



รายงานสหกิจศึกษาฉบับสมบูรณ์

การออกแบบและพัฒนาระบบการเตือนและการตอบรับของปัญหา
ที่กระทบต่อธุรกิจหลักของบริษัท

Escalation process design and improvement

นายพิภกรี เจาะสำเภา

ภาควิชาวิศวกรรมคอมพิวเตอร์ สาขาวิชาวิศวกรรมสารสนเทศ

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2559



รายงานสหกิจศึกษาฉบับสมบูรณ์

การออกแบบและพัฒนาระบบการเตือนและการตอบรับของปัญหา
ที่กระทบต่อธุรกิจหลักของบริษัท
Escalation process design and improvement

นายพิภกร เจ๊ะสาเมาะ

เลขหมู่..... 148601
เลขทะเบียน.....
วันเดือนปี..... 6 ม.ย. 2560

b. 128726257
l.....

ภาควิชาวิศวกรรมคอมพิวเตอร์ สาขาวิชาวิศวกรรมสารสนเทศ

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2559

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อโครงการสหกิจศึกษา การออกแบบและพัฒนาระบบการเตือนและการตอบรับของปัญหาที่กระทบ
ต่อธุรกิจหลักของบริษัท

ชื่อ - สกุล นักศึกษา นายพิภกรี เจ๊ะสาม๊ะ

คณะ วิศวกรรมศาสตร์

ภาควิชา วิศวกรรมคอมพิวเตอร์

ชื่อ - สกุล อาจารย์นิเทศ ดร.พิภูลแก้ว ตังติสานนท์

ชื่อ - สกุล ผู้นิเทศงาน นายรณดลย์ นาคภพ

สถานประกอบการ บริษัท เอ็กซอนโมบิล จำกัด

บทคัดย่อ

ในทุก ๆ บริษัท การดูแลธุรกิจหลักให้ดำเนินไปอย่างราบรื่นนั้นถือเป็นเรื่องที่สำคัญมาก สำหรับบริษัท เอ็กซอนโมบิล จำกัด เมื่อเกิดเหตุฉุกเฉินที่ส่งผลกระทบต่อธุรกิจหลักของบริษัท ผู้ที่พบผลกระทบดังกล่าว หรือผู้ใช้ (User) จะทำการสร้างตั๋วงาน (ticket) ขึ้นมาในระบบที่เรียกว่า ไอทีเซอร์วิสเมเนจเม้นท์ (IT Service Management) เพื่อให้ผู้ที่เกี่ยวข้องกับเหตุฉุกเฉินดังกล่าวเข้ามาแก้ไข โดยตั๋วงานนั้นก็จะมี ความสำคัญต่างกัน ยิ่งตั๋วงานมีความสำคัญสูงมากเท่าใด พนักงานที่เกี่ยวข้องก็ต้องรีบแก้ไขให้ทันท่วงที มากเท่านั้น ในวันทำงานปกติ พนักงานส่วนมากจะคอยดูแลตั๋วงานอยู่ตลอดเวลา ทว่าในวันหยุดสุด สัปดาห์นั้น พนักงานไม่ได้คอยดูแลตั๋วงานตลอดเวลาเหมือนในวันทำงาน ดังนั้น หากมีตั๋วงานความสำคัญ สูงถูกส่งมาในวันหยุดสุดสัปดาห์ เหตุฉุกเฉินดังกล่าวอาจไม่ได้รับการแก้ไขให้ทันท่วงที ซึ่งอาจส่งผลเสียต่อ ธุรกิจหลักของบริษัทได้ ดังนั้น ทีมรีเทล (Retail) ของบริษัท เอ็กซอนโมบิล จำกัด จึงได้สร้างแอปพลิเคชัน ขึ้นมาหนึ่งตัว เรียกว่า ซิงเกิลออนคอลเซอร์วิส (Single On-call Service) หรือเอสโอเอส (SOS) ซึ่งมี กระบวนการทำงานคือ เมื่อระบบเอสโอเอสตรวจพบว่า มีตั๋วงานที่มีความสำคัญสูงถูกส่งมาในระบบ เอสโอ เอสจะทำการส่งอีเมลล์ไปแจ้งเตือนพนักงานที่เกี่ยวข้องกับตั๋วงานดังกล่าว ซึ่งเรียกพนักงานเหล่านี้ว่า พนักงานออนคอล (On-call) โดยจะคอยส่งเมลล์ย้ำเป็นระยะ ๆ จนกว่าพนักงานออนคอลจะทำการตอบ อีเมลล์ดังกล่าว งานวิจัยชิ้นนี้จะเป็นการพัฒนาแอปพลิเคชันดังกล่าวให้มีประสิทธิภาพมากขึ้น ซึ่งจะส่งผล ให้ประสบการณ์การใช้งานดีขึ้นทั้งฝั่งผู้ใช้และฝั่งพนักงานออนคอล

คำสำคัญ: แอปพลิเคชัน, ตั๋วงาน, ซิงเกิลออนคอลเซอร์วิส, พนักงานออนคอล, อีเมลล์, เหตุฉุกเฉิน

Cooperative Title: Escalation process and improvement

Student intern name: Fikkree Jehsamoh

Faculty: Engineering

Department: Computer Engineering

Advisor name: Dr. Pikulkaew Tangtisanon

Mentor name: Ronnadul Nakabhop

Company: ExxonMobil Co., Ltd.

