()))),
// successful call ratio >= 90%
9, IF(DIVIDE(VOICE_COMPLETE, VOICE_SAMPLE) >= 0.9 &&
DIVIDE(VOICE_COMPLETE, VOICE_SAMPLE) <= 1.0, "#C1E1C1",
IF(DIVIDE(VOICE_COMPLETE, VOICE_SAMPLE) < 0.9, "#FFB6B6", BLANK()))),
// 90% bad call rate <= 2%
11, IF(DIVIDE(VOICE_BAD, VOICE_SAMPLE) >= 0.9 && DIVIDE(VOICE_BAD,
VOICE_SAMPLE) <= 1.0, "#FFB6B6",
IF(DIVIDE(VOICE_BAD, VOICE_SAMPLE) < 0.9, "#C1E1C1", BLANK()))
)

CF_DTN_voc =
VAR VOICE_SAMPLE = CALCULATE(COUNTROWS('Database'),
'Database'[TYPE] = "Voice Successful", 'Database'[operator1] = "DTAC")
VAR VOICE_COMPLETE = CALCULATE(COUNTROWS('Database'),
'Database'[TYPE] = "Voice Successful", 'Database'[operator1] = "DTAC", 'Database'[TP] =
"Complete")
VAR VOICE_BLOCK = CALCULATE(COUNTROWS('Database'), 'Database'[TYPE]
= "Voice Successful", 'Database'[operator1] = "DTAC", 'Database'[TP] = "Block")
VAR VOICE_DROP = CALCULATE(COUNTROWS('Database'), 'Database'[TYPE]
= "Voice Successful", 'Database'[operator1] = "DTAC", 'Database'[TP] = "Drop")
VAR VOICE_BAD = CALCULATE(COUNTROWS('Database'), 'Database'[TYPE] =
"Voice Successful", 'Database'[operator1] = "DTAC", 'Database'[TP] = "Bad")

```

```

RETURN
SWITCH(
    SELECTEDVALUE('QoS_Parameter(Voice)'[Index]),
    // ให้ success call setup >= 90%
    4, IF(DIVIDE(VOICE_COMPLETE + VOICE_DROP, VOICE_SAMPLE) >= 0.9 &&
DIVIDE(VOICE_COMPLETE + VOICE_DROP, VOICE_SAMPLE) <= 1.0, "#C1E1C1",
    IF(DIVIDE(VOICE_COMPLETE + VOICE_DROP, VOICE_SAMPLE) < 0.9,
"#FFB6B6", BLANK())),
    // drop call rate <= 2%
    7, IF(DIVIDE(VOICE_DROP, VOICE_SAMPLE - VOICE_BLOCK) >= 0.9 &&
DIVIDE(VOICE_DROP, VOICE_SAMPLE - VOICE_BLOCK) <= 1.0, "#FFB6B6",
    IF(DIVIDE(VOICE_DROP, VOICE_SAMPLE - VOICE_BLOCK) < 0.9, "#C1E1C1",
BLANK())),
    // successful call ratio >= 90%
    9, IF(DIVIDE(VOICE_COMPLETE, VOICE_SAMPLE) >= 0.9 &&
DIVIDE(VOICE_COMPLETE, VOICE_SAMPLE) <= 1.0, "#C1E1C1",
    IF(DIVIDE(VOICE_COMPLETE, VOICE_SAMPLE) < 0.9, "#FFB6B6", BLANK())),
    // ให้ bad call rate <= 2%
    11, IF(DIVIDE(VOICE_BAD, VOICE_SAMPLE) >= 0.9 && DIVIDE(VOICE_BAD,
VOICE_SAMPLE) <= 1.0, "#FFB6B6",
    IF(DIVIDE(VOICE_BAD, VOICE_SAMPLE) < 0.9, "#C1E1C1", BLANK()))
)

CF_TOT_voc =
VAR    VOICE_SAMPLE    =    CALCULATE(COUNTROWS('Database'),
'Database'[TYPE] = "Voice Successful", 'Database'[operator1] = "TOT")
VAR    VOICE_COMPLETE    =    CALCULATE(COUNTROWS('Database'),
'Database'[TYPE] = "Voice Successful", 'Database'[operator1] = "TOT", 'Database'[TP] =
"Complete")
VAR VOICE_BLOCK = CALCULATE(COUNTROWS('Database'), 'Database'[TYPE]
= "Voice Successful", 'Database'[operator1] = "TOT", 'Database'[TP] = "Block")

```

```

VAR VOICE_DROP = CALCULATE(COUNTROWS('Database'), 'Database'[TYPE]
= "Voice Successful", 'Database'[operator1] = "TOT", 'Database'[TP] = "Drop")

VAR VOICE_BAD = CALCULATE(COUNTROWS('Database'), 'Database'[TYPE] =
"Voice Successful", 'Database'[operator1] = "TOT", 'Database'[TP] = "Bad")

