

การพัฒนาระบบเสียงรอสายและเลิกไม่รับสาย ในโครงข่าย VoIP
ตามสถาปัตยกรรม NGN



บุญรัตน์ ขวัญคุณ

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิศวกรรมคอมพิวเตอร์และโทรคมนาคม คณะวิศวกรรมศาสตร์ มหาวิทยาลัยธุรกิจบัณฑิตย์

พ.ศ. 2555

**Development of Ring Back Tone and Call Screening Service System in VoIP
Network complied to Next Generation Network Architecture**



Bunyarat Kwandussadee

**A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Engineering
Department of Computer and Telecommunication Engineering
Faculty of Engineering, Dhurakij Pundit University**

2012

กิตติกรรมประกาศ

วิทยานิพนธ์นี้สำเร็จลุล่วงไปได้ด้วยความกรุณาเป็นอย่างยิ่งจาก อาจารย์ ดร.ชนัญ จารุวิทย์โกวิท อาจารย์ที่ปรึกษาร่วมวิทยานิพนธ์ ที่คอยให้คำแนะนำ ให้คำปรึกษา ตลอดจนแนวทางในการแก้ไขปัญหาต่างๆ ขอขอบคุณ อาจารย์ ดร.ชัยพร เขมะภักตะพันธ์ อาจารย์ที่ปรึกษา วิทยานิพนธ์ที่ให้ข้อคิดเห็นและคำแนะนำที่เป็นประโยชน์ต่องานวิจัย และเอาใจใส่นักศึกษาเสมอมา

ขอขอบคุณ รศ.ดร.ไพบูรณ์ พงษ์สุนันท์ ดร.ประศาสน์ จันทราทิพย์ กรรมการสอบ วิทยานิพนธ์ ที่สละเวลามาเป็นกรรมการสอบวิทยานิพนธ์ และให้คำแนะนำที่เป็นประโยชน์ต่อ งานวิจัย และขอบคุณเจ้าหน้าที่ทุกท่านที่ช่วยดำเนินเรื่องต่างๆ ให้เป็นอย่างดี

ขอขอบคุณรุ่นพี่ คุณ ประณตพล ดลคูติตา ที่คอยให้คำแนะนำในการแก้ปัญหาตลอด ขอขอบคุณเพื่อนๆ ร่วมรุ่น น้องๆ ทุกๆ คน ที่คอยช่วยเหลือกันมาตลอด

ขอขอบคุณมหาวิทยาลัยธุรกิจบัณฑิตย์

ท้ายสุดนี้ขอขอบคุณ คุณพ่อ คุณแม่ ตลอดจนคนในครอบครัวของผู้วิจัย ที่คอยให้ กำลังใจ และสนับสนุนผู้วิจัยในทุกๆ ด้าน ตลอดระยะเวลาการศึกษาจนสำเร็จการศึกษา

บุญรัตน์ ขวัญคุณฤ์

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	ฅ
บทคัดย่อภาษาอังกฤษ	ง
กิตติกรรมประกาศ	จ
สารบัญตาราง	ซ
สารบัญรูป	ฌ
บทที่	
1. บทนำ	1
1.1 ความเป็นมาและความสำคัญของปัญหา	1
1.2 วัตถุประสงค์ของงานวิจัย	4
1.3 ขอบเขตงานวิจัย	4
1.4 ประโยชน์ที่คาดว่าจะได้รับ	6
2. ทฤษฎี และงานวิจัยที่เกี่ยวข้อง	7
2.1 ความรู้พื้นฐานของระบบโครงข่าย NGN	7
2.2 ความรู้พื้นฐานของบริการเสียงเพลงรอสายของระบบ โทรศัพท์มือถือปัจจุบัน	10
2.3 ความรู้พื้นฐานของบริการเลือกไม่รับสาย ของระบบ โทรศัพท์มือถือปัจจุบัน	19
2.4 ผลงานวิจัย/ผลิตภัณฑ์ ที่เกี่ยวข้อง	20
3. การออกแบบและพัฒนาระบบ	23
3.1 แนวทางการวิจัยและพัฒนา	23
3.2 เครื่องมือที่ใช้ในงานวิจัย	24
3.3 แผนการดำเนินงาน	25
3.4 ขั้นตอนและวิธีดำเนินงาน	26
4. การทดสอบระบบ	45
5. สรุปผลและข้อเสนอแนะ	73
5.1 สรุปผลการวิจัย	73
5.2 ข้อจำกัดของระบบ	75
5.3 ข้อเสนอแนะ	76

สารบัญ (ต่อ)

	หน้า
บรรณานุกรม	77
ภาคผนวก	81
ประวัติผู้เขียน	99

สารบัญตาราง

ตารางที่	หน้า
2.1 แสดงการเปรียบเทียบคุณลักษณะของงานวิจัยและบริการที่เกี่ยวข้องกับ วิทยานิพนธ์นี้.....	22
3.1 แผนการดำเนินงาน.....	26
3.2 ตาราง cs.....	39
3.3 ตาราง rbt	39
3.4 ตาราง sipusers	40
3.5 ตาราง application_table	41
3.6 ตาราง myalbum.....	41
3.7 ตาราง device	42
4.1 ผลการทดสอบการติดตั้งเครื่อง IP-PBX ด้วย Asterisk 1.6.....	49
4.2 ผลการทดสอบการติดตั้งเครื่อง IP-PBX ด้วย Asterisk 1.8.....	50
4.3 ผลการทดสอบการกำหนดเสียงเพลงรอสายในระบบ	51
4.4 ผลการทดสอบการกำหนดเสียงตอบรับอัตโนมัติในระบบ.....	53
4.5 ผลการทดสอบการตั้งค่าผู้ใช้งาน	54
4.6 ผลการทดสอบการตั้งค่าเพลงในอัลบั้มตัวเอง	56
4.7 ผลการทดสอบการตั้งค่าเสียงเพลงรอสายสำหรับเบอร์พิเศษ	58
4.8 ผลการทดสอบการตั้งค่าเลือกไม่รับสายสำหรับเบอร์พิเศษ.....	60
4.9 สรุปผลการทดสอบการใช้ทรัพยากรของเครื่อง Application server.....	67
4.10 ตารางที่ 4.10 สรุปผลการทดสอบการใช้ทรัพยากรของเครื่อง Asterisk 1.8.3.....	71
5.1 สรุปผลการทำงานตามขอบเขตส่วนบริการเสริมสำหรับผู้ทั่วไป	74
5.2 สรุปผลการทำงานตามขอบเขตส่วนการกำหนดค่าในระบบ สำหรับผู้ดูแลระบบ.....	75

สารบัญรูป

รูปที่	หน้า
1.1 ระดับชั้นการทำงานของงานวิจัยที่ผ่านตามสถาปัตยกรรม NGN	2
1.2 ระดับชั้นการทำงานของงานวิจัยที่นำเสนอตามสถาปัตยกรรม NGN	3
2.1 สถาปัตยกรรมโครงข่าย NGN	9
2.2 แสดงการเชื่อมต่อของระบบเพลงรอสาย ในขณะที่เส้นปะแสดงการทำงานของระบบ เสียงเพลงรอสาย	11
2.3 แสดงรายละเอียดลำดับการทำงานของระบบเสียงเพลงรอสาย.....	12
2.4 แสดงการเชื่อมต่อระบบเสียงเพลงรอสายเข้ากับชุมสายของผู้ให้บริการปลายทาง.....	14
2.5 แสดงลำดับการทำงานของเสียงเพลงรอสายที่ชุมสายปลายทาง	15
2.6 การตั้งค่าเพลงพิเศษคนพิเศษ	17
2.7 การตั้งค่าเปลี่ยนอารมณ์ตามเวลา	17
2.8 การตั้งค่าเพลงพิเศษ วันพิเศษ	18
2.9 โครงสร้างการทำงานของบริการเลือกไม่รับสาย	19
2.10 การตั้งค่า Call screening.....	20
2.11 ภาพรวมการทำงานของระบบสนับสนุนฯ.....	21
3.1 ภาพรวมของระบบตามสถาปัตยกรรมโครงข่าย NGN	27
3.2 การทำงานของไฟล์ SIP.CONF และ EXTENSIONS.CONF ตามสถาปัตยกรรม NGN	28
3.3 module ต่างๆ ที่งานวิจัยนี้ได้พัฒนาขึ้น	29
3.4 แผนภาพส่วนการกำหนดค่าการทำงานของระบบ	31
3.5 แผนภาพส่วนควบคุมการใช้งาน	34
3.6 การทำงานของระบบบริการเสียงเพลงรอสาย.....	35
3.7 การทำงานของระบบบริการเลือกไม่รับสาย.....	37
3.8 ความสัมพันธ์ระหว่างตาราง	43
4.1 ภาพรวมการทดสอบระบบ	45
4.2 แสดงตัวอย่างหน้าจอการตั้งค่าการใช้งานเครื่อง IP-PBX เบื้องต้น	48
4.3 แสดงข้อมูลเครื่อง IP-PBX ทั้งหมดในระบบ	48
4.4 แสดงไฟล์ callscreening.php.....	48

สารบัญรูป(ต่อ)

รูปที่	หน้า
4.5 แสดงไฟล์ IVR.....	49
4.6 แสดงเสียงเพลงรอสายในระบบ	51
4.7 แสดงเสียงเพลงรอสายที่ path /usr/src/rbt ของ application server	51
4.8 แสดงเสียงตอบรับอัตโนมัติในระบบ	52
4.9 แสดงเสียงตอบรับอัตโนมัติที่ path /usr/src/ivr ของ application server	52
4.10 แสดงตัวอย่างหน้าจอการตั้งค่าผู้ใช้งานในระบบ	54
4.11 แสดงข้อมูลผู้ใช้งานในระบบ	54
4.12 แสดงเสียงเพลงรอสายในอัลบั้มของตัวเอง	55
4.13 แสดงเสียงเพลงรอสายในอัลบั้มของตัวเองแยก path ที่เครื่อง IP-PBX ตาม ผู้ใช้งานแต่ละคน	55
4.14 แสดงตัวอย่างหน้าจอการตั้งค่าเสียงเพลงรอสาย	57
4.15 แสดงข้อมูลการตั้งค่าเสียงเพลงรอสาย	57
4.16 แสดงตัวอย่างหน้าจอการตั้งค่าเลือกไม่รับสาย.....	59
4.17 แสดงข้อมูลการตั้งค่าเลือกไม่รับสาย.....	59
4.18 ผลการทดสอบจากหมายเลข 5000 และ 6000 โทรหากันในช่วงเวลาและนอก ช่วงเวลา.....	61
4.19 แสดงตัวอย่างรายละเอียด CDR.....	62
4.20 ผลการทดสอบจากหมายเลข 5000 และ 6000 โทรหากันในช่วงเวลาและนอก ช่วงเวลา.....	63
4.21 หน้าจอ mailbox กรณีที่เลือกตั้งค่าแบบ VM	64
4.22 แสดงตัวอย่างรายละเอียด CDR.....	64
4.23 ค่าเริ่มต้นของ CPU และ RAM.....	66
4.24 การใช้ทรัพยากรของ CPU และ RAM กรณีมีการใช้งานในระบบพร้อมกัน 4 คู่สาย.....	66
4.25 กราฟสรุปการใช้ทรัพยากรของเครื่อง Application Server.....	67
4.26 ภาพรวมการทดสอบระบบ โดยใช้ SIPp	68
4.27 แสดงการใช้ทรัพยากร RAM ก่อนการใช้งาน โดยเรียกดูด้วย phpinfo	69

สารบัญรูป(ต่อ)

รูปที่	หน้า
4.28 แสดงการใช้ Bandwidth ก่อนการใช้งาน	69
4.29 แสดงการใช้โปรแกรม SIPp ในการสร้าง call จำนวน 30 calls	70
4.30 แสดงผลที่ได้จากโปรแกรม SIPp ในการสร้าง call จำนวน 30 calls	70
4.31 แสดงการใช้ทรัพยากร RAM ในการสร้าง call จำนวน 30 calls	70
4.32 แสดงการใช้ Bandwidth ในการสร้าง call จำนวน 30 calls.....	71
4.33 กราฟสรุปการใช้ทรัพยากร RAM ของเครื่อง Asterisk 1.8.3	72
4.34 กราฟสรุปการใช้ Bandwidth ของเครื่อง Asterisk 1.8.3	72

หัวข้อวิทยานิพนธ์	การพัฒนาระบบเสียงรอสายและเลือกไม่รับสายในโครงข่าย VoIP ตามสถาปัตยกรรม NGN
ชื่อผู้เขียน	บุญรัตน์ ขวัญคุชฎี
อาจารย์ที่ปรึกษา	ดร.ชัยพร เขมะภาคะพันธ์
อาจารย์ที่ปรึกษาร่วม	ดร.ธนัญ จารุวิทย์โกวิท
สาขาวิชา	วิศวกรรมคอมพิวเตอร์และโทรคมนาคม
ปีการศึกษา	2554

บทคัดย่อ

งานวิจัยนี้นำเสนอแนวทางการใช้งานบริการเสริมในโครงข่าย VoIP ด้วยระบบ Asterisk เพื่อให้เป็นไปตามสถาปัตยกรรม NGN โดยบริการเสริมต่าง ๆ จะถูกรวมศูนย์โดยจะติดตั้งอยู่ที่เครื่องแม่ข่ายเพียงระบบเดียว ซึ่งทำงานในระดับชั้น Service Layer ผู้ใช้งานที่ลงทะเบียนอยู่กับเครื่อง Asterisk ต่าง ๆ ในระบบสามารถเข้าถึงเพื่อใช้งานบริการเสริมดังกล่าวได้อย่างเท่าเทียมกัน โดยบริการเสริมที่ใช้เป็นกรณีศึกษาในงานวิจัยนี้ได้แก่ บริการเสียงรอสาย และบริการเลือกไม่รับสาย นอกจากนั้นงานวิจัยนี้ยังได้พัฒนาระบบจัดการผู้ใช้งาน และการตั้งค่าการใช้งานของระบบแบบรวมศูนย์ซึ่งทำให้ผู้ดูแลระบบสามารถควบคุมการทำงานของระบบได้ที่จุดเดียว โดยระบบพัฒนาด้วยโปรแกรม PHP, JavaScript และใช้ MySQL เป็นฐานข้อมูล

ผลการทดสอบพบว่า Asterisk สามารถเข้าถึงเพื่อใช้งานบริการเสริมดังกล่าวได้ ทำให้ระบบดังกล่าวมีความคล่องตัวในการใช้งาน โดยเฉพาะอย่างยิ่งระบบที่มีการใช้งานเครื่องแม่ข่าย Asterisk หลาย ๆ เครื่องพร้อมกัน จากการทดสอบการใช้บริการเสียงเพลงรอสายและเลือกไม่รับสาย ด้วยเครื่องแม่ข่าย Asterisk เวอร์ชัน 1.6 และ 1.8 พบว่าระบบที่พัฒนาสามารถทำงานได้ตามขอบเขตที่ตั้งไว้

Thesis Title	Development of Ring Back Tone and Call Screening Service System in VoIP Network complied to Next Generation Network Architecture
Author	Bunyarat Kwandussadee
Thesis Advisor	Chiyaporn Khemapatapan, Ph.D
Co-Thesis Advisor	Tanun Jaruvitayakovit, Ph.D
Department	Computer and Telecommunication Engineering
Academic Year	2011

ABSTRACT

This thesis designs and develops value added service (VAS) in VoIP network to enhance the functionality of Asterisk. Normally, Asterisk is working in the control layer according to the next generation network (NGN) architecture. This research purposes the method to add more VASs on VoIP network based on Asterisk follow the NGN architecture. So, all VASs will be centralized in the application servers located in service layer. The users that register with Asterisk servers in the system can access and utilize the applications simultaneously. The case study applications that this research develops are call screening (CS) and ring back tone (RBT). In addition, this research develops the centralized user configuration and management system. So, the administrator can control the system from one place. The system was developed by using PHP, JavaScript and MySQL as a database system.

Testing results shown that users can access to use application with multiple Asterisk servers. As the result, the system is very flexible to be implemented in the system that consists of many Asterisk servers. Testing results indicate that RBT and CS service in the developed system can work well with Asterisk server version 1.6 and 1.8. The system can fully function as the targeted purposes.

บทที่ 1

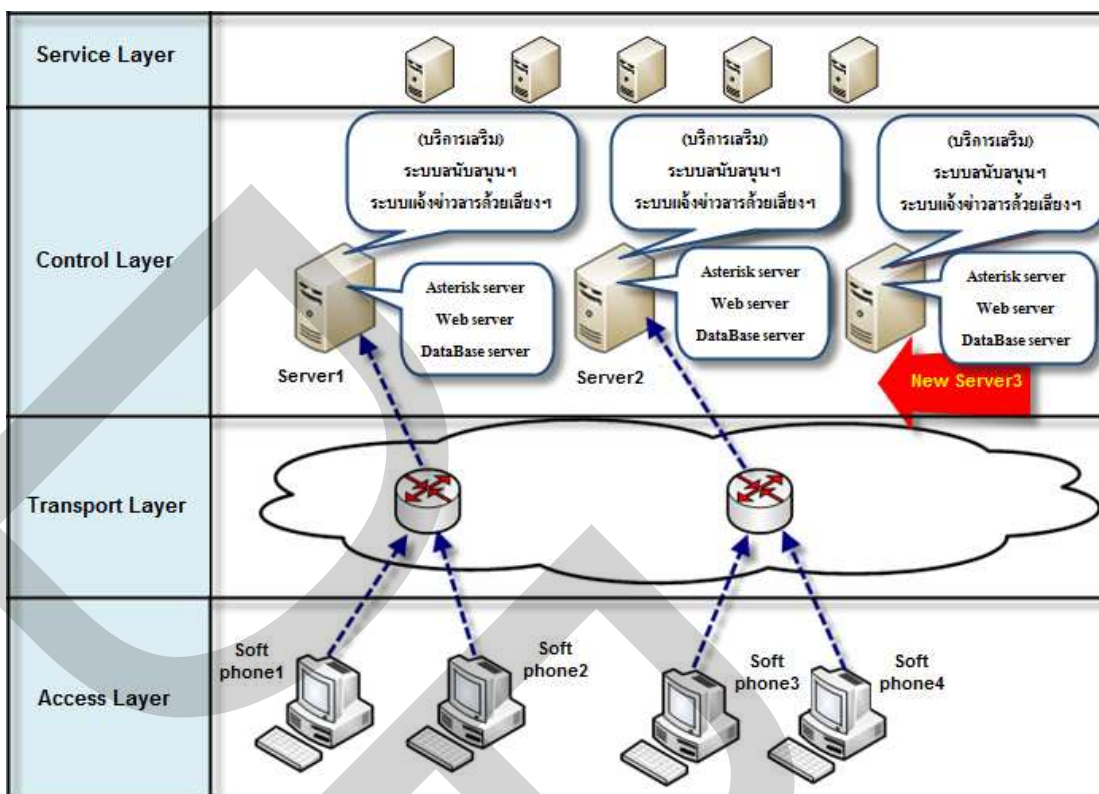
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ปัจจุบันมีการใช้งานเทคโนโลยี VoIP มากขึ้นเนื่องจากมีค่าบริการต่ำ ในขณะที่คุณภาพเสียงที่ได้ก็ใกล้เคียงกับโทรศัพท์พื้นฐานเดิม (PSTN) นอกเหนือจากอุปกรณ์ VoIP ที่มีในท้องตลาดแล้วยังมีซอฟต์แวร์ IP-PBX ระบบเปิด (open source) อยู่หลายประเภท โดยซอฟต์แวร์ที่เป็นที่นิยมใช้กันมากได้แก่ Asterisk ซึ่งมีฟังก์ชันการทำงานพื้นฐานค่อนข้างครบถ้วน และสามารถเพิ่มฟังก์ชันการทำงานให้หลากหลายมากยิ่งขึ้น จากงานวิจัยที่ผ่านมาได้มีการพัฒนาฟังก์ชันเพื่อเพิ่มประสิทธิภาพการทำงานของ Asterisk ให้ครอบคลุมและมีประสิทธิภาพในการทำงานมากยิ่งขึ้น เช่น ระบบสนับสนุนการกำหนดค่าการทำงานของ Asterisk¹ โดยใช้ฟังก์ชัน AGI (Asterisk Gateway Interface) ทำงานร่วมกับ PHP พัฒนาเป็นเว็บเพื่อใช้ในการกำหนดค่าต่างๆ ในระบบ Asterisk และการเพิ่มฟังก์ชันการทำงานของแอสเทอริกส์ด้วยพีเอชพี : กรณีสึกษาระบบแจ้งข่าวสารด้วยเสียงอัตโนมัติผ่านโทรศัพท์ไอพี² โดยใช้ AMI (Asterisk Manager Interface) เป็นตัวจัดการ API ที่ใช้สื่อสารกับ Asterisk พัฒนาเป็นเว็บไซต์ โดยมีการทำงานอยู่ในระดับชั้น Control Layer ตามสถาปัตยกรรม NGN ดังแสดงในรูปที่ 1.1

¹ ประจวบคณ ดลคุสิตา. (2552). ระบบสนับสนุนการกำหนดค่าการทำงานของไอพีพีบีเอ็กซ์แบบแอสเทอริกส์.

² ชานนทร์ อยู่ภูติมากร. (2553). การพัฒนาระบบแจ้งข่าวสารด้วยเสียงอัตโนมัติผ่านโทรศัพท์ไอพี.

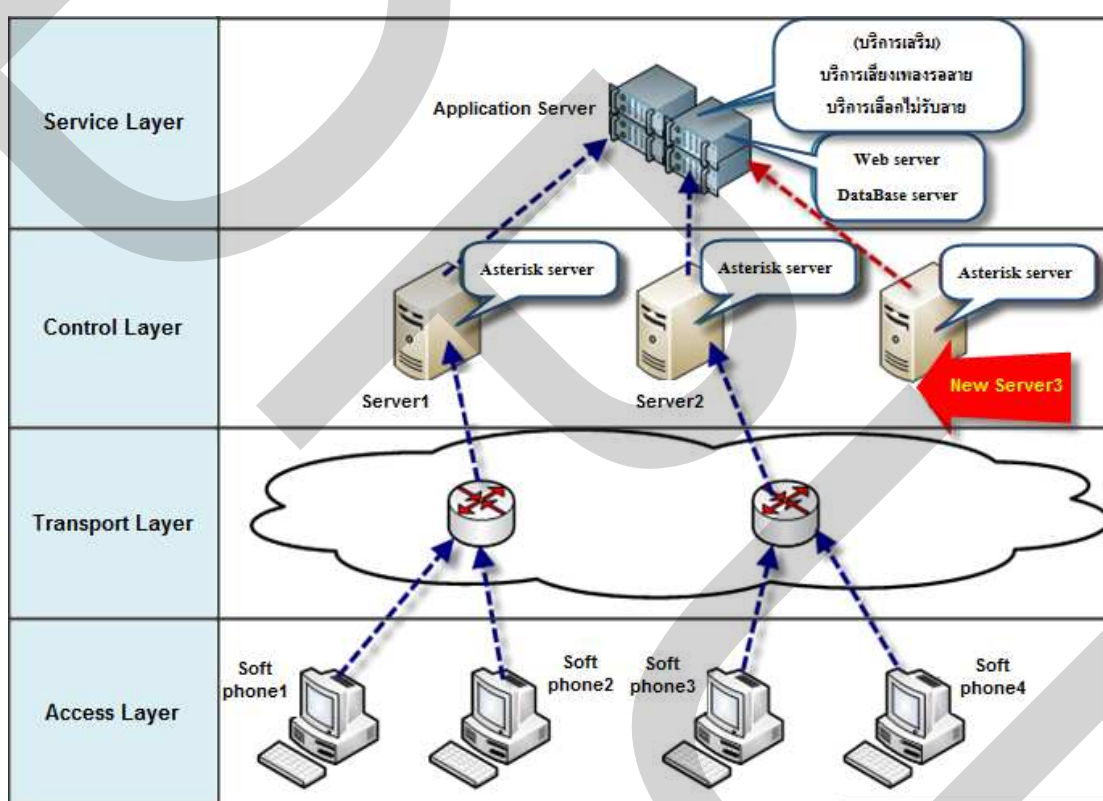


รูปที่ 1.1 ระดับชั้นการทำงานของงานวิจัยที่ผ่านมาตามสถาปัตยกรรม NGN

1. ข้อดี ของระบบบริการเสริมในงานวิจัยที่ผ่านมา
 - 1.1 การใช้งาน Asterisk สามารถทำได้ง่ายขึ้น
 - 1.2 มีฟังก์ชันการทำงานมากขึ้น
2. ข้อจำกัด ของระบบบริการเสริมในงานวิจัยที่ผ่านมา
 - 2.1 ในกรณีที่ Softphone 1 และ 2 มีการเชื่อมต่อกับ Server 1 อยู่และต้องการใช้งานระบบสนับสนุนฯและระบบแจ้งข่าวสารด้วยเสียงฯ ที่เครื่อง Server 1 จำเป็นต้องติดตั้งบริการเสริมดังกล่าวทั้ง 2 บริการ เพื่อให้ Softphone 1 และ 2 สามารถใช้งานได้ และในกรณีที่ Softphone 3 และ 4 มีการเชื่อมต่อกับ Server 2 อยู่และต้องการใช้งานระบบสนับสนุนฯและระบบแจ้งข่าวสารด้วยเสียงฯ ที่เครื่อง Server 2 จำเป็นต้องติดตั้งบริการเสริมดังกล่าวทั้ง 2 บริการ เพื่อให้ Softphone 3 และ 4 สามารถใช้งานได้ และเมื่อมี Asterisk Server 3 เพิ่มเข้ามาเพื่อให้ Softphone 3 และ 4 เปลี่ยนไปใช้งาน Server 3 ได้ ที่เครื่อง Server 3 จำเป็นต้องติดตั้งบริการเสริมดังกล่าวทั้ง 2 บริการ ซึ่งทำให้ไม่สามารถให้บริการที่สะดวก รวดเร็ว

2.2 เมื่อระบบสนับสนุนฯ และระบบแจ้งข่าวสารด้วยเสียงฯ มีการปรับปรุง หรือเปลี่ยนแปลง ผู้ดูแลระบบก็จำเป็นต้องปรับปรุงหรือเปลี่ยนแปลงที่เครื่อง Asterisk ทั้ง 3 เครื่องที่ใช้งาน

งานวิจัยนี้จึงนำเสนอการออกแบบและพัฒนาระบบบริการเสริมในโครงข่าย VoIP ตามสถาปัตยกรรม NGN โดยใช้บริการเสริมเป็นกรณีศึกษา 2 บริการ ระบบบริการเลือกไม่รับสาย (Call screening) และบริการเสียงเพลงรอสาย (Ring back tone) โดยมีการทำงานอยู่ในระดับชั้น Service Layer ตามสถาปัตยกรรม NGN ดังแสดงในรูปที่ 1.2



รูปที่ 1.2 ระดับชั้นการทำงานของงานวิจัยที่นำเสนอตามสถาปัตยกรรม NGN

3. ข้อดี ของงานวิจัยที่นำเสนอ

3.1 บริการเสียงเพลงรอสาย และบริการเลือกไม่รับสายจะถูกผนวกรวมบริการเข้าด้วยกันไว้ที่เครื่องแม่ข่ายศูนย์กลางในระดับชั้น Service Layer เพื่อง่ายต่อการดูแลรักษา ลดค่าใช้จ่าย และรองรับการพัฒนาบริการใหม่ๆที่จะเกิดขึ้นได้ในอนาคต

3.2 มีการพัฒนาเป็นเว็บไซต์ ทำให้สามารถใช้งาน Asterisk ได้ง่าย

3.3 ในกรณีที่ Softphone 1 และ 2 มีการเชื่อมต่อกับ Server 1 อยู่และต้องการใช้งานระบบบริการเสียงเพลงรอสายและบริการเลือกไม่รับสาย เครื่อง Server 1 สามารถเชื่อมต่อไปที่ Server A เพื่อให้ Softphone 1 และ 2 สามารถใช้งานได้ และในกรณีที่ Softphone 3 และ 4 มีการเชื่อมต่อกับ Server 2 อยู่และต้องการใช้งานระบบบริการเสียงเพลงรอสายและบริการเลือกไม่รับสาย เครื่อง Server 2 สามารถเชื่อมต่อไปที่ Server A เพื่อให้ Softphone 3 และ 4 สามารถใช้งานได้ และเมื่อมี Asterisk Server 3 เพิ่มเข้ามา เพื่อให้ Softphone 3 และ 4 เปลี่ยนไปใช้งาน Server 3 ได้ เครื่อง Server 3 สามารถเชื่อมต่อไปที่ Server A เพื่อให้ Softphone 3 และ 4 สามารถใช้งานได้ ทำให้สามารถให้บริการได้สะดวก รวดเร็ว

3.4 เมื่อระบบบริการเสียงเพลงรอสายและบริการเลือกไม่รับสาย มีการปรับปรุงหรือเปลี่ยนแปลง ผู้ดูแลระบบสามารถแก้ไขได้ที่เครื่อง Server A เพียงจุดเดียว

1.2 วัตถุประสงค์ของงานวิจัย

1. เพื่อออกแบบและพัฒนาระบบบริการเสริม 2 แบบคือ บริการเลือกไม่รับสาย และบริการเสียงเพลงรอสายในโครงข่าย VoIP ให้มีการทำงานอยู่ในระดับชั้น Service Layer ตามสถาปัตยกรรมของโครงข่าย NGN ได้
2. เพื่อให้เครื่องแม่ข่าย Asterisk มากกว่า 1 ระบบ สามารถเข้าถึง (access) เพื่อใช้งานระบบบริการเสริมที่รวมศูนย์อยู่ได้

1.3 ขอบเขตงานวิจัย

วิทยานิพนธ์ฉบับนี้มีขอบเขตการวิจัยดังนี้

1. ออกแบบ และพัฒนาบริการเสริมสำหรับการใช้งาน IP-PBX (เลือกใช้ Asterisk ซึ่งเป็นซอฟต์แวร์ระบบเปิด) โดยบริการเสริมดังกล่าวจะเป็นไปตามโครงสร้างของโครงข่าย NGN (Next Generation Network – NGN) กล่าวคือบริการเสริมจะถูกแยกการทำงานไปอยู่ในชั้น (layer) ที่สูงกว่าชั้น Control Layer ซึ่งจะทำให้ระบบ IP-PBX หลายๆ ระบบสามารถเข้าใช้บริการเสริมที่พัฒนาขึ้นได้เหมือนกัน
2. บริการเสริมที่จะพัฒนาในวิทยานิพนธ์นี้มี 2 บริการ ได้แก่ บริการเสียงเพลงรอสาย (Ring Back Tone – RBT) และบริการเลือกไม่รับสาย (Call screening)
3. ผู้ใช้งานในระบบมี 2 ประเภท ได้แก่ ผู้ดูแลระบบ และผู้ใช้งานทั่วไป
4. ผู้ดูแลระบบ (Administrator) สามารถ
 - 4.1 เพิ่ม/ลด ผู้ใช้งานในระบบได้

4.2 แก้ไขสิทธิ์ของผู้ใช้ได้

4.3 เพิ่ม/ลด จำนวนเสียงข้อความต้อนรับให้กับระบบสำหรับบริการ Call-Screening แบบ IVR ได้

4.4 เพิ่ม/ลด จำนวนเสียงเพลงรอสายให้กับระบบสำหรับบริการ Ring Back Tone ได้

4.5 กำหนดค่าที่จำเป็นสำหรับเครื่อง Asterisk เครื่องใหม่ผ่านหน้าเว็บได้

5. ผู้ใช้งานทั่วไป (User) สามารถใช้งานบริการเสริม โดยการกำหนดรายละเอียดดังต่อไปนี้ผ่านเว็บ

5.1 บริการเพลงรอสาย

5.1.1 สามารถเพิ่ม/ลด/แก้ไข/เรียกดู การตั้งค่าเพลงรอสายสำหรับเบอร์ที่โทรเข้ามาได้

5.1.2 สามารถกำหนดเพลงรอสายให้กับเบอร์ที่โทรเข้า ตามวัน เวลา และเทศกาลได้

5.1.3 ระบบจะมีเพลงให้เลือก (default) อย่างน้อย 5 เพลง โดยผู้ใช้งานสามารถเพิ่มเพลงใหม่เข้าไปในระบบส่วนตัวของตนเอง

5.2 บริการเลือกไม่รับสาย

5.2.1 สามารถเพิ่ม/ลด/แก้ไข/เรียกดู เบอร์ที่ไม่ต้องการรับสายได้

5.2.2 ในกรณีที่ ไม่ต้องการรับสาย ผู้ใช้งานสามารถเลือกให้ผู้โทรเข้าไปได้ยินเสียงสายไม่ว่าง (busy), โอนสายเข้าระบบฝากข้อความ (voice mail) หรือมีระบบตอบรับอัตโนมัติ (Interactive Voice Response – IVR) โดยการเลือกไม่รับสายนั้นผู้ใช้งานอาจจะกำหนดจากเบอร์โทรเข้า วัน เวลา หรือเทศกาลได้

5.2.3 สำหรับกรณีระบบ IVR ผู้ใช้งานสามารถเลือกเสียง IVR ที่มีอยู่ในระบบอย่างน้อย 5 เสียง

6. วิทยานิพนธ์ฉบับนี้จะสาธิตการใช้งานกับเครื่อง Asterisk server 2 รุ่น ได้แก่รุ่น 1.6 และ 1.8 เพื่อแสดงให้เห็นว่าผู้ใช้งานที่อยู่ภายใต้เครื่องแม่ข่ายแตกต่างกันสามารถเข้าใช้บริการเสริมที่พัฒนาขึ้นได้เหมือนกัน

7. ระบบมีการเก็บบันทึกข้อมูลการใช้งานแบบรวมศูนย์

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1. สามารถพัฒนาโปรแกรมประยุกต์เพื่อใช้กำหนดค่าเบื้องต้นให้กับ Asterisk ให้สามารถใช้งานร่วมกับเครื่องแม่ข่ายศูนย์กลางได้ แทนการกำหนดค่าแบบเดิมที่เป็นแบบ Command Line ทำให้ง่ายต่อการใช้งานในกรณีที่มีเครื่อง Asterisk ใหม่ในระบบ
2. ตัวโปรแกรมมีความง่ายต่อการใช้งาน ไม่ขึ้นกับระบบ เพราะเป็นการควบคุมผ่านเว็บเบราว์เซอร์
3. ผู้ใช้สามารถใช้งานระบบบริการเสียงเพลงรอสายและบริการเลือกไม่รับสาย ได้

บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

ในวิทยานิพนธ์นี้ มีทฤษฎีที่เกี่ยวข้องในด้านต่างๆ เพื่อช่วยในการพัฒนาระบบ คือ สถาปัตยกรรมโครงข่าย NGN, Ring Back Tone, Call Screening รวมถึงงานวิจัยและผลงานที่เกี่ยวข้อง

2.1 ความรู้พื้นฐานของระบบโครงข่าย NGN¹

Next Generation Network (NGN) โครงข่ายสื่อสารโทรคมนาคม ที่มีการใช้งานอยู่ปัจจุบัน 2 ประเภท คือ โครงข่ายให้บริการเสียงพูด (Voice Network) และโครงข่ายเพื่อให้บริการข้อมูล (Data Network) เนื่องจากการพัฒนาเทคโนโลยีการเปลี่ยนแปลงทางการตลาดในการให้บริการโทรคมนาคมที่ต้องการความรวดเร็ว รวมทั้งการดูแลบำรุงรักษาโครงข่ายชนิดต่างๆ ทำให้เกิดความสิ้นเปลือง และเป็นภาระที่จะสร้างบริการใหม่ให้ทัน และเพียงพอต่อความต้องการ ดังนั้นจึงได้เกิดโครงข่ายใหม่ที่รวมเอา Voice Network และ Data Network เข้าด้วยกัน จึงเกิด Next Generation Network (NGN) หรือ เรียกว่า Voice Data Convergence ไปในทิศทางเดียวกัน คือ สามารถทำงานร่วมกันได้เสมือนเป็นโครงข่ายเดียวกันเพื่อประโยชน์ในด้านต่างๆ เช่น ง่ายต่อการดูแลบำรุงรักษา ลดค่าใช้จ่าย และรองรับการพัฒนาบริการใหม่ๆ ที่จะเกิดขึ้นได้ในอนาคตได้อย่างรวดเร็วทันใจ และรองรับความต้องการของลูกค้าได้อย่างรวดเร็ว และความต้องการการใช้ในอนาคตได้เพิ่มขึ้นอย่างรวดเร็ว โดยเฉพาะอย่างยิ่งการเพิ่มขึ้นของ Multimedia Traffic บนอินเทอร์เน็ต ดังนั้นการให้บริการด้านโทรคมนาคมจึงมีความจำเป็นต้องเปลี่ยนแปลงทางด้านสถาปัตยกรรมโครงข่ายใหม่ เพื่อให้เกิดการบริการที่หลากหลาย มีการควบคุมจากศูนย์กลาง โดยให้โครงข่ายอยู่ใกล้ผู้ใช้มากที่สุด หรือ เรียกว่า Edge Switch การรวมโครงข่ายในลักษณะนี้ทำให้เกิดการลดต้นทุนให้ต่ำลง เพื่อทำให้เกิดความสามารถในการแข่งขัน เพิ่มความคล่องตัวในการบริหารจัดการโครงข่าย

Next Generation Network (NGN) ถูกนิยามโดย ITU-T คือโครงข่ายโทรคมนาคมแบบ packet-based ที่สามารถรองรับบริการอันหลากหลายในการสื่อสารโทรคมนาคมได้ และสามารถเชื่อมต่อใช้โครงข่ายสื่อสารความเร็วสูงได้หลากหลาย โดยมีคุณภาพของข้อมูลการสื่อสาร

¹ ปรมะสาร กุมาบุญ. (2552). จาก http://www.torakom.com/article_index.php?sub=article_show&art=21.