Abstract

The main purpose of every company in the world is to take care of their respective business in order to make it run smoothly. In ExxonMobil Corporate, whenever there are any incident that impact company's core business, the person who found it will create a ticket and submit it into core system called "IT Service Management" and the person who responsible for the incident will need to respond and resolve the issue. Each ticket will have its own priority. If the ticket's priority is high or critical, the in charged-employee will have to immediately check and fix it as soon as possible. In work hour, every employee will always monitor for the ticket, but in weekends, they are not always around their computer like in work hour so they might missed some high priority ticket. This project is developed in order to solve this problem. The application will track high or above priority ticket and repeatedly send SOS email to the in charged-employee until that person reply one of SOS email. This project is focus on improving SOS's efficiency so that it'll result in good experience for both user and On-call.

Keywords: application, ticket, Single On-call Service, On-call, Email, incident

กิตติกรรมประกาศ

การปฏิบัติงานสหกิจศึกษา ณ บริษัท เอ็กซอนโมบิล จำกัด ในหัวข้อโครงการ “การออกแบบและพัฒนาระบบการเตือนและการตอบรับของปัญหาที่ส่งผลกระทบต่อธุรกิจหลักของบริษัท” นั้นสามารถสำเร็จลุล่วงไปได้ด้วยดี โดยได้รับความช่วยเหลือและการสนับสนุนจากหลายฝ่าย ดังนี้

1. คุณอัจฉราวรรณ ฉายสุนทรสิริ ตำแหน่ง Supervisor
2. คุณศรัณย์ ว่องธนาการ ตำแหน่ง Developer
3. คุณทศวิน จ้างประเสริฐ ตำแหน่ง Developer
4. คุณรณดลย์ นาคภพ ตำแหน่ง Developer
5. คุณธัญธร พัฒนานันต์วงศ์ ตำแหน่ง ผู้ช่วยทดสอบงาน
6. คุณวรุฒม์ วรปราชญา ตำแหน่ง ผู้ประสานงานโครงการฝึกงาน
7. นักศึกษาสหกิจศึกษาในโครงการ Internship ของบริษัทเอ็กซอนโมบิลทุกท่าน

และบุคลากรท่านอื่น ๆ ทั้งในและนอกประเทศที่ได้กล่าวถึง ผู้วิจัยขอขอบคุณทุกท่านที่ได้ให้คำแนะนำ ความช่วยเหลือ รวมไปถึงข้อเสนอแนะอันเป็นประโยชน์อย่างยิ่งตลอดการจัดทำโครงการวิจัยชิ้นนี้ นอกจากนี้ยังขอขอบคุณผู้ที่เกี่ยวข้องท่านอื่น ๆ ทั้งดร.พิกุลแก้ว ตั้งติสานนท์ ที่คอยให้คำแนะนำตลอดการปฏิบัติสหกิจศึกษา ณ บริษัท เอ็กซอนโมบิล จำกัด และสมาชิกในครอบครัวทุกท่าน ที่คอยให้กำลังใจเสมอมา

พิภกรี เจ้สาเม้าะ

ผู้วิจัย

สารบัญ

หน้า

บทคัดย่อ.....	I
Abstract.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VII
สารบัญภาพ.....	VIII
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญ.....	1
1.2 วัตถุประสงค์ของการวิจัย.....	3
1.3 ขอบเขตของการวิจัย.....	3
1.4 วิธีการดำเนินการวิจัย.....	4
1.5 ประโยชน์ที่คาดว่าจะได้รับ.....	6
บทที่ 2 แนวคิด ทฤษฎีและงานวิจัยที่เกี่ยวข้อง.....	8
2.1 ยูเอ็มแอล (Unified Modeling Language: UML).....	8
2.1.1 แผนภาพยูสเคส (Use case diagram).....	8
2.1.2 แผนภาพกิจกรรม (Activity diagram).....	9
2.2 ทฤษฎีที่เกี่ยวข้องในด้านภาษาคอมพิวเตอร์.....	10
2.2.1 ภาษาไฮเปอร์เท็กซ์มาร์กอัพ (Hypertext Markup Language).....	10
2.2.2 แคสเคดดิ้งสไตล์ชีท (Cascading Style Sheet).....	11
2.2.3 ภาษาจาวาสคริปต์ (JavaScript Language).....	11

