

บทที่ 2

วรรณกรรมและงานวิจัยที่เกี่ยวข้อง

2.1 ทฤษฎีและแนวคิดที่เกี่ยวข้อง

ทฤษฎีและแนวคิดที่เกี่ยวข้องกับระบบโทรศัพท์ตอบรับอัตโนมัติ IVR, VoIP, Asterisk และเทคโนโลยีเว็บเซิร์ฟเวอร์ ซึ่งทั้งหมดมีรายละเอียดของเนื้อหาดังต่อไปนี้

2.1.1 Voice over Internet Protocol (VoIP)

การใช้งานโทรศัพท์บนเครือข่ายอินเทอร์เน็ตโดยใช้โปรโตคอลชื่อ IP (Internet Protocol) โดยต้องอาศัยอุปกรณ์ (Hardware) หรือโปรแกรมคอมพิวเตอร์ทำงานร่วมกัน เทคโนโลยีนี้ถูกคิดค้นขึ้นโดยองค์กร Advanced Research Project Agency Network (ARPANET) เป็นเทคโนโลยีที่ช่วยประหยัดต้นทุน เทคโนโลยีนี้นิยมใช้ในองค์กรเพื่อประหยัดค่าใช้จ่ายสำหรับการโทรภายในองค์กร

2.1.2 Interactive Voice Response (IVR) หรือ เทคโนโลยีระบบโทรศัพท์ตอบรับอัตโนมัติ

ซอฟต์แวร์ประยุกต์ที่ขอรับการนำเข้าเสียงและกดปุ่มให้การตอบสนองตามความเหมาะสมในรูปแบบของเสียง การเรียกกลับ และตัวกลางอื่นๆ โปรแกรมประยุกต์ IVR เป็นการให้บริการข้อมูลและทำการธุรกิจอัตโนมัติ โดยจะเชื่อมต่อกับระบบฐานข้อมูลโทรศัพท์ เพื่อให้บริการแบบออนไลน์ ระบบ IVR จะเก็บเสียงที่บันทึกไว้ล่วงหน้า โดยจัดเก็บเป็นส่วนๆตามเนื้อหาของบริการ เมื่อมีสามเข้ามายัง PBX มายัง IVR ระบบจะรับคำสั่งรูปแบบของ DTMF จากแป้นโทรศัพท์ และไปค้นหาข้อมูลจากฐานข้อมูลมาตอบในรูปแบบของเสียงตามที่ได้นับที่ไว้ล่วงหน้า เช่น รายงานสภาพอากาศ รายงานผลการแข่งขันกีฬา เป็นต้น

กระบวนการขอใช้บริการผ่านทางโทรศัพท์ตอบรับอัตโนมัติ ประกอบด้วย 3 ส่วนหลัก คือ

1. ลูกค้า (Customer) คือผู้ขอใช้บริการ IVR
2. ผู้ควบคุมการให้บริการ (Operator) เช่น AIS, DTAC, TRUEMOVE
3. ผู้ให้บริการ (Content Provider) หรือ CP คือผู้ที่เปิดให้บริการ IVR

ตัวอย่าง เช่น บริการดูดวงผ่านระบบโทรศัพท์ตอบรับอัตโนมัติ โดยกดหมายเลข *000 แล้วกดโทรออก เมื่อลูกค้ากดหมายเลข IVR ตามตัวอย่าง ทาง Operator จะเทียบหมายเลข IVR กับการขอเปิดหมายเลข IVR ที่มาจาก Content Provider เพื่อทำการ Route call ไปยังปลายทางได้ถูกต้อง โดย

ระบบการให้บริการขึ้นอยู่กับ Content Provider ในการพัฒนาโปรแกรมเพื่อรองรับบริการในรูปแบบใด ในส่วนของรายได้ที่เกิดขึ้นจะขึ้นอยู่กับค่าธรรมเนียมการตรงลงตอนทำสัญญากันระหว่าง Operator กับ Content Provider เมื่อคู่สายของลูกค้ามาถึงปลายทางได้ถูกต้องแล้ว ต่อไปคือการใช้บริการเช่น เมื่อเข้าสู่ระบบลูกค้าจะได้ยินเสียงเมนูเพื่อเลือกทำรายการที่ต้องการ หลังจากที่ลูกค้าเลือกแล้วระบบจะเล่นเสียงตรงตามกับเมนูที่เลือก

ข้อดีของ IVR คือ สามารถให้บริการข้อมูลได้อย่างรวดเร็ว เพราะสามารถใช้บริการผ่านโทรศัพท์มือถือได้ทุกเครื่อง