ที่มีประสิทธิภาพ (QoS; Quality of Service) อย่างดีในการส่งข้อมูลผ่านเครือข่ายโทรคมนาคมประเภทต่างๆ โดยไม่มีข้อจำกัดในการเข้าถึงเครือข่ายของผู้ให้บริการที่มีความแตกต่างๆ ทั้งยังรองรับการใช้งานแบบเคลื่อนที่ซึ่งรองรับให้ผู้ให้บริการสามารถเข้าใช้โครงข่ายได้ทุกหนทุกแห่ง

2.1.1 ปัจจัยที่ผลักดันให้เกิด NGN²

2.1.1.1 การเพิ่มขึ้นของข้อมูลชนิดมัลติมีเดียโดยการเพิ่มขึ้นในด้านปริมาณ และประเภทของการบริการ ดังนั้นโครงข่าย NGN สามารถรองรับแบนวิธ (Bandwidth) ที่มากขึ้น และประเภทที่เกิดขึ้นทั้ง เสียง วิดีโอ และข้อมูล

2.1.1.2 ระดับความต้องการบริการที่หลากหลาย (Multiple Degrees Services Aspects) เนื่องจากการเพิ่มขึ้นของการทำงานในด้านข้อมูล จากการใช้งานจากเครือข่ายคอมพิวเตอร์ โครงข่าย LAN และอินเทอร์เน็ต ดังนั้นโครงข่ายโทรคมนาคมจึงมีความจำเป็นในการเตรียมขีดความสามารถในการรองรับความต้องการดังกล่าว เช่น การกำหนดให้มีการรับประกันคุณภาพบริการในหลายระดับ (Multiple – Class Of Quality Of Services : QoS) การมีระดับการรับประกันความปลอดภัยของข้อมูล (Security Insurance Levels) และการกำหนดขอบข่ายการเคลื่อนย้ายอุปกรณ์ (Ranges Of Mobility) ซึ่งโครงข่าย NGN มีความสามารถในการตอบสนองลักษณะดังกล่าวได้ โดยการรวมเอา IP Technology และ ATM Technology ซึ่งเป็นทางออกของความต้องการดังกล่าว ดังนั้นผู้ให้บริการจำเป็นต้องรองรับความต้องการดังกล่าว

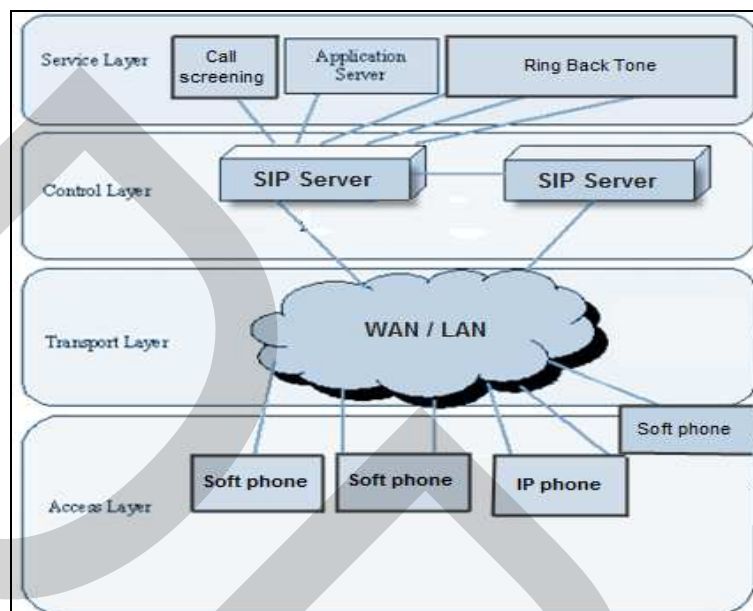
2.1.1.3 จำนวนปริมาณข้อมูลประเภท Packet เพิ่มมากขึ้น

2.1.1.4 การพัฒนาที่ก้าวหน้าของเทคโนโลยี IP, ATM และ Voice Over Packet Network เนื่องจากในปัจจุบัน ได้มีการพัฒนามาตรฐานการเชื่อมต่ออย่างสมบูรณ์ ระหว่างผู้ผลิตอุปกรณ์ กับการพัฒนาเทคโนโลยี Voice Packet Network เช่น VoIP, VoATM, VoDSL, VoIP mobile.

2.1.1.5 การพัฒนาอย่างรวดเร็วทางด้านเทคโนโลยีไร้สาย (Wireless Technology) จากการพัฒนาของโครงข่ายไร้สายได้มีการพัฒนาอย่างรวดเร็วในการรับส่งข้อมูล

² ปริญญา สายอรุณ. (2553). จาก <http://ongosk.blogspot.com/2010/07/itm-640-id.html>.

2.1.2 สถาปัตยกรรมโครงข่าย NGN³



รูปที่ 2.1 สถาปัตยกรรมโครงข่าย NGN

ที่มา : <http://www.ipv4.com/articles/general/Next-Generation-Networking.htm>

จากรูปที่ 2.1 โครงข่าย NGN สามารถแบ่งได้เป็น 4 ระดับชั้น ดังนี้

2.1.2.1 Access Layer เป็นระดับชั้นอุปกรณ์ปลายทาง (Client)

เนื่องจากสถาปัตยกรรมจะสามารถรองรับการติดต่อเข้าใช้งานหลายๆ ระบบ เช่น Enterprise Customer, Remote Office / SOHO, Resident Users, Mobile ลักษณะการรับส่ง จะเป็น Packet

2.1.2.2 Transport Layer หรือ Infrastructure จะประกอบไปด้วย Media Gateway ซึ่งทำหน้าที่เชื่อมต่อกับโครงข่าย Access หรือ Transport โดยที่ Media Gateway จะถูกควบคุมการทำงานด้วย Media Controller หรือ Softswitch ใน Control Layer

2.1.2.3 Control Layer หรือ Call Processing โดยทั่วไปจะประกอบด้วย Softswitch, Media Gateway Controller หรือ Call Agent มีหน้าที่ในการเลือกเส้นทางโดยการรองรับสัญญาณควบคุมในมาตรฐานต่างๆ ทั้งแบบ Circuit Switching และ Packet Switching เช่น

³ ประเมศวร์ กุमारบุญ. (2552). จาก http://www.torakom.com/article_index.php?sub=article_show&art=21.

Signaling System No.7 (SS7), Session Initiation Protocol (SIP), Media Gateway Control (MEGACO), Media Gateway Control Protocol (MGCP) และ H.323 และอื่นๆ และยังทำหน้าที่แปลงสัญญาณควบคุมจากมาตรฐานหนึ่งไปมาตรฐานหนึ่ง

2.1.2.4 Service Layer หรือ Applications โดยทั่วไปจะประกอบด้วยส่วนของ Application Service ซึ่งทำหน้าที่เป็นการให้บริการเสริม หรือ Voice Added ต่างๆ ซึ่งมีลักษณะคล้ายกับโครงข่าย PSTN ที่มีการใช้ร่วมกับโครงข่าย IN (Intelligence Network) ส่วนนี้จะควบคุมโดย Media Gateway Controller ที่มีอยู่ใน Softswitch

2.1.3 ข้อดีของโครงข่าย NGN

2.1.3.1 โครงสร้างของโครงข่ายจะเปลี่ยนไปจากเดิมที่แยกตามบริการแต่ละประเภท เป็นโครงข่ายหลักเดียวสำหรับบริการทุกประเภท ทั้งบริการโทรศัพท์บ้าน บริการต่อเข้าโครงข่ายอินเทอร์เน็ต หรือแม้แต่บริการติดต่อสื่อสารจากโทรศัพท์เคลื่อนที่ โดยข้อมูลจะถูกขนส่งโดยใช้โครงข่ายไอพีเป็นหลัก ซึ่งข้อมูลทุกชนิดจะถูกส่งเป็นกลุ่มข้อมูลหรือแพ็คเกจ (Packet)

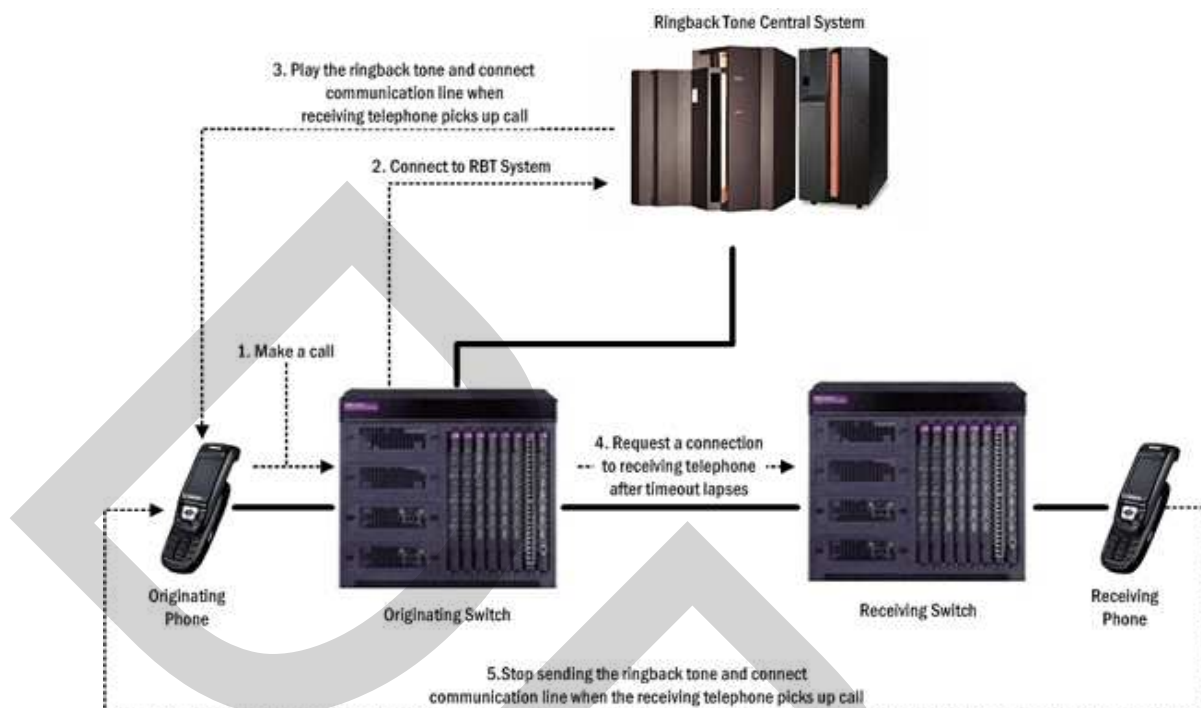
2.1.3.2 โครงข่าย NGN จะมีการแบ่งชั้นทำหน้าที่ต่างๆกันอย่างชัดเจน

2.1.3.3 จุดเชื่อมต่อต่างๆ (ระหว่างชั้นภายในโครงข่าย ระหว่างโครงข่ายกับผู้ใช้ หรือระหว่างโครงข่ายกับโครงข่ายอื่น) เป็นจุดเชื่อมต่อด้วยมาตรฐานเปิดที่ถูกกำหนดไว้ชัดเจน ทำให้การเชื่อมต่อและทำงานร่วมกันไม่มีปัญหา

2.2 ความรู้พื้นฐานของบริการเสียงเพลงรอสาย ของระบบโทรศัพท์มือถือปัจจุบัน

เสียงรอสาย (Ring Back Tone-RBT) เป็นการบ่งชี้ด้วยเสียงที่ปลายทางจะได้ยินบนสายโทรศัพท์ หลังจากต้นทางได้กดเลขหมาย และก่อนที่อีกฝ่ายจะรับสาย เสียงรอสายนี้มักจะเล่นซ้ำเพื่อให้แน่ใจว่าปลายทางรู้ว่าสายของต้นทางกำลังมีการเรียกเข้า ในปัจจุบันเสียงรอสายได้ทำให้เป็นส่วนตัวมากขึ้น โดยต้นทางสามารถเลือกเสียงรอสายได้เอง

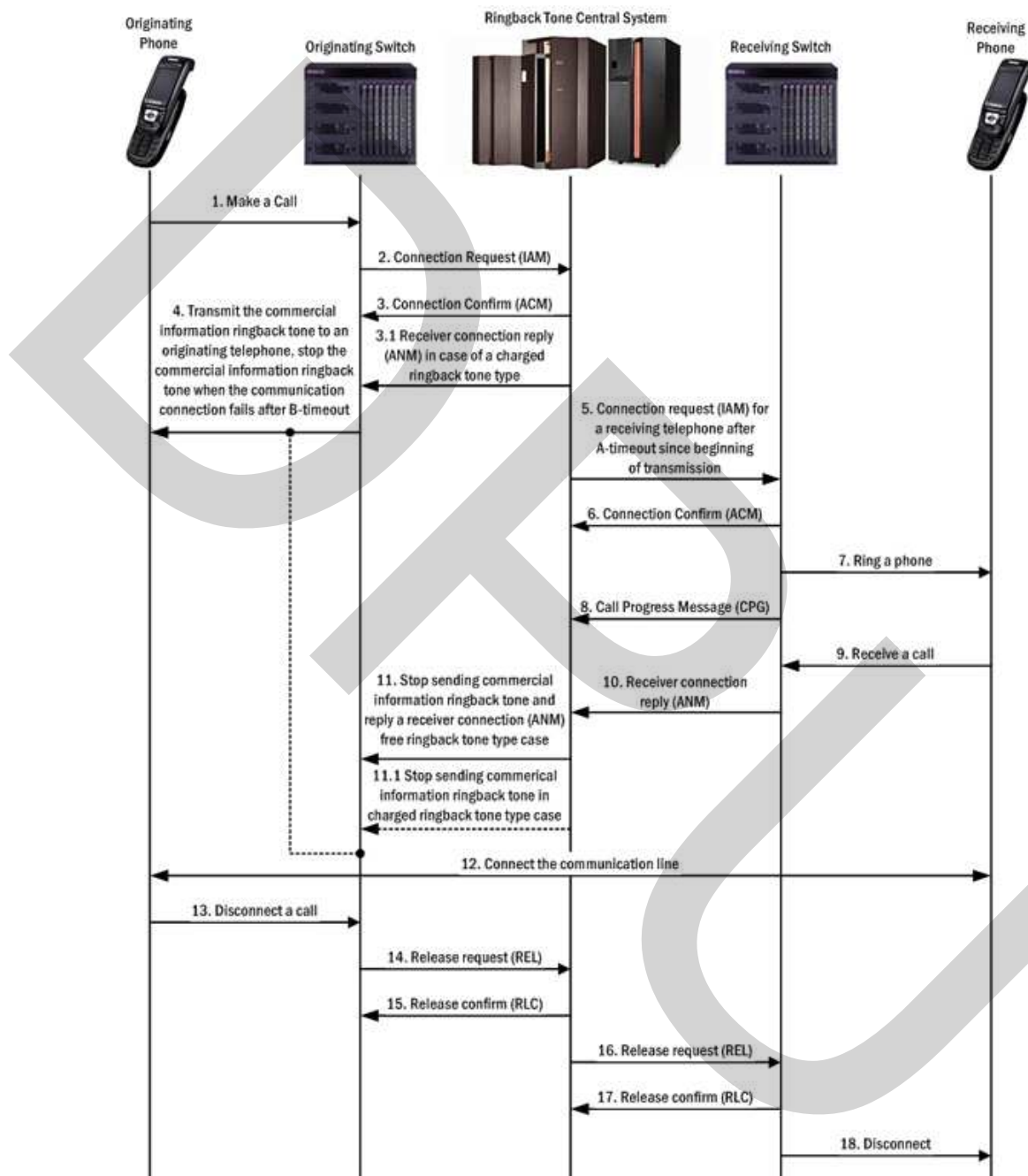
การบริการที่เรียกว่าเสียงเพลงรอสายนั้นจะเกิดขึ้นเมื่อมีการโทร ชุมสายจะทำการเชื่อมต่อไปยังระบบเสียงรอสาย เพื่อสร้างเสียงเพลงออกมา (โดยระบบจะตรวจสอบได้จากหมายเลขโทรศัพท์ของผู้โทรเข้ามา) เมื่อปลายทางรับสายระบบจะตัดการเชื่อมต่อให้เป็นแบบปกติ มีหลายวิธีการที่จะทำให้ชุมสายแทนที่เสียงรอสายแบบเดิมที่เป็นแบบ “เสียงกริ่ง” เป็นสัญญาณเสียงรอสายตามที่ลูกค้าต้องการ ซึ่งบริการเสียงเพลงรอสายนั้นจะได้รับความคุ้มครองทางกฎหมายภายในเครือข่ายของบริการนั้น ในการให้บริการนั้นจะประกอบไปด้วยส่วนสำคัญ 2 ส่วนที่เชื่อมต่อกับระบบชุมสาย คือส่วนที่เชื่อมต่อกับชุมสายต้นทาง หรือชุมสายปลายทาง และระบบเสียงเพลงรอสาย ซึ่งโครงสร้างและการทำงานดังแสดงในรูปที่ 2.2



รูปที่ 2.2 แสดงการเชื่อมต่อของระบบเพลงรอสายในขณะที่เส้นปะ แสดงการทำงานของระบบเสียงเพลงรอสาย

ที่มา : http://rbth.com/ringbacktones/ringbacktones_call_flow.htm

รายละเอียดลำดับการทำงานของระบบเสียงเพลงรอสาย ดังแสดงในรูปที่ 2.3



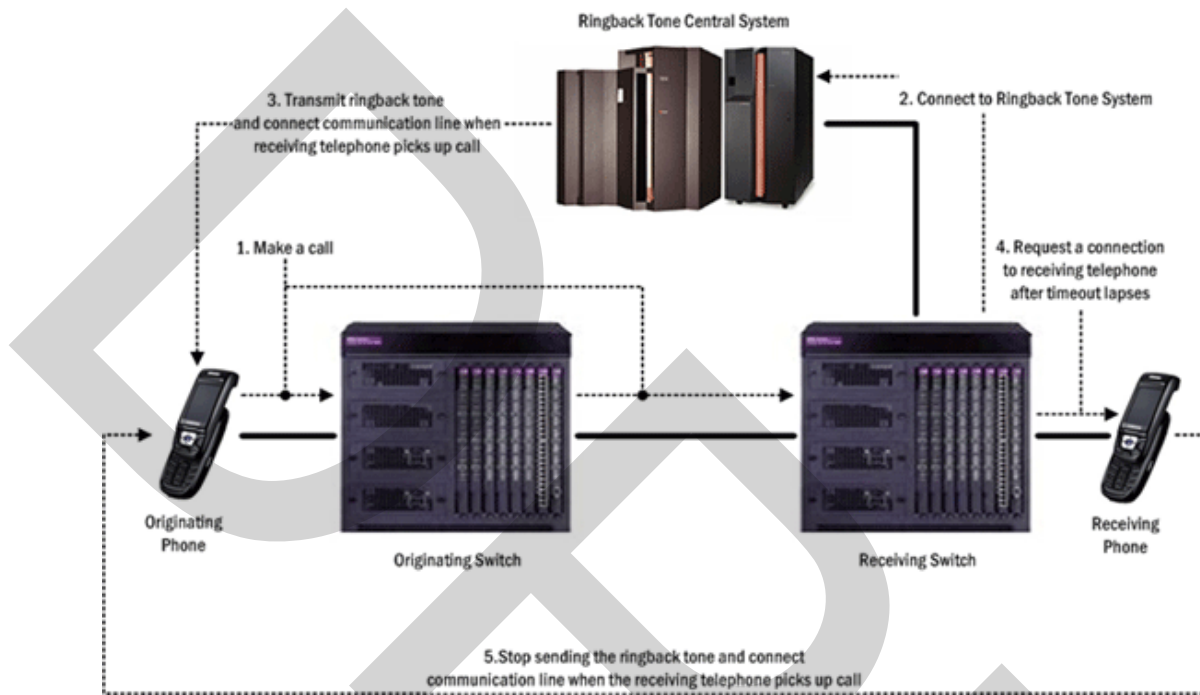
รูปที่ 2.3 แสดงรายละเอียดลำดับการทำงานของระบบเสียงเพลงรอสาย

ที่มา : http://rbth.com/ringbacktones/ringbacktones_call_flow.htm

จากรูปที่ 2.3 อธิบายการทำงานของระบบเสียงเพลงรอสาย ดังนี้

- 1) เมื่อหมายเลขต้นทางโทรออก จะมีการเชื่อมต่อกับชุมสายต้นทาง
- 2) ชุมสายต้นทางเชื่อมต่อไปที่ระบบเสียงเพลงรอสายเพื่อร้องขอการติดต่อกับชุมสายปลายทาง
- 3) ระบบเสียงเพลงรอสายตอบรับการร้องขอการติดต่อไปที่ชุมสายต้นทาง
 - 3.1) สร้างวงจรรองรับสำหรับสัญญาณเสียงในกรณีที่มีการตั้งเสียงเพลงรอสาย
- 4) ชุมสายต้นทางส่งสัญญาณเสียงเพลงรอสายไปยังหมายเลขต้นทาง และหยุดเล่นเสียงเพลงรอสายเมื่อปลายทางไม่รับสายในเวลาที่กำหนด
- 5) ระบบเสียงเพลงรอสายทำการร้องขอการติดต่อกับชุมสายปลายทาง
- 6) ชุมสายปลายทางตอบรับการร้องขอการติดต่อไปที่ระบบเสียงเพลงรอสาย
- 7) ชุมสายปลายทางส่งสัญญาณเสียงกริ่งไปที่หมายเลขปลายทาง
- 8) ชุมสายปลายทางแจ้งสถานะของหมายเลขปลายทางว่าว่างหรือไม่ไปที่ระบบเสียงเพลงรอสาย
 - 9) หมายเลขปลายทางรับสาย
 - 10) ชุมสายปลายทางสร้างวงจรรองรับสำหรับสัญญาณเสียงเพื่อการสนทนา
 - 11) ระบบเสียงเพลงรอสายหยุดการเล่นเสียงเพลงรอสายและตอบรับสัญญาณเสียงเพื่อการสนทนา
 - 11.1) หยุดการเล่นเสียงเพลงรอสายตามที่ได้ตั้งไว้
 - 12) เชื่อมต่อเพื่อสนทนา
 - 13) หมายเลขต้นทางยกเลิกการติดต่อ
 - 14) ชุมสายต้นทางส่งข้อความไปบอกที่ระบบเสียงเพลงรอสายเพื่อยกเลิกการติดต่อ
 - 15) ระบบเสียงเพลงรอสายตอบรับข้อความและยกเลิกการใช้งาน Timeslot นั้นๆ
 - 16) ระบบเสียงเพลงรอสายส่งข้อความไปบอกที่ชุมสายปลายทางเพื่อยกเลิกการติดต่อ
 - 17) ชุมสายปลายทางตอบรับข้อความและยกเลิกการใช้งาน Timeslot นั้นๆ
 - 18) ชุมสายปลายทางยกเลิกการเชื่อมต่อ

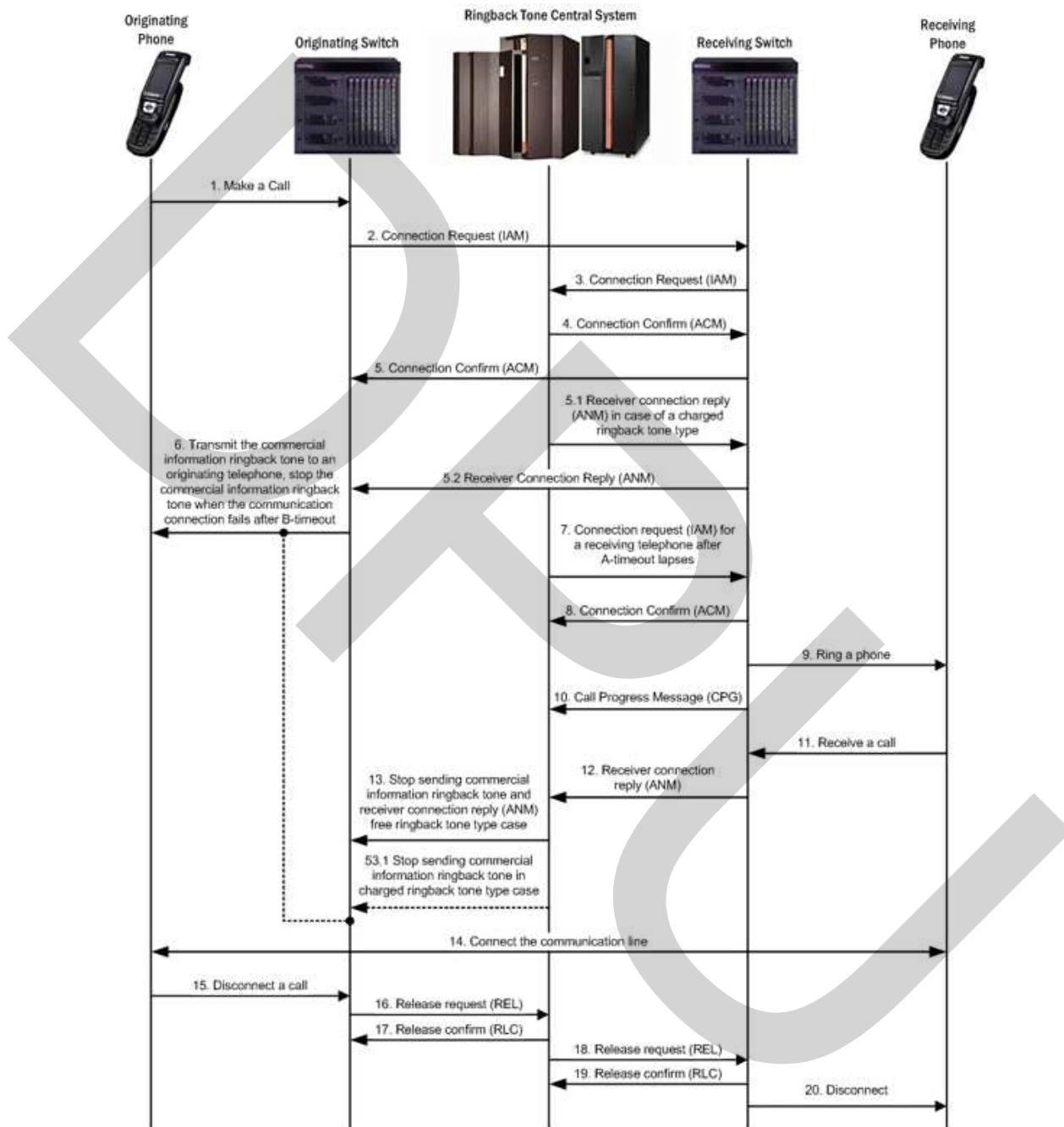
การเชื่อมต่อระบบเสียงเพลงรอสายเข้ากับชุมสายของผู้ให้บริการปลายทาง
 ดังแสดงในรูปที่ 2.4



รูปที่ 2.4 แสดงการเชื่อมต่อระบบเสียงเพลงรอสายเข้ากับชุมสายของผู้ให้บริการปลายทาง

ที่มา : http://rbth.com/ringbacktones/ringbacktones_call_flow.htm

ลำดับการทำงานของเสียงเพลงรอสายที่ชุมสายปลายทาง ดังแสดงในรูปที่ 2.5



รูปที่ 2.5 แสดงลำดับการทำงานของเสียงเพลงรอสายที่ชุมสายปลายทาง

ที่มา : http://rbth.com/ringbacktones/ringbacktones_call_flow.htm

จากรูปที่ 2.5 อธิบายการทำงานของเสียงเพลงรอสายที่ชุมสายปลายทาง ดังนี้

- 1) เมื่อหมายเลขต้นทางโทรออก จะมีการเชื่อมต่อกับชุมสายต้นทาง
- 2) ชุมสายต้นทางเชื่อมต่อไปที่ชุมสายปลายทางเพื่อร้องขอการติดต่อกับหมายเลขปลายทาง
- 3) ชุมสายปลายทางเชื่อมต่อไปที่ระบบเสียงเพลงรอสายเพื่อร้องขอการติดต่อกับชุมสายต้นทาง
- 4) ระบบเสียงเพลงรอสายตอบรับการร้องขอการติดต่อไปที่ชุมสายปลายทาง
- 5) ชุมสายปลายทางตอบรับการร้องขอการติดต่อไปที่ชุมสายต้นทาง
 - 5.1) สร้างวงจรรองรับสำหรับสัญญาณเสียงในกรณีที่มีการตั้งเสียงเพลงรอสาย
 - 5.2) ชุมสายปลายทางตอบรับและสร้างวงจรรองรับสัญญาณเสียงไปที่ชุมสายต้นทาง
- 6) ชุมสายต้นทางส่งสัญญาณเสียงเพลงรอสายไปยังหมายเลขต้นทางและหยุดเล่นเสียงเพลงรอสายเมื่อปลายทางไม่รับสายในเวลาที่กำหนด
- 7) ระบบเสียงเพลงรอสายทำการร้องขอการติดต่อกับชุมสายปลายทาง
- 8) ชุมสายปลายทางตอบรับการร้องขอการติดต่อไปที่ระบบเสียงเพลงรอสาย
- 9) ชุมสายปลายทางส่งสัญญาณเสียงกริ่งไปที่หมายเลขปลายทาง
- 10) ชุมสายปลายทางแจ้งสถานะของหมายเลขปลายทางว่าว่างหรือไม่ไปที่ระบบเสียงเพลงรอสาย
 - 11) หมายเลขปลายทางรับสาย
 - 12) ชุมสายปลายทางสร้างวงจรรองรับสำหรับสัญญาณเสียงเพื่อการสนทนา
 - 13) ระบบเสียงเพลงรอสายหยุดการเล่นเสียงเพลงรอสายและตอบรับสัญญาณเสียงเพื่อการสนทนา
 - 13.1) หยุดการเล่นเสียงเพลงรอสายตามที่ได้ตั้งไว้
 - 14) เชื่อมต่อเพื่อสนทนา
 - 15) หมายเลขต้นทางยกเลิกการติดต่อ
 - 16) ชุมสายต้นทางส่งข้อความไปบอกที่ระบบเสียงเพลงรอสายเพื่อยกเลิกการติดต่อ
 - 17) ระบบเสียงเพลงรอสายตอบรับข้อความและยกเลิกการใช้งาน Timeslot นั้นๆ
 - 18) ระบบเสียงเพลงรอสายส่งข้อความไปบอกที่ชุมสายปลายทางเพื่อยกเลิกการติดต่อ
 - 19) ชุมสายปลายทางตอบรับข้อความและยกเลิกการใช้งาน Timeslot นั้นๆ
 - 20) ชุมสายปลายทางยกเลิกการเชื่อมต่อ

ซึ่งการเชื่อมต่อและการตั้งค่าต่างๆดังกล่าวมาแล้วนั้นเป็นสิ่งจำเป็นเมื่อต้องการให้บริการเสียงเพลงรอสาย

ตัวอย่าง ผู้ใช้ที่เป็นสมาชิกบริการเสียงเพลงรอสายในระบบโทรศัพท์มือถือ DTAC สามารถกำหนดค่าต่างๆ ผ่านหน้า web page ได้ดังนี้

- 1) เพลงพิเศษคนพิเศษ โดยมีการกำหนดเบอร์และเสียงเพลงรอสาย

• เพลงพิเศษ คนพิเศษ (Special Song Special Friend)

หมายเลขโทรศัพท์	ชื่อเพลง
0868916669	รักไม่ต้องการเวลา (Ost.กวน มึน โฮ)
	-- select --
	-- select --
	-- select --
	-- select --
	-- select --
	-- select --
	-- select --
	-- select --

บันทึกการตั้งค่า

รูปที่ 2.6 การตั้งค่าเพลงพิเศษคนพิเศษ

ที่มา : <http://www.dtacmusic.com/ring4u/web/index.jsp>

- 2) เปลี่ยนอารมณ์ตามเวลา โดยมีการกำหนดช่วงเวลาและเสียงเพลงรอสาย

• เปลี่ยนอารมณ์ ตามเวลา (Your Song Your Time)

เวลา	ชื่อเพลง
Time # 1 : 09 : 00 ~ 23 : 55	ที่คิดถึง...เพราะรักเธอใช่ไหม
Time # 2 : --- : --- ~ --- : ---	-- select --
Time # 3 : --- : --- ~ --- : ---	-- select --

บันทึกการตั้งค่า

รูปที่ 2.7 การตั้งค่าเปลี่ยนอารมณ์ตามเวลา

ที่มา : <http://www.dtacmusic.com/ring4u/web/index.jsp>

3) เพลงพิเศษวันพิเศษ โดยมีการกำหนดวันที่-เดือน-ปี และเสียงเพลงรอสาย

• เพลงพิเศษ วันพิเศษ (Special Song Special Day)

วันที่			ชื่อเพลง
03	/	05	หนึ่งวันที่ยาวนาน
---	/	---	-- select --

บันทึกการตั้งค่า

รูปที่ 2.8 การตั้งค่าเพลงพิเศษ วันพิเศษ

ที่มา : <http://www.dtacmusic.com/ring4u/web/index.jsp>

การทำงานของบริการเสียงเพลงรอสาย

1) เมื่อผู้ใช้งานโทรศัพท์ทำการโทรหาเบอร์ที่ไม่มีเสียงเพลงรอสายในระบบจะได้ยิน ringing

2) เมื่อผู้ใช้งานโทรศัพท์ทำการโทรหาเบอร์ที่มีเสียงเพลงรอสายในระบบ

2.1) กรณีไม่มีการกำหนดค่า เพลงพิเศษคนพิเศษ เปลี่ยนอารมณ์ตามเวลา และ เพลงพิเศษวันพิเศษ หรืออยู่นอกช่วงเวลาที่ตั้งไว้ จะได้ยินเสียงเพลงรอสายในระบบวนไปเรื่อยๆ และในแต่ละครั้งที่ทำการโทรจะได้ยินเพลงที่ 1 โดยการสุ่ม