RETURN

SWITCH(
    SELECTEDVALUE('QoS_Parameter(Voice)'[Index]),
    // ให้ success call setup >= 90%
    4, IF(DIVIDE(VOICE_COMPLETE + VOICE_DROP, VOICE_SAMPLE) >= 0.9 &&
DIVIDE(VOICE_COMPLETE + VOICE_DROP, VOICE_SAMPLE) <= 1.0, "#C1E1C1",
    IF(DIVIDE(VOICE_COMPLETE + VOICE_DROP, VOICE_SAMPLE) < 0.9,
"#FFB6B6", BLANK()))),
    // drop call rate <= 2%
    7, IF(DIVIDE(VOICE_DROP, VOICE_SAMPLE - VOICE_BLOCK) >= 0.9 &&
DIVIDE(VOICE_DROP, VOICE_SAMPLE - VOICE_BLOCK) <= 1.0, "#FFB6B6",
    IF(DIVIDE(VOICE_DROP, VOICE_SAMPLE - VOICE_BLOCK) < 0.9, "#C1E1C1",
BLANK()))),
    // successful call ratio >= 90%
    9, IF(DIVIDE(VOICE_COMPLETE, VOICE_SAMPLE) >= 0.9 &&
DIVIDE(VOICE_COMPLETE, VOICE_SAMPLE) <= 1.0, "#C1E1C1",
    IF(DIVIDE(VOICE_COMPLETE, VOICE_SAMPLE) < 0.9, "#FFB6B6", BLANK()))),
    // ให้ bad call rate <= 2%
    11, IF(DIVIDE(VOICE_BAD, VOICE_SAMPLE) >= 0.9 && DIVIDE(VOICE_BAD,
VOICE_SAMPLE) <= 1.0, "#FFB6B6",
    IF(DIVIDE(VOICE_BAD, VOICE_SAMPLE) < 0.9, "#C1E1C1", BLANK()))
)

CF_TRUE_voc =
VAR VOICE_SAMPLE = CALCULATE(COUNTROWS('Database'),
'Database'[TYPE] = "Voice Successful", 'Database'[operator1] = "TRUE-H")

```

```

VAR    VOICE_COMPLETE    =    CALCULATE(COUNTROWS('Database'),
'Database'[TYPE] = "Voice Successful", 'Database'[operator1] = "TRUE-H", 'Database'[TP]
= "Complete")

VAR VOICE_BLOCK = CALCULATE(COUNTROWS('Database'), 'Database'[TYPE]
= "Voice Successful", 'Database'[operator1] = "TRUE-H", 'Database'[TP] = "Block")

VAR VOICE_DROP = CALCULATE(COUNTROWS('Database'), 'Database'[TYPE]
= "Voice Successful", 'Database'[operator1] = "TRUE-H", 'Database'[TP] = "Drop")

VAR VOICE_BAD = CALCULATE(COUNTROWS('Database'), 'Database'[TYPE] =
"Voice Successful", 'Database'[operator1] = "TRUE-H", 'Database'[TP] = "Bad")

RETURN

SWITCH(
    SELECTEDVALUE('QoS_Parameter(Voice)')[Index]),
    // ให้ success call setup >= 90%
    4, IF(DIVIDE(VOICE_COMPLETE + VOICE_DROP, VOICE_SAMPLE) >= 0.9 &&
DIVIDE(VOICE_COMPLETE + VOICE_DROP, VOICE_SAMPLE) <= 1.0, "#C1E1C1",
    IF(DIVIDE(VOICE_COMPLETE + VOICE_DROP, VOICE_SAMPLE) < 0.9,
"#FFB6B6", BLANK())),
    // drop call rate <= 2%
    7, IF(DIVIDE(VOICE_DROP, VOICE_SAMPLE - VOICE_BLOCK) >= 0.9 &&
DIVIDE(VOICE_DROP, VOICE_SAMPLE - VOICE_BLOCK) <= 1.0, "#FFB6B6",
    IF(DIVIDE(VOICE_DROP, VOICE_SAMPLE - VOICE_BLOCK) < 0.9, "#C1E1C1",
BLANK())),
    // successful call ratio >= 90%
    9, IF(DIVIDE(VOICE_COMPLETE, VOICE_SAMPLE) >= 0.9 &&
DIVIDE(VOICE_COMPLETE, VOICE_SAMPLE) <= 1.0, "#C1E1C1",
    IF(DIVIDE(VOICE_COMPLETE, VOICE_SAMPLE) < 0.9, "#FFB6B6", BLANK())),
    // ให้ bad call rate <= 2%
    11, IF(DIVIDE(VOICE_BAD, VOICE_SAMPLE) >= 0.9 && DIVIDE(VOICE_BAD,
VOICE_SAMPLE) <= 1.0, "#FFB6B6",
    IF(DIVIDE(VOICE_BAD, VOICE_SAMPLE) < 0.9, "#C1E1C1", BLANK()))
)

```