2.1.3 Asterisk

2.1.3.1 Asterisk คือ ซอฟต์แวร์ระบบโทรศัพท์แบบ IP-PBX สมบูรณ์แบบซึ่งสามารถทำงานได้บนหลายๆ ระบบปฏิบัติการ เช่น Linux, Mac OSX, OpenBSD, FreeBSD และ Sun solaris โดยได้มีการจัดเตรียมฟังก์ชันการใช้งานของผู้สาขาโทรศัพท์ PBX (Private Branch Solaris) คุณภาพสูงไว้ในตัว Asterisk รองรับกับระบบ VoIP (Voice over IP) หลายโปรโตคอล เช่น SIP, H.323, IAX, MGCP, SCCP (Cisco Skinny) ซึ่งรองรับกับอุปกรณ์โทรศัพท์ที่เป็นมาตรฐานและใช้ฮาร์ดแวร์ที่ราคาไม่แพง Asterisk มีการเผยแพร่แบบ Open source ภายใต้ GNU General Public License (GPL) นั้นหมายความว่า เราสามารถที่จะดาวน์โหลดโปรแกรม Asterisk มาใช้งานได้ฟรีตามข้อกำหนด Asterisk เป็นซอฟต์แวร์ที่ทำงานเป็น Process อยู่บนเครื่องคอมพิวเตอร์ โดยการตั้งค่าของ Asterisk เพื่อให้ระบบทำงานได้มีอยู่ 2 แบบคือ

1. แบบ .conf ไฟล์ การตั้งค่าที่นิยมกันแพร่หลายไม่ว่าจะเป็นการตั้งค่าโดยผ่านเว็บอินเตอร์เฟส เช่น FreePBX หรือการเขียนลงบน Asterisk Dial plan โดยระบบจะทำหน้าที่ในการควบคุม Asterisk ให้ทำงานตามที่เราต้องการ

2. แบบ database เป็นลักษณะการตั้งค่าบางประเภทของ Asterisk ข้อดีคือสามารถปรับเปลี่ยนได้ทันทีโดยไม่จำเป็นต้อง Reload Asterisk

2.1.3.2 Asterisk Dial plan เป็นกลไกควบคุมการทำงานของ Call Flow ให้ทำงานตามที่ออกแบบระบบไว้ ไฟล์ที่ใช้ควบคุม dial plan คือ extension.conf และสำหรับที่อยู่ของไฟล์ etc/asterisk/extension.conf Dialplan จะประกอบด้วย 4 ส่วน ได้แก่

1. contexts คือ ส่วนย่อยของ dial plan ซึ่งแต่ละ contexts จะแยกจากกันและไม่สามารถสร้าง contexts ที่มีชื่อซ้ำกันได้ เนื่องจากในระบบจะไม่ทราบว่าต้องอ้างถึง contexts ตัวใด ในส่วนของชื่อ contexts จะสามารถประกอบด้วย A_Z, a-z, 0-9, - (hyphen), _ (underscore) และชื่อต้องอยู่ภายในสัญลักษณ์ [(Right square bracket),] (Left square bracket)

ตัวอย่างการตั้งชื่อ contexts

```
[incoming]
[dial-out]
[call_out]
```

ภาพที่ 2.1 ตัวอย่างการตั้งชื่อ contexts

2. extensions คือ ขั้นตอนในการทำงานของระบบเช่น เมื่อมีสายเข้ามาระบบต้องจัดการให้เข้าใช้งาน contextsใด และextensionsในcontextsจะเป็นประตัวประมวลผลในการทำงานตามลำดับความสำคัญ (Priorities) ที่กำหนดไว้ รูปแบบการเขียน extension คือ
 exten => name, priority, application

```
exten => 100,1,Answer()
extension name = 100
priority = 1
application = Answer()
```

ภาพที่ 2.2 ตัวอย่างรูปแบบการเขียน extensions

3. priority คือ ลำดับขั้นตอนการทำงานในแต่ละ contexts โดยปกติแล้ว Asterisk จะทำงานจากบนลงล่าง แต่ถ้าในกรณีมีการกำหนด Priority ไว้ ระบบจะสามารถทำงานข้ามขั้นตอนหรือข้ามไปในส่วนที่ต้องการได้ทันที

```
[incoming]
exten => 100,1,Answer()
exten => 100,2,Playback()
exten => 100,3,Hangup()
```

ภาพที่ 2.3 ตัวอย่างรูปแบบการเขียน priority

กรณี contexts มีหลายๆขั้นตอนการทำงาน ปัญหาที่เกิดขึ้นเมื่อเขียน Priority เป็นลำดับตัวเลขคือ เมื่อมีการแก้ไขลำดับต้องแก้ไขเลขเพื่อเรียงลำดับใด ในกรณีนี้จึงนิยมเขียนแบบ unnumbered priority คือ การเขียน Priority เริ่มต้นเป็นเลข 1 จากนั้น Priority ถัดมาจะเป็นตัว n ระบบจะเข้าใจว่าให้ทำงานบรรทัดต่อไปเรื่อยๆ ตัวอย่างเช่น

```
[incoming]
exten => 100,1,Answer()
exten => 100,n,Playback()
exten => 100,n,Hangup()
```