2.2) กรณีมีการกำหนดค่าไว้

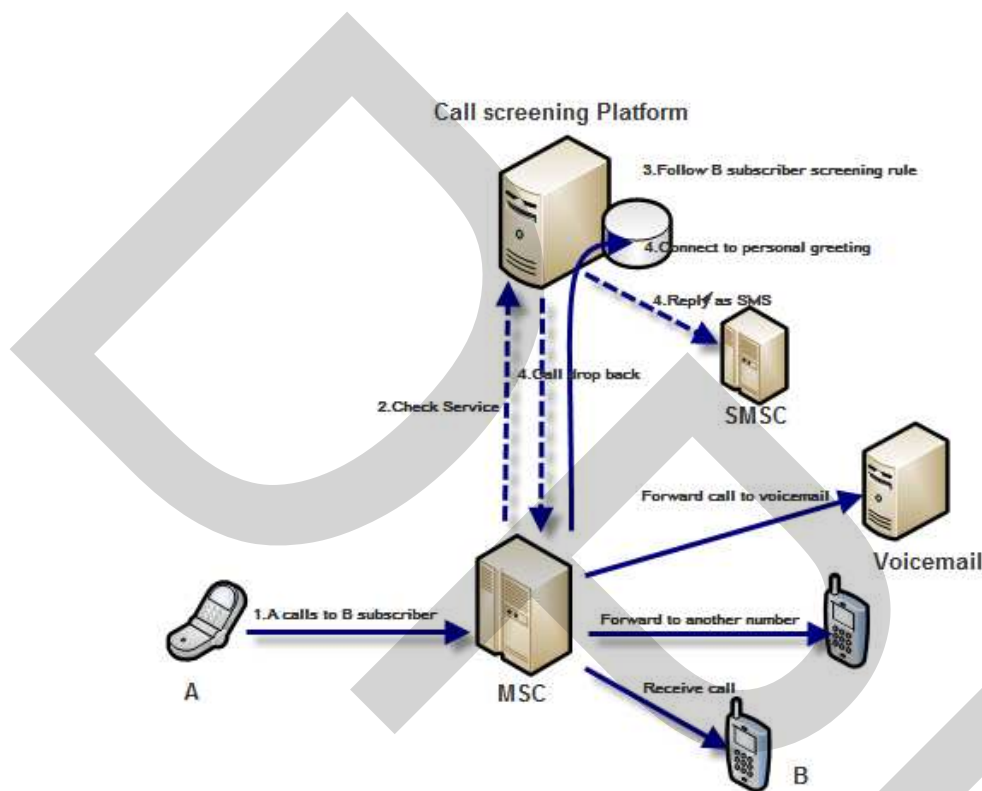
2.2.1) เพลงพิเศษคนพิเศษ ไม่ว่าจะช่วงเวลาใดถ้าเบอร์พิเศษทำการโทรก็จะได้ยิน แต่เพลงที่ปลายทางกำหนดไว้ตลอด

2.2.2) เปลี่ยนอารมณ์ตามเวลา ทุกเบอร์ที่โทรเข้ามาในช่วงเวลาที่กำหนดไว้ จะได้ยินเสียงเพลงที่ตั้งไว้

2.2.3) เพลงพิเศษวันพิเศษ ทุกเบอร์ที่โทรเข้ามาในวันพิเศษที่กำหนดไว้จะได้ยินเสียงเพลงที่ตั้งไว้

2.3 ความรู้พื้นฐานของบริการเลือกไม่รับสาย ของระบบโทรศัพท์มือถือปัจจุบัน

บริการเลือกไม่รับสาย (Call Screening) คือบริการที่ช่วยให้คุณเลือกไม่รับสายในช่วงเวลาที่ไม่สะดวก หรือไม่สามารถรับสายได้ โดยมีโครงสร้างการทำงานดังแสดงในรูปที่ 2.9



รูปที่ 2.9 โครงสร้างการทำงานของบริการเลือกไม่รับสาย

จากรูปที่ 2.9 อธิบายการทำงานของบริการเลือกไม่รับสายได้ดังนี้

- 1) เมื่อหมายเลขต้นทางโทรออก จะทำการเชื่อมต่อไปที่ชุมสายต้นทาง
- 2) ชุมสายทำการตรวจสอบการตั้งค่าเลือกไม่รับสายจากระบบ Call screening
- 3) ระบบ Call screening จะมีข้อมูลการตั้งค่าเลือกไม่รับสายของเบอร์ปลายทาง
- 4) ตอบกลับบริการไปที่ชุมสาย

บริการเลือกไม่รับสาย ของระบบโทรศัพท์มือถือ ปัจจุบันมี 3 แบบ ดังนี้

- 1) เสียงสายไม่ว่าง (Busy tone)
- 2) ระบบฝากข้อความเสียง (Voicemail)
- 3) ระบบเสียงตอบรับอัตโนมัติ (IVR) โดยจะมีให้ผู้ใช้เลือก 5 ข้อความเสียง

ตัวอย่างการตั้งค่าผ่าน web page ของผู้ใช้ระบบโทรศัพท์มือถือ DTAC

รูปที่ 2.10 การตั้งค่า Call screening

ที่มา : www.dtac.co.th

การทำงานของบริการเลือกไม่รับสาย

- 1) ผู้ใช้จะต้องทำการตั้งช่วงเวลาเลือกไม่รับสายสำหรับแต่ละหมายเลขที่ไม่ต้องการรับสาย พร้อมกับเลือกรูปแบบของบริการเลือกไม่รับสายให้กับหมายเลขนั้นๆ
- 2) เมื่อมีการโทรเข้ามาในช่วงเวลาที่ได้กำหนดไว้ ก็จะเข้าสู่บริการเลือกไม่รับสายตามรูปแบบต่างๆที่ได้กำหนดไว้ในช่วงเวลานั้น

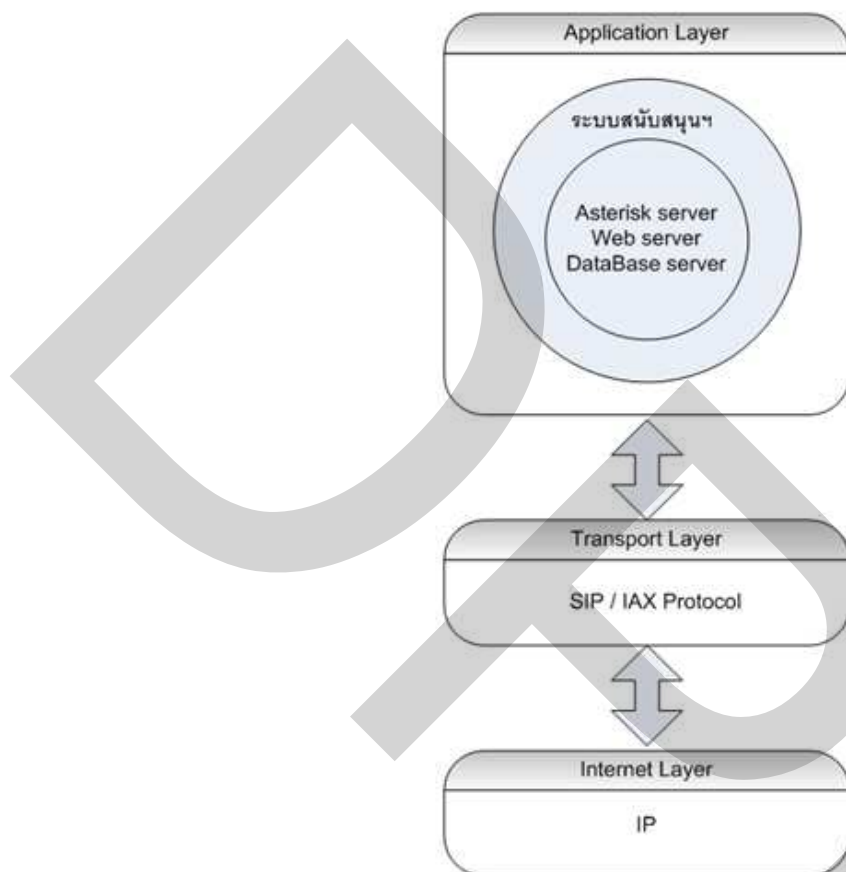
2.4 ผลงานวิจัย/ผลิตภัณฑ์ ที่เกี่ยวข้อง

2.4.1 ระบบสนับสนุนการกำหนดค่าการทำงานของ Asterisk⁴

ในการทำงานของระบบสนับสนุนนั้น จะทำงานในระดับ Application Layer หรือในระดับชั้น Control Layer ตามสถาปัตยกรรมโครงข่าย NGN โดยระบบสนับสนุนฯ จะทำหน้าที่กำหนดการทำงานต่างๆ ของ Asterisk ผ่านทาง Web Interface โดยที่ผู้ใช้งานสามารถเรียกใช้งานผ่าน Browser ทั่วไปได้ เช่น Internet Explorer หรือ Fire Fox เป็นต้น ซึ่งระบบที่พัฒนาจะมี Interface เป็นภาษาไทย ทำให้ง่ายต่อการทำความเข้าใจ และง่ายต่อการใช้งาน โดยตัวโปรแกรมจะสร้างไฟล์ที่จำเป็นต่อการทำงานของ Asterisk ขึ้นมาจากค่าที่ผู้ใช้กำหนดไว้ จากนั้น

⁴ ประจวบตผล ดลคุณิตตา. (2552). ระบบสนับสนุนการกำหนดค่าการทำงานของไอพีพีบีเอ็กซ์แบบแอสเทอริคส์.

ตัวโปรแกรมจะทำหน้าที่นำไฟล์ที่สร้างขึ้นไปทับไฟล์ของเดิมที่มีอยู่ ก็จะทำให้ Asterisk สามารถทำงานได้ตามความต้องการ ดังแสดงในรูปที่ 2.11



รูปที่ 2.11 ภาพรวมการทำงานของระบบสนับสนุน

2.4.2 การเพิ่มฟังก์ชันการทำงานของแอสเทริกส์ด้วยพีเอชพี : กรณีศึกษาระบบแจ้งข่าวสารด้วยเสียงอัตโนมัติผ่านโทรศัพท์ไอพี

ลักษณะการทำงาน คือใช้ PHP สร้างหน้าเว็บอินเตอร์เฟส เพื่อจัดการฟังก์ชันการทำงานต่างๆ ซึ่งจะทำงานร่วมกับฐานข้อมูลในการเก็บข้อมูลต่างๆของระบบ ส่วนการทำงานของ Asterisk จะใช้คำสั่ง AMI สื่อสารกับ PHP ผ่านทาง Socket ในส่วนของฐานข้อมูล Asterisk จะติดต่อผ่านทาง AGI (Asterisk Gateway Interface) ซึ่งเป็นช่องทางที่ทำให้สามารถสื่อสารผ่านทาง Stdin และ Stdout ด้วย PHP ทำให้สามารถติดต่อกับฐานข้อมูลด้วย PHP ได้ และการทำงานในส่วนของการสื่อสารกับโทรศัพท์ไอพีจะทำงานผ่านระบบ IVR (Interactive Voice Response) ตอบโต้กันระหว่าง Asterisk กับโทรศัพท์ไอพี กรณีศึกษาระบบแจ้งข่าวสารด้วยเสียงอัตโนมัติ

⁵ ชานนท์ อยู่ญาติมาก. (2553). การพัฒนาระบบแจ้งข่าวสารด้วยเสียงอัตโนมัติผ่านโทรศัพท์ไอพี.

ผ่านโทรศัพท์ไอพี เป็นการนำเทคโนโลยี VoIP มาประยุกต์ใช้งานโดยใช้ Asterisk มาเป็นศูนย์กลางการรับฝากข้อความเสียง มีการจัดการควบคุมระบบผ่านหน้าเว็บไซต์ และระบบ IVR ผู้ใช้งานสามารถฝากข้อความเสียงผ่านทางเว็บเพจหรือระบบโทรศัพท์ไอพี เมื่อถึงเวลาที่กำหนดระบบจะสามารถโทรศัพท์ออกไปแจ้งให้ผู้รับทราบโดยอัตโนมัติ ซึ่งมีการทำงานอยู่ในระดับชั้น Control Layer ตามสถาปัตยกรรมโครงข่าย NGN

จากการศึกษาการใช้งานบริการต่างๆและงานวิจัยที่เกี่ยวข้องกับวิทยานิพนธ์ที่นำเสนอ สามารถเปรียบเทียบคุณสมบัติได้ดังตารางที่ 2.1

ตารางที่ 2.1 แสดงการเปรียบเทียบคุณลักษณะของงานวิจัยและบริการที่เกี่ยวข้องกับวิทยานิพนธ์นี้

คุณสมบัติ	ระบบสนับสนุนการกำหนดค่าการทำงานของ Asterisk	กรณีศึกษาระบบแจ้งข่าวสารด้วยเสียงอัตโนมัติผ่านโทรศัพท์ไอพี	งานวิจัยที่นำเสนอ
1. ติดตั้งบริการเสริมที่เครื่องแม่ข่ายศูนย์กลาง			✓
2. ติดตั้ง Database Server ที่เครื่องแม่ข่ายศูนย์กลาง			✓
3. ข้อมูลการใช้งาน(CDR) รวมอยู่ที่เครื่องแม่ข่ายจุดเดียว			✓
4. เมื่อมีการปรับปรุงหรือเปลี่ยนแปลงทำที่จุดเดียว			✓
5. ผู้ใช้งานสามารถเปลี่ยนค่าการใช้งานของตนเองได้	✓	✓	✓
6. สร้างไฟล์ที่จำเป็นต่อการใช้งาน Asterisk	✓	✓	✓
7. สร้าง SIP User	✓	✓	✓
8. สร้างกลุ่ม User	✓	✓	✓
9. รองรับการเพิ่มเครื่อง IP-PBX server ต่างรุ่นได้			✓

บทที่ 3

การออกแบบและพัฒนาระบบ

3.1 แนวทางการวิจัยและพัฒนา

งานวิจัยนี้มีวัตถุประสงค์เพื่อศึกษา ออกแบบ และพัฒนาระบบบริการเสียงรอสายและเลือกไม่รับสายเป็นกรณีศึกษาในโครงข่าย VoIP ให้มีโครงสร้างการทำงานเป็นไปตามรูปแบบสถาปัตยกรรม NGN คือระบบมีการควบคุมและจัดการจากเครื่องแม่ข่ายศูนย์กลางเพียงจุดเดียว เพื่อให้เครื่อง Asterisk สามารถเข้าถึงข้อมูลได้อย่างรวดเร็ว การติดตั้งระบบทำได้ง่ายและรองรับการเพิ่มขึ้นของเครื่อง Asterisk เครื่องใหม่ โดยมีแนวทางในการวิจัยและพัฒนาดังนี้

3.1.1 ศึกษาและรวบรวมข้อมูล

- 1) ศึกษาการใช้งานและทำงานของโปรแกรม Asterisk ในแบบ realtime
- 2) ศึกษาเพิ่มเติมการใช้งานภาษา PHP ฐานข้อมูล MySQL การติดตั้ง PHP MyAdmin การติดตั้ง mail Server และการติดตั้ง Apache บนระบบปฏิบัติการ Linux
- 3) ศึกษาการสร้างระบบฐานข้อมูลเพื่อเก็บข้อมูลสำหรับการเรียกใช้งานในแบบ realtime
- 4) ศึกษาโครงสร้างสถาปัตยกรรม NGN
- 5) ศึกษาเพิ่มเติมการทำงานระบบบริการเสียงรอสายและเลือกไม่รับสาย

3.1.2 การออกแบบระบบงาน

- 1) ออกแบบหน้า Web page สำหรับส่วนติดต่อกับผู้ใช้
- 2) ออกแบบฐานข้อมูล Asterisk เพื่อให้ Asterisk สามารถเรียกใช้งานในแบบ realtime
- 3) ออกแบบแผนการโทร (dial plan) ระบบบริการเสียงรอสายและเลือกไม่รับสายให้มีการทำงานตามวัน-เวลา หรือเทศกาล

3.1.3 พัฒนาระบบงาน

ทำการพัฒนาระบบให้สามารถทำงานได้ตามวัตถุประสงค์ที่ตั้งไว้ มีการทดสอบความสามารถของระบบเพื่อหาข้อผิดพลาดต่างๆ ภายในระบบแล้วทำการแก้ไข

3.1.4 ทดสอบการใช้งาน

มีการทดสอบในแต่ละส่วนย่อย และทดสอบการทำงานของทั้งระบบเพื่อดูประสิทธิภาพการทำงานของระบบ รวมทั้งความถูกต้องในการทำงานของระบบตามวัน-เวลา หรือเทศกาล

3.1.5 สรุปผลการพัฒนา

นำข้อมูลที่ได้ในการทดสอบมาสรุปผล เพื่อใช้ในการวิเคราะห์การทำงานและประเมินประสิทธิภาพของระบบ

3.2 เครื่องมือที่ใช้ในงานวิจัย

3.2.1 Hardware

1) เครื่องคอมพิวเตอร์เพื่อจำลองเป็นเครื่อง Application server ในระดับชั้น Service Layer จำนวน 1 เครื่อง

CPU 1.5 GHz, RAM 512 MB, Harddisk 40 GB

2) เครื่องคอมพิวเตอร์เพื่อจำลองเป็นเครื่อง IP PBX ในระดับชั้น Control Layer จำนวน 2 เครื่อง

เครื่องที่ 1 สำหรับติดตั้ง Asterisk 1.6.2.17.2

CPU Core2 Duo 2.26 GHz, RAM 2 GB, Harddisk 320 GB

เครื่องที่ 2 สำหรับติดตั้ง Asterisk 1.8.3

CPU 1.5 GHz, RAM 512 MB, Harddisk 60 GB

3) เครื่องคอมพิวเตอร์เพื่อจำลองเป็นเครื่อง Client ในระดับชั้น Access Layer จำนวน 2 เครื่อง

เครื่องที่ 1 สำหรับติดตั้ง X-Lite

CPU Core i5 2.26 GHz, RAM 2 GB, Harddisk 500 GB

เครื่องที่ 2 สำหรับติดตั้ง X-Lite

CPU Core2 1.66 GHz, RAM 1.5 GB, Harddisk 160 GB

4) เครื่องคอมพิวเตอร์สำหรับเขียนชุดคำสั่งจำนวน 1 เครื่อง

CPU Intel Core i5 2.26 GHz, RAM 2 GB, Harddisk 500 GB

3.2.2 Software

1) Linux Ubuntu 10.1 server

2) Asterisk (เวอร์ชัน 1.6.2.17.2 และเวอร์ชัน 1.8.3)

3) Apache 2.0

4) MySQL 5.0

5) PHP5, JavaScript

6) Squirrelmail เพื่อใช้เป็น Mail Server กรณีที่มีการส่งข้อความเข้า voice mail

7) Softphone (X-Lite)

3.3 แผนการดำเนินงาน

3.3.1 ศึกษาทฤษฎีเบื้องต้น

ทฤษฎี VoIP, โครงสร้างสถาปัตยกรรม NGN, หลักการคอนฟิกและการทำงานของโปรแกรม Asterisk ในแบบ realtime และศึกษาการทำงานของระบบบริการเสียงรอสายและเลือกไม่รับสาย

3.3.2 การรวบรวมข้อมูล

- 1) รวบรวมข้อมูลที่เกี่ยวข้องกับ VoIP ว่ามีหลักการทำงานอย่างไรเพื่อให้ง่ายกับการทำงานวิจัย
- 2) รวบรวมข้อมูลที่เกี่ยวข้องกับสถาปัตยกรรม NGN ว่ามีกี่ระดับชั้น มีหลักการทำงานและการเชื่อมต่ออย่างไรเพื่อให้ง่ายกับการทำงานวิจัย
- 3) รวบรวมข้อมูลการคอนฟิก Asterisk ในแบบ realtime และการสร้างฐานข้อมูล Asterisk

3.3.3 การวิเคราะห์ข้อมูล

นำข้อมูลที่ได้มาทำการวิเคราะห์ในส่วนของการกำหนดค่าในฐานข้อมูล การเขียน dial plan การกำหนดค่าการทำงานระบบบริการฯ และการสร้างฐานข้อมูลเพิ่มเติม เพื่อให้ Asterisk ทำงานตามความต้องการของระบบได้

3.3.4 การออกแบบโปรแกรม

ออกแบบส่วนการรับค่าจากผู้ใช้งาน และการจัดเก็บข้อมูลลงในฐานข้อมูล

3.3.5 การพัฒนา – ทดสอบ

- 1) การเขียนโปรแกรม
- 2) การทดสอบย่อย
- 3) การทดสอบรวม

3.3.6 การจัดทำเอกสาร

นำข้อมูลที่ได้ในการทดสอบมาสรุปผล เพื่อใช้เป็นข้อมูลในการจัดทำวิทยานิพนธ์ ดังแสดงในตารางที่ 3.1

ตารางที่ 3.1 แผนการดำเนินงาน

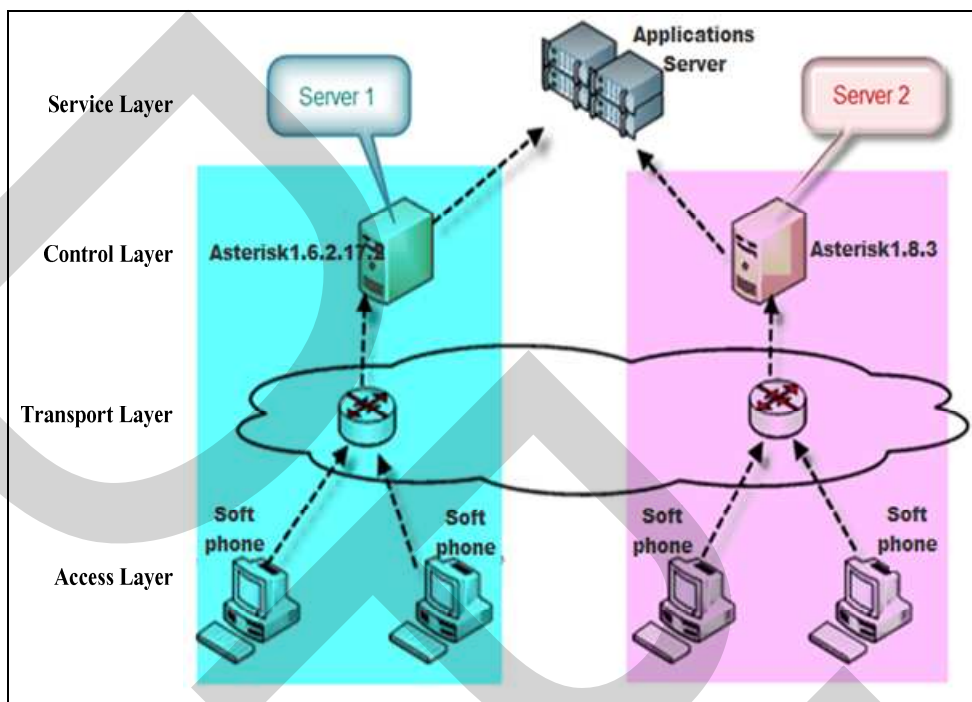
งาน	เดือน	ม.ค. - มี.ค.	เม.ย. - มิ.ย.	ก.ค. - ก.ย.	ต.ค. - ธ.ค.	ม.ค. - มี.ค.
		54	54	54	54	55
ศึกษาทฤษฎีเบื้องต้น						
การรวบรวมข้อมูล						
การวิเคราะห์ข้อมูล						
ออกแบบโปรแกรม						
การพัฒนา - ทดสอบ						
การเขียนโปรแกรม						
การทดสอบย่อย						
การทดสอบรวม						
การจัดทำเอกสาร						

3.4 ขั้นตอนและวิธีดำเนินงาน

3.4.1 แนวคิดในการพัฒนาระบบเสียงรอสายและเลือกไม่รับสาย

แนวคิดในการพัฒนาระบบเสียงรอสายและเลือกไม่รับสายนั้น จะทำงานในระดับชั้น Service Layer ตามสถาปัตยกรรมโครงข่าย NGN โดยระบบบริการเสริมจะทำหน้าที่กำหนดการทำงานต่างๆ ของ Asterisk ผ่านทาง Web Interface โดยที่ผู้ใช้งานสามารถเรียกใช้งานผ่าน Browser ทั่วไปได้ เช่น Internet Explorer หรือ Fire Fox เป็นต้น ซึ่งระบบที่พัฒนาจะมี Interface เป็นภาษาไทย ทำให้ง่ายต่อการทำความเข้าใจ และง่ายต่อการใช้งาน โดยขั้นตอนการติดตั้งเครื่อง Asterisk ในครั้งแรกตัวโปรแกรมจะสร้างไฟล์ที่จำเป็นต่อการทำงานของ Asterisk แต่ละ version (ในงานวิจัยนี้ระบบที่พัฒนาจะรองรับ Asterisk1.6 และ Asterisk1.8 เท่านั้น) จากนั้นเมื่อผู้ดูแลระบบจัดการในส่วนของบริษัทเสริม ระบบที่พัฒนาจะสร้างไฟล์ที่เกี่ยวข้อง และ

ตั้งเชื่อมต่อเพื่อเขียนข้อมูลดังกล่าวแทนไฟล์ของเดิมที่มีอยู่ ก็จะทำให้ระบบ Asterisk สามารถทำงานได้ตามความต้องการ ดังแสดงในรูป 3.1



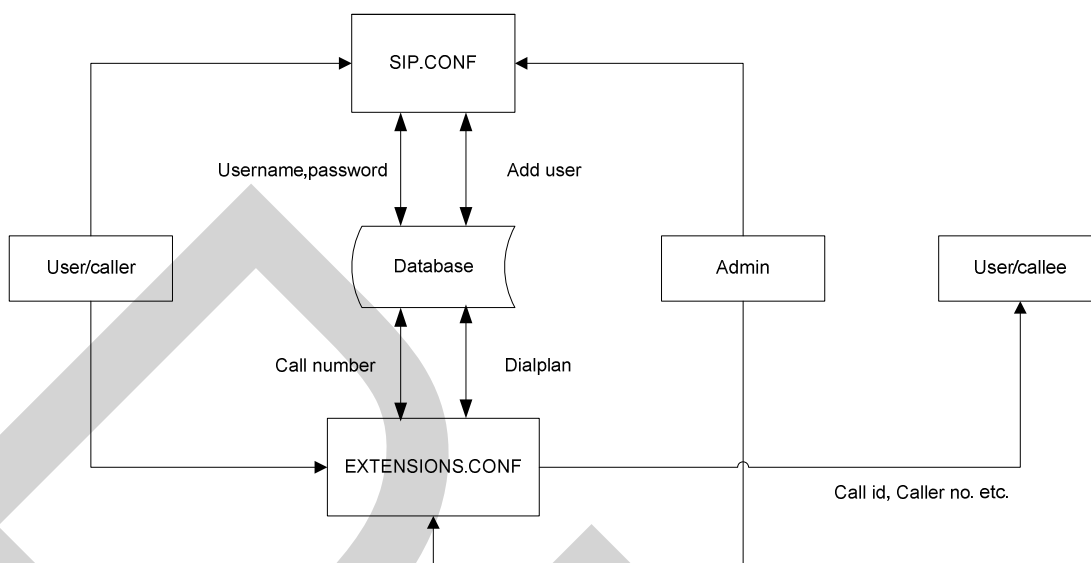
รูปที่ 3.1 ภาพรวมของระบบตามสถาปัตยกรรมโครงข่าย NGN

จากรูปที่ 3.1 เพื่อให้เครื่อง Asterisk แต่ละ version สามารถใช้งาน Database ซึ่งอยู่ใน Application Server ในระดับชั้น Service Layer ได้ นั้น จะต้องมีการกำหนดค่าที่เครื่อง Application Server ดังนี้

File : /etc/mysql/my.cnf มีการกำหนดรายละเอียดเพิ่มเติมดังนี้

```
[mysqld]
#bind-address = 127.0.0.1
```

การทำงานของ Asterisk ในระดับชั้น Control Layer จะแบ่งการทำงานออกเป็น ส่วนๆ ในลักษณะของไฟล์โปรแกรม การทำงานจะมีความสัมพันธ์ระหว่างไฟล์ก็ต่อเมื่อมีการเรียกใช้ไฟล์นั้น จากการศึกษาพบว่า ในการใช้งาน Asterisk เพื่อให้เป็นไปตามสถาปัตยกรรม NGN มีไฟล์ที่สำคัญและเพียงพอต่อการใช้งานคือไฟล์ sip.conf และ extensions.conf ซึ่งการทำงานของไฟล์ทั้งสองดังแสดงในรูปที่ 3.2



ภาพที่ 3.2 การทำงานของไฟล์ SIP.CONF และ EXTENSIONS.CONF ตามสถาปัตยกรรม NGN

จากรูปที่ 3.2 สามารถอธิบายหลักการทำงานของระบบได้ดังนี้

1) เมื่อผู้ใช้งานชื่อเข้าใช้จาก Softphone เช่น X-lite จะต้องใส่ชื่อผู้ใช้และรหัสผ่าน(ถ้ามี) ซึ่งข้อมูลนี้จะถูกส่งไปทำงานที่ไฟล์ sip.conf

2) ไฟล์ sip.conf จะตรวจสอบข้อมูลใน database ว่าตรงกับบัญชีผู้ใช้ที่มีหรือไม่ หากการลงชื่อเข้าใช้ไม่สำเร็จ ผู้ใช้จะโทรออกหรือรับสายไม่ได้ เมื่อผู้ใช้งานชื่อเข้าใช้สำเร็จก็พร้อมที่จะเรียกใช้งานโทรออกหรือรับสายเข้าต่อไป

3) เมื่อผู้ใช้งานต้องการโทรออก ข้อมูลที่สำคัญคือ หมายเลขปลายทางจะถูกส่งไปทำงานที่ไฟล์ extensions.conf

4) ไฟล์ extensions.conf จะเรียกใช้งาน database ซึ่งตารางที่เกี่ยวข้อง คือ extensions_table ตารางนี้จะเป็นไฟล์หลักที่กำหนดรูปแบบการโทรออกต่างๆ เช่น กำหนดให้มีเสียงเรียก (กริ่ง) นานกี่วินาที กำหนดเสียง BUSY Tone กำหนดการฝากข้อความเข้าระบบรับฝากข้อความเสียง กำหนดการเล่นไฟล์เสียงระบบตอบรับอัตโนมัติ กำหนดการเล่นไฟล์เสียงเพลงรอสาย เป็นต้น

3.4.2 การออกแบบระบบ

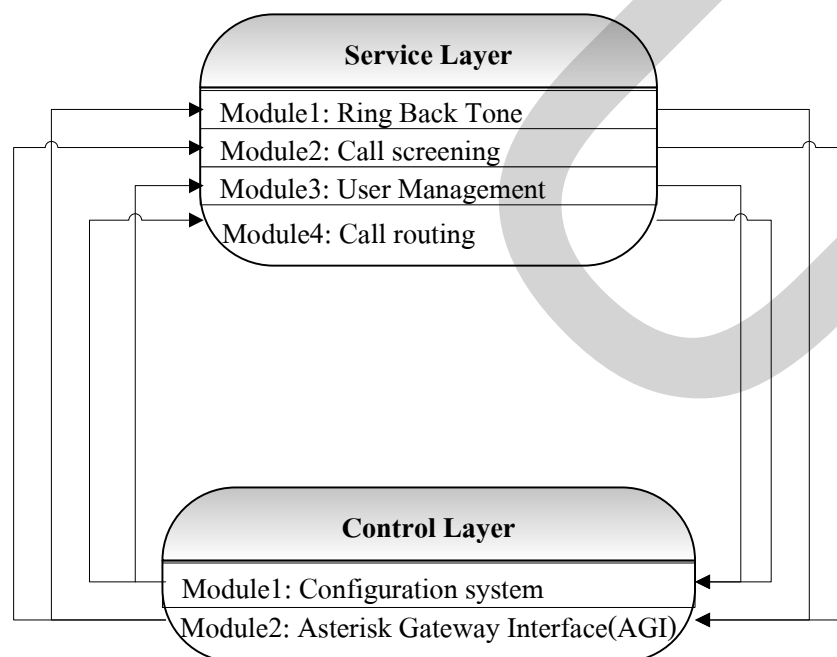
เพื่อให้การทำงานของระบบเป็นไปอย่างถูกต้อง จึงได้มีการออกแบบและวิเคราะห์ความต้องการของระบบเพื่อให้สามารถทำงานได้อย่างถูกต้องและมีความรวดเร็วในการสร้างระบบ ซึ่งในการพัฒนาระบบต้องเน้นให้ใช้งานได้ง่าย สะดวก มีการกำหนดค่าใช้งานเท่าที่จำเป็นต่อระบบ และระบบมีความน่าเชื่อถือ สามารถให้ผลลัพธ์ของการทำงานได้ถูกต้องตามที่ผู้ใช้งานกำหนด จะแบ่งเป็น 5 ขั้นตอน ได้แก่

- 1) การออกแบบและพัฒนา module ต่างๆ
- 2) การออกแบบกระบวนการทำงานของโปรแกรม
- 3) การออกแบบการทำงานของระบบบริการเสียงรอสาย
- 4) การออกแบบการทำงานของระบบบริการเลือกไม่รับสาย
- 5) การออกแบบส่วนติดต่อผู้ใช้แบบเว็บ (Web Interface)
- 6) การออกแบบฐานข้อมูล

มีขั้นตอนออกแบบการทำงานดังขั้นตอนที่ 3.4.2.1 - 3.4.2.6

3.4.2.1 ขั้นตอนการออกแบบและพัฒนา module ต่างๆ

ขั้นตอนการออกแบบและพัฒนา module ต่างๆ เพื่อให้มีการเรียกใช้งานตามรูปแบบสถาปัตยกรรม NGN แสดงดังรูปที่ 3.3



รูปที่ 3.3 module ต่างๆ ที่งานวิจัยนี้ได้พัฒนาขึ้น

จากรูปที่ 3.3 สามารถอธิบายรายละเอียดได้ดังนี้

Service Layer มี module ที่ได้พัฒนาขึ้นมาจำนวน 4 module ดังนี้

Module 1: Ring Back Tone เป็นส่วนการรับค่าจากผู้ใช้งาน แล้วนำข้อมูลเก็บใน database โดยจะมีการเรียกใช้งานจาก Module 5: Asterisk Gateway Interface (AGI) และมีหน้าที่ตอบกลับข้อมูลการตั้งค่าเสียงเพลงรอสาย เช่น หมายเลขต้นทาง, เพลง, วัน, วันที่, เดือน และเวลา

Module 2: Call Screening เป็นส่วนการรับค่าจากผู้ใช้งาน แล้วนำข้อมูลเก็บใน database โดยจะมีการเรียกใช้งานจาก Module 5: Asterisk Gateway Interface (AGI) และมีหน้าที่ตอบกลับข้อมูลการตั้งค่าเลือกไม่รับสาย เช่น หมายเลขต้นทาง, IVR, BUSY tone, Voicemail, วัน, วันที่, เดือน และเวลา

Module 3: User Management เป็นส่วนการรับค่าจากผู้ใช้งาน แล้วนำข้อมูลเก็บใน database โดยจะมีการเรียกใช้งานจาก Module 4: Configuration และมีหน้าที่ตอบกลับข้อมูลผู้ใช้งาน เช่น หมายเลขผู้ใช้, สิทธิการใช้งาน และหมายเลขเครื่อง IP PBX

Module 4: Call routing เป็นรูปแบบการโทร มีการกำหนดไว้ใน database โดยจะมีการเรียกใช้งานจาก Module 4: Configuration system และมีหน้าที่ตอบกลับข้อมูลการโทร

Control Layer มี module ที่ได้พัฒนาขึ้นมาจำนวน 2 module ดังนี้

Module 1: Configuration system เป็นการ config ไฟล์ที่จำเป็นต่อการใช้งาน Asterisk ในกรณีนี้คือไฟล์ sip.conf และ extensions.conf ให้มีการเก็บข้อมูลของผู้ใช้ และ dial plan ไว้ใน database ที่อยู่ใน Service Layer เมื่อมีการทำงาน Asterisk จะเรียกใช้งานจาก Module 3: User Management เพื่อตรวจสอบผู้ใช้งาน และ Module 4: Call routing เพื่อตรวจสอบ dial plan

Module 2: Asterisk Gateway Interface (AGI) เป็นส่วนการตรวจสอบการตั้งค่าบริการ เมื่อมีการทำงาน Asterisk จะเรียกใช้งานจาก Module 1: Ring Back Tone เพื่อตรวจสอบการตั้งค่าบริการเสียงรอสาย และ Module 2: Call Screening เพื่อตรวจสอบการตั้งค่าบริการเลือกไม่รับสาย

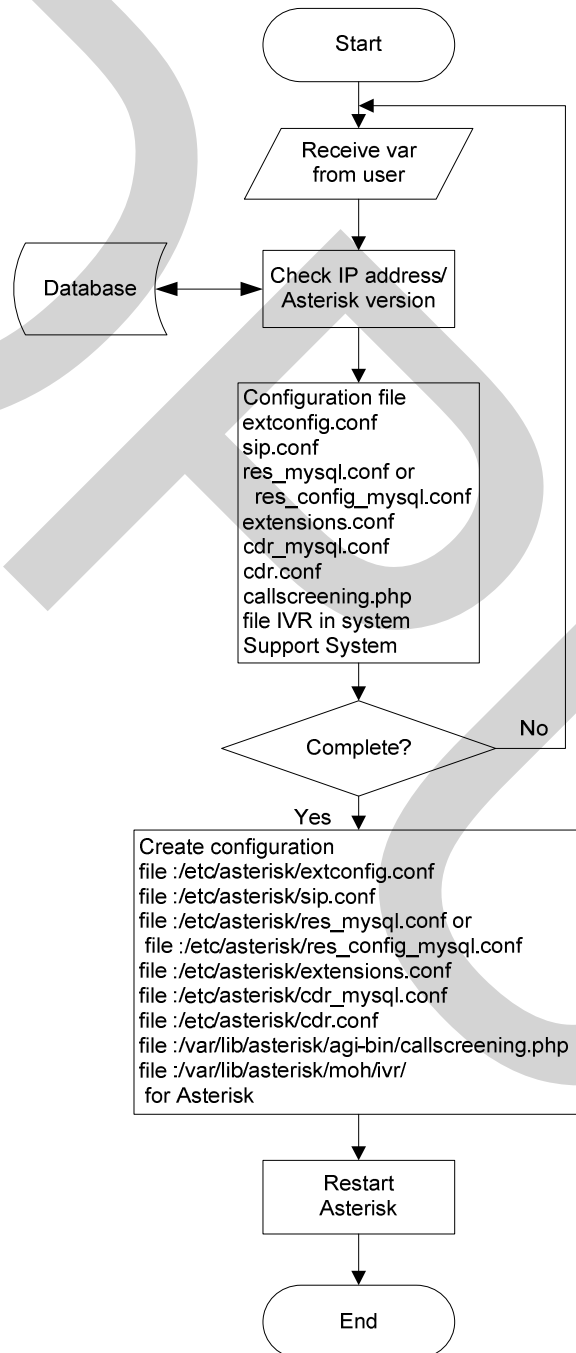
3.4.2.2 ขั้นตอนการออกแบบกระบวนการทำงานของโปรแกรม

แบ่งเป็น 2 ส่วน คือ

- 1) ส่วนการกำหนดค่าการทำงานของระบบ ซึ่งจะทำงานก่อนที่จะเริ่มใช้ระบบ
- 2) ส่วนควบคุมการใช้งาน ซึ่งจะทำงานในขณะที่มีการใช้งานระบบ

3.4.2.2.1 การออกแบบส่วนการกำหนดค่าการทำงานของระบบ

การออกแบบส่วนการกำหนดค่าการทำงานของระบบ จะเป็นการตั้งค่าการใช้งานเครื่อง IP-PBX ก่อนที่จะเริ่มใช้ระบบ โดยในส่วนนี้ผู้ดูแลระบบจะกำหนดค่าต่างๆ ในช่องรับข้อความที่กำหนดไว้ให้ผ่านทางหน้าเว็บ เช่น รุ่นเครื่อง IP-PBX, IP address, login และ password จากนั้นระบบจะทำงานตามขั้นตอนในรูปที่ 3.4



รูปที่ 3.4 แผนภาพส่วนการกำหนดค่าการทำงานของระบบ

จากรูปที่ 3.4 ส่วนการกำหนดค่าการทำงานของระบบ เป็นการกำหนดค่าเบื้องต้นที่จำเป็นต่อการใช้งาน Asterisk แต่ละ version โดยที่ตัว Application server จะตรวจสอบ IP address ของ Control Layer (ในกรณีนี้คือ Asterisk) แล้ว Application server จะสร้างไฟล์ที่จำเป็นต่อการใช้งานของระบบและเชื่อมต่อเพื่อเขียนข้อมูลดังกล่าวแทนที่ไฟล์เดิมที่มีอยู่ เพื่อให้ระบบสามารถใช้งานได้

การกำหนดค่าไฟล์ที่จำเป็นในเครื่อง Asterisk1.6 และ Asterisk1.8 ให้สามารถใช้งานระบบได้มีดังนี้

1) File : /etc/asterisk/extconfig.conf จะบอกให้ Asterisk ดึงข้อมูลจากฐานข้อมูล มีการกำหนดรายละเอียดเพิ่มเติมดังนี้

```
[settings]
sipusers => mysql,general,sipusers
sippeers => mysql,general,sipusers
voicemail => mysql,general,voicemail_users
extensions => mysql,general,extensions_table
musiconhold => mysql,general,musiconhold
```

2) File : /etc/asterisk/res_mysql.conf เป็นไฟล์ config เมื่อต้องการให้ Asterisk ดึง config มาจาก database แทนที่จะดึงมาจากไฟล์.conf สำหรับ Asterisk1.6 และ res_config_mysql.conf สำหรับ Asterisk1.8 มีการกำหนดรายละเอียดเพิ่มเติมเหมือนกันดังนี้

```
[general]
dbhost = 192.168.1.4
dbname = asterisk
dbuser = root
dbpass = 123
dbport = 3306
dbsock = /var/run/mysqld/mysqld.sock
```

3) File : /etc/asterisk/extensions.conf เพื่อให้ Asterisk เรียกใช้งาน dial plan ในแบบ realtime มีการกำหนดรายละเอียดเพิ่มเติมดังนี้

```
[default]
;include => demo
switch => Realtime/kataa@extensions
```

4) File : /etc/asterisk/cdr_mysql.conf เป็นไฟล์ config เพื่อให้ Asterisk เก็บรายการโทร (CDR) ไว้ใน database และจะเขียนเว็บให้ดึง CDR มาโชว์ได้ มีการกำหนดรายละเอียดเพิ่มเติมดังนี้

```
[global]
hostname=192.168.1.4
dbname=asterisk
table=cdr
password=123
user=root
port=3306
sock=/var/run/mysql/mysql.sock
```

5) File : /etc/asterisk/cdr.conf ไฟล์นี้บอกให้ Asterisk ส่ง CDR ไปยัง database มีการกำหนดรายละเอียดเพิ่มเติมดังนี้

```
[general]
enable=yes
unanswered = yes
```

6) File : /etc/asterisk/sip.conf เป็นไฟล์ config สำหรับโปรโตคอล SIP ของ Asterisk เพื่อให้ใช้งานในแบบ realtime ได้ มีการกำหนดรายละเอียดเพิ่มเติมดังนี้

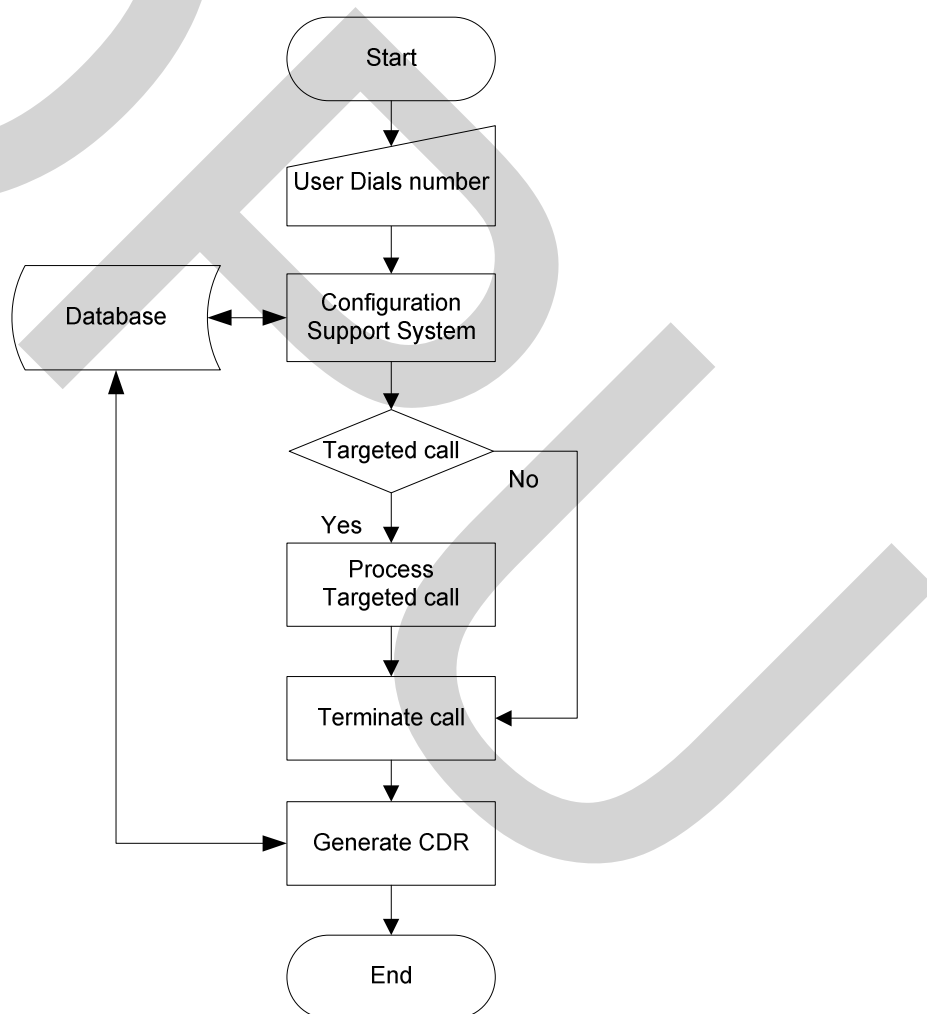
```
rtcachefriends = yes
```

7) File : /var/lib/asterisk/agi-bin/callscreening.php ไฟล์นี้ได้พัฒนาขึ้นเพื่อให้ Asterisk เรียกใช้งานผ่านฟังก์ชัน AGI ในการตรวจสอบข้อมูลการตั้งค่าบริการเสียงรอสายและเลือกไม่รับสายจาก database

8) File : /var/lib/asterisk/moh/ivr/ เป็นไฟล์เสียงระบบตอบรับอัตโนมัติของระบบ

3.4.2.2.2 การออกแบบส่วนควบคุมการใช้งาน

การทำงานในส่วนนี้จะทำงานไปพร้อมกันกับ Asterisk เพื่อควบคุมการใช้งาน โดยระหว่างทำงานจะมีการรับ-ส่งค่า ระหว่างโปรแกรมกับ Asterisk และจะสิ้นสุดการทำงานในส่วนนี้เมื่อสิ้นสุดการติดต่อสื่อสารของผู้ใช้ โดยมีการทำงานตามขั้นตอนในรูปที่ 3.5



รูปที่ 3.5 แผนภาพส่วนควบคุมการใช้งาน

จากรูปที่ 3.5 สามารถอธิบายรายละเอียดได้ดังนี้

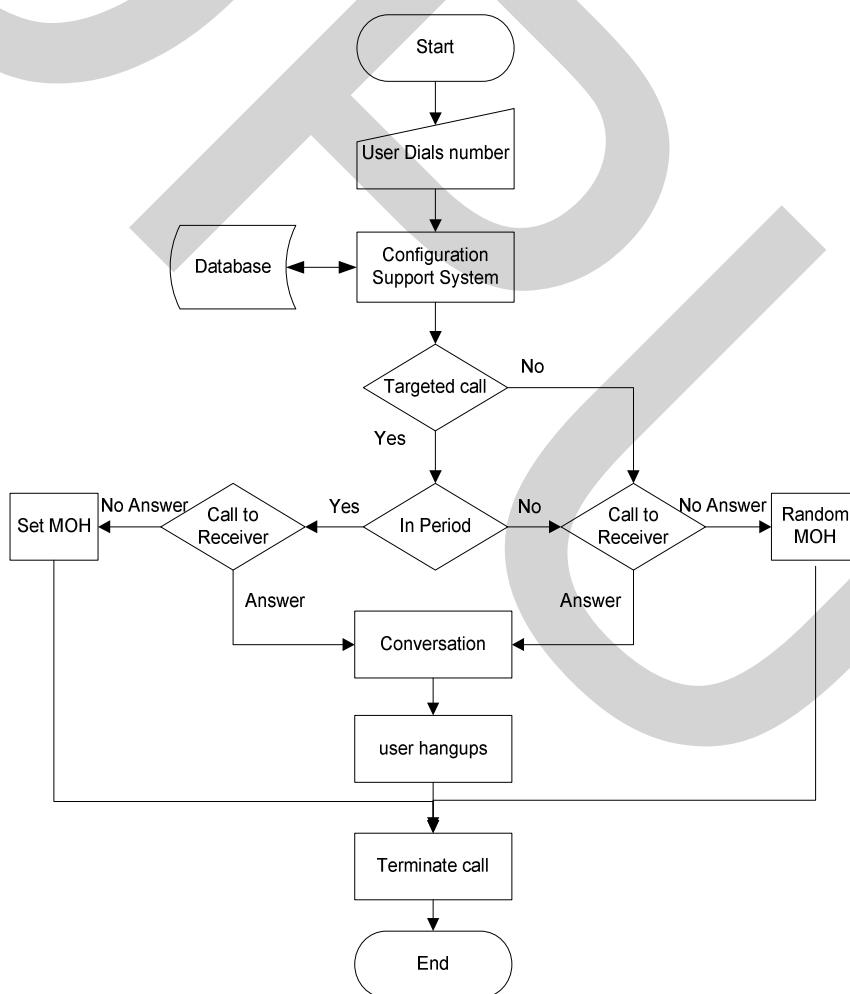
1) เมื่อเบอร์ต้นทางโทรหาเบอร์ปลายทาง ระบบจะทำการดึงข้อมูลจาก database และตรวจสอบการตั้งค่าของเบอร์ปลายทาง ในกรณีนี้คือ Targeted call

2) กรณีที่เป็น Targeted call ระบบจะทำงานตามรูปแบบที่ได้กำหนดไว้

3) เมื่อสิ้นสุดการโทรจะมีการเก็บรายละเอียด CDR

3.4.2.3 ขั้นตอนการออกแบบการทำงานของระบบบริการเสียงเพลงรอสาย สำหรับเบอร์พิเศษ

ในส่วนของการทำงานตาม วัน-เวลา เสียงเพลงรอสายสำหรับเบอร์พิเศษ ระบบจะทำการตรวจสอบวัน-เวลา ปัจจุบัน ว่าอยู่ในช่วงวัน-เวลา ที่ได้กำหนดหรือไม่ โดยมีการรับส่งค่าระหว่าง Asterisk และ AGI ดังแสดงในรูปที่ 3.6



รูปที่ 3.6 การทำงานของระบบบริการเสียงเพลงรอสาย

จากรูปที่ 3.6 อธิบายการทำงานของระบบบริการเสียงเพลงรอสาย ได้ดังนี้

- 1) ผู้ใช้งานโทรเข้า
- 2) ระบบอ่านข้อมูลจาก database ของระบบบริการเสริม
- 3) ระบบตรวจสอบการตั้งค่าบริการเสียงเพลงรอสาย

3.1) กรณีที่ไม่ได้ตั้งค่าไว้

ถ้ามีเพลงในอัลบั้มจะได้ยินเสียงเพลงรอสายของปลายทางวนไปเรื่อยๆ และในแต่ละครั้งที่ทำการโทรจะได้ยินเพลงที่ 1 โดยการสุ่ม (Random Music On Hold: MOH)

3.2) กรณีมีการตั้งค่าไว้และระบบจะตรวจสอบวัน-เวลา ปัจจุบัน

3.2.1) กรณีที่ไม่อยู่ในช่วงวัน-เวลาที่ตั้งไว้

เมื่อโทรและปลายทางยังไม่รับสายจะได้ยิน เสียงเพลงรอสายของปลายทางวนไปเรื่อยๆ และในแต่ละครั้งที่ทำการโทรจะได้ยินเพลงที่ 1 โดยการสุ่ม

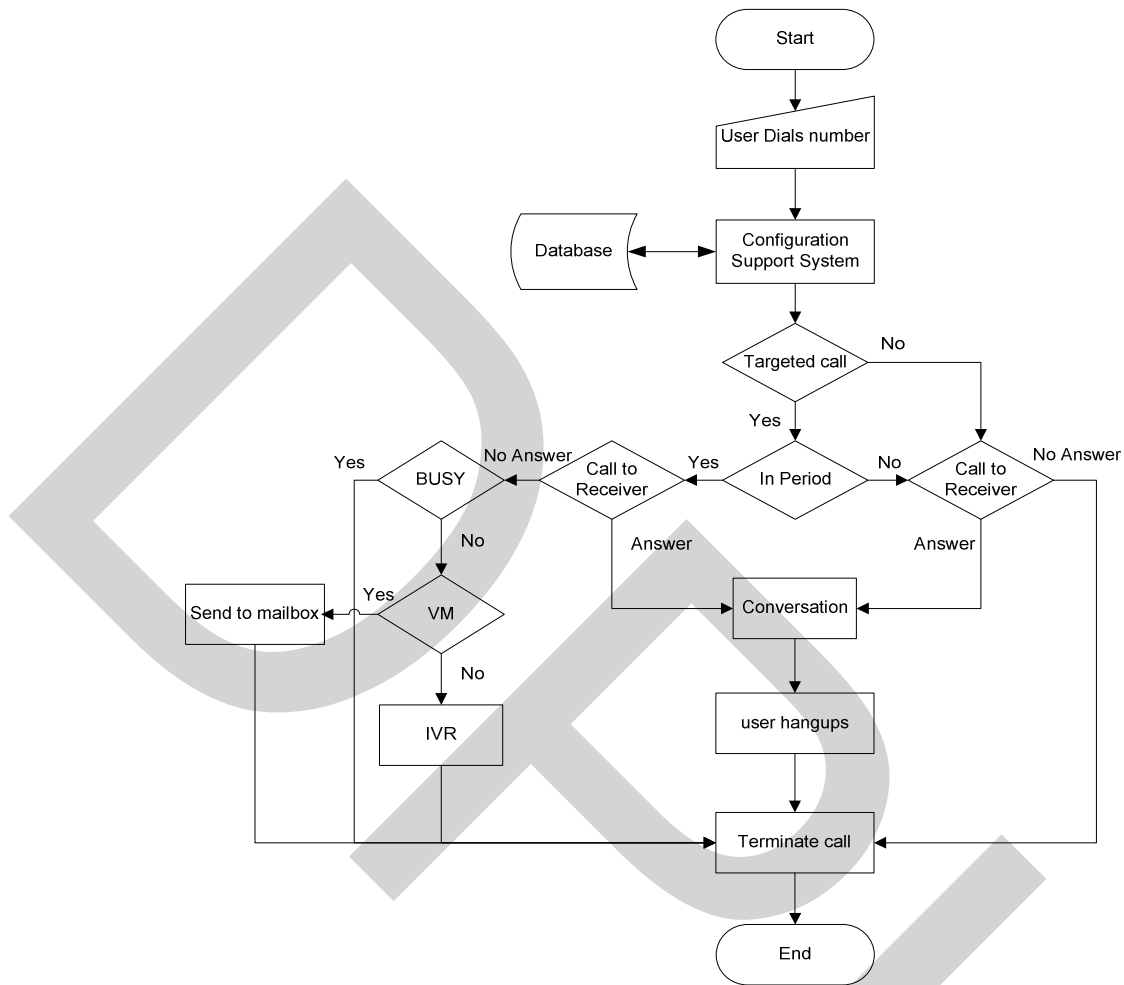
3.2.2) กรณีอยู่ในช่วงวัน-เวลาที่ตั้งไว้

เมื่อโทรและปลายทางยังไม่รับสายจะได้ยิน เสียงเพลงรอสายที่ปลายทางตั้งไว้

4) ผู้ใช้งานวางสายภายหลังสิ้นสุดการสนทนา

3.4.2.4 ขั้นตอนการออกแบบการทำงานของระบบบริการเลือกไม่รับสาย สำหรับเบอร์พิเศษ

ในส่วนของการทำงานตาม วัน-เวลา เลือกไม่รับสายสำหรับเบอร์พิเศษ ระบบจะทำการตรวจสอบวัน-เวลา ปัจจุบัน ว่าอยู่ในช่วงวัน-เวลา ที่ได้กำหนดหรือไม่ โดยมีการรับส่งค่าระหว่าง Asterisk และ AGI ดังแสดงในภาพที่ 3.7



รูปที่ 3.7 การทำงานของระบบบริการเลือกไม่รับสาย

จากรูปที่ 3.7 อธิบายการทำงานของระบบบริการเลือกไม่รับสาย ได้ดังนี้

- 1) ผู้ใช้งานโทรเข้า
- 2) ระบบอ่านข้อมูลจาก database ของระบบบริการเสริม
- 3) ระบบตรวจสอบการตั้งค่าบริการเลือกไม่รับสาย

กรณีมีการตั้งค่า และอยู่ในช่วงวัน-เวลาที่ตั้งไว้ ระบบจะตรวจสอบประเภทบริการเลือกไม่รับสายที่ตั้งไว้ ดังนี้

- 3.1) กรณีที่ตั้ง BUSY จะได้ยินเสียง BUSY
- 3.2) กรณีที่ตั้ง VM จะส่งข้อความเข้า mailbox เบอร์ปลายทาง
- 3.3) กรณีที่ตั้ง IVR จะได้ยินเสียงระบบตอบรับที่ได้ตั้งไว้
- 4) ผู้ใช้งานวางสายภายหลังสิ้นสุดการสนทนา

3.4.2.5 ขั้นตอนการออกแบบส่วนติดต่อผู้ใช้แบบเว็บ (Web Interface)

การออกแบบในส่วนนี้จะใช้รูปแบบของ Graphic User Interface เพื่อให้ง่ายต่อการใช้งาน โดยให้มีช่องรับข้อความที่เพียงพอและเป็นต่อการใช้งานระบบเสียงรอสายและเลือกไม่รับสายร่วมกับ Asterisk โดยใช้ภาษา HTML, PHP script และ Java script เพื่อช่วยในการประมวลผล สำหรับการออกแบบและพัฒนาส่วนติดต่อผู้ใช้ จะแบ่งเป็น 2 ส่วน คือส่วนผู้ดูแลระบบ และส่วนผู้ใช้งานทั่วไป โดยมีรายละเอียดส่วนประกอบหน้าเว็บ ดังนี้

ส่วนผู้ดูแลระบบ ประกอบด้วย

1) หน้าหลัก (Menu) ในส่วนนี้ผู้ดูแลระบบสามารถดูข้อมูลทั่วไป แก่ใจ user และ password ของตนเองได้

2) การตั้งค่าการใช้งานเครื่อง IP-PBX ผู้ดูแลระบบจะกำหนดข้อมูลของเครื่องที่ต้องการติดตั้ง เช่นหมายเลข IP address, login, password และ version asterisk ที่ได้ทำการติดตั้งไว้ในครั้งแรก เพื่อทำการปรับปรุงไฟล์เก่าและสร้างไฟล์ใหม่ที่เครื่องนั้นๆ

3) การกำหนดค่าในระบบ มีดังนี้

3.1) เสียงเพลงรอสาย ผู้ดูแลระบบจะต้องกำหนดเสียงเพลงรอสายเข้าระบบเพื่อให้ผู้ใช้งานทั่วไปเลือกเสียงเพลงจากในระบบได้

3.2) เสียงตอบรับอัตโนมัติ ผู้ดูแลระบบจะต้องกำหนดเสียงตอบรับอัตโนมัติเข้าระบบเพื่อให้ผู้ใช้งานไปตั้งค่า Call Screening แบบ IVR ได้ ในขั้นตอนนี้เมื่อผู้ดูแลระบบเพิ่มหรือเปลี่ยนเสียงตอบรับอัตโนมัติ ระบบจะตรวจสอบว่ามีเครื่อง IP-PBX หมายเลขอะไรบ้างที่ได้ติดตั้งอยู่ในระบบ เพื่อที่จะเพิ่มหรือเปลี่ยนเสียงตอบรับอัตโนมัติที่เครื่องนั้น ๆ

3.3) การตั้งค่าผู้ใช้งาน ผู้ดูแลระบบจะกำหนดค่าผู้ใช้งานเพื่อให้สามารถใช้งานบริการได้

4) บริการเสริม

4.1) อัลบั้มเพลงส่วนตัว ในส่วนนี้ผู้ดูแลระบบหรือผู้ใช้งานสามารถกำหนดเพลงเข้าอัลบั้มเพื่อที่สามารถนำไปตั้งค่าเสียงเพลงรอสายได้ โดยสามารถเลือกเพลงจากเพลงในระบบหรือเพิ่มเองได้

4.2) บริการเสียงเพลงรอสาย ผู้ดูแลระบบหรือผู้ใช้งานสามารถตั้งค่าตามวัน – เวลา แก่ใจ หรือลบได้

4.3) บริการเลือกไม่รับสาย ผู้ดูแลระบบหรือผู้ใช้งานสามารถตั้งค่าตามวัน – เวลา แก่ใจ หรือลบได้

5) ปริมาณการใช้งาน CDR เพื่อให้สามารถดูรายละเอียดการโทรของผู้ใช้งานแต่ละคนได้

ส่วนผู้ใช้ทั่วไป ประกอบด้วย

1) หน้าหลัก สามารถแก้ไข user และ password ได้

2) บริการเสริม

2.1) อัลบั้มเพลงส่วนตัว สามารถกำหนดเพลงเข้าอัลบั้มจากเพลงในระบบ หรือเพิ่มเองได้

2.2) บริการเสียงเพลงรอสาย สามารถตั้งค่าตามวัน – เวลา แก้ไข หรือลบได้

2.3) บริการเลือกไม่รับสาย สามารถตั้งค่าตามวัน – เวลา แก้ไข หรือลบได้

2.4) ปริมาณการใช้งาน CDR เพื่อให้สามารถดูรายละเอียดการโทรของตัวเองได้

3.4.2.6 ขั้นตอนการออกแบบฐานข้อมูล

ในส่วนของฐานข้อมูล Asterisk จะประกอบด้วย 4 ตารางได้แก่ ตาราง voicemail_users, cdr, musiconhold และ extensions_table ดังแสดงในตารางที่ ข.1- ข.4 ในภาคผนวก ข ผู้วิจัยได้สร้างตารางเพิ่มเติมอีก 6 ตาราง ได้แก่ ตาราง cs, rbt, sipusers, application_table, myalbum และ device เพื่อเก็บรายละเอียดต่าง ๆ ดังแสดงในพจนานุกรมข้อมูล ในตารางที่ 3.2 - 3.7

ตารางที่ 3.2 ตาราง cs

ชื่อคอลัมน์	ชนิดข้อมูล	ค่าจำเพาะ	คำอธิบาย
id	int(11)		ลำดับ
message	varchar(20)	NOT NULL	ชื่อ message ของระบบ
description	varchar(20)	NOT NULL	ชื่อข้อความที่แสดง

ตารางที่ 3.3 ตาราง rbt

ชื่อคอลัมน์	ชนิดข้อมูล	ค่าจำเพาะ	คำอธิบาย
id	int(11)		ลำดับ
songdefault	varchar(20)	NOT NULL	ชื่อ เพลง ของระบบ
songname	varchar(20)	NOT NULL	ชื่อเพลงที่แสดง

ตารางที่ 3.4 ตาราง sipusers

ชื่อคอลัมน์	ชนิดข้อมูล	ค่าจำเพาะ	คำอธิบาย
id	int(11)	AUTO_INCREMENT	ลำดับ
name	varchar(10)	NOT NULL	หมายเลข extension เช่น 5000
ipaddr	int(5)	NOT NULL	IP address IP PBX
port	varchar(20)	NOT NULL	Port
regseconds	int(11)	NOT NULL	เวลาในการ register
defaultuser	varchar(10)	NOT NULL	หมายเลข extension เช่น 5000
fullcontact	varchar(35)	NULL	
regserver	varchar(20)	NULL	
useragent	varchar(20)	NOT NULL	softphone
lastms	int(11)	NOT NULL	
host	varchar(40)	NOT NULL	กำหนดเป็น dynamic
type	enum('friend','user','peer')	NOT NULL	กำหนดเป็น friend
context	varchar(40)	NOT NULL	ชื่อ context
secret	varchar(40)	NOT NULL	รหัสผ่าน
qualify	varchar(40)	NOT NULL	กำหนดเป็น yes
canreinvite	varchar(10)	NOT NULL	กำหนดเป็น yes
authu	varchar(10)	NOT NULL	สิทธิ์การใช้งานระบบ เป็น admin หรือ user
IP-PBX	varchar(20)	NOT NULL	หมายเลขเครื่อง IP-PBX ที่ connect

ตารางที่ 3.5 ตาราง application_table

ชื่อคอลัมน์	ชนิดข้อมูล	ค่าจำเพาะ	คำอธิบาย
id	int(11)		ลำดับ
own_extension	varchar(10)	NOT NULL	หมายเลข extension ผู้ใช้
blockExten	varchar(10)	NOT NULL	หมายเลข extension คนพิเศษ
activity	varchar(10)	NOT NULL	ประเภทการตั้งค่า
song	varchar(20)	NOT NULL	เพลง หรือ message
start_time	varchar(5)	NOT NULL	เวลาเริ่มต้น
stop_time	varchar(5)	NOT NULL	เวลาสิ้นสุด
start_date	varchar(5)	NOT NULL	วันเริ่มต้น
stop_date	varchar(5)	NOT NULL	วันสิ้นสุด
start_day	varchar(5)	NOT NULL	วันที่เริ่มต้น
stop_day	varchar(5)	NOT NULL	วันที่สิ้นสุด
month	varchar(5)	NOT NULL	เดือน
album	varchar(10)	NOT NULL	ชื่ออัลบั้มเพลงของแต่ละหมายเลข

ตารางที่ 3.6 ตาราง myalbum

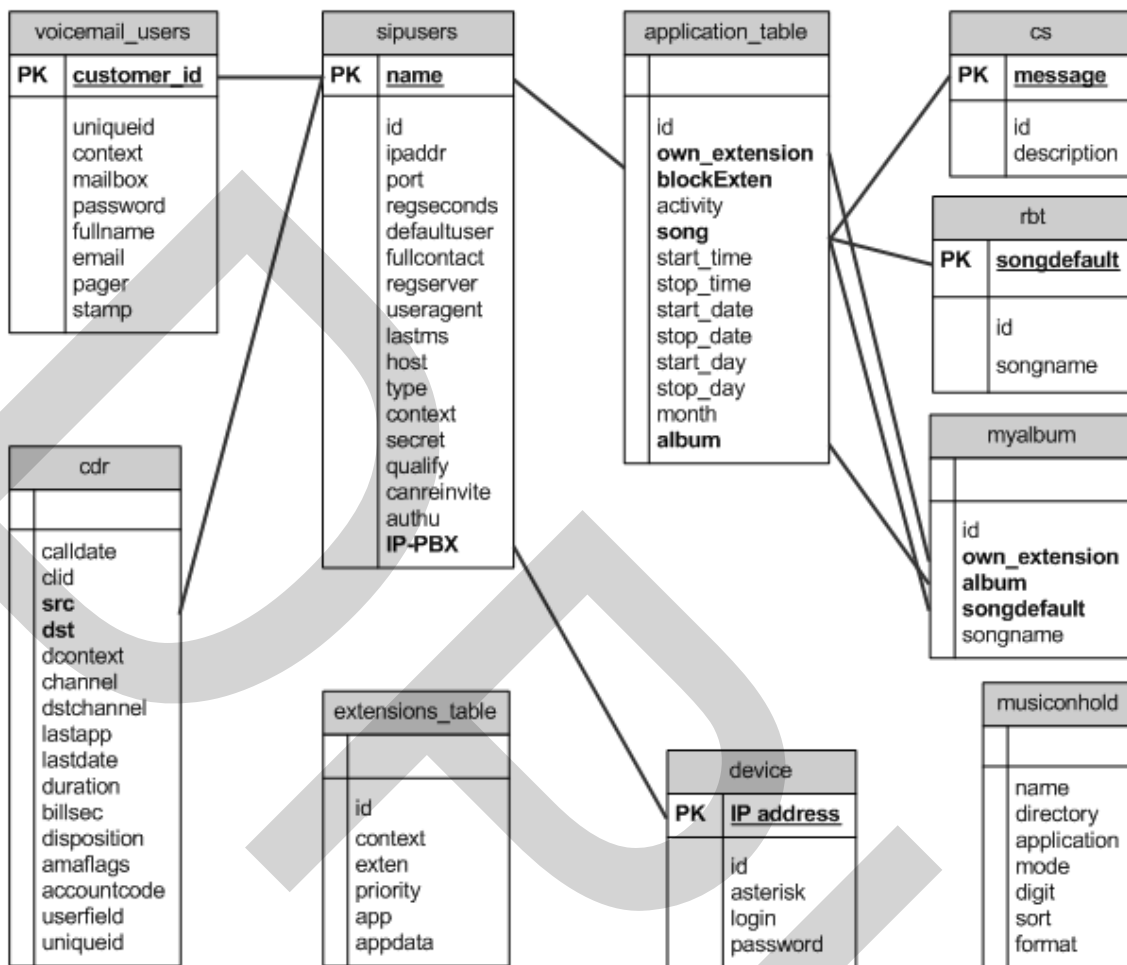
ชื่อคอลัมน์	ชนิดข้อมูล	ค่าจำเพาะ	คำอธิบาย
id	int(11)		ลำดับ
own_extension	varchar(20)	NOT NULL	หมายเลข extension ผู้ใช้
album	varchar(20)	NOT NULL	ชื่ออัลบั้มเพลงของแต่ละหมายเลข
songdefault	varchar(20)	NOT NULL	ชื่อเพลงของระบบ
songname	varchar(20)	NOT NULL	ชื่อเพลงที่แสดง

ตารางที่ 3.7 ตาราง device

ชื่อคอลัมน์	ชนิดข้อมูล	ค่าจำเพาะ	คำอธิบาย
id	int(11)		ลำดับ
asterisk	varchar(20)	NOT NULL	version ของ IP-PBX
IP address	varchar(20)	NOT NULL	หมายเลขเครื่อง IP-PBX
login	varchar(20)	NOT NULL	User name login server
password	varchar(20)	NOT NULL	Password login server

จากที่ผู้วิจัยได้สร้างฐานข้อมูลเพิ่มเติมจากฐานข้อมูล Asterisk โดยมีวัตถุประสงค์ ดังนี้

- 1) ตาราง sipusers ได้เพิ่มข้อมูลพืว authu และ IP-PBX เพื่อทำการตรวจสอบสิทธิ์ผู้ใช้งาน และเครื่อง IP-PBX ที่ register
- 2) ตาราง application_table เพื่อเก็บข้อมูลการตั้งค่าบริการเสียงรอสายและเลือกไม่รับสาย ที่ผู้ใช้งานกำหนดขึ้นเองจากหน้าเว็บรับข้อมูล
- 3) ตาราง cs เพื่อเก็บรายละเอียดเสียงข้อความตอบรับอัตโนมัติ ที่ผู้ดูแลระบบกำหนดขึ้นผ่านหน้าเว็บรับข้อมูล
- 4) ตาราง rbt เพื่อเก็บรายละเอียดเสียงเพลงรอสาย ที่ผู้ดูแลระบบกำหนดขึ้นผ่านหน้าเว็บรับข้อมูล
- 5) ตาราง myalbum เพื่อเก็บรายละเอียดเสียงเพลงรอสายของแต่ละผู้ใช้ ที่ผู้ใช้ได้ทำการเลือกจากระบบหรือกำหนดขึ้นเองผ่านหน้าเว็บรับข้อมูล
- 6) ตาราง device เพื่อเก็บข้อมูลของเครื่อง IP-PBX ที่มีการใช้งานระบบ ซึ่งผู้ดูแลระบบกำหนดขึ้นผ่านหน้าเว็บรับข้อมูล



รูปที่ 3.8 ความสัมพันธ์ระหว่างตาราง

จากภาพที่ 3.8 แสดงความสัมพันธ์ระหว่างตารางดังนี้

- 1) ตาราง cdr สัมพันธ์กับตาราง sipusers โดยฟิลด์ src กับ dst จัดเก็บหมายเลข Extensions ที่เก็บอยู่ในฟิลด์ name ของผู้ใช้แต่ละคน สำหรับการแสดงข้อมูลการโทร
- 2) ตาราง voicemail_users สัมพันธ์กับตาราง sipusers โดยฟิลด์ customer_id จัดเก็บหมายเลข Extensions ที่เก็บอยู่ในฟิลด์ name ของผู้ใช้สำหรับการส่งข้อมูลเข้า mailbox
- 3) ตาราง sipusers สัมพันธ์กับตาราง device โดยฟิลด์ IP-PBX จัดเก็บหมายเลขเครื่อง IP address ที่ผู้ใช้งานเชื่อมต่ออยู่ สำหรับการส่งข้อมูล
- 4) ตาราง application_table สัมพันธ์กับตาราง sipusers โดยฟิลด์ own_extension กับ blockExten จัดเก็บหมายเลข Extensions ที่เก็บอยู่ในฟิลด์ name ของผู้ใช้แต่ละคน สำหรับการตั้งค่า Ring Back Tone และ Call Screening

5) ตาราง application_table สัมพันธ์กับตาราง cs ซึ่งเป็นตารางเก็บข้อความ message ของระบบ โดยฟิลด์ song จัดเก็บ message ตามที่ผู้ใช้เลือกเพื่อตั้งค่า Call Screening แบบ IVR