ภาพที่ 2.4 ตัวอย่างรูปแบบการเขียน priority แบบ unnumbered priority

4. Application คือ โปรแกรมที่ทำงานต่างๆ เช่น Answer() จะเป็น Application ที่ตอบรับ call ที่เข้ามาที่ channel, Background(sound file) จะเป็น Application สำหรับเล่น sound file แต่สามารถรับ Input จากผู้ใช้งานได้ในกรณีกดปุ่มโทรศัพท์, Playback(sound file) จะเป็น Application สำหรับเล่น sound file แต่ไม่สามารถรับอินพุตจากผู้ใช้งานได้

2.1.3.3 Asterisk Gateway Interface (AGI) คือ เป็นช่องทางหรือชุดคำสั่งที่ทำให้เราสามารถควบคุมการทำงานของ Asterisk ได้มีประสิทธิภาพมากยิ่งขึ้น สามารถใช้ได้หลายภาษาไม่ว่าเป็น PERL, PHP, Java, C/C++, .NET ไม่เว้นแต่ shell ต่างๆภาษาเหล่านี้สามารถติดต่อกับ Asterisk ได้อย่างง่ายดาย สำหรับที่อยู่ของไฟล์ AGI คือ var/lib/asterisk/agi-bin

PHP AGI เป็น Class PHP ซึ่งจะช่วยในการเขียนแอปพลิเคชันมาสั่งงาน Asterisk ได้ง่ายและสะดวกมากยิ่งขึ้น PHP AGI เป็น Class สำเร็จรูปสามารถเรียกใช้งานได้ทันที หลังจากการติดตั้ง เริ่มต้นต้อง require phpagi.php ก่อนเพื่อให้สามารถเรียกใช้งาน Class ของ phpagi ได้ จากนั้นเขียน PHP Script ตามด้วยการเขียน dial plan เพื่อสั่งให้ Asterisk ดึง PHP Script มาใช้งาน

```

extension.conf :
[incoming]
exten => 100,1,AGI(script.php)
exten => 100,n,Set(Caller=${caller})
exten => 100,n,Hangup()

script.php :
#!/usr/bin/php -q
<?
require 'phpagi.php';
$agi = new AGI();
$db = 'asterisk';
$dbuser = 'xxx';
$dbpass = 'xxx';
$dbhost = 'localhost';
mysql_connect($dbhost,$dbuser,$dbpass);
mysql_select_db("$db"); //or die("could not open database");
$row=mysql_query("select caller from clients LIMIT 1");
$fetch = mysql_fetch_array($row);
$caller = $fetch[caller];
if (mysql_num_rows($row) > 0)
{
    $agi -> set_variable ("caller",$caller);
}
?>

```

ภาพที่ 2.5 ตัวอย่างการเรียกใช้ AGI สำหรับเชื่อมต่อกับฐานข้อมูลเพื่อดึงข้อมูลหมายเลขโทรศัพท์

2.2 งานวิจัยที่เกี่ยวข้อง

กล่าวถึงงานวิจัยที่มีความคล้ายคลึงกับงานวิจัยของผู้พัฒนา โดยจะแบ่งออกเป็นงานวิจัยที่เกี่ยวกับการให้บริการข้อมูล งานวิจัยที่เกี่ยวข้องกับการให้บริการระบบโทรศัพท์ตอบรับอัตโนมัติและระบบจัดการโทรศัพท์แบบ VoIP (Voice Over IP)

2.2.1 ธนวัฒน์ กันพานิชย์และนายวิวัฒน์ วะสุรี (2551) ระบบตอบรับอัตโนมัติสำหรับแจ้งผลการเรียนของนักศึกษา มหาวิทยาลัยขอนแก่น

ผู้พัฒนาได้พัฒนาระบบตอบรับอัตโนมัติ เพื่อแจ้งผลการเรียนของนักศึกษา มหาวิทยาลัยขอนแก่น ระบบนี้ถูกพัฒนาขึ้นมาเพื่ออำนวยความสะดวกให้แก่นักศึกษา และเพิ่มช่องทางในการเช็คผลการเรียน โดยระบบจะแจ้งผลการเรียน ภาคเรียนล่าสุด นอกจากนี้ระบบยังสามารถแจ้งข่าวสารหรือประกาศต่างของทางมหาวิทยาลัยได้ ในการพัฒนาระบบนี้ใช้เทคโนโลยีสื่อสารด้วยเสียงผ่านเครือข่าย (VoIP Over IP) VoIP และรวมถึงนำเทคโนโลยีสารสังเคราะห์เสียงพูด มาใช้ในการอ่านผลการเรียนผ่านระบบตอบรับอัตโนมัติ