6) ตาราง application_table สัมพันธ์กับตาราง rbt ซึ่งเป็นตารางเก็บเพลงรอสาย song ของระบบ โดยฟิลด์ song จัดเก็บ song ตามที่ผู้ใช้เลือกเพื่อตั้งค่า Ring Back Tone

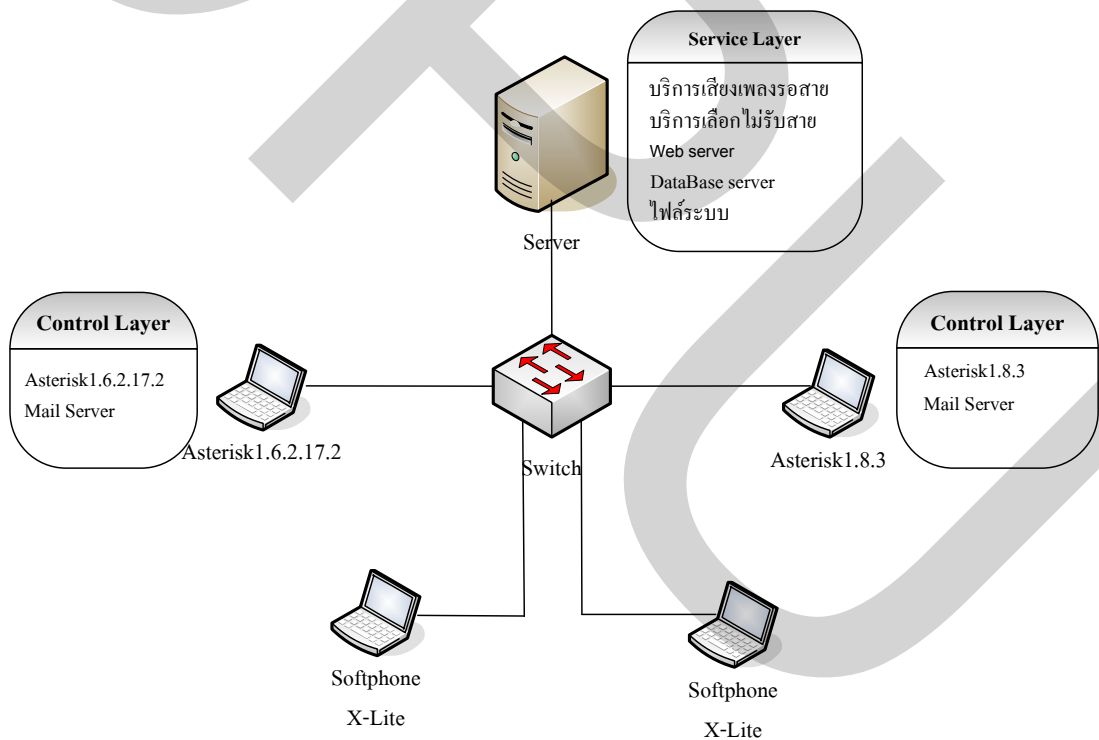
7) ตาราง application_table สัมพันธ์กับตาราง myalbum ซึ่งเป็นตารางเก็บเพลงรอสาย song ของผู้ใช้แต่ละคน สำหรับการตั้งค่า Ring Back Tone

สำหรับตาราง extensions_table และตาราง musiconhold ไม่มีความสัมพันธ์กับตารางใดเลย เนื่องจากข้อมูลที่จัดเก็บเป็นข้อมูลที่ใช้เฉพาะกรณี คือ การเรียกใช้งาน dial plan และการกำหนดค่าทั่วไปในการเรียกใช้งานจาก Asterisk

บทที่ 4

การทดสอบระบบ

ในการทดสอบระบบบริการเสียงรอสายและเลือกไม่รับสายในโครงข่าย VoIP ตามสถาปัตยกรรม NGN ได้ทดสอบโดยเริ่มตั้งแต่การตั้งค่าการใช้งานเครื่อง IP-PBX การกำหนดเสียงเพลงรอสายเข้าระบบ การกำหนดข้อความตอบรับอัตโนมัติเข้าระบบ การสร้างผู้ใช้งาน โดยผู้ดูแลระบบ การตั้งค่าเพลงในอัลบั้มส่วนตัว การตั้งค่าบริการเสียงรอสาย การตั้งค่าบริการเลือกไม่รับสาย โดยผู้ใช้ทั่วไป และการดูรายละเอียดการใช้งาน CDR ซึ่งทางผู้วิจัยได้ติดตั้งระบบเพื่อใช้ในการทดสอบดังแสดงรูปที่ 4.1



รูปที่ 4.1 ภาพรวมการทดสอบระบบ

จากรูปที่ 4.1 ภาพรวมการทดสอบระบบประกอบด้วย

1) เครื่องคอมพิวเตอร์ในระดับชั้น Service Layer มีการติดตั้งบริการเสียงเพลงรอสาย, บริการเลือกไม่รับสาย, Web server, DataBase server และเก็บข้อมูลไฟล์ที่จำเป็นต่อการใช้งานระบบ

2) เครื่องคอมพิวเตอร์ Server ในระดับชั้น Control Layer

เครื่องที่ 1 มีการติดตั้ง Asterisk 1.6.2.17.2 และ Mail Server

เครื่องที่ 2 มีการติดตั้ง Asterisk 1.8.3 และ Mail Server

3) เครื่องคอมพิวเตอร์ Client ในระดับชั้น Access Layer จำนวน 2 เครื่อง ติดตั้ง X-Lite

แบ่งการทดสอบออกเป็น 11 หัวข้อ พร้อมกับวัตถุประสงค์การทดสอบของแต่ละหัวข้อดังต่อไปนี้คือ

1) การทดสอบการตั้งค่าการใช้งานเครื่อง IP-PBX จะทำหลังจากที่ได้มีการติดตั้ง Software ต่าง ๆ ที่เครื่อง IP-PBX เรียบร้อยแล้ว โดยการตั้งค่าส่วนนี้จะเป็นการปรับปรุงไฟล์เก่าที่ได้ติดตั้งไว้ในครั้งแรก การสร้างไฟล์ใหม่และการคัดลอกไฟล์เสียง IVR ที่จำเป็นต่อการใช้งานระบบ เพื่อให้ระบบสามารถเรียกใช้งานที่ตัว Application server ได้ ในส่วนนี้ผู้วิจัยได้ทดสอบตั้งค่าที่เครื่อง Asterisk 1.6 และ Asterisk 1.8

2) การทดสอบการกำหนดเสียงเพลงรอสายในระบบ ผู้ดูแลระบบจะกำหนดเสียงเพลงรอสายเข้าไปในระบบ เพื่อให้ผู้ใช้สามารถเลือกเพลงรอสายจากในระบบไปเก็บในอัลบั้มของตัวเองได้

3) การทดสอบการกำหนดเสียงตอบรับอัตโนมัติในระบบ ผู้ดูแลระบบจะกำหนดเสียงตอบรับอัตโนมัติเข้าไปในระบบ เพื่อให้สามารถใช้งานบริการเลือกไม่รับสายแบบ IVR ได้

4) การทดสอบการตั้งค่าผู้ใช้งานในระบบ จะตั้งค่าผู้ใช้งานก่อนที่จะมีการใช้งานบริการเสียงรอสายและเลือกไม่รับสาย

5) การทดสอบการตั้งค่าอัลบั้มเพลงส่วนตัว เป็นการทดสอบการกำหนดเสียงเพลงรอสายเก็บในอัลบั้มเพลงของตัวเองจะต้องทำการใช้งานบริการเสียงเพลงรอสาย ผู้ใช้ทั่วไปสามารถเลือกเพลงรอสายเก็บในอัลบั้มได้จาก 2 รูปแบบ คือ เลือกเพลงรอสายจากในระบบที่ผู้ดูแลระบบได้กำหนดไว้ตามการทดสอบในหัวข้อการกำหนดเสียงเพลงรอสายในระบบ และเลือกจากการ upload ได้ด้วยตัวเอง

6) การทดสอบตั้งค่าบริการเสียงเพลงรอสาย เป็นการทดสอบตั้งค่าเมื่อมีเพลงในอัลบั้มส่วนตัวแล้ว โดยกำหนดวัน-เวลา พร้อมกับเสียงเพลงรอสาย

7) การทดสอบการตั้งค่าบริการเลือกไม่รับสาย จะทำการทดสอบ 3 รูปแบบ คือ BUSY Tone , IVR และ Voice Mail Box ในกรณีที่ต้องการตั้งค่าเป็น IVR จะทำหลังจากที่ผู้ดูแลระบบเพิ่มเสียงตอบรับอัตโนมัติเข้าในระบบเรียบร้อยแล้ว

8) การทดสอบการใช้งานโทรออก หลังจากที่ได้มีการตั้งค่าบริการเสียงรอสายเรียบร้อยแล้ว

9) การทดสอบการใช้งานโทรออก หลังจากที่ได้มีการตั้งค่าบริการเลือกไม่รับสายเรียบร้อยแล้ว

10) การทดสอบประสิทธิภาพการทำงานของเครื่องแม่ข่าย Application server

11) การทดสอบประสิทธิภาพการทำงานของเครื่องแม่ข่าย Asterisk 1.8.3

การทดสอบที่ 1 การตั้งค่าการใช้งานเครื่อง IP-PBX

วัตถุประสงค์

1) เพื่อทดสอบว่าระบบสามารถปรับปรุงไฟล์ extconfig.conf , sip.conf, extensions.conf, cdr_mysql.conf, cdr.conf, และ res_mysql.conf ที่เครื่อง Asterisk 1.6 ให้โดยอัตโนมัติ

2) เพื่อทดสอบว่าระบบสามารถปรับปรุงไฟล์ extconfig.conf , sip.conf, extensions.conf, cdr_mysql.conf, cdr.conf, และ res_config_mysql.conf ที่เครื่อง Asterisk 1.8 ให้โดยอัตโนมัติ

3) เพื่อทดสอบว่าระบบสามารถสร้างไฟล์ callscreening.php ให้โดยอัตโนมัติ

4) เพื่อทดสอบว่าระบบสามารถคัดลอกไฟล์เสียง IVR จาก application server ไปที่เครื่อง IP-PBX ให้โดยอัตโนมัติ

5) เพื่อทดสอบว่าระบบสามารถจัดเก็บข้อมูลเครื่อง IP-PBX ในฐานข้อมูลให้โดยอัตโนมัติ

การทดสอบนี้จะดำเนินการที่ตัว Application server โดยผู้ดูแลระบบกรอกรายละเอียดข้อมูลเบื้องต้นของเครื่องที่ต้องการติดตั้ง ซึ่งได้ทำการทดสอบตั้งค่าโดยใช้ Asterisk 1.6.2.17.2 จำนวน 15 ครั้ง และ Asterisk 1.8.3 จำนวน 15 ครั้ง รายละเอียดดังแสดงในรูปที่ 4.2 จะได้ผลดังรูปที่ 4.3, 4.4 และ 4.5 ตามลำดับ

การตั้งค่าการใช้งานเครื่อง IP-PBX เมืองต้น

รุ่นเครื่อง IP-PBX :	asterisk 1.6
IP address :	192.168.1.6 *ระบุหมายเลข IP address เช่น 192.168.1.6
login :	pc16
password :	123
<input type="button" value="บันทึก"/> <input type="button" value="ยกเลิก"/>	

รูปที่ 4.2 แสดงตัวอย่างหน้าจอการตั้งค่าการใช้งานเครื่อง IP-PBX เมืองต้น

การพัฒนาระบบเสียงรอสายและเลือกไม่รับสายในโครงข่าย VoIP ตามสถาปัตยกรรม NGN
Development of Ring Back Tone and Call Screening Service System in VoIP Network
complied to Next Generation Network Architecture

นางสาว มยุรัตน์ ขวัญคุณฐิติ รหัส 515620009

การตั้งค่าการใช้งานเครื่อง IP-PBX เมืองต้น

>> [ตั้งค่า IP-PBX เครื่องใหม่](#) <<

รุ่นเครื่อง IP-PBX	IP address	login	password	แก้ไข	ลบ
asterisk 1.6	192.168.1.6	pc16	123	✎	✖

รูปที่ 4.3 แสดงข้อมูลเครื่อง IP-PBX ทั้งหมดในระบบ

```

root@pc16:/var/lib/asterisk/agi-bin# ls
agi-test.agi  callscreening.php  eagi-s
root@pc16:/var/lib/asterisk/agi-bin#
  
```

ระบบสร้างไฟล์ callscreening.php jukebox.agi

รูปที่ 4.4 แสดงไฟล์ callscreening.php

```

root@pc16:/var/lib/asterisk# cd moh
root@pc16:/var/lib/asterisk/moh# ls
5000 5000song1 5000song2 5000song3 5000song4 macroform-cold_day.wav
5000song1 CHANGE macroform-robot_dity.wav
5000song2 CREDIT macroform-the_simplicity.wav
5000song3 ivr macroform-the_simplicity.wav
5000song4 LICENSE-asterisk-moh-opsound-wav
root@pc16:/var/lib/asterisk/moh# cd ivr
root@pc16:/var/lib/asterisk/moh/ivr# ls
message1.wav message2.wav message3.wav message4.wav message5.wav
root@pc16:/var/lib/asterisk/moh/ivr#

```

สร้าง folder ivr เพื่อเก็บไฟล์เสียง ivr

ไฟล์เสียง ivr ในระบบที่เก็บใน folder ivr

รูปที่ 4.5 แสดงไฟล์เสียง IVR

ตารางที่ 4.1 ผลการทดสอบการติดตั้งเครื่อง IP-PBX ด้วย Asterisk 1.6

หัวข้อการทดสอบ	ผลการทำงาน 15 ครั้ง		ความถูกต้องคิดเป็นเปอร์เซ็นต์
	ถูก	ผิด	
1. สามารถปรับปรุงไฟล์ extconfig.conf ได้	15	0	100%
2. สามารถปรับปรุงไฟล์ sip.conf ได้	15	0	100%
3. สามารถปรับปรุงไฟล์ res_mysql.conf ได้	15	0	100%
4. สามารถปรับปรุงไฟล์ extensions.conf ได้	15	0	100%
5. สามารถปรับปรุงไฟล์ cdr_mysql.conf ได้	15	0	100%
6. สามารถปรับปรุงไฟล์ cdr.conf ได้	15	0	100%
7. สามารถสร้างไฟล์ callscreening.php ได้	15	0	100%
8. สามารถคัดลอกไฟล์เสียง IVR จาก application server ไปที่เครื่อง IP-PBX ได้	15	0	100%
9. สามารถจัดเก็บข้อมูลเครื่อง IP-PBX ใน database ได้	15	0	100%

ตารางที่ 4.2 ผลการทดสอบการติดตั้งเครื่อง IP-PBX ด้วย Asterisk 1.8

หัวข้อการทดสอบ	ผลการทำงาน 15 ครั้ง		ความถูกต้องคิด เป็นเปอร์เซ็นต์
	ถูก	ผิด	
1. สามารถปรับปรุงไฟล์ extconfig.conf ได้	15	0	100%
2. สามารถปรับปรุงไฟล์ sip.conf ได้	15	0	100%
3. สามารถปรับปรุงไฟล์ res_config_mysql.conf ได้	15	0	100%
4. สามารถปรับปรุงไฟล์ extensions.conf ได้	15	0	100%
5. สามารถปรับปรุงไฟล์ cdr_mysql.conf ได้	15	0	100%
6. สามารถปรับปรุงไฟล์ cdr.conf ได้	15	0	100%
7. สามารถสร้างไฟล์ callscreening.php ได้	15	0	100%
8. สามารถคัดลอกไฟล์เสียง IVR จาก application server ไปที่เครื่อง IP-PBX ได้	15	0	100%
9. สามารถจัดเก็บข้อมูลเครื่อง IP-PBX ใน database ได้	15	0	100%

จากผลการทดสอบในตารางที่ 4.1 และ 4.2 พบว่าระบบสามารถทำงานได้ถูกต้องทั้งในเครื่อง Asterisk 1.6 และ Asterisk 1.8

การทดสอบที่ 2 การกำหนดเสียงเพลงรอสายในระบบ

วัตถุประสงค์

1) เพื่อทดสอบว่าระบบสามารถจัดเก็บเพลงที่เพิ่มเข้าระบบไว้ที่ path /usr/src/rbt ของ application server ให้โดยอัตโนมัติ

2) เพื่อทดสอบว่าระบบสามารถเก็บรายชื่อเพลงไว้ใน database ให้โดยอัตโนมัติ

ทดสอบโดยการเพิ่มเสียงเพลงรอสายเข้าในระบบจากตัว Application server โดยการเพิ่ม หรือเปลี่ยนเสียงเพลงรอสายจำนวน 30 ครั้ง ได้ผลดังแสดงในรูปที่ 4.6, 4.7 และตารางที่ 4.3

หน้าหลัก		เสียงเพลงรอสาย				
การกำหนดค่าเริ่มต้นอัตโนมัติ การตั้งค่าการใช้งานเครื่อง IP-PBX		เพลงที่	ชื่อเพลง	เพิ่ม	เปลี่ยน	ลบ
การกำหนดค่าในระบบ - เสียงเพลงรอสาย - เสียงตอบรับอัตโนมัติ - การตั้งค่าผู้ใช้งาน		1	Closing Time.wav			
บริการเสริม - อัลบั้มเพลงส่วนตัว - บริการเสียงเพลงรอสาย - บริการเลือกในรับสาย		2	How Do I Live.wav			
		3	When You Say Nothing At All.wav			
		4	เก็บดาว.wav			
		5	ความอ่อนแอ.wav			
		6	คำตอบสุดท้าย.wav			
		7	คิดมาก.wav			
		8	คิดเหมือนกันหรือเปล่า.wav			
		9	ทันอู่ในใจ.wav			
		10	สัมหวนน้ำตาลเบรียว.wav			

รูปที่ 4.6 แสดงเสียงเพลงรอสายในระบบ

```

root@DBserver:/usr/src/rbt# is
song10.wav song2.wav song4.wav song6.wav song8.wav
song1.wav song3.wav song5.wav song7.wav song9.wav
root@DBserver:/usr/src/rbt#

```

รูปที่ 4.7 แสดงเสียงเพลงรอสายที่ path /usr/src/rbt ของ application server

ตารางที่ 4.3 ผลการทดสอบการกำหนดเสียงเพลงรอสายในระบบ

หัวข้อการทดสอบ	ผลการทำงาน 30 ครั้ง		ความถูกต้องคิดเป็นเปอร์เซ็นต์
	ถูก	ผิด	
1. เพลงที่เพิ่มเข้าระบบมีการจัดเก็บที่ path /usr/src/rbt	30	0	100%
2. สามารถเก็บรายชื่อเพลงไว้ใน database	30	0	100%

จากผลการทดสอบในตารางที่ 4.3 พบว่าระบบสามารถทำงานได้ถูกต้อง โดยตรวจสอบจำนวนเพลงที่ได้เพิ่มในระบบ โดยตรวจสอบทั้งจาก database ที่จัดเก็บรายชื่อเพลงไว้ และจำนวนเพลงที่ path /usr/src/rbt พบว่าระบบทำงานได้ถูกต้อง 100%

การทดสอบที่ 3 การกำหนดเสียงตอบรับอัตโนมัติในระบบ

วัตถุประสงค์

1) เพื่อทดสอบว่าระบบสามารถจัดเก็บข้อความที่เพิ่มเข้าระบบไว้ที่ path /usr/src/ivr ของ application server และที่ path /var/lib/asterisk/moh/ivr/ ของเครื่อง IP-PBX ทั้งหมดที่มีอยู่ในระบบให้โดยอัตโนมัติ

2) เพื่อทดสอบว่าระบบสามารถเก็บชื่อข้อความไว้ใน database ให้โดยอัตโนมัติ ทดสอบโดยการเพิ่มเสียงเพลงรอสายเข้าในระบบจากตัว application server โดยการเพิ่ม หรือเปลี่ยนเสียงเพลงรอสาย จำนวน 30 ครั้ง ได้ผลดังแสดงในรูปที่ 4.8, 4.9 และตารางที่ 4.4

หน้าหลัก		เสียงตอบรับอัตโนมัติ			
การกำหนดค่าเริ่มต้นอัตโนมัติ	ข้อความที่	ข้อความ	เพิ่ม	เปลี่ยน	ลบ
การตั้งค่าการไต่ถามเครื่อง IP-PBX	1	กำลังประชุม.wav			
การกำหนดค่าในระบบ	2	ตั้งใจเรียน.wav			
- เสียงเพลงรอสาย	3	มีปัญหาหัวใจ.wav			
- เสียงตอบรับอัตโนมัติ	4	ไม่สะดวกรับสาย.wav			
- การตั้งค่ามูโฆงาน	5	อยู่ต่างประเทศ.wav			

รูปที่ 4.8 แสดงเสียงตอบรับอัตโนมัติในระบบ

```

root@DBserver:~# cd /usr/src/ivr/
root@DBserver:/usr/src/ivr# ls
message1.wav message2.wav message3.wav message4.wav message5.wav
root@DBserver:/usr/src/ivr#

```

ไฟล์เสียง ivr ในระบบทั้งหมด

รูปที่ 4.9 แสดงเสียงตอบรับอัตโนมัติที่ path /usr/src/ivr ของ application server

ตารางที่ 4.4 ผลการทดสอบการกำหนดเสียงตอบรับอัตโนมัติในระบบ

หัวข้อการทดสอบ	ผลการทำงาน 30 ครั้ง		ความถูกต้องคิดเป็นเปอร์เซ็นต์
	ถูก	ผิด	
1. ข้อความที่เพิ่มเข้าระบบมีการจัดเก็บที่ path /usr/src/ivr	30	0	100%
2. ข้อความที่เพิ่มเข้าระบบมีการจัดเก็บที่ path /var/lib/asterisk/moh/ivr/	30	0	100%
3. สามารถเก็บรายชื่อข้อความไว้ใน database	30	0	100%

จากผลการทดสอบในตารางที่ 4.4 พบว่าระบบสามารถทำงานได้ถูกต้อง โดยตรวจสอบจำนวนเสียงข้อความที่ได้เพิ่มในระบบ โดยตรวจสอบทั้งจาก database ที่จัดเก็บรายชื่อข้อความไว้จำนวนข้อความที่ path /usr/src/ivr ที่ตัว Application server และจำนวนข้อความที่ path /var/lib/asterisk/moh/ivr/ ที่ตัว IP-PBX พบว่าระบบทำงานได้ถูกต้อง 100%

การทดสอบที่ 4 การตั้งค่าผู้ใช้งานในระบบ

วัตถุประสงค์

1) เพื่อทดสอบว่าระบบสามารถจัดเก็บรายชื่อผู้ใช้งานไว้ใน database ได้ถูกต้อง การทดสอบการตั้งค่าผู้ใช้งานโดยการสมมุติค่าหมายเลขผู้ใช้งานเป็นตัวเลข 4 หลัก ทำการเพิ่มหรือแก้ไขการตั้งค่า จำนวน 30 ครั้ง โดยใส่ข้อมูลรายละเอียดดังรูปที่ 4.10

การตั้งค่าผู้ใช้งานในระบบ	
หมายเลขโทรศัพท์ :	5000 * ระบุหมายเลข 4 หลัก
ชื่อผู้ใช้ :	5000 สิทธิ์ : Admin ▼
รหัสผ่าน :	5000
การใช้งานการเชื่อมต่อ IP-PBX :	192.168.1.6 ▼
การใช้งานส่งข้อความไป E-mail Address :	pc16@pc16 * ระบุ E-mail เช่น pc16@pc16
รหัสผ่านสำหรับเข้ากล่องข้อความเสียง :	5000
<input type="button" value="บันทึก"/> <input type="button" value="ยกเลิก"/>	

รูปที่ 4.10 แสดงตัวอย่างหน้าจอการตั้งค่าผู้ใช้งานในระบบ

หน้าหลัก	ผู้ใช้งานในระบบ																																								
การกำหนดค่าเริ่มต้นอัตโนมัติ การตั้งค่าการใช้งานเครื่อง IP-PBX การกำหนดค่าในระบบ - เสียงเพลงรอสาย - เสียงตอบรับอัตโนมัติ - การตั้งค่าผู้ใช้งาน	>> เพิ่มผู้ใช้งานใหม่ << <table border="1"> <thead> <tr> <th>หมายเลข</th> <th>ชื่อที่แสดง</th> <th>สิทธิ์</th> <th>IP-PBX</th> <th>อีเมลล์</th> <th>กล่องรับข้อความ</th> <th>แก้ไข</th> <th>ลบ</th> </tr> </thead> <tbody> <tr> <td>4000</td> <td>cat</td> <td>admin</td> <td>192.168.1.6</td> <td>pc16@pc16</td> <td>4000</td> <td></td> <td></td> </tr> <tr> <td>5000</td> <td>5000</td> <td>admin</td> <td>192.168.1.6</td> <td>pc16@pc16</td> <td>5000</td> <td></td> <td></td> </tr> <tr> <td>6000</td> <td>dog</td> <td>user</td> <td>192.168.1.6</td> <td>pc16@pc16</td> <td>6000</td> <td></td> <td></td> </tr> <tr> <td>8000</td> <td>8000</td> <td>user</td> <td>192.168.1.8</td> <td>pc18@pc18</td> <td>8000</td> <td></td> <td></td> </tr> </tbody> </table>	หมายเลข	ชื่อที่แสดง	สิทธิ์	IP-PBX	อีเมลล์	กล่องรับข้อความ	แก้ไข	ลบ	4000	cat	admin	192.168.1.6	pc16@pc16	4000			5000	5000	admin	192.168.1.6	pc16@pc16	5000			6000	dog	user	192.168.1.6	pc16@pc16	6000			8000	8000	user	192.168.1.8	pc18@pc18	8000		
หมายเลข	ชื่อที่แสดง	สิทธิ์	IP-PBX	อีเมลล์	กล่องรับข้อความ	แก้ไข	ลบ																																		
4000	cat	admin	192.168.1.6	pc16@pc16	4000																																				
5000	5000	admin	192.168.1.6	pc16@pc16	5000																																				
6000	dog	user	192.168.1.6	pc16@pc16	6000																																				
8000	8000	user	192.168.1.8	pc18@pc18	8000																																				

รูปที่ 4.11 แสดงข้อมูลผู้ใช้งานในระบบ

ตารางที่ 4.5 ผลการทดสอบการตั้งค่าผู้ใช้งาน

หัวข้อการทดสอบ	ผลการทำงาน 30 ครั้ง		ความถูกต้องคิดเป็นเปอร์เซ็นต์
	ถูก	ผิด	
1. สามารถจัดเก็บรายชื่อผู้ใช้ไว้ใน database ได้	30	0	100%

จากผลการทดสอบในตารางที่ 4.5 พบว่าระบบสามารถทำงานได้ถูกต้อง โดยตรวจสอบจำนวนผู้ใช้งานที่ได้เพิ่มในระบบ โดยตรวจสอบทั้งจาก database ที่จัดเก็บข้อมูลผู้ใช้งานไว้จากรูปที่ 4.11 พบว่าระบบทำงานได้ถูกต้อง 100%

การทดสอบที่ 5 การตั้งค่าอัลบั้มเพลงส่วนตัว

วัตถุประสงค์

- 1) เพื่อทดสอบว่าเพลงที่ผู้ใช้แต่ละคนเพิ่มเข้าไปในอัลบั้มมีการเก็บแยก path ที่เครื่อง IP-PBX ตามผู้ใช้งานแต่ละคนได้
- 2) เพื่อทดสอบว่าระบบสามารถเก็บรายชื่อเพลงไว้ใน database โดยแยกแต่ละผู้ใช้ได้ ทดสอบโดยเลือกเพิ่มเสียงเพลงรอสายเข้าไปในอัลบั้มเพลงส่วนตัวจากตัว application server โดยการเพิ่ม หรือเปลี่ยนเสียงเพลงรอสายในอัลบั้ม จำนวน 30 ครั้ง ได้ผลดังแสดงในรูปที่ 4.12, 4.13 และตารางที่ 4.6

หน้าหลัก		อัลบั้มเพลงส่วนตัว				
การกำหนดค่าเริ่มต้นอัตโนมัติ		เพลงที่	ชื่อเพลง	เพิ่ม	เปลี่ยน	ลบ
การตั้งค่าการไม่งานเครื่อง IP-PBX		1	Closing Time.wav			
การกำหนดค่าในระบบ		2	คือเธอ.wav			
- เสียงเพลงรอสาย		3	เก็บดาว.wav			
- เสียงตอบรับอัตโนมัติ		4	มากกว่ารัก.wav			
- การตั้งค่าผู้ใช้งาน		5	คำตอบสุดท้าย.wav			
บริการเสริม						
- อัลบั้มเพลงส่วนตัว						
- บริการเสียงเพลงรอสาย						
- บริการเลือกไม่รับสาย						

รูปที่ 4.12 แสดงเสียงเพลงรอสายในอัลบั้มของตัวเอง

```

root@pc16:/var/lib/asterisk/moh# ls
5000  6000song1  macroform-cold_day.wav
5000song1  6000song2  macroform-robot_dity.wav
5000song2  6000song3  macroform-the_simplicity.wav
5000song3  CHANGELOG  nolo_camp-morning_coffee.wav
5000song4  CREDITS-ast  no_project-system.wav
5000song5  ivr
6000  LICENSE-asterisk-moh-opsound-wav
root@pc16:/var/lib/asterisk/moh#
  
```

เสียงเพลงรอสายของผู้ใช้งานหมายเลข 5000 และ 6000

รูปที่ 4.13 แสดงเสียงเพลงรอสายในอัลบั้มของตัวเองแยก path ที่เครื่อง IP-PBX ตามผู้ใช้งานแต่ละคน

ตารางที่ 4.6 ผลการทดสอบการตั้งค่าเพลงในอัลบั้มตัวเอง

หัวข้อการทดสอบ	ผลการทำงาน 30 ครั้ง		ความถูกต้องคิด เป็นเปอร์เซ็นต์
	ถูก	ผิด	
1. เพลงที่เพิ่มเข้าระบบมีการจัดเก็บที่ path ตามแต่ละ ผู้ใช้ถูกต้อง	30	0	100%
2. สามารถเก็บรายชื่อเพลงไว้ใน database	30	0	100%

จากผลการทดสอบในตารางที่ 4.6 พบว่าระบบสามารถทำงานได้ถูกต้อง โดยตรวจสอบจำนวนเพลงที่ได้เพิ่มในระบบ โดยตรวจสอบทั้งจาก database ที่จัดเก็บรายชื่อเพลงไว้และจำนวนเพลงที่ path เครื่อง IP-PBX ตามแต่ละผู้ใช้ พบว่าระบบทำงานได้ถูกต้อง 100%

การทดสอบที่ 6 การตั้งค่าบริการเสียงเพลงรอสาย

วัตถุประสงค์

- 1) เพื่อทดสอบว่าระบบสามารถเลือกเพลงจากอัลบั้มของตัวเองมาทำการตั้งค่าได้
- 2) เพื่อทดสอบว่าระบบสามารถเก็บรายละเอียดการตั้งค่าตามวัน-เวลา เทศกาลไว้ใน database แยกตามผู้ใช้ได้

ทดสอบโดยตั้งค่าบริการเสียงเพลงรอสายสำหรับหมายเลขพิเศษ ตามวัน-เวลา หรือเทศกาล โดยเพิ่มหรือแก้ไขการตั้งค่าเดิม เพื่อให้การตั้งค่าครอบคลุมตามรูปแบบจึงได้ทำการทดสอบจำนวน 36 ครั้ง ได้ผลดังแสดงในรูปที่ 4.14, 4.15 และตารางที่ 4.7

ตั้งค่า Ring Back Tone

หมายเลขผู้ใช้ :	5000
เบอร์คนที่เศษ :	6000 * ระบุหมายเลข 4 หลัก
ประเภทบริการ :	RBT
ตั้งค่าช่วงเวลา :	<input type="radio"/> ไม่ใช้ <input checked="" type="radio"/> เวลาเริ่มต้น 09 : 00 เวลาสิ้นสุด 20 : 00
ตั้งค่าวัน :	<input type="radio"/> ไม่ใช้ <input type="radio"/> วัน --เลือก-- <input checked="" type="radio"/> วันเริ่มต้นเสาร์ วันสิ้นสุดอาทิตย์ <input type="radio"/> เทศกาล --เลือก--
ตั้งค่าวันที่ :	<input type="radio"/> ไม่ใช้ <input type="radio"/> วันที่ 01 <input checked="" type="radio"/> วันที่เริ่มต้น 01 วันที่สิ้นสุด 31
ตั้งค่าเดือน :	<input type="radio"/> ไม่ใช้ <input checked="" type="radio"/> เดือน มกราคม
เพลงรอสาย :	When You Say Nothing At All.wav
<input type="button" value="บันทึก"/> <input type="button" value="ยกเลิก"/>	

รูปที่ 4.14 แสดงตัวอย่างหน้าจอการตั้งค่าเสียงเพลงรอสาย

หน้าหลัก		Ring Back Tone											
การกำหนดค่าเริ่มต้นอัตโนมัติ การตั้งค่าการใช้งานเครื่อง IP-PBX		เบอร์ที่เศษ	ชื่อเพลง	เวลาเริ่มต้น	เวลาสิ้นสุด	วันเริ่มต้น	วันสิ้นสุด	วันที่เริ่มต้น	วันที่สิ้นสุด	เดือน	เพิ่ม	แก้ไข	ลบ
การกำหนดค่าในระบบ - เสียงเพลงรอสาย - เสียงตอบรับอัตโนมัติ - การตั้งค่าผู้ใช้งาน		6000	คือเธอ.wav	09:00	23:00			01	20	jan			
บริการเสริม - อัลบั้มเพลงส่วนตัว - บริการเสียงเพลงรอสาย - บริการเลือกไม่รับสาย													
คุณสมบัติเพิ่มเติม - ปริมาณการใช้งาน (CDR)													
ออกจากระบบ													

รูปที่ 4.15 แสดงข้อมูลการตั้งค่าเสียงเพลงรอสาย

ตารางที่ 4.7 ผลการทดสอบการตั้งค่าเสียงเพลงรอสายสำหรับเบอร์พิเศษ

หัวข้อการทดสอบ	ผลการทำงาน 36 ครั้ง		ความถูกต้องคิด เป็นเปอร์เซ็นต์
	ถูก	ผิด	
1. สามารถเลือกเพลงจากอัลบั้มของตัวเองมาทำการตั้งค่าได้	36	0	100%
2. สามารถเก็บรายละเอียดการตั้งค่าตามวัน-เวลา เทศกาลไว้ใน database แยกตามผู้ใช้ได้	36	0	100%

จากผลการทดสอบในตารางที่ 4.7 พบว่าระบบสามารถทำงานได้ถูกต้อง โดยตรวจสอบจาก database ที่จัดเก็บรายละเอียดการตั้งค่าเสียงเพลงรอสายตามแต่ละผู้ใช้ พบว่าระบบทำงานได้ถูกต้อง 100%

การทดสอบที่ 7 การตั้งค่าบริการเลือกไม่รับสาย

วัตถุประสงค์

- 1) เพื่อทดสอบว่าระบบสามารถเลือกข้อความจากในระบบมาทำการตั้งค่าได้
- 2) เพื่อทดสอบว่าระบบสามารถเก็บรายละเอียดการตั้งค่าตามวัน-เวลา เทศกาลไว้ใน database แยกตามผู้ใช้ได้

ทดสอบโดยตั้งค่าบริการเลือกไม่รับสายสำหรับหมายเลขพิเศษ ตามวัน-เวลา หรือ เทศกาล โดยเพิ่มหรือแก้ไขการตั้งค่าเดิม เพื่อให้การตั้งค่าครอบคลุมตามรูปแบบ จึงได้ทำการทดสอบจำนวน 36 ครั้งจาก 3 ประเภท คือ Voice Mail, IVR และ BUSY Tone ได้ผลดังแสดงในรูปที่ 4.16, 4.17 และตารางที่ 4.8

ตั้งค่า Call Screening

หมายเลขผู้ใช้ :	5000
เบอร์คนพิเศษ :	6000 * ระบุหมายเลข 4 หลัก
ประเภทบริการ :	IVR
ตั้งค่าช่วงเวลา :	<input checked="" type="radio"/> ไม่ใช้ <input type="radio"/> เวลาเริ่มต้น 00 : 00 เวลาสิ้นสุด 00 : 00
ตั้งค่าวัน :	<input checked="" type="radio"/> ไม่ใช้ <input checked="" type="radio"/> วัน เสาร์ <input type="radio"/> วันเริ่มต้น --เลือก-- วันสิ้นสุด --เลือก-- <input type="radio"/> เทศกาล --เลือก--
ตั้งค่าวันที่ :	<input checked="" type="radio"/> ไม่ใช้ <input checked="" type="radio"/> วันที่ 01 <input type="radio"/> วันที่เริ่มต้น 01 วันที่สิ้นสุด 01
ตั้งค่าเดือน :	<input checked="" type="radio"/> ไม่ใช้ <input type="radio"/> เดือน --เลือก--
ข้อความต้อนรับ :	กำลังประชุม.wav

รูปที่ 4.16 แสดงตัวอย่างหน้าจอการตั้งค่าเลือกไม่รับสาย

หน้าหลัก		Call Screening										
เบอร์พิเศษ	ประเภท	ข้อความ	เวลาเริ่มต้น	เวลาสิ้นสุด	วันเริ่มต้น	วันสิ้นสุด	วันที่เริ่มต้น	วันที่สิ้นสุด	เดือน	เพิ่ม	แก้ไข	ลบ
4000	IVR	ไม่สะดวกรับสาย.wav			mon	fri			jan			
6000	VM				sun	thu						

การกำหนดค่าเริ่มต้นอัตโนมัติ

การตั้งค่าการใช้งานเครื่อง IP-PBX

การกำหนดค่าในระบบ

- เสียงเพลงรอสาย
- เสียงตอบรับอัตโนมัติ
- การตั้งค่าผู้ใช้งาน

บริการเสริม

- อัลบั้มเพลงส่วนตัว
- บริการเสียงเพลงรอสาย
- บริการเลือกไม่รับสาย

คุณสมบัติเพิ่มเติม

- ปริมาณการใช้งาน (CDR)

ออกจากระบบ

รูปที่ 4.17 แสดงข้อมูลการตั้งค่าเลือกไม่รับสาย

ตารางที่ 4.8 ผลการทดสอบการตั้งค่าเลือกไม่รับสายสำหรับเบอร์พิเศษ

หัวข้อการทดสอบ	ผลการทำงาน 36 ครั้ง		ความถูกต้องคิดเป็นเปอร์เซ็นต์
	ถูก	ผิด	
1. สามารถเลือกข้อความจากระบบมาทำการตั้งค่าได้	36	0	100%
2. สามารถเก็บรายละเอียดการตั้งค่าตามวัน-เวลา เทศกาลไว้ใน database แยกตามผู้ใช้ได้	36	0	100%

จากผลการทดสอบในตารางที่ 4.8 พบว่าระบบสามารถทำงานได้ถูกต้อง โดยตรวจสอบจาก database ที่จัดเก็บรายละเอียดการตั้งค่าเสียงเพลงรอสายตามแต่ละผู้ใช้ พบว่าระบบทำงานได้ถูกต้อง 100%

การทดสอบที่ 8 ทดสอบการโทรหลังจากตั้งค่าบริการเสียงเพลงรอสายเรียบร้อยแล้ว

วัตถุประสงค์

- 1) เพื่อทดสอบว่าระบบสามารถทำงานได้ถูกต้องตามวัน-เวลา เทศกาลที่ผู้ใช้กำหนด
- 2) เพื่อทดสอบว่าระบบแสดง CDR ของผู้ใช้ในการโทรออก การทำงานที่หมายเลข extension สดท้าย ระยะเวลาที่ใช้สาย และคำสั่งสุดท้ายที่ทำงาน ได้ถูกต้อง

ทดสอบโดยใช้ X-Lite จากเครื่อง client ทั้ง 2 เครื่อง ซึ่งได้ลงทะเบียนเรียบร้อยแล้ว ทดสอบโทรหากันตามการตั้งค่าในการทดลองที่ 6 โดยได้เพิ่มหรือเปลี่ยนวัน-เวลาในการทดสอบที่ 6 ทั้งหมดจำนวน 36 รูปแบบ และทดสอบโทรใน-นอกช่วงเวลา ทั้งหมด 72 ครั้ง ได้ผลการทดสอบดังรูปที่ 4.18 และ 4.19

ข้อมูลทดสอบ												ผลการทำงาน			
หมายเลขผู้รับ	กรณี	เบอร์โทร	ประเภท	ชื่อเพลง	เวลาเริ่มต้น	เวลาสิ้นสุด	วันเริ่มต้น	วันสิ้นสุด	วันที่เริ่มต้น	วันที่สิ้นสุด	เดือน	ในช่วงเวลา		นอกช่วงเวลา	
												ถูก	ผิด	ถูก	ผิด
5000	1	6000	RBT	When You Say Nothing At All.wav	9:00	23:00						✓		✓	
5000	2	6000	RBT	Closing Time.wav			fri					✓		✓	
5000	3	6000	RBT	มากกว่ารัก.wav			mon	fri				✓		✓	
5000	4	6000	RBT	หนึ่งจ.wav						30		✓		✓	
5000	5	6000	RBT	อยู่ ๆ ก็มาปรากฏตัวในหัวใจ.wav						1	31	✓		✓	
5000	6	6000	RBT	When You Say Nothing At All.wav							jan	✓		✓	
5000	7	6000	RBT	Closing Time.wav	9:00	23:00	fri					✓		✓	
5000	8	6000	RBT	มากกว่ารัก.wav	9:00	23:00	wed	fri				✓		✓	
5000	9	6000	RBT	หนึ่งจ.wav	9:00	23:00				30		✓		✓	
5000	10	6000	RBT	อยู่ ๆ ก็มาปรากฏตัวในหัวใจ.wav	9:00	23:00				1	31	✓		✓	
5000	11	6000	RBT	When You Say Nothing At All.wav	9:00	23:00					jan	✓		✓	
5000	12	6000	RBT	Closing Time.wav			fri			30		✓		✓	
5000	13	6000	RBT	มากกว่ารัก.wav			fri			1	31	✓		✓	
5000	14	6000	RBT	หนึ่งจ.wav			fri				jan	✓		✓	
5000	15	6000	RBT	อยู่ ๆ ก็มาปรากฏตัวในหัวใจ.wav			mon	sat		30		✓		✓	
5000	16	6000	RBT	When You Say Nothing At All.wav			mon	sat		1	31	✓		✓	
6000	17	5000	RBT	คำขอบคุณ.wav			mon	sat			jan	✓		✓	
6000	18	5000	RBT	หุ้มอยู่ในใจ.wav						30	jan	✓		✓	
6000	19	5000	RBT	สมหมาย.wav						1	31	jan	✓		✓
6000	20	5000	RBT	This I promise you.wav	9:00	23:00	fri			30		✓		✓	
6000	21	5000	RBT	มากกว่ารัก.wav	9:00	23:00	fri			1	31	✓		✓	
6000	22	5000	RBT	คำขอบคุณ.wav	9:00	23:00	fri				jan	✓		✓	
6000	23	5000	RBT	หุ้มอยู่ในใจ.wav	9:00	23:00	mon	sat		30		✓		✓	
6000	24	5000	RBT	สมหมาย.wav	9:00	23:00	mon	sat		1	31	✓		✓	
6000	25	5000	RBT	This I promise you.wav	9:00	23:00	mon	sat			jan	✓		✓	
6000	26	5000	RBT	มากกว่ารัก.wav	9:00	23:00				30	jan	✓		✓	
6000	27	5000	RBT	คำขอบคุณ.wav	9:00	23:00				1	31	jan	✓		✓
6000	28	5000	RBT	หุ้มอยู่ในใจ.wav			fri			30	jan	✓		✓	
6000	29	5000	RBT	สมหมาย.wav			fri			1	31	jan	✓		✓
6000	30	5000	RBT	This I promise you.wav			mon	sat		30	jan	✓		✓	
6000	31	5000	RBT	มากกว่ารัก.wav			mon	sat		1	31	jan	✓		✓
6000	32	5000	RBT	คำขอบคุณ.wav	9:00	23:00	fri			30	jan	✓		✓	
5000	33	6000	RBT	มากกว่ารัก.wav	9:00	23:00	fri			1	31	jan	✓		✓
5000	34	6000	RBT	หนึ่งจ.wav	9:00	23:00	mon	sat		30	jan	✓		✓	
5000	35	6000	RBT	อยู่ ๆ ก็มาปรากฏตัวในหัวใจ.wav	9:00	23:00	mon	sat		1	31	jan	✓		✓
5000	36	6000	RBT	When You Say Nothing At All.wav								✓		✓	

รูปที่ 4.18 ผลการทดสอบจากหมายเลข 5000 และ 6000 โทรหากันในช่วงเวลาและนอกช่วงเวลา

ปริมาณการใช้งาน (CDR)

เริ่มจากวันที่ : 1 มกราคม 2555 ถึงวันที่ : 1 มกราคม 2555 แสดงข้อมูลของ : ทั้งหมด

ค้นหา

วันที่/เวลา	หมายเลขต้นทาง	หมายเลขปลายทาง	เวลา(นาที:วินาที)	คำสั่งสุดท้ายที่ทำงาน
14 ม.ค. 2555 19:06:35	5000	700	0 : 7	Dial
14 ม.ค. 2555 19:02:14	5000	700	0 : 14	Dial
04 ม.ค. 2555 21:22:28	5000	700	0 : 34	Hangup
03 ม.ค. 2555 13:03:59	5000	700	0 : 7	Dial
03 ม.ค. 2555 13:02:05	5000	700	0 : 35	Hangup
03 ม.ค. 2555 13:00:13	5000	700	0 : 5	Dial
03 ม.ค. 2555 12:58:49	5000	700	0 : 6	Dial
01 ม.ค. 2555 20:24:38	5000	901	0 : 12	Dial

รูปที่ 4.19 แสดงตัวอย่างรายละเอียด CDR

จากผลการทดสอบในรูปที่ 4.18 และ 4.19 พบว่าระบบสามารถทำงานได้ถูกต้อง หมายเลข 5000 และ 6000 สามารถโทรหากันได้ทุกกรณี กรณีที่อยู่ในช่วงวัน-เวลาระบบจะเล่นไฟล์เสียงเพลงที่ได้กำหนดไว้ในช่วงวัน-เวลานั้นๆ และกรณีที่ไม่มีอยู่ในช่วงวัน-เวลาที่กำหนดไว้ระบบจะเล่นไฟล์เสียงที่มีอยู่ในอัลบั้มของผู้ใช้ โดยการสุ่มเล่นไฟล์เสียงเพลงที่อยู่ในอัลบั้มแต่ละครั้งวนไปเรื่อยๆ และมีการเก็บรายละเอียด CDR การโทรออกของหมายเลข 5000 และ 6000 ได้ โดยการทำงานคิดเป็นความถูกต้อง 100%

การทดสอบที่ 9 ทดสอบการโทร หลังจากตั้งค่าบริการเลือกไม่รับสายเรียบร้อยแล้ว

วัตถุประสงค์

- 1) เพื่อทดสอบว่าระบบสามารถทำงานได้ถูกต้องตามวัน-เวลา เทศกาลที่ผู้ใช้กำหนด
- 2) เพื่อทดสอบว่าในกรณีตั้ง voice mail ระบบสามารถส่งข้อความเข้า mail box ได้
- 3) เพื่อทดสอบว่าระบบแสดง CDR ของผู้ใช้ในการโทรออก การทำงานที่หมายเลข extension สุดท้าย ระยะเวลาที่ใช้สาย และคำสั่งสุดท้ายที่ทำงาน ได้ถูกต้อง

ทดสอบโดยใช้ X-Lite จากเครื่อง client ทั้ง 2 เครื่อง ซึ่งได้ลงทะเบียนเรียบร้อยแล้ว ทดสอบโทรหากันตามการตั้งค่าในการทดลองที่ 7 โดยได้ทำการเพิ่มหรือเปลี่ยนวัน-เวลา ในการทดสอบที่ 7 ทั้งหมดจำนวน 36 รูปแบบ ทำการทดสอบโทรในช่วงเวลาและนอกช่วงเวลา ทั้งหมด 72 ครั้ง ได้ผลการทดสอบดังรูปที่ 4.20, 4.21 และ 4.22

ข้อมูลทดสอบ											ผลการทำงาน				
หมายเลขผู้โทร	กรณี	เบอร์พิเศษ	ประเภท	ข้อความ	เวลาเริ่มต้น	เวลาสิ้นสุด	วันเริ่มต้น	วันสิ้นสุด	วันที่เริ่มต้น	วันที่สิ้นสุด	เดือน	ในช่วงเวลา		นอกช่วงเวลา	
												ถูก	ผิด	ถูก	ผิด
5000	1	6000	BUSY		9:00	23:00						✓		✓	
5000	2	6000	IVR	กำลังประชุม.wav			fri					✓		✓	
5000	3	6000	VM				mon	fri				✓		✓	
5000	4	6000	BUSY						30			✓		✓	
5000	5	6000	IVR	ตั้งใจเรียน.wav					1	31		✓		✓	
5000	6	6000	VM							jan		✓		✓	
5000	7	6000	BUSY		9:00	23:00	fri					✓		✓	
5000	8	6000	IVR	มีปัญหากับใจ.wav	9:00	23:00	wed	fri				✓		✓	
5000	9	6000	VM		9:00	23:00			30			✓		✓	
5000	10	6000	BUSY		9:00	23:00			1	31		✓		✓	
5000	11	6000	IVR	ไม่สะดวกรับสาย.wav	9:00	23:00					jan	✓		✓	
5000	12	6000	VM				fri		30			✓		✓	
5000	13	6000	BUSY				fri		1	31		✓		✓	
5000	14	6000	IVR	อยู่ต่างประเทศ.wav			fri				jan	✓		✓	
5000	15	6000	VM				mon	sat	30			✓		✓	
5000	16	6000	BUSY				mon	sat	1	31		✓		✓	
6000	17	5000	IVR	กำลังประชุม.wav			mon	sat			jan	✓		✓	
6000	18	5000	VM						30		jan	✓		✓	
6000	19	5000	BUSY						1	31	jan	✓		✓	
6000	20	5000	IVR	ตั้งใจเรียน.wav	9:00	23:00	fri		30			✓		✓	
6000	21	5000	VM		9:00	23:00	fri		1	31		✓		✓	
6000	22	5000	BUSY		9:00	23:00	fri				jan	✓		✓	
6000	23	5000	IVR	มีปัญหากับใจ.wav	9:00	23:00	mon	sat	30			✓		✓	
6000	24	5000	VM		9:00	23:00	mon	sat	1	31		✓		✓	
6000	25	5000	BUSY		9:00	23:00	mon	sat			jan	✓		✓	
6000	26	5000	IVR	ไม่สะดวกรับสาย.wav	9:00	23:00			30		jan	✓		✓	
6000	27	5000	VM		9:00	23:00			1	31	jan	✓		✓	
6000	28	5000	BUSY				fri		30		jan	✓		✓	
6000	29	5000	IVR	อยู่ต่างประเทศ.wav			fri		1	31	jan	✓		✓	
6000	30	5000	VM				mon	sat	30		jan	✓		✓	
6000	31	5000	BUSY				mon	sat	1	31	jan	✓		✓	
6000	32	5000	IVR	กำลังประชุม.wav	9:00	23:00	fri		30		jan	✓		✓	
5000	33	6000	VM		9:00	23:00	fri		1	31	jan	✓		✓	
5000	34	6000	BUSY		9:00	23:00	mon	sat	30		jan	✓		✓	
5000	35	6000	IVR	ตั้งใจเรียน.wav	9:00	23:00	mon	sat	1	31	jan	✓		✓	
5000	36	6000	VM									✓		✓	

รูปที่ 4.20 ผลการทดสอบจากหมายเลข 5000 และ 6000 โทรหากันในช่วงเวลาและนอกช่วงเวลา

Subject: [PBX]: New message 1 in mailbox 5000
From: "Asterisk PBX" <asterisk@pc16>
Date: Tue, January 31, 2012 12:37 am
To: pc16@pc16
Priority: Normal
Options: [View Full Header](#) | [View Printable Version](#) | [Download this as a file](#)

Dear :

Just wanted to let you know you were just left a 0:05 long message (number 1) in mailbox 5000 from 6000, on Tuesday, January 31, 2012 at 12:37:51 AM so you might want to check it when you get a chance. Thanks!

--Asterisk

Attachments: ไฟล์เสียงที่ฝากไว้
[msg0000.WAV](#) 12 k [audio/x-wav] Voicemail sound attachment.

รูปที่ 4.21 หน้าจอ mail box กรณีที่เลือกตั้งค่าแบบ VM

ปริมาณการใช้งาน (CDR)

เริ่มจากวันที่ : 1 มกราคม 2555 ถึงวันที่ : 1 มกราคม 2555 แสดงข้อมูลของ : ทั้งหมด ค้นหา

วันที่/เวลา	หมายเลขต้นทาง	หมายเลขปลายทาง	เวลา(นาที:วินาที)	คำสั่งสุดท้ายที่ทำงาน
05 ม.ค. 2555 21:21:51	6000	5000	0 : 21	VoiceMail
05 ม.ค. 2555 21:14:09	6000	5000	0 : 27	VoiceMail
05 ม.ค. 2555 21:12:07	6000	5000	0 : 20	VoiceMail
05 ม.ค. 2555 06:58:13	6000	5000	0 : 20	VoiceMail
05 ม.ค. 2555 06:53:11	6000	5000	0 : 20	VoiceMail
02 ม.ค. 2555 08:17:42	6000	5000	0 : 20	VoiceMail
02 ม.ค. 2555 08:08:17	6000	100	0 : 14	Busy
02 ม.ค. 2555 08:07:44	6000	100	0 : 14	Busy
02 ม.ค. 2555 08:06:55	6000	100	0 : 19	Busy
02 ม.ค. 2555 08:05:35	6000	100	0 : 19	Busy
02 ม.ค. 2555 08:04:08	6000	510	0 : 11	Hangup
02 ม.ค. 2555 08:03:27	6000	901	0 : 17	Dial
02 ม.ค. 2555 08:02:43	6000	510	0 : 10	Hangup

รูปที่ 4.22 แสดงตัวอย่างรายละเอียด CDR

จากผลการทดสอบในรูปที่ 4.20, 4.21 และ 4.22 พบว่าระบบสามารถทำงานได้ถูกต้อง หมายเลข 5000 และ 6000 สามารถโทรหากันได้ทุกกรณี กรณีที่อยู่ในช่วงวัน-เวลา และตั้ง IVR ระบบจะเล่นไฟล์เสียงข้อความตอบรับอัตโนมัติที่ได้กำหนดไว้ในช่วงวัน-เวลานั้นๆ ตั้ง BUSY ระบบจะได้ยิน BUSY Tone ตั้ง VM จะมีการส่งข้อความเข้า mail box ดังแสดงในรูปที่ 4.21 และกรณีที่ไม่วางอยู่ในช่วงวัน-เวลาที่กำหนดไว้ระบบจะ ringing และมีการเก็บรายละเอียด CDR การโทรออกของหมายเลข 5000 และ 6000 ได้ โดยการทำงานคิดเป็นความถูกต้อง 100%

การทดสอบที่ 10 การทดสอบประสิทธิภาพการทำงานของเครื่องแม่ข่าย application server

วัตถุประสงค์

1) เพื่อทดสอบการใช้ทรัพยากรระบบ โดยการวัดค่า CPU และ RAM ของ application server ในขณะที่มีการโทรเข้ามาในระบบที่เครื่อง application server

ทดสอบการใช้งานร่วมกัน 2 Server คือ application server และ Asterisk 1.8.3 มีสเปคเครื่องดังต่อไปนี้คือ

1.1) เครื่อง application server ในระดับชั้น Service Layer

CPU 1.5 GHz, RAM 512 MB, Harddisk 40 GB

1.2) เครื่อง IP PBX ติดตั้ง Asterisk 1.8.3 ในระดับชั้น Control Layer

CPU 1.5 GHz, RAM 512 MB, Harddisk 60 GB

และใช้ Softphone 10 เลขหมายในการทดสอบ โดย 1 Softphone มี 2 คู่สาย ทำให้สามารถโทรออกและรับสายได้พร้อมกัน โดยทางผู้วิจัยได้ทดสอบโทรเข้ามาใช้งานพร้อมกันด้วยจำนวนหมายเลขที่เพิ่มขึ้น 4, 6, 8, 10, 12, 14, 16, 18 และ 20 คู่สาย ตามลำดับ

ตัวแปรที่ใช้ในการนำมาพิจารณาเรื่องประสิทธิภาพของระบบคือ การใช้งาน CPU และ RAM ในการทดสอบตอนเริ่มต้นกรณีที่ไม่มีหมายเลขใดโทรเข้ามา เครื่อง application server ค่า CPU อยู่ที่ 0.3% (100- 99.7) และ RAM อยู่ที่ 30.4 % (191028 / 628216) ดังแสดงในรูปที่ 4.23

```

root@db: ~
top - 01:39:23 up 15 min, 1 user, load average: 0.00, 0.01, 0.03
Tasks: 85 total, 1 running, 84 sleeping, 0 stopped, 0 zombie
Cpu(s): 0.0%us, 0.1%sy, 0.0%ni, 99.7%id, 0.2%wa, 0.0%hi, 0.0%si, 0.0%st
Mem: 628216k total, 191028k used, 437188k free, 21164k buffers
Swap: 644092k total, 0k used, 644092k free, 100148k cached

  PID USER      PR  NI  VIRT  RES  SHR  S  %CPU  %MEM    TIME+  COMMAND
 888 root        20   0  4352 1316 1100  S   0   0.2   0:00.01  master
   1 root        20   0  3328 1812 1232  S   0   0.3   0:00.78  init
   2 root        20   0   0     0   0     S   0   0.0   0:00.00  kthreadd
   3 root        20   0   0     0   0     S   0   0.0   0:00.03  ksoftirqd/0
   6 root        RT   0   0     0   0     S   0   0.0   0:00.00  migration/0
   7 root        RT   0   0     0   0     S   0   0.0   0:00.00  migration/1
   8 root        20   0   0     0   0     S   0   0.0   0:00.00  kworker/1:0

```

รูปที่ 4.23 ค่าเริ่มต้นของ CPU และ RAM

การทดสอบเริ่มต้นกรณีมีการใช้งานในระบบพร้อมกัน 4 คู่สาย

ผลการทดสอบการใช้ทรัพยากรออกมาดังนี้คือ เครื่อง application server ค่า CPU อยู่ที่ 0.1% (100- 99.9) และ RAM อยู่ที่ 30.4 % (191028 / 628216) ดังแสดงในรูปที่ 4.24 จากนั้นทำการทดสอบโดยเพิ่มผู้ใช้งานในระบบพร้อมกันเป็น 6, 8, 10, 12, 14, 16, 18 และ 20 คู่สายตามลำดับ โดยผลการทดสอบที่ได้สรุปดังแสดงในตารางที่ 4.9

```

root@db: ~
top - 01:50:14 up 26 min, 1 user, load average: 0.00, 0.01, 0.03
Tasks: 85 total, 1 running, 84 sleeping, 0 stopped, 0 zombie
Cpu(s): 0.0%us, 0.1%sy, 0.0%ni, 99.9%id, 0.0%wa, 0.0%hi, 0.0%si, 0.0%st
Mem: 628216k total, 191028k used, 437188k free, 21364k buffers
Swap: 644092k total, 0k used, 644092k free, 100164k cached

  PID USER      PR  NI  VIRT  RES  SHR  S  %CPU  %MEM    TIME+  COMMAND
 728 mysql    20   0 134m  18m 5696  S   0   2.9   0:00.45  mysqld
   1 root        20   0  3328 1812 1232  S   0   0.3   0:00.78  init
   2 root        20   0   0     0   0     S   0   0.0   0:00.00  kthreadd
   3 root        20   0   0     0   0     S   0   0.0   0:00.03  ksoftirqd/0
   6 root        RT   0   0     0   0     S   0   0.0   0:00.00  migration/0
   7 root        RT   0   0     0   0     S   0   0.0   0:00.00  migration/1
   8 root        20   0   0     0   0     S   0   0.0   0:00.00  kworker/1:0

```

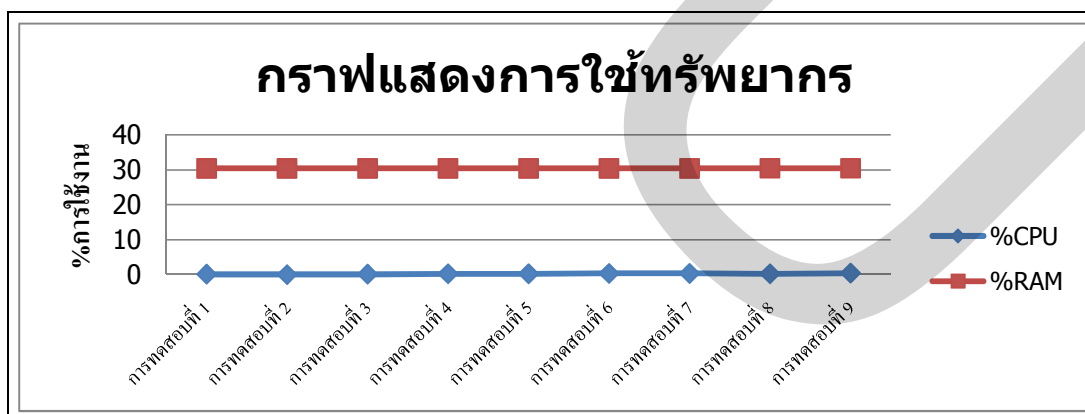
รูปที่ 4.24 การใช้ทรัพยากรของ CPU และ RAM กรณีมีการใช้งานในระบบพร้อมกัน 4 คู่สาย

ตารางที่ 4.9 สรุปผลการทดสอบการใช้ทรัพยากรของเครื่อง application server

การทดลองที่	%การใช้งาน CPU	%การใช้งาน RAM
1.	0.1%	30.40%
2.	0.1%	30.40%
3.	0.1%	30.40%
4.	0.2%	30.42%
5.	0.2%	30.42%
6.	0.3%	30.42%
7.	0.3%	30.42%
8.	0.2%	30.44%
9.	0.4%	30.44%

หมายเหตุ : Server มีสเปค CPU 1.5 GHz, RAM 512 MB, Harddisk 40 GB

การทดสอบการใช้ทรัพยากรของเครื่อง application server สามารถสรุปเป็นกราฟได้
ดังรูปที่ 4.25



รูปที่ 4.25 กราฟสรุปการใช้ทรัพยากรของเครื่อง application server

จากผลการทดสอบในตารางที่ 4.9 และรูปที่ 4.25 พบว่า RAM มีการใช้ทรัพยากรมากกว่า CPU แต่จากสภาพแวดล้อมของเครื่อง application server ที่ทดสอบ พบว่าการใช้ทรัพยากรของ RAM มีผลกระทบต่อระบบค่อนข้างน้อย ผู้วิจัยจึงเพิ่มการทดสอบวัดการใช้งานของ RAM และ bandwidth ที่เครื่อง Asterisk 1.8.3 เพิ่มเติม เพื่อหาผลกระทบที่มีต่อประสิทธิภาพการทำงานของระบบ

การทดสอบที่ 11 การทดสอบประสิทธิภาพการทำงานของเครื่องแม่ข่าย Asterisk 1.8.3

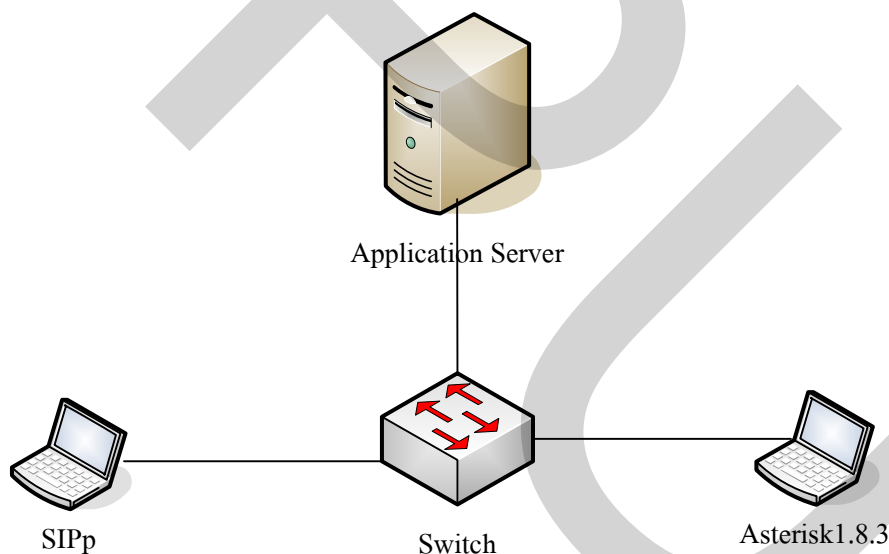
วัตถุประสงค์

1) เพื่อทดสอบการใช้ทรัพยากรระบบโดยการวัดค่า RAM และ Bandwidth ของ Server ในขณะที่มีการโทรเข้ามาในระบบที่เครื่อง Asterisk 1.8.3

ทดสอบการใช้งานที่เครื่อง Asterisk 1.8.3 มีสเปคเครื่องดังนี้

CPU 1.5 GHz, RAM 512 MB, Harddisk 60 GB, Bandwidth 100 Mbps

และใช้ โปรแกรม SIPp ซึ่งเป็นโปรแกรมจำลองการสร้างจำนวน call ในการจำลองการโทรไปยังเครื่อง Asterisk 1.8.3 โดยมีโครงสร้างการทำงานของโปรแกรม SIPp ดังแสดงในรูป 4.26



จากรูปที่ 4.26 ภาพรวมการทดสอบระบบโดยใช้ SIPp

จากรูปที่ 4.26 โปรแกรม SIPp มีขั้นตอนการทำงานดังนี้

- 1) โทรจาก SIPp ไปยังเครื่อง Asterisk
- 2) รอสายเป็นเวลา 20 วินาที
- 3) วางสาย

ผู้วิจัยได้ทดสอบโทรเข้ามาด้วยจำนวน call พร้อมกันเพิ่มขึ้นเป็น 30, 50, 90, 110 และ 120 คู่สาย ตามลำดับ

ตัวแปรที่ใช้ในการนำมาพิจารณาเรื่องประสิทธิภาพของระบบคือ การใช้งาน RAM และ Bandwidth เท่านั้น เนื่องจากค่า CPU ที่ทดสอบพบว่าการเปลี่ยนแปลงโดยเฉลี่ยอยู่ที่ 0.1 %

ในการทดสอบตอนเริ่มต้นกรณีที่ไม่มีความหมายเลขใดโทรเข้ามา เครื่อง Asterisk 1.8.3 มีการใช้ RAM อยู่ที่ 51% และ Bandwidth ด้านรับ (downlink) อยู่ที่ 0.52 %

$[8 * (65.03 * 10^3) / 100 * 10^6 * 100]$ และด้านส่ง (uplink) อยู่ที่ 2.24%

$[8 * (280.52 * 10^3) / 100 * 10^6 * 100]$ ดังแสดงในรูปที่ 4.27 และ 4.28 ตามลำดับ

MEMORY USAGE				
Type	Usage	Free	Used	Size
Physical Memory	51%	244.33 MiB	251.87 MiB	496.20 MiB
Disk Swap	0%	401.00 MiB	0.00 KiB	401.00 MiB

รูปที่ 4.27 แสดงการใช้ทรัพยากร RAM ก่อนการใช้งานโดยเรียกดูด้วย phpsysinfo

<http://phpsysinfo.sourceforge.net/> ผ่าน Browser

จากรูปที่ 4.27 MiB มาจาก Mega Binary Byte = Mebibyte และ KiB มาจาก Kilo

Binary Byte = Kibibyte

NETWORK USAGE			
Device	Received	Sent	Err/Drop
lo	7.13 KiB	7.13 KiB	0/0
eth0	65.03 KiB	280.52 KiB	0/0

รูปที่ 4.28 แสดงการใช้ Bandwidth ก่อนการใช้งาน

การทดสอบเริ่มต้นกรณีมีการใช้งานในระบบ 30 calls

ผลการทดสอบสร้าง call โดยใช้ SIPp จำนวน 30 calls โทรเข้ามาในระบบดังแสดง

รูปที่ 4.29, 4.30, 4.31 และ 4.32 ตามลำดับ


```

----- Scenario Screen ----- [1-9]: Change Screen --
Call-rate(length)  Port  Total-time  Total-calls  Remote-host
10.0(20000 ms)/1.000s  5061  181.07 s  270  192.168.1.5:5060(UDP)

0 new calls during 1.000 s period  4 ms scheduler resolution
29 calls (limit 30)  Peak was 30 calls, after 3 s
0 Running, 60 Paused, 1 Woken up
0 dead call msg (discarded)  0 out-of-call msg (discarded)
3 open sockets

Messages  Retrans  Timeout  Unexpected-Msg
INVITE ----->  270  0  0
100 <-----  270  0  0
180 <-----  0  0  0
183 <-----  0  0  0
200 <-----  E-RTD1 270  0  0
ACK ----->  270  0
Pause [ 20.0s]  270
BYE ----->  241  0
200 <-----  241  0

----- [+-|*|/]: Adjust rate ---- [q]: Soft exit ---- [p]: Pause traffic -----

```

รูปที่ 4.29 แสดงการใช้โปรแกรม SIPp ในการสร้าง call จำนวน 30 calls

```

----- Statistics Screen ----- [1-9]: Change Screen --
Start Time      | 2012-04-28 00:52:34:964 | 1335549154.964556
Last Reset Time | 2012-04-28 00:56:10:077 | 1335549370.077413
Current Time    | 2012-04-28 00:56:11:014 | 1335549371.014088
-----
Counter Name    | Periodic value          | Cumulative value
-----
Elapsed Time    | 00:00:00:936           | 00:03:36:049
Call Rate       | 0.000 cps              | 1.527 cps
-----
Incoming call created | 0                      | 0
OutGoing call created | 0                      | 330
Total Call created  |                        | 330
Current Call      | 30                      |
-----
Successful call  | 0                      | 300
Failed call      | 0                      | 0
-----
Response Time 1  | 00:00:00:000          | 00:00:00:001
Call Length     | 00:00:00:000          | 00:00:20:007
-----
Test Terminated

```

รูปที่ 4.30 แสดงผลที่ได้จากโปรแกรม SIPp ในการสร้าง call จำนวน 30 calls

MEMORY USAGE				
Type	Usage	Free	Used	Size
# Physical Memory	52%	242.69 MIB	253.52 MIB	496.20 MIB
# Disk Swap	0%	401.00 MIB	0.00 KIB	401.00 MIB

รูปที่ 4.31 แสดงการใช้ทรัพยากร RAM ในการสร้าง call จำนวน 30 calls

NETWORK USAGE			
Device	Received	Sent	Err/Drop
lo	9.05 KiB	9.05 KiB	0/0
eth0	546.54 KiB	1.07 MiB	0/0

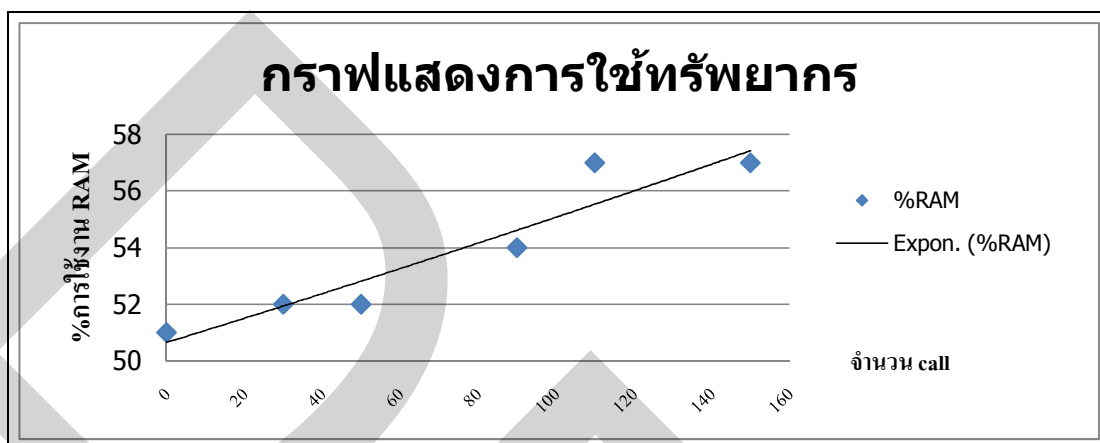
รูปที่ 4.32 แสดงการใช้ Bandwidth ในการสร้าง call จำนวน 30 calls