2.2.2 ชัยวัฒน์ พาภิบุญ (2554) ระบบจัดการโทรศัพท์แบบวีโอไอพี

เนื่องจากผู้พัฒนาเห็นวิวัฒนาการของระบบการสื่อสารในปัจจุบันมีการพัฒนามากกว่าเดิม จากแต่ก่อนเป็นระบบอนาล็อก (Analog) เปลี่ยนมาเป็นระบบดิจิทัล (Digital) ซึ่งวิธีการเหล่านี้เรียกว่า VoIP (Voice Over IP) เป็นการเพิ่มเติมมูลค่าอินเทอร์เน็ตเน็ทเดิมๆ ที่มีอยู่แล้วในการดูข้อมูล รับ ส่งข้อมูลต่างๆในชีวิตประจำวันให้สามารถจัดส่งข้อมูลประเภทเสียงไปยังปลายทางได้ เพื่อให้ทันต่อยุคและเทคโนโลยีใหม่ๆที่ถูกพัฒนาขึ้นมา

2.2.3 พันธุ์เทพ วิมุกตะลพ (2550) การศึกษาแนวทางการนำระบบ Automatic Call Center มาใช้ในระบบ Call Center ในมหาวิทยาลัยหอการค้าไทย

การศึกษานี้ได้จัดทำขึ้นเพื่อศึกษาการออกแบบระบบ Call Center ที่จะนำระบบ Automatic Call Center เข้ามาใช้งาน ศึกษาการวางรูปแบบและการทำงานของระบบ Automatic Call Center และศึกษาระบบตอบรับอัตโนมัติตั้งแต่อุปกรณ์หลัก, อุปกรณ์, การออกแบบระบบ IVR การวิจัยครั้งนี้ได้ใช้วิธีสัมภาษณ์ทั้งหมด 3-ท่านคือ (1) หัวหน้าฝ่ายประชาสัมพันธ์ มหาวิทยาลัยหอการค้าไทย (2) ประธานบริษัท Tell Voice (3) ประธาน บริษัทโทรศัพท์เคลื่อนที่รายใหญ่แห่งหนึ่ง โดยนำข้อมูลมาวิเคราะห์ในเรื่องของ งบประมาณความเป็นไปได้ในการลงทุน ผลการวิจัยพบว่า หัวหน้าฝ่ายประชาสัมพันธ์ มีแนวคิดที่จะลงทุนในระบบ Automatic Call Center เพื่อเพิ่มประสิทธิภาพให้กับผู้มาติดต่อสื่อสารและเป็นการลดค่าใช้จ่ายอีกทางหนึ่ง

2.2.4 ญัตินันท์ ทัดพิทักษ์กุล (2551) ระบบตอบรับโทรศัพท์อัตโนมัติเพื่อการรายงานข้อมูลแก่นักเดินทาง

การรายงานข้อมูลการเดินทางในกรุงเทพมหานคร ปัจจุบันเป็นการรับฟังทางวิทยุในรถ ซึ่งการสื่อสาร เป็นแบบทางเดียว ผู้เดินทางไม่สามารถร้องขอข้อมูลที่ตรงตามความต้องการได้ หรือต้องใช้เวลารอสาย Call-center ที่มีอยู่จำกัด การดูข้อมูลผ่านโทรศัพท์มือถือหรือระบบนำทาง อาจไม่มีข้อมูล real time หรือต้องละสายตาในการดูแผนที่ซึ่งอันตรายต่อการขับขี่ และยากต่อการค้นหาการใช้ voice command ในการร้องขอข้อมูลผ่าน call-center หรือ IVR จึงเป็นทางเลือกที่มีประสิทธิภาพและใช้ในหลายประเทศเช่น On Star ของ GM ปัญหาที่จะเกิดขึ้นคือการบริการที่จำกัดทั้งจำนวน operator ที่จะตอบและเวลาในการปฏิบัติงาน เช่นนอกเวลางาน การพัฒนาต้นแบบระบบ IVR แบบอัตโนมัติจึงเป็นสิ่งท้าทายที่จะเกิดขึ้นในอนาคต งานวิจัยนี้เสนอการพัฒนาระบบ IVR โดยรับเสียงจากโทรศัพท์ ทำการแปลงเสียงเป็นข้อความ เพื่อค้นหาคำตอบ และทำการอ่านคำตอบกลับไปทางโทรศัพท์โดยไม่อาศัย Operator สำหรับข้อมูลที่ถาม-ตอบในโครงการนี้จำกัดเฉพาะสภาพจราจรในกรุงเทพมหานคร