จากผลการทดสอบรูปที่ 4.30 พบว่าจาก 30 calls ที่ใช้ทดสอบไม่มีการ drop call และมีการใช้ทรัพยากรออกมาดังนี้คือ RAM อยู่ที่ 52% และ Bandwidth ด้านรับ (downlink) อยู่ที่ 4.37% $[8 \cdot (546.54 \cdot 10^3) / 100 \cdot 10^6 \cdot 100]$ ด้านส่ง (uplink) อยู่ที่ 8.56% $[(8 \cdot 1.07) / 100 \cdot 100]$ จากนั้นทำการทดสอบโดยเพิ่มจำนวน call ในระบบเป็น 50, 70, 90, 110 และ 120 calls ตามลำดับ โดยผลการทดสอบที่ได้สรุปดังแสดงในตารางที่ 4.10

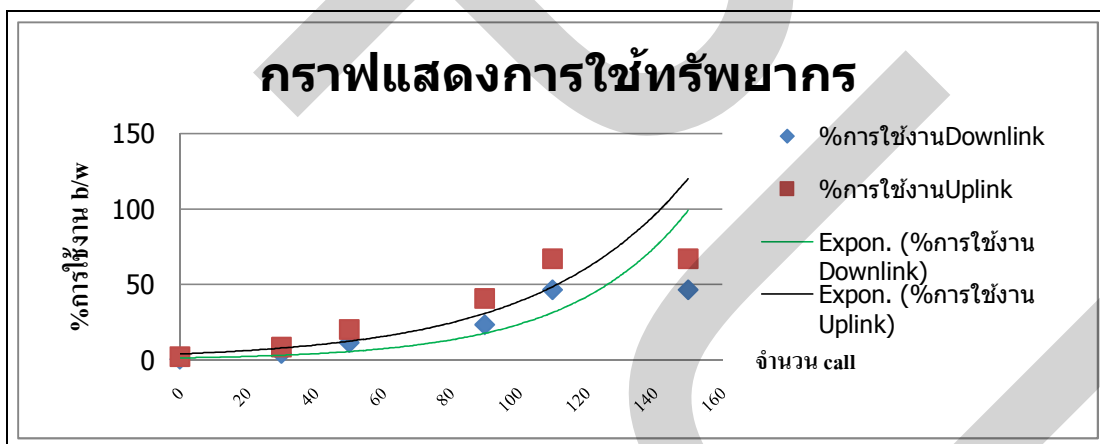
ตารางที่ 4.10 สรุปผลการทดสอบการใช้ทรัพยากรของเครื่อง Asterisk 1.8.3

จำนวน call	%การใช้งาน RAM	%การใช้งาน Bandwidth ด้าน Downlink	%การใช้งาน Bandwidth ด้าน Uplink
0	51%	0.52%	2.24%
30	52%	4.37%	8.56%
50	52%	11.52%	20.32%
90	54%	23.36%	40.8%
110	57%	46.56%	67.12%
120	57%	46.56%	67.12%

การทดสอบการใช้ทรัพยากรของเครื่อง Asterisk 1.8.3 สามารถสรุปเป็นกราฟได้ดังรูปที่ 4.33 และ 4.34



รูปที่ 4.33 กราฟสรุปการใช้ทรัพยากร RAM ของเครื่อง Asterisk 1.8.3



รูปที่ 4.34 กราฟสรุปการใช้ Bandwidth ของเครื่อง Asterisk 1.8.3

จากกราฟผลการทดสอบในรูปที่ 4.33 และ 4.34 พบว่าในการใช้งานที่ 60% เท่ากัน การใช้งาน Bandwidth ด้าน Uplink มีการใช้งานทรัพยากรมากกว่า RAM กล่าวคือ Bandwidth ด้าน Uplink มีผลกระทบต่อการใช้ทรัพยากรมากกว่า ซึ่งสามารถประมาณการได้ว่าถ้ามีผู้โทรเข้ามาในระบบมากกว่า 125 สาย $[(120/67.12)*70]$ ขึ้นไป ค่า ณ ที่ uplink utilization 70% อาจจะทำให้ระบบที่เครื่อง Asterisk server เกิดการ Delay รวมถึงอาจทำให้ประสิทธิภาพการทำงานของระบบค่อยลงได้

บทที่ 5

สรุปผล และข้อเสนอแนะ

ในบทนี้เป็นการอภิปรายเพื่อสรุปผลที่ได้จากการทดสอบงานวิจัย รวมทั้งข้อจำกัดของระบบที่พบจากการทดสอบระบบ และข้อเสนอแนะสำหรับแนวทางในการพัฒนางานวิจัยนี้ต่อไป เพื่อแก้ข้อบกพร่องของระบบให้มีประสิทธิภาพมากขึ้น

5.1 สรุปผลการวิจัย

5.1.1 สรุปผลตามวัตถุประสงค์ของงานวิจัย

5.1.1.1 สามารถออกแบบและพัฒนาระบบเสียงรอสายและเลือกไม่รับสายในโครงข่าย VoIP ตามสถาปัตยกรรม NGN ตามวัตถุประสงค์ของงานวิจัย คือให้ระบบมีการทำงานอยู่ในระดับชั้น Service Layer ให้มีการจัดการที่ศูนย์กลาง

5.1.1.2 สามารถพัฒนาระบบให้รองรับเครื่องแม่ข่าย Asterisk มากกว่า 1 ระบบ

5.1.2 สรุปผลตามขอบเขตของงานวิจัย

หลังจากทดสอบระบบในด้านต่างๆ แล้วนั้น พบว่าโปรแกรมระบบเสียงรอสายและเลือกไม่รับสาย ฯ สามารถทำงานได้ตามขอบเขตของงานวิจัยที่ได้กำหนดไว้ โดยแบ่งผลสรุปเป็น 2 ส่วน คือ ส่วนบริการเสริมสำหรับผู้ทั่วไป และส่วนการกำหนดค่าในระบบ สำหรับผู้ดูแลระบบ ดังแสดงในตารางที่ 5.1 และตารางที่ 5.2

ตารางที่ 5.1 สรุปผลการทำงานตามขอบเขตส่วนบริการเสริมสำหรับผู้ทั่วไป

หัวข้อตามขอบเขต	ผลการทำงาน	
	ได้	ไม่ได้
บริการเพลงรอสาย		
1. เพิ่ม/ลด/แก้ไข/เรียกดู การตั้งค่าเพลงรอสายสำหรับเบอร์ที่โทรเข้ามาได้	✓	
2. กำหนดเพลงรอสายให้กับเบอร์ที่โทรเข้า ตามวัน เวลา และเทศกาลได้	✓	
3. ระบบจะมีเพลงให้เลือก (default) อย่างน้อย 5 เพลง โดยผู้ใช้งานสามารถเพิ่มเพลงใหม่เข้าในระบบส่วนตัวของตนเอง	✓	
บริการเลือกไม่รับสาย		
4. เพิ่ม/ลด/แก้ไข/เรียกดู เบอร์ที่ไม่ต้องการรับสายได้	✓	
5. กรณีที่ไม่ต้องการรับสาย ผู้ใช้งานสามารถเลือกให้ผู้โทรเข้าไปได้ยินเสียงสายไม่ว่าง (busy) ได้ โดยสามารถกำหนดจากเบอร์โทรเข้า วัน เวลา หรือเทศกาลได้	✓	
6. กรณีที่ไม่ต้องการรับสาย ผู้ใช้งานสามารถเลือกให้ผู้โทรเข้าถูกโอนสายเข้าระบบฝากข้อความ (voice mail) ได้ โดยสามารถกำหนดจากเบอร์โทรเข้า วัน เวลา หรือเทศกาลได้	✓	
7. กรณีที่ไม่ต้องการรับสาย ผู้ใช้งานสามารถเลือกให้ผู้โทรเข้าไปได้ยินเสียงระบบตอบรับอัตโนมัติ (Interactive Voice Response – IVR) ได้โดยสามารถกำหนดจากเบอร์โทรเข้า วัน เวลา หรือเทศกาลได้	✓	
8. ผู้ใช้งานสามารถเลือกเสียง IVR ที่มีอยู่ในระบบอย่างน้อย 5 เสียง	✓	
9. ระบบมีการเก็บบันทึกข้อมูลการใช้งานแบบรวมศูนย์ได้	✓	

ตารางที่ 5.2 สรุปผลการทำงานตามขอบเขตส่วนการกำหนดค่าในระบบ สำหรับผู้ดูแลระบบ

หัวข้อขอบเขต	ผลการทำงาน	
	ได้	ไม่ได้
1. เพิ่ม/ลด ผู้ใช้งานในระบบได้	✓	
2. แก้ไขสิทธิ์ของผู้ใช้ได้	✓	
3. เพิ่ม/ลด จำนวนเสียงข้อความต้อนรับให้กับระบบสำหรับบริการ Call-Screening แบบ IVR ได้	✓	
4. เพิ่ม/ลด จำนวนเสียงเพลงรอสายให้กับระบบสำหรับบริการ Ring Back Tone ได้	✓	
5. กำหนดค่าที่จำเป็นสำหรับเครื่อง asterisk1.6 เครื่องใหม่ผ่านหน้าเว็บได้	✓	
6. กำหนดค่าที่จำเป็นสำหรับเครื่อง asterisk1.8 เครื่องใหม่ผ่านหน้าเว็บได้	✓	

จากตารางที่ 5.1 และ 5.2 พบว่าโปรแกรมระบบบริการเสียงรอสายและเลือกไม่รับสาย สามารถทำงานได้ตามขอบเขตที่กำหนดไว้

5.2 ข้อจำกัดของระบบ

ข้อจำกัดของระบบบริการเสียงรอสายและเลือกไม่รับสายในโครงข่าย VoIP ตามสถาปัตยกรรม NGN สามารถแยกข้อจำกัดออกเป็นข้อๆ ได้ดังต่อไปนี้คือ

5.2.1 การเล่นไฟล์เสียงเพลงรอสาย และเสียงตอบรับอัตโนมัติ ในงานวิจัยนี้ได้เลือกใช้ไฟล์ .wav (8000Hz mono 16-bit) เพื่อให้การทำงานถูกต้อง เนื่องจากในการทดสอบด้วยไฟล์ .mp3 พบว่าในการโทรแต่ละครั้งระบบไม่สามารถเล่นไฟล์เสียงได้หลายเพลงแบบต่อเนื่อง และเมื่อสายที่ 1 โทรเข้ามาซึ่งกำลังเล่นไฟล์เสียงอยู่ พอมีสายที่ 2 เรียกเข้ามาก็จะได้ยินไฟล์เสียงเดียวกับสายที่ 1 และเล่นไฟล์นั้นไปพร้อมกัน (parallel)

5.2.2 ในกรณีที่เครื่อง Asterisk มีปัญหา softphone ที่ได้กำหนดให้ register กับเครื่องนั้น จะไม่สามารถใช้งานได้ ระบบไม่สามารถปรับให้ softphone เปลี่ยนข้อมูล register แบบอัตโนมัติ

5.2.3 softphone ที่ได้ register ที่เครื่อง Asterisk ต่างเครื่องกันไม่สามารถโทรหากันได้

5.2.4 หน้าเว็บการกำหนดค่าในระบบ และการตั้งค่าต่างๆ จะมีจำนวนจำกัดไม่สามารถปรับขยายได้แบบอัตโนมัติ

5.3 ข้อเสนอแนะ

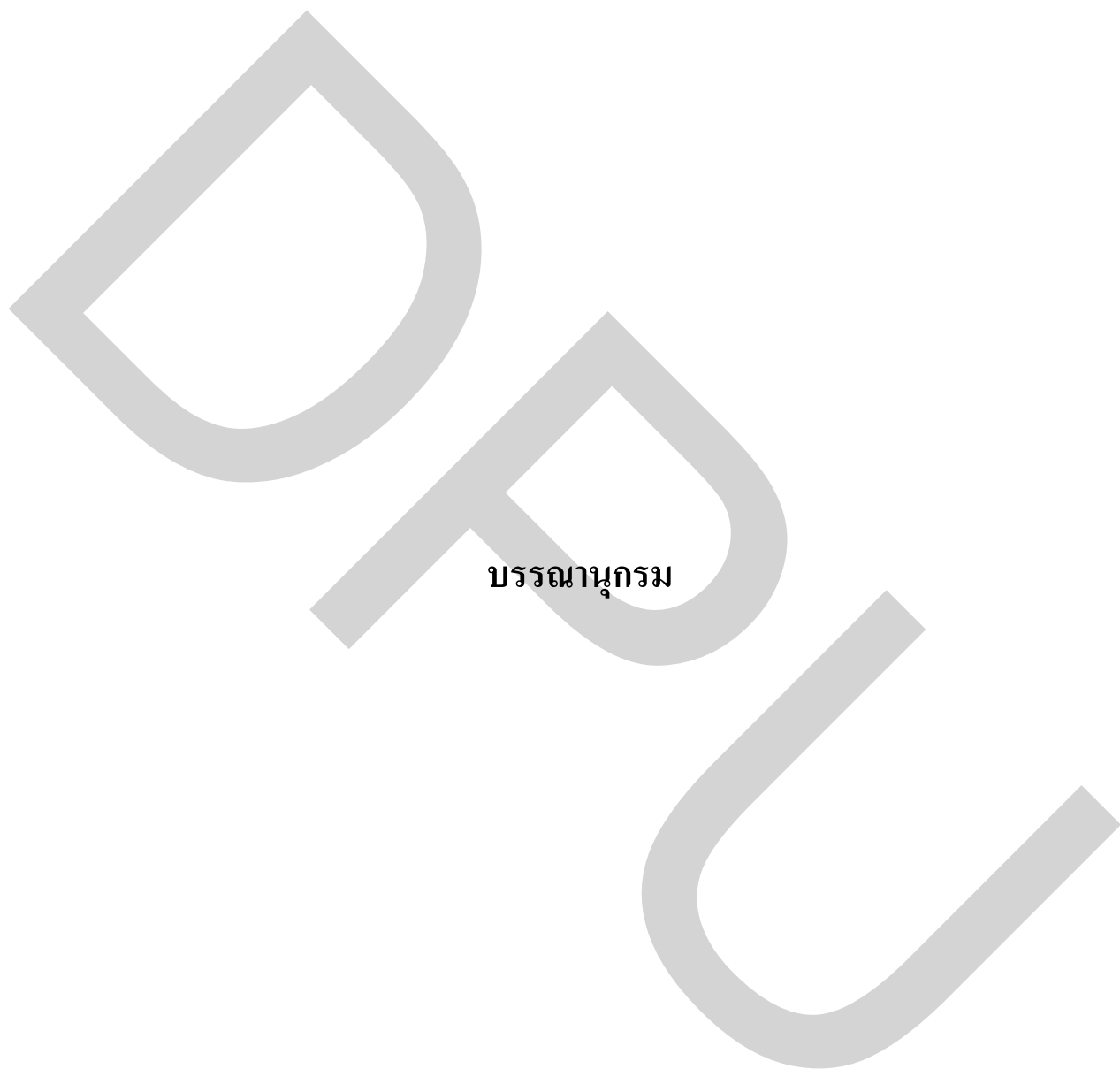
ข้อเสนอแนะของระบบบริการเสียงรอสายและเลือกไม่รับสายในโครงข่าย VoIP ตามสถาปัตยกรรม NGN สามารถแยกข้อเสนอแนะออกเป็นข้อๆ ได้ดังต่อไปนี้คือ

5.3.1 ระบบควรจะพัฒนาให้สามารถรองรับการเล่นไฟล์เสียงได้หลายรูปแบบ

5.3.2 ระบบควรจะพัฒนาให้ softphone ที่ได้ register ที่เครื่อง Asterisk ต่างเครื่องกันสามารถโทรหากันได้ โดยทำ SIP Trunk

5.3.3 ระบบควรจะพัฒนาหน้าเว็บให้สามารถ config maximum ค่าต่างๆ ได้ แบบอัตโนมัติ เช่น จำนวนเพลงรอสายในระบบ, จำนวนข้อความตอบรับอัตโนมัติ, จำนวนเพลงในอัลบั้มตัวเอง, จำนวนช่วงเวลาที่สามารถตั้งค่า RBT และ CS ได้

5.3.4 ระบบควรจะพัฒนาให้สามารถเขียน dial plan ได้จากหน้าเว็บในลักษณะ Flowchart โดยการพัฒนาฟังก์ชันต่างๆ ของ Asterisk ให้เป็นรูปสัญลักษณ์ เมื่อต้องการใช้งานก็เลือกมาวางหน้า page แล้วกำหนด properties ค่าที่จำเป็นของฟังก์ชันนั้น



บรรณานุกรม

บรรณานุกรม

ภาษาไทย

หนังสือ

กิตติพงษ์ สุวรรณราช. (2551). การออกแบบและติดตั้งระบบโทรศัพท์ IP-PBX ด้วย Asterisk.

กรุงเทพฯ: ออฟเซ็ท เพรส.

กิตติศักดิ์ เจริญโกคานนท์. (2543). คู่มือเรียนเขียนเว็บอิคอมเมอร์ซด้วย PHP 5.

กรุงเทพฯ: บริษัท ซัคเซส มีเดีย จำกัด.

วิทยานิพนธ์

ชานนทร์ อยู่ญาติมาก. (2553). การพัฒนาระบบแจ้งข่าวสารด้วยเสียงอัตโนมัติผ่านโทรศัพท์ไอพี.

ปริญญาานิพนธ์วิศวกรรมศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์และโทรคมนาคม. กรุงเทพฯ: มหาวิทยาลัยธุรกิจบัณฑิต.

ประณตพล ดลคู่สิตา. (2552). ระบบสนับสนุนการกำหนดค่าการทำงานของไอพีพีบี

เอ็กซ์แบบแอสเทอริกส์. ปริญญาานิพนธ์วิศวกรรมศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์และโทรคมนาคม. กรุงเทพฯ: มหาวิทยาลัยธุรกิจบัณฑิต.

สารสนเทศจากสื่ออิเล็กทรอนิกส์

ปริญญ์ สายอรุณ. (2553). โครงข่ายโทรคมนาคมยุคหน้า Next Generation Network (NGN).

สืบค้นเมื่อ 9 เมษายน 2554,

จาก <http://ongosk.blogspot.com/2010/07/itm-640-id.html>

ปรเมศวร์ กุมารบุญ. (2552). Next Generation Network (NGN) คืออะไร.

สืบค้นเมื่อ 10 กุมภาพันธ์ 2554,

จาก http://www.torakom.com/article_index.php?sub=article_show&art=21

รังสิมา เกียรติยุทธชาติ, สมิตธิชัย ไชยวงศ์. (2554). เทคโนโลยี VoIP. Standard of VoIP

Technology. สืบค้นเมื่อ 20 เมษายน 2554,

จาก <http://www.vcharkarn.com/varticle/17875#P2>

ระบบ Asterisk คืออะไร และทำงานอย่างไร.(2554). สืบค้นเมื่อ 25 มกราคม 2554,

จาก <http://www.asteriskdiy.com/index.php>

ระบบการติดต่อสื่อสารผ่าน IP Network. (2550). สืบค้นเมื่อ 20 มกราคม 2554,

จาก <http://6211-group3.exteen.com/20071215/ip-network>

วิกิพีเดีย. (2554). ภาษาพีเอชพี สืบค้นเมื่อ 25 มกราคม 2554,

จาก <http://th.wikipedia.org/wiki/>

ภาษาต่างประเทศ

BOOKS

Jim Van Meggelen, Leif Madsen, and Jared Smith. (2007). **Asterisk : The Future of Telephony** (2nd Edition). United States of America: O'REILLY.

Nir Simionovich. (2009). **Asterisk Gateway Interface 1.4 and 1.6 Programming**.
Birmingham: PACKT Publishing.

ELECTRONIC SOURCES

Asteriskdiy. (2011). Asterisk. Retrieved Feb 10, 2011,

From <http://www.asteriskdiy.com/index.php>

Asterisk GotoIfTime. (2011). Retrieved Feb 12, 2011,

From <http://www.voip-info.org/wiki/view/Asterisk+cmd+GotoIfTime>

CallScreening. (2011). Retrieved June 5, 2011, From www.dtac.co.th

Lakshmi Tech. (2011). Next Generation Networking. Retrieved June 6, 2011,

From <http://www.ipv4.com/articles/general/Next-Generation-Networking.htm>

Rbth. (2011). Ring Back Tone Holdings Limited. Retrieved June 10, 2011,

From http://rbth.com/ringbacktones/ringbacktones_call_flow.htm

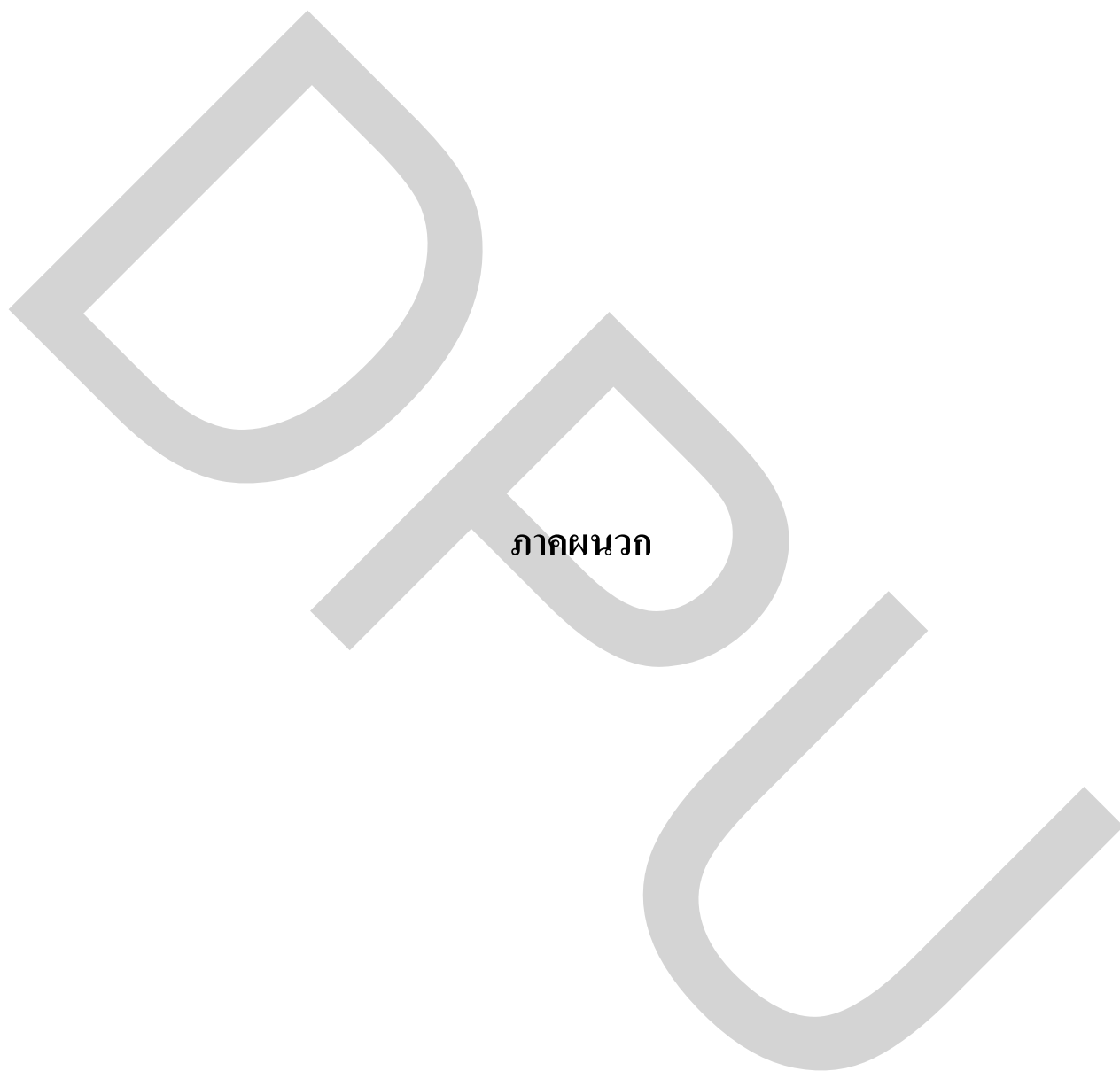
Ring4U. (2011). Retrieved June 8, 2011,

From <http://www.dtacmusic.com/ring4u/web/index.jsp>

SIP Methods (Basic CALL EXAMPLE) [online]. (2011). Retrieved Feb 4, 2011,

From <http://www.ithome.com.tw/plog/index.php?>

Wikipedia. (2011). Retrieved May 8, 2011, From <http://th.wikipedia.org/wiki/>



ภาคผนวก

ภาคผนวก ก ทฤษฎีที่เกี่ยวข้องกับงานวิจัย

ระบบ Voice over IP (VoIP)¹

วีโอไอพี VoIP ย่อมาจาก วอยซ์โอเวอร์ไอพี (Voice over Internet Protocol) (หรือชื่ออื่น IP Telephony, Internet telephony, หรือ Digital Phone) เป็นการสื่อสารทางเสียงผ่านโครงข่ายอินเทอร์เน็ต หรือโครงข่ายอื่นๆที่ใช้อินเทอร์เน็ตโพรโทคอล สัญญาณเสียงจะถูกตัดแบ่งเป็นแพ็กเก็ตวิ่งผ่านไปบนโครงข่ายที่ใช้สำหรับการสื่อสารข้อมูลทั่วไป แทนการใช้วงจรเฉพาะตามวิธีการสื่อสารในระบบโทรศัพท์แบบดั้งเดิม ข้อดีของวีโอไอพี คือการที่สามารถใช้โครงข่ายได้อย่างมีประสิทธิภาพ ทำให้สามารถให้บริการได้ในอัตราค่าบริการที่ถูกลงมาก

สำหรับมาตรฐานที่มีการใช้งานอยู่บนเทคโนโลยี VoIP นั้นโดยทั่วไปจะมีอยู่ 3 มาตรฐานด้วยกัน แต่ที่นิยมได้แก่มาตรฐาน H.323 และมาตรฐาน SIP มาตรฐานเหล่านี้ เราสามารถเรียกได้อีกอย่างหนึ่งว่า “Call Control Technologies” ซึ่งถือว่าเป็นส่วนประกอบสำคัญสำหรับการนำเทคโนโลยี VoIP มาใช้งาน

SIP (Session Initiation Protocol)

มาตรฐาน SIP เป็นมาตรฐานภายใต้ IETF Standard ซึ่งถูกออกแบบมาสำหรับการเชื่อมต่อ VoIP มาตรฐาน SIP นั้นจะเป็นมาตรฐาน Application Layer Control Protocol สำหรับการเริ่มต้น (Creating), การปรับเปลี่ยน (Modifying) และการสิ้นสุด (Terminating) ของ Session หรือการติดต่อสื่อสารหนึ่งครั้ง มาตรฐาน SIP จะมีสถาปัตยกรรมการทำงานคล้ายคลึงการทำงานแบบ Client-Server Protocol เป็นมาตรฐานที่มี Reliability ที่ค่อนข้างสูง

คุณสมบัติของ SIP

1. เป็นโปรโตคอลระดับ Application ซึ่งอยู่เหนือโปรโตคอลระดับ Transport โดยอนุญาตให้สามารถทำการส่งสัญญาณโดยใช้โปรโตคอลระดับ Transport ได้ทั้งชนิด TCP (Transmission Control Protocol) และ UDP (User Datagram Protocol)
2. รูปแบบสัญญาณที่นิยามตามมาตรฐาน SIP มีลักษณะเป็นข้อความ (Text-Based) ซึ่งถูกเรียกว่า SIP Message โดยรูปแบบและไวยากรณ์ของสัญญาณมีลักษณะคล้ายกับรูปแบบสัญญาณของโปรโตคอล HTTP (Hypertext Transfer Protocol) ทำให้ผู้พัฒนาสามารถพัฒนาได้ง่าย และเหตุที่มีการใช้สัญญาณ SIP Message ตลอดขั้นตอนการส่งสัญญาณควบคุมรวมถึงลำดับขั้นตอนการส่งสัญญาณตามมาตรฐาน SIP ไม่ซับซ้อนจึงทำให้โปรโตคอล SIP สามารถทำงานได้รวดเร็วกว่ามาตรฐาน H.323

¹ Wikipedia, 2554. จาก <http://th.wikipedia.org/wiki/>.

3. ตามมาตรฐานของ SIP จะรับผิดชอบขั้นตอนการส่งสัญญาณควบคุมในส่วนก่อนและหลังการสนทนาเท่านั้น โดยในขั้นตอนการส่งข้อมูลมัลติพหุสื่อ (Media Stream) จะใช้โปรโตคอล RTP (Real-Time Transfer Protocol) และในขั้นตอนการแลกเปลี่ยนความสามารถในการส่งข้อมูลมัลติพหุสื่อ (Media Capabilities Exchange) จะใช้โปรโตคอล SDP (Session Description Protocol) ทำงานร่วมด้วย

4. สถาปัตยกรรมตามมาตรฐาน SIP เป็นแบบ Client/Server โดยมีการอ้างอิงตัว SIP Client โดยใช้ชื่ออ้างอิงถึงที่อยู่ของตัว SIP Client นั้นๆ ซึ่งถูกเรียกว่า SIP URL (Uniform Resource Locators) ทำให้เกิดความสะดวกและความยืดหยุ่นในการใช้บริการเพิ่มมากขึ้น

ส่วนประกอบของ Protocol SIP

SIP ประกอบด้วย 2 ส่วนดังนี้ คือ User Agent และ Network Server โดยในแต่ละส่วนมีการทำงานดังนี้

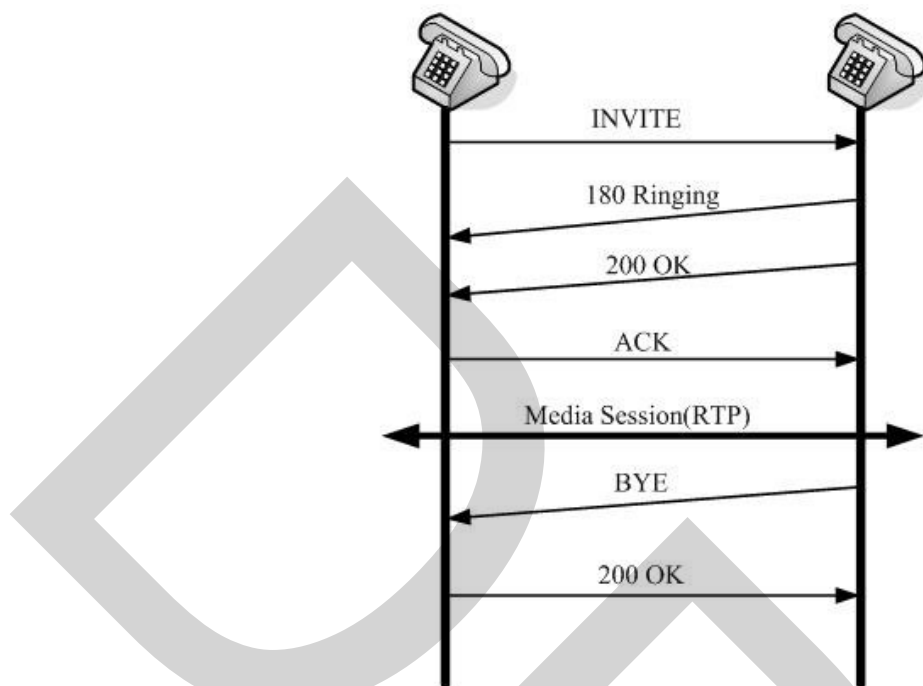
1. User Agents ยังประกอบด้วยส่วนย่อยๆ ดังนี้ User Agent Client และ User Agent Server มีหน้าที่สร้าง Requests และ Response โดยส่วนนี้จะเหมือนกับ end-point

2. Network Server เป็นหัวใจของ SIP ประกอบด้วย Server 3 ชนิด ได้แก่ Proxy Server , Redirect Server และ Registrar Server โดยที่ Proxy Server กระทำเหมือนกับ HTTP Proxy ซึ่งมีหน้าที่ Requests ไปยัง Hop ถัดไปหรือ Proxy ตัวอื่นๆ หรือ Redirect Server ไปยังจุดหมายปลายทาง โดยค้นหาจาก Domain Name Server (DNS)

2.1 Proxy Server เปรียบเสมือน Router ซึ่งการ Requests / Response ครั้งหนึ่งๆ อาจจะมีการกระทำผ่าน Proxy Server หลายๆ ตัว ก่อนจะถึงจุดหมายปลายทางที่แท้จริง

2.2 Redirect Server จะไม่มีการ Forward Requests ไปยัง Hop ถัดไป แต่จะส่ง Response กลับไปยัง Client ของตำแหน่งที่แท้จริงทันที

2.3 Registrar คือการยืนยันการลงทะเบียนกับ SIP Domain โดย UA จะลงทะเบียนโดยใช้ Uniform Resource Identifier (URI) เช่น IP การ Link ไปยังตำแหน่ง Web-page หรือการใช้ E-mail link ตัวอย่างเช่น mailto:use@domain และการติดต่อสื่อสารจะเก็บไว้ที่ Server ตัวอื่นๆ ที่เรียกว่า Location Server (LS) แต่โดยทั่วไปจะอยู่เครื่องเดียวกับเครื่อง Registrar การแลกเปลี่ยนข่าวสาร (Message) อย่างง่าย



รูปที่ ก.1 การแลกเปลี่ยนข่าวสาร(Message) ของ SIP โพรโตคอล

ที่มา : <http://www.ithome.com.tw/plog/index.php?op=ViewArticle&articleId=797&blogId=70>

จากรูปที่ ก.1 เมื่อ A ยกหูเพื่อจะสนทนากับ B จะมีข่าวสารต่าง ๆ ตามลำดับดังนี้

1. ข่าวสาร INVITE จาก A ไปหา B เพื่อสร้างการเชื่อมต่อก่อนที่จะสนทนา ข้อมูลใน SIP INVITE จะบอกถึงชนิดของการสนทนา ยกตัวอย่าง การสนทนาโดยใช้เสียง หรือการสื่อสารแบบมัลติมีเดีย เช่น Video Conference หรือ เกม เป็นต้น

2. ข่าวสาร 180 Ringing ถูกส่งมาจาก B เพื่อตอบสนองข่าวสาร INVITE ว่า B ได้รับข่าวสาร INVITE ทั้งนี้ B จะได้รับสัญญาณซึ่งอาจอยู่ในรูปแบบต่าง ๆ เช่น เสียงกริ่ง หรือ ข้อความกระพริบนำจอโทรศัพท์ของ B

3. เมื่อโทรศัพท์ด้าน B ยอมรับการเรียกแล้ว B จะส่งข่าวสาร 200 OK เพื่อแสดงว่าสามารถทำการสื่อสารในรูปแบบตามชนิดของการสนทนาที่ส่งมา

4. เพื่อที่จะแสดงให้ B มั่นใจว่า A สามารถรองรับการสนทนากับ B ได้ A จะส่งข่าวสาร ACK ไปหา B

5. หลังจากนั้นจะมีการสร้างเซสชันการสื่อสาร (Media Session) โดยใช้โปรโตคอล RTP (สำหรับตัวอย่างนี้)

6. B ส่งข่าวสาร BYE มาที่ A เพื่อขอยุติเซสชันการสนทนา
7. A ทำการส่งสัญญาณ 200 OK เพื่อยอมรับการยกเลิกเซสชันการสนทนา
 - ข่าวสาร INVITE (Invite Message)
 - 1) INVITE sip: Bob@Broadway.com SIP/2.0
 - 2) Via: SIP/2.0/UDPaudiocodes.com:5060
 - 3) Form: Alice
 - 4) To: Bob
 - 5) Call-ID: abfd43978343@audiocodes.com
 - 6) Max-Forwards: 70
 - 7) CSeq: 1 INVITE
 - 8) Contact: Alice
 - 9) Content-Type: application/sdp
 - 10) Content-Length: 142

ฟิลด์ INVITE : แสดงถึง URI (Uniform Resource Indicator) ในที่นี้คือ Bob@Broadway.com ตามด้วยเวอร์ชันของ SIP (SIP/2.0)

ฟิลด์ Via : แสดงถึงชื่อโฮสต์ (ในที่นี้คือ audiocodes.com) ให้ค่า IP Address ตามหลังด้วยหมายเลขพอร์ตของ SIP เพื่อใช้ในการส่งข้อมูลร่วมกับ UDP (User Data Protocol) หรือ TCP (Transmission Control Protocol)

ฟิลด์ From : แสดงชื่อของผู้เรียก (Alice)

ฟิลด์ To : แสดงชื่อผู้รับ (Bob)

ฟิลด์ Call-ID : มีรูปแบบคล้ายกับการกำหนดที่อยู่ของ E-mail ถูกระบุขึ้นเพื่อใช้ติดตามเซสชัน SIP โดยเฉพาะ การรวมกันของฟิลด์ To , From และ Call-ID เรียกว่า “Call leg” เพื่อระบุการเรียกที่เกิดขึ้นครั้งนี้ทั้งสองฝ่าย เนื่องจากเป็นไปได้ว่าจะเกิดการเรียกหลายครั้งระหว่างผู้ใช้ทั้งสอง การร้องขอการเรียกครั้งต่อไปจะอ้างอิงลำดับจาก Call leg นี้

ฟิลด์ Max Forward : คือจำนวนสูงสุดที่ยินยอมให้ส่งค่าได้ เพื่อป้องกันการเกิดวนซ้ำ

ฟิลด์ CSeq : คือ ลำดับคำสั่ง ตามด้วยวิธีการ ซึ่งหมายเลขจะเพิ่มจำนวนเมื่อมีการส่งข่าวสารออกไป

ฟิลด์ Contact : บรรจุ URL ของ SIP เพื่อใช้ในการบอกเส้นทางไปยังปลายทาง

ฟิลด์ Content-Type และ Content-Length : ระบุเนื้อหาข่าวสารในที่นี้เป็น SDP (Session Description Protocol) ซึ่งบรรจุข้อมูลทั้งสิ้น 158 ออกเตต โดยนับออกเตตบรรทัด v=0 จนถึง

บรรทัด a=trpmap:0 PCMU/8000 โดยที่ฟิลด์ Via , To , From ,Call-ID และ CSeq นั้น จำเป็นต้องมีในการกำหนดข่าวสาร SIP ส่วน ฟิลด์อื่นเป็นเพียงออปชั่น

- ข่าวสาร 180 Ringing (Message 180 Ringing)

Message 180 Ringing คือ Message ที่ส่งจากปลายทางเพื่อบอกต้นทางว่าได้รับ Message Invite แล้ว ประกอบด้วยฟิลด์ต่างๆ ดังต่อไปนี้

- 1) SIP/2.0 180 Ringing
- 2) From: UserA <sip:UAA@example.com>;tag=589304
- 3) To: UserB <sip:UAB@example.com>;tag=314159
- 4) Call-ID: 8204589102@example.com
- 5) CSeq: 1 INVITE
- 6) Content Length: 0

- ข่าวสาร 200 OK (Message 200 OK)

ที่ปลายทางจะมีการส่ง Message 200 OK เพื่อเป็นการบอกต้นทางว่าชนิดของเซสชันที่ต้นทางร้องขอได้รับการยอมรับจากปลายทางแล้ว ประกอบด้วยฟิลด์ต่อไปนี้

- 1) SIP/2.0 200 OK
- 2) From: UserA <sip:UAA@example.com>;tag=589304
- 3) To: UserB <sip:UAB@example.com>;tag=314159
- 4) Call-ID: 8204589102@example.com
- 5) CSeq: 1 INVITE
- 6) Contact: <sip:UserB@10.20.30.41>
- 7) Content-Type: application/sdp
- 8) Content-Length: 140

- ข่าวสาร ACK (Message ACK)

ขั้นตอนสุดท้ายของการให้คำยืนยันการใช้เซสชันสื่อสารคือการส่ง MessageACK (Acknowledgement) หมายความว่าต้นทางสามารถรองรับเซสชันสื่อสารของปลายทางได้ซึ่ง Message ACK ประกอบด้วย ฟิลด์ดังนี้

- 1) ACK sip:UAB@example.com SIP/2.0
- 2) Via: SIP/2.0/UDP 10.20.30.41:5060
- 3) Route: <sip:UserB@10.20.30.41>
- 4) From: UserA <sip:UAA@example.com>;tag=589304

5) To: UserB <sip:UAB@example.com>;tag=314159

6) Call-ID: 8204589102@example.com

7) CSeq: 1 ACK

8) Content-Length: 0

- ข่าวสาร BYE (Message BYE)

เมื่อต้นทางส่ง INVITE เสมือนว่าต้นทางเป็น Client และเมื่อปลายทางตอบสนองก็เสมือนเป็น SIP Server หลังจากที่มีการสร้างเซสชันสื่อสารและสนทนากันแล้วโดยโปรโตคอล RTP เมื่อการสนทนาจบแล้วทางปลายทางจะส่ง Message BYE ซึ่งตอนนี้ปลายทางจะทำหน้าที่เป็น Client ส่วนต้นทางจะทำหน้าที่ Server นี่เป็นสาเหตุหนึ่งที่อุปกรณ์ SIP ต้องมีซอฟต์แวร์ที่รองรับการทำงานทั้ง Client และ Server ในเครื่องเดียว ซึ่งข่าวสาร BYE ประกอบด้วยฟิลด์ต่อไปนี้

1) BYE sip:UAB@example.com SIP/2.0

2) Via: SIP/2.0/UDP 10.20.30.41:5060

3) To: UserB <sip:UAB@example.com>;tag=314159

4) From: UserA <sip:UAA@example.com>;tag=589304

5) Call-ID: 8204589102@example.com

6) CSeq: 1 BYE

7) Content-Length: 0

Asterisk²

Asterisk คือ opensource software ที่ทำหน้าที่หลักเป็น Softswitch, IP-PBX หรือที่เรียกว่าตู้ชุมสายโทรศัพท์ระบบ IP ซึ่งมีหน้าที่ในการควบคุมและจัดการบริหาร การเชื่อมต่อระหว่างอุปกรณ์โทรศัพท์ผ่านเครื่องข่ายเนทเวอร์ค อีกทั้งยังสามารถเพิ่มเติมประสิทธิภาพและความสามารถในการทำงานได้โดยง่าย

Asterisk คือ Software ระบบโทรศัพท์แบบ IP-PBX สมบูรณ์แบบ ซึ่งสามารถทำงานได้หลายๆ ระบบปฏิบัติการ เช่น Linux, Mac OS X, OpenBSD, FreeBSD และ Sun Solaris โดยได้มีการจัดเตรียมฟังก์ชันการใช้งานของผู้สาขาโทรศัพท์ PBX (Private Branch eXchange) คุณภาพสูงไว้ในตัว Asterisk รองรับระบบ VoIP (Voice over IP) หลายโปรโตคอล เช่น SIP,

² Jim Van Meggelen, Leif Madsen, and Jared Smith. (2007). Asterisk : The Future of Telephony, Second Edition.

H.323, IAX, MGCP, SCCP (Cisco Skinny) ซึ่งรองรับกับอุปกรณ์โทรศัพท์ที่เป็นมาตรฐานและใช้ฮาร์ดแวร์ที่ราคาไม่แพง

Asterisk ถูกพัฒนาขึ้นโดย Mark Spencer จากบริษัท Digium Inc. เป็น Software ที่ทำงานบนระบบปฏิบัติการหลายระบบ ทั้ง Linux, Max OS X, OpenBSD, FreeBSD และ Sun Solaris แต่ Distribution ของ Linux ที่เป็นที่นิยมสำหรับติดตั้ง Asterisk ก็คือ CentOS ซึ่งเป็น Linux distribution ที่พัฒนามาจาก Red Hat Enterprise Linux (RHEL)

ความสามารถของ Asterisk

1. Switch (PBX) ตู้ชุมสาย

Asterisk สามารถทำหน้าที่เป็นอุปกรณ์สลับสายโทรศัพท์ไม่ว่าจะเป็นระบบ IP หรือ hybrid, สามารถทำการตั้งค่าเส้นทางของการโทรศัพท์โดยตัวเอง, สามารถเพิ่มเติม feature ได้เช่น (ระบบ Voicemail, IVR), รองรับการเชื่อมต่อกับระบบโทรศัพท์พื้นฐานไม่ว่าจะเป็นแบบ analog หรือ digital (ISDN)

2. Gateway

สามารถทำหน้าที่เป็นอุปกรณ์ที่ใช้ในการเชื่อมต่อระหว่างระบบโทรศัพท์พื้นฐานกับระบบ VoIP

3. Feature & Media Server

อีกความสามารถของ Asterisk ก็คือสามารถทำเป็น ระบบตอบรับหรือระบบการประชุมทางโทรศัพท์ เพื่อให้ทำงานเข้ากับระบบโทรศัพท์ที่มีอยู่เดิม ได้อีกด้วย ตัวอย่างการ implementation เช่น สามารถทำเป็น IVR หรือระบบตอบรับ ให้กับตู้ชุมสาย (PABX) เดิมที่ไม่มีระบบตอบรับ

4. Call Center

รองรับการทำงานของระบบ Call-Center อย่างเต็มรูปแบบ เช่น ACD, Queue, IVR, Skill-based routing เป็นต้น

5. CODEC

CODEC ย่อมาจาก “Coder-Decoder” ซึ่งหมายถึง กลไก (ของฮาร์ดแวร์ หรือซอฟต์แวร์) การเข้ารหัสหรือถอดรหัส และการบีบอัด และคลายข้อมูล โดย CODEC จะสามารถใช้ได้กับข้อมูลที่เป็นออดิโอ ข้อความ และวิดีโอ ซึ่งเราจะใช้ CODEC ทำงานร่วมกับแทรค (track) เสียงข้อความที่เป็นซิปโตเทิล และวิดีโอ ในแอปพลิเคชันการประชุมผ่านระบบวิดีโอ และการถ่ายทอดสื่อต่างๆ ด้วยกลไกสตรีมมิ่ง ซึ่ง Asterisk รองรับ CODEC ต่อไปนี้ ADPCM, G.711 (A-Law & μ -Law), G.722, G.723.1, G.726, G.729, GSM, iLBC, Linear, LPC-10 และ Speex

6. Protocols

โพรโทคอล (Protocol) คือระเบียบวิธีการในการติดต่อสื่อสาร เมื่อมาใช้กับเทคโนโลยีสื่อสารโทรคมนาคม จึงหมายถึงขั้นตอนการติดต่อสื่อสาร ซึ่งรวมถึง กฎ ระเบียบ และข้อกำหนดต่าง ๆ รวมถึงมาตรฐานที่ใช้ เพื่อให้ตัวรับและตัวส่งสามารถดำเนินกิจกรรมทางด้านสื่อสารได้สำเร็จ โดย Asterisk รองรับ โพรโทคอลดังนี้ IAX™ (Inter-Asterisk Exchange) , H.323, SIP (Session Initiation Protocol), MGCP (Media Gateway Control Protocol), SCCP (Cisco® Skinny®), E&M, E&M Wink, Feature Group D และ GR-303

การทำงานของ Asterisk

ในการทำงานของ Asterisk ผู้ใช้จำเป็นต้องสามารถที่จะเข้าใจการทำงานและตั้งค่าต่างๆ ของ Asterisk ได้ โดยการตั้งค่าของ Asterisk เพิ่มที่จะทำให้ ระบบทำงานได้หรือเป็นไปตามที่ผู้ใช้ต้องการนั้นมีอยู่ 2 ลักษณะ คือ

1. แบบการกำหนดค่าในไฟล์ระบบ ซึ่งการตั้งค่าแบบนี้ใช้งานกันอย่างแพร่หลายไม่ว่าจะเป็นการตั้งโดยผ่าน Web-Interface เช่น FreePBX โดย ไฟล์ที่เขียนลงไปในระบบจะทำหน้าที่ในการบอกให้ Asterisk ทำงานอย่างที่เราต้องการ

2. แบบใช้ฐานข้อมูล เป็นอีกลักษณะในการตั้งค่าบางประเภทของ Asterisk โดยการตั้งค่าลักษณะนี้มีข้อดีในการที่เราสามารถที่จะปรับเปลี่ยนค่าได้ทันที โดยไม่จำเป็นต้องสั่งให้ Asterisk ทำการอ่านค่าจาก file อีกครั้ง (reload) โดยหลังจากที่การตั้งค่าต่างๆเสร็จสิ้น Asterisk ก็จะสามารถทำงานได้ทันทีโดยในกรณีที่เป็น ระบบ SIP/IAX/etc. เครื่องลูกข่ายต่างๆก็จะสามารถที่จะเชื่อมต่อเข้ามายังระบบ ผ่านระบบเครือข่าย อนึ่งหากต้องการที่จะให้ Asterisk ทำงานกับ Telephony Hardware เช่น การ์ดสายนอก หรือกล่องสายนอกต่างๆ ก็ต้องทำการติดตั้งค่าใน Asterisk ด้วย

PHP³

ภาษาพีเอชพี ในชื่อภาษาอังกฤษว่า PHP ซึ่งใช้เป็นคำย่อแบบกล่าวซ้ำ จากคำว่า PHP Hypertext Preprocessor หรือชื่อเดิม Personal Home Page PHP เป็นภาษาจำพวก scripting language คำสั่งต่างๆจะเก็บอยู่ในไฟล์ที่เรียกว่าสคริปต์ (script) และเวลาใช้งานต้องอาศัยตัวแปลชุดคำสั่ง ตัวอย่างของภาษาสคริปต์ก็เช่น JavaScript, Perl เป็นต้น ลักษณะของ PHP ที่แตกต่างจากภาษาสคริปต์แบบอื่นๆ คือ PHP ได้รับการพัฒนาและออกแบบมาเพื่อใช้งานในการสร้างเอกสารแบบ HTML โดยสามารถ สอดแทรกหรือแก้ไขเนื้อหาได้โดยอัตโนมัติ ดังนั้นจึงกล่าวว่า PHP เป็นภาษา

³ วิกิพีเดีย. (2554). จาก <http://th.wikipedia.org/wiki/>.

ที่เรียกว่า server-side หรือ HTML-embedded scripting language เป็นเครื่องมือที่สำคัญชนิดหนึ่ง ที่ช่วยให้สามารถสร้างเอกสารแบบ Dynamic HTML ได้อย่างมีประสิทธิภาพ

PHP ได้รับการเผยแพร่เป็นครั้งแรกในปีค.ศ.1994 จากนั้นก็มีการพัฒนาต่อมาตามลำดับ เป็นเวอร์ชัน 1 ในปี 1995 เวอร์ชัน 2 (ตอนนั้นใช้ชื่อว่า PHP/FI) ในช่วงระหว่าง 1995-1997 และเวอร์ชัน 3 ช่วง 1997 ถึง 1999 จนถึงเวอร์ชัน 5 ในปัจจุบัน PHP เป็นผลงานที่เติบโตมาจากกลุ่มของนักพัฒนาในเชิงเปิดเผยรหัสต้นฉบับ หรือ OpenSource ดังนั้น PHP จึงมีการพัฒนาไปอย่างรวดเร็วและแพร่หลายโดยเฉพาะอย่างยิ่งเมื่อใช้ร่วมกับ Apache Webserver ระบบปฏิบัติการอย่างเช่น Linux หรือ FreeBSD เป็นต้น ในปัจจุบัน PHP สามารถใช้ร่วมกับ Web Server หลายๆตัวบนระบบปฏิบัติการอย่างเช่น Windows 95/98/NT เป็นต้น

ตัวอย่างภาษาพีเอชพี

ภาษาพีเอชพี จะเป็นส่วนประกอบภายในเว็บเพจ โดยคำสั่งจะปรากฏระหว่าง

<?php ... ?> เช่น

```
<?php
echo "Hello, World!"; ?>
```

หรือ

```
<SCRIPT LANGUAGE = 'php'>
echo "Hello World.";
</SCRIPT>
```

หรือ

```
<%
echo "Hello World.";
%>
```

คุณสมบัติ

การแสดงผลของพีเอชพี จะปรากฏในลักษณะ HTML ซึ่งจะไม่ได้แสดงคำสั่งที่ผู้ใช้เขียน ซึ่งเป็นลักษณะเด่นที่พีเอชพีแตกต่างจากภาษาในลักษณะไคลเอนต์-ไซด์ สคริปต์ เช่น ภาษาจาวาสคริปต์ ที่ผู้ชมเว็บไซต์สามารถอ่าน ดูและคัดลอกคำสั่งไปใช้เองได้ นอกจากนี้พีเอชพียังเป็นภาษาที่เรียนรู้และเริ่มต้นได้ไม่ยากโดยมีเครื่องมือช่วยเหลือและคู่มือที่สามารถหาอ่านได้ฟรีบนอินเทอร์เน็ต ความสามารถการประมวลผลหลักของพีเอชพี ได้แก่ การสร้างเนื้อหาอัตโนมัติจัดการคำสั่ง การอ่านข้อมูลจากผู้ใช้และประมวลผล การอ่านข้อมูลจากฐานข้อมูล ความสามารถจัดการกับ

คู่มือ ซึ่งทำงานเช่นเดียวกับโปรแกรมในลักษณะ CGI คุณสมบัติอื่นเช่นการประมวลผลตามบรรทัด คำสั่ง (command line scripting) ทำให้ผู้เขียน โปรแกรมสร้างสคริปต์พีเอชพี ทำงานผ่านพีเอชพี พาร์เซอร์ (PHP parser) โดยไม่ต้องผ่านเซิร์ฟเวอร์หรือเบราวเซอร์ ซึ่งมีลักษณะเหมือนกับ Cron (ในยูนิกซ์หรือลินุกซ์) หรือ Task Scheduler (ในวินโดวส์) สคริปต์เหล่านี้สามารถนำไปใช้ในแบบ Simple text processing tasks ได้

การแสดงผลของพีเอชพี ถึงแม้ว่าจุดประสงค์หลักใช้ในการแสดงผล HTML แต่ยังสามารถสร้าง XHTML หรือ XML ได้ นอกจากนี้สามารถทำงานร่วมกับคำสั่งเสริมต่างๆ ซึ่งสามารถแสดงผลข้อมูลหลัก PDF แฟลช (โดยใช้ libswf และ Ming) พีเอชพีมีความสามารถอย่างมากในการทำงานเป็นประมวลผลข้อความ จาก POSIX Extended หรือ รูปแบบ Perl ทั่วไป เพื่อแปลงเป็นเอกสาร XML ในการแปลงและเข้าสู่เอกสาร XML เรารองรับมาตรฐาน SAX และ DOM สามารถใช้รูปแบบ XSLT ของเราเพื่อแปลงเอกสาร XML เมื่อใช้พีเอชพีในการทำอีคอมเมิร์ซ สามารถทำงานร่วมกับ โปรแกรมอื่น เช่น Cybercash payment, CyberMUT, VeriSign Payflow Pro และ CCVS functions เพื่อใช้ในการสร้างโปรแกรมทำธุรกรรมทางการเงิน

การรองรับพีเอชพี

คำสั่งของพีเอชพี สามารถสร้างผ่านทางโปรแกรมแก้ไขข้อความทั่วไป เช่น Notepad หรือ vi ซึ่งทำให้การทำงานของพีเอชพี สามารถทำงานได้ในระบบปฏิบัติการหลักเกือบทั้งหมด โดยเมื่อเขียนคำสั่งแล้วนำมาประมวลผล Apache, Microsoft Internet Information Services (IIS) , Personal Web Server, Netscape และ iPlanet servers, Oreilly Website Pro server, Caudium, Xitami, OmniHTTPd, และอื่นๆ อีกมากมาย. สำหรับส่วนหลักของ PHP ยังมี Module ในการรองรับ CGI มาตรฐาน ซึ่ง PHP สามารถทำงานเป็นตัวประมวลผล CGI ด้วย และด้วย PHP, คุณมีอิสรภาพในการเลือก ระบบปฏิบัติการ และ เว็บเซิร์ฟเวอร์ นอกจากนี้คุณยังสามารถใช้สร้างโปรแกรมโครงสร้าง สร้างโปรแกรมเชิงวัตถุ (OOP) หรือสร้างโปรแกรมที่รวมทั้งสองอย่างเข้าด้วยกัน แม้ว่าความสามารถของคำสั่ง OOP มาตรฐานในเวอร์ชันนี้ยังไม่สมบูรณ์ แต่ตัวไลบรารีทั้งหลายของโปรแกรม และตัวโปรแกรมประยุกต์ (รวมถึง PEAR library) ได้ถูกเขียนขึ้นโดยใช้รูปแบบการเขียนแบบ OOP เท่านั้น

พีเอชพีสามารถทำงานร่วมกับฐานข้อมูลได้หลายชนิด ซึ่งฐานข้อมูลส่วนหนึ่งที่รองรับได้แก่ Oracle, dBase, PostgreSQL, IBM, DB2, MySQL, Informix, ODBC โครงสร้างของฐานข้อมูลแบบ DBX ซึ่งทำให้พีเอชพีใช้กับฐานข้อมูลอะไรก็ได้ที่รองรับรูปแบบนี้ และ PHP ยังรองรับ ODBC (Open DataBase Connection) ซึ่งเป็นมาตรฐานการเชื่อมต่อฐานข้อมูลที่ใช้กันแพร่หลายอีกด้วย

พีเอชพียังสามารถรองรับการสื่อสารกับการบริการในโปรโตคอลต่างๆ เช่น LDAP, IMAP, SNMP, NNTP, POP3, HTTP, COM (บนวินโดวส์) และอื่นๆ อีกมากมาย สามารถเปิด Socket บนเครือข่ายโดยตรง และ ตอบโต้โดยใช้ โปรโตคอลใดๆ ก็ได้ PHP มีการรองรับสำหรับการแลกเปลี่ยนข้อมูลแบบ WDDX Complex กับ Web Programming อื่นๆ ทั่วไปได้ ในส่วน Interconnection, พีเอชพีมีการรองรับสำหรับ Java objects ให้เปลี่ยนเป็น PHP Object แล้วใช้งาน และสามารถใช้รูปแบบ CORBA เพื่อเข้าสู่ Remote Object ได้เช่นกัน

MySQL⁴

MySQL (มายเอสคิวแอล) เป็นระบบจัดการฐานข้อมูลโดยใช้ภาษา SQL. แม้ว่า MySQL เป็นซอฟต์แวร์โอเพนซอร์ส แต่แตกต่างจากซอฟต์แวร์โอเพนซอร์สทั่วไป โดยมีการพัฒนาภายใต้บริษัท MySQL AB ในประเทศสวีเดน โดยจัดการ MySQL ทั้งในแบบที่ให้ใช้ฟรี และแบบที่ใช้ในเชิงธุรกิจ

MySQL สร้างขึ้นโดยชาวสวีเดน 2 คน และชาวฟินแลนด์ ชื่อ David Axmark, Allan Larsson และ Michael "Monty" Widenius.

MySQL เป็นที่นิยมใช้กันมากสำหรับฐานข้อมูลสำหรับเว็บไซต์ เช่น มีเดียวิกิ และ phpBB และนิยมใช้งานร่วมกับภาษาโปรแกรม PHP ซึ่งมักจะได้ชื่อว่าเป็นคู่ จะเห็นได้จากคู่มือคอมพิวเตอร์ต่างๆ ที่จะสอนการใช้งาน MySQL และ PHP ควบคู่กันไป นอกจากนี้ หลายภาษาโปรแกรมที่สามารถทำงานร่วมกับฐานข้อมูล MySQL ซึ่งรวมถึง C, C++, Pascal, C#, Java, Perl, PHP, Python, Ruby เป็นต้น ใช้งานผ่าน API สำหรับโปรแกรมที่ติดต่อผ่าน ODBC หรือ ส่วนเชื่อมต่อกับภาษาอื่น (DataBase connector) เช่น เอเอสพี สามารถเรียกใช้ MySQL ผ่านทาง MyODBC, ADO, ADO.NET เป็นต้น

โปรแกรมช่วยในการจัดการฐานข้อมูล และ ทำงานกับฐานข้อมูล

ในการจัดการฐานข้อมูล MySQL สามารถใช้โปรแกรมแบบ command-line เพื่อจัดการฐานข้อมูล (โดยใช้คำสั่ง: mysql และ mysqladmin เป็นต้น) หรือจะดาวน์โหลดโปรแกรมจัดการฐานข้อมูลแบบ GUI จากเว็บไซต์ของ MySQL ซึ่งคือโปรแกรม MySQL Administrator และ MySQL Query Browser เป็นต้น

ส่วนเชื่อมต่อกับภาษาการพัฒนาด้านอื่น (database connector)

มีส่วนติดต่อ (Interface) เพื่อเชื่อมต่อกับภาษาในการพัฒนา อื่นๆ เพื่อให้เข้าถึงฟังก์ชันการทำงานกับฐานข้อมูล MySQL ได้เช่น ODBC (Open DataBase Connector) อันเป็นมาตรฐาน

⁴ วิกิพีเดีย. (2554). จาก <http://th.wikipedia.org/wiki/>.

กลางที่กำหนดมาเพื่อให้ใช้เป็นสะพานในการเชื่อมต่อกับโปรแกรมหรือระบบอื่นๆ เช่น MyODBC อันเป็นไดรเวอร์เพื่อใช้สำหรับการเชื่อมต่อในระบบปฏิบัติการวินโดวส์, JDBC คลาสส่วนเชื่อมต่อสำหรับ Java เพื่อใช้ในการติดต่อกับ MySQL และมี API (Application Programming Interface) ต่างๆมีให้เลือกใช้มากมายในการที่เข้าถึง MySQL โดยไม่ขึ้นอยู่กับภาษาการพัฒนาใดภาษาหนึ่ง นอกเหนือจาก ตัวเชื่อมต่อกับภาษาอื่น (Connector) ที่ได้กล่าวมาแล้ว ยังมี API ที่สนับสนุนในขณะนี้คือ

- DBI สำหรับการเชื่อมต่อกับ ภาษา perl
- Ruby สำหรับการเชื่อมต่อกับ ภาษา ruby
- Python สำหรับการเชื่อมต่อกับ ภาษา python
- .NET สำหรับการเชื่อมกับ ภาษา .NET framework
- MySQL++ สำหรับการเชื่อมต่อกับ ภาษา C++
- Ch สำหรับการเชื่อมต่อกับ Ch (C/C++ interpreter)

ยังมีโปรแกรมอีกตัว เป็นโปรแกรมบริหารพัฒนาโดยผู้อื่น ซึ่งใช้กันอย่างแพร่หลายและนิยมกันเขียนในภาษาพีเอชที เป็น โปรแกรมเว็บแอปพลิเคชัน ชื่อ phpMyAdmin

AGI - Asterisk Gateway Interface⁵

AGI (Asterisk Gateway Interface) เป็น interface ที่สามารถเพิ่มฟังก์ชันการทำงานให้กับ Asterisk ซึ่งฟังก์ชันการทำงานนี้สามารถเขียนได้หลายภาษา เช่น Perl, PHP, C, Pascal

สามารถแบ่งชนิดของ AGI ได้หลายประเภท เช่น

- AGI จะติดต่อกับ dialplan และจะถูกเรียกใช้จาก extensions.conf
- EAGI จะติดต่อกับ channel
- DeadAGI จะใช้สำหรับติดต่อกับ channel หลังการวางหู
- FastAGI ช่วยให้สามารถติดต่อ AGI ผ่าน TCP ได้

AGI ชนิดที่จะนำมาใช้งานวิจัยนี้จะเป็น AGI ซึ่งสามารถควบคุม dialplan หรือแผนการโทร โดยการตรวจสอบวัน-เวลา เลือกไม่รับสาย และ เสียงเพลงรอสาย

⁵ Nir Simionovich. (199x). Asterisk Gateway Interface 1.4 and 1.6 Programming.

ภาคผนวก ข Database ที่เกี่ยวข้องกับงานวิจัย

ตารางรายละเอียดการใช้งาน

เป็นตารางที่อธิบายรายละเอียดการใช้งานระบบ IP-PBX ของ Asterisk โดยจะบันทึกการใช้งานระบบ เช่น หมายเลขต้นทาง หมายเลขปลายทาง เวลาที่ใช้ทั้งหมด เป็นต้น โดยใช้ชื่อตารางว่า cdr ดังแสดงในตารางที่ ข.1

ตารางที่ ข.1 ตาราง cdr

ชื่อคอลัมน์	ชนิดข้อมูล	ค่าจำเพาะ	คำอธิบาย
calldate	Datetime	NOT NULL	วันและเวลาที่ใช้งาน
clid	varchar(80)	NOT NULL	ชื่อที่แสดงหน้าจอ
src	varchar(80)	NOT NULL	หมายเลขต้นทาง
dst	varchar(80)	NOT NULL	หมายเลขปลายทาง
dcontext	varchar(80)	NOT NULL	Context ที่ใช้
channel	varchar(80)	NOT NULL	ช่องสื่อสารที่ใช้
dstchannel	varchar(80)	NOT NULL	ช่องสื่อสารที่เหมาะสม
lastapp	varchar(80)	NOT NULL	คำสั่งสุดท้ายที่ใช้
lastdata	varchar(80)	NOT NULL	ข้อมูลสุดท้ายที่ใช้
duration	int(11)	NOT NULL default '0'	ระยะเวลาที่ใช้ทั้งหมด
billsec	int(11)	NOT NULL default '0'	ระยะเวลาที่สร้างช่องสื่อสาร
disposition	varchar(45)	NOT NULL	การตอบสนองต่อสาย
amaflags	int(11)	NOT NULL default '0'	การเรียกเก็บค่าบริการ
accountcode	varchar(200)	NOT NULL	การเรียกเก็บค่าบริการ
userfield	varchar(255)	NOT NULL	สำรองสำหรับผู้ใช้
uniqueid	varchar(32)	NOT NULL	ตัวระบุให้ไม่ซ้ำ Channel

ตารางระบบฝากข้อความ

เป็นตารางที่อธิบายเกี่ยวกับรายละเอียดการฝากข้อความของหมายเลขที่ไม่ได้รับสาย เช่นกำหนดฝากข้อความไปที่อีเมลเป็นต้น โดยใช้ชื่อตาราง voicemail_users ดังแสดงในตารางที่ ข.2

ตารางที่ ข.2 ตาราง voicemail_users

ชื่อคอลัมน์	ชนิดข้อมูล	ค่าจำเพาะ	คำอธิบาย
uniqueid	int(11)	NOT NULL	
customer_id	int(11)	NOT NULL	หมายเลข extension ผู้ใช้
context	varchar(50)	NOT NULL	ชื่อ context
mailbox	int(5)	NOT NULL	ชื่อ mailbox
password	varchar(10)	NOT NULL	รหัสผ่าน
fullname	varchar(50)	NULL	
email	varchar(50)	NOT NULL	อีเมล
pager	varchar(50)	NULL	
stamp	timestamp	NOT NULL	

ตารางการเล่นไฟล์เสียง

เป็นตารางที่อธิบายเกี่ยวกับที่เก็บไฟล์เสียงของผู้ใช้ระบบบริการเสริมฯ โดยที่เมื่อมีผู้ใช้รายอื่นทำการโทรเข้ามา และได้มีการตั้งค่าเสียงเพลงให้กับหมายเลขที่โทรเข้ามา ข้อมูลที่สำคัญคือชื่อเพลงจะถูกส่งมาทำการค้นหาในตาราง musiconhold ดังแสดงในตารางที่ ข.3 เช่นที่เก็บไฟล์รูปแบบการเล่นไฟล์เสียง เป็นต้น

ตารางที่ ข.3 ตาราง musiconhold

ชื่อคอลัมน์	ชนิดข้อมูล	ค่าจำเพาะ	คำอธิบาย
name	varchar(80)	NOT NULL	ชื่อ folder ที่เก็บเพลง
directory	varchar(255)	NOT NULL	path ที่เก็บเพลง
application	varchar(255)	NULL	
mode	varchar(80)	NOT NULL	กำหนดเป็น files
digit	char(1)	NULL	
sort	varchar(16)	NOT NULL	กำหนดเป็น random
format	varchar(16)	NOT NULL	format file เสียง กำหนดเป็น wav

ตารางแผนการโทร

เป็นตารางที่อธิบายเกี่ยวกับผู้ใช้ระบบบริการเสริมฯ โดยที่เมื่อผู้ใช้ทำการลงชื่อเข้าใช้งานสำเร็จต้องการที่โทรออก ข้อมูลที่สำคัญคือ หมายเลขปลายทางจะถูกส่งมาทำงานที่ตาราง extensions_table ดังแสดงในตารางที่ ข.4 ซึ่งตารางนี้จะเป็นตารางหลักที่กำหนดรูปแบบการโทรออกตามเงื่อนไขต่างๆ เช่น มีการตรวจสอบวัน-เวลา เล่นเสียงเพลงรอสาย เสียงสายไม่ว่าง เสียงระบบตอบรับอัตโนมัติ กำหนดให้มีเสียงเรียก(กริ่ง)นานกี่วินาที กำหนดการฝากข้อความเข้าระบบรับฝากข้อความเสียง เป็นต้น

ตารางที่ ข.4 ตาราง extensions_table

ชื่อคอลัมน์	ชนิดข้อมูล	ค่าจำเพาะ	คำอธิบาย
id	int(11)	AUTO_INCREMENT	ลำดับ
context	varchar(20)	NOT NULL	ชื่อ context
exten	varchar(20)	NOT NULL	หมายเลข extension
priority	tinyint(4)	NOT NULL	ลำดับความสำคัญ
app	varchar(20)	NOT NULL	application ที่ใช้
appdata	varchar(128)	NOT NULL	ข้อมูลที่ใช้

ภาคผนวก ค

เสียงเพลงรอสายที่ใช้ทดสอบงานวิจัย

ตารางที่ ค.5 เสียงเพลงรอสายที่ใช้ทดสอบระบบ

ชื่อเพลง	เนื้อเพลง
When You Say Nothing At All.wav	“All day long I can hear people talking out loud ...”
Closing Time.wav	“Closing time ...”
How Do I Live.wav	“How Do I Get Through ...”
ความอ่อนแอ.wav	“ความอ่อนแอที่มีวันนี้มันทำให้ฉันกลัว ...”
เก็บดาว.wav	“เก็บดาวบนท้องฟ้า ...”
คำตอบสุดท้าย.wav	“โปรดอย่าปล่อยมือฉัน ...”
คิดมาก.wav	“อยากจะบอกให้เธอรู้...”
สัมหวนน้ำตาลเปรี้ยว.wav	“ก็ตอนนั้นฉันเองไม่คิดเลย ...”
คิดเหมือนกันหรือเปล่า.wav	“คิดเหมือนกันหรือเปล่า ...”
หุ้มอยู่ในใจ.wav	“วันที่ความฝัน ...”
คือเธอ.wav	“ทุกวันที่หมุนผ่าน ก็ยังทำให้เข้าใจ ...”
มากกว่ารัก.wav	“เธอเป็นมากกว่ารัก เพราะ ...”
รักไม่ต้องการเวลา.wav	“ลมหายใจ ...”
เรือลำหนึ่ง.wav	“เราคงเป็นดั่งเรือลำหนึ่ง ...”
หวัง.wav	“ทุกๆครั้งที่เธอนั้นส่งรอยยิ้มเข้ามาทักทาย ...”
อยู่ ๆ ก็มาปรากฏตัวในหัวใจ.wav	“อยู่ ๆ ก็มีแต่เธอมาปรากฏตัวในหัวใจ...”
เฉียด.wav	“อยากจะขอได้ไหม หยุดเวลานี้ไว้ ...”
This I promise you.wav	“And I will take you in my arms ...”
ลมหนาว.wav	“ใกล้หน้าหนาวทุกครั้ง ...”
สักวันฉันจะดีพอ.wav	“อยากดีพอ ให้เธอได้มั่นใจ ...”

เสียงตอบรับอัตโนมัติที่ใช้ทดสอบงานวิจัย

ตารางที่ ค.6 เสียงตอบรับอัตโนมัติที่ใช้ทดสอบระบบ

ชื่อข้อความ	ข้อความ
กำลังประชุม.wav	“ขออภัยค่ะขณะนี้กำลังประชุมไม่สะดวกรับสาย”
ตั้งใจเรียน.wav	“ขออภัยค่ะไม่สามารถรับสายได้คนขยันกำลังตั้งใจเรียน”
มีปัญหาหัวใจ.wav	“ขออภัยค่ะเบอร์นี้มีปัญหาหัวใจไม่พร้อมคุยกับใครนะ”
ไม่สะดวกรับสาย.wav	“ขออภัยค่ะขณะนี้ไม่สะดวกรับสาย”
อยู่ต่างประเทศ.wav	“ขออภัยค่ะขณะนี้อยู่ต่างประเทศไม่สะดวกรับสาย”

ประวัติผู้เขียน

ชื่อ-นามสกุล

นางสาว บุญรัตน์ ขวัญคุณฤ์

ประวัติการศึกษา

วิทยาศาสตรบัณฑิต สาขาวิทยาการคอมพิวเตอร์
มหาวิทยาลัยสยาม

ตำแหน่งและสถานที่ทำงานปัจจุบัน

โปรแกรมเมอร์ บริษัท ซอร์ฟมีเดียพลัส จำกัด