2.2.4 ภาษาซีชาร์ป (C#).....	12
2.2.5 ภาษาเอสคิวแอล (SQL).....	12
2.3 ทฤษฎีกระบวนการออกแบบ (Design process)	13
2.3.1 การออกแบบโดยคำนึงถึงผู้ใช้เป็นศูนย์กลาง (User-Centered Design)	13
2.3.2 แนวคิดการพัฒนาแบบอไจล์ (Agile concept)	14
2.3.3 ทฤษฎีการออกแบบประสบการณ์การใช้งานของผู้ใช้ (User Experience) และส่วนต่อประสานกับผู้ใช้ (User Interface) ด้วยแมททีเรียลดีไซน์ (Material Design).....	15
2.4 ทฤษฎีที่เกี่ยวข้องในด้านซอฟต์แวร์คอมพิวเตอร์.....	16
2.4.1 โปรแกรมไมโครซอฟต์วิซวลสตูดิโอ (Microsoft Visual Studio).....	16
2.4.2 โปรแกรมไมโครซอฟต์เอสคิวแอลเซิร์ฟเวอร์เมนเนจเมนต์สตูดิโอ (Microsoft SQL Server Management Studio).....	19
2.4.3 โปรแกรมไมโครซอฟต์เอ้าท์ลุค (Microsoft Outlook).....	20
2.4.4 สภาพแวดล้อมการพัฒนาสำหรับนักพัฒนาทั่วไป (Common Developer Environment)	21
2.5 ทฤษฎีที่เกี่ยวข้องในด้านเฟรมเวิร์ค	25
2.5.1 เจควีรี่ (jQuery)	25
2.5.2 แมททีเรียลไลซ์ (Materialize).....	25
2.5.3 ฟูลคาลินดาร์ (fullcalendar).....	26
2.5.4 เอเอสพีดีเอทเน็ต (ASP.NET)	26
2.6 ทฤษฎีที่เกี่ยวข้องอื่นๆ.....	26
2.6.1 ไมโครซอฟต์เอ็กซ์เชนจ์เซิร์ฟเวอร์ (Microsoft Exchange Server).....	26
บทที่ 3 วิธีดำเนินการวิจัย.....	27
3.1 โครงสร้างทางสถาปัตยกรรมของระบบ	27
3.2 กระบวนการทำงานของระบบ	28

3.2.1 การพัฒนาในส่วนของ front-end	31
3.2.2 การพัฒนาในส่วนของ back-end.....	35
3.2.3 การสนับสนุนแอปพลิเคชันในส่วนของฐานข้อมูล.....	38
3.2.4 การสนับสนุนผู้ใช้เมื่อมีทีมอื่นมาติดต่อขอใช้งานแอปพลิเคชัน	39
บทที่ 4 ผลการวิจัย.....	40
4.1 การพัฒนาในส่วนของ front-end	40
4.1.1 การพัฒนาในส่วนอินเทอร์เฟซ (interface).....	42
4.1.2 การพัฒนาในส่วนสคริปต์ (script).....	56
4.2 การพัฒนาในส่วนของ back-end	60
4.2.1 การเพิ่มเติมฟังก์ชันในส่วนต่อประสานโปรแกรมประยุกต์ (Application Programming Interface: API).....	60
4.2.2 การเพิ่มเติมฟังก์ชันในส่วนควบคุมอีเมลล์ (SOS Mailer).....	63
4.2.3 การติดตั้งสภาพแวดล้อมสำหรับการพัฒนาขึ้นใหม่	65
4.3 การสนับสนุนแอปพลิเคชันในส่วนของฐานข้อมูล	67
4.4 การสนับสนุนผู้ใช้เมื่อมีทีมอื่นมาติดต่อขอใช้งานแอปพลิเคชัน	69
บทที่ 5 สรุปผลการวิจัยและข้อเสนอแนะ	74
5.1 สรุปผลการวิจัย	74
5.2 ปัญหาที่พบในระหว่างการทำเนิงานวิจัย	74
5.3 แนวทางการแก้ไขปัญหา	74
5.4 แนวทางการพัฒนาต่อและนำไปใช้.....	75
เอกสารอ้างอิง	76
ภาคผนวก ก Poster	78

สารบัญตาราง

หน้า

ตารางที่ 2.1 สัญลักษณ์ที่ใช้ในแผนภาพยุคศส.....	9
ตารางที่ 2.2 สัญลักษณ์ที่ใช้ในแผนภาพกิจกรรม.....	10



สารบัญภาพ

หน้า

บทที่ 2 แนวคิด ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

ภาพที่ 2.1 ภาพรวมของยูซีดี.....	13
ภาพที่ 2.2 ตัวอย่างแมทที่เรียลตี้ไลน์.....	15
ภาพที่ 2.3 สัญลักษณ์ของโปรแกรมไมโครซอฟต์วิซวลสตูดิโอ	16
ภาพที่ 2.4 ส่วนแก้ไขโค้ดของวิซวลสตูดิโอ	17
ภาพที่ 2.5 โปรแกรมไมโครซอฟต์เอสคิวแอลเซิร์ฟเวอร์เมนเนจเม้นต์สตูดิโอ	19
ภาพที่ 2.6 โปรแกรมไมโครซอฟต์เอ้าท์ลุค.....	20
ภาพที่ 2.7 เซิร์ฟเวอร์สภาพแวดล้อมสำหรับนักพัฒนา.....	21
ภาพที่ 2.8 การเชื่อมต่อโปรแกรมรีโมทเดสก์ท็อปกอนเน็คชั่นด้วยไอดีพีเคซ.....	22
ภาพที่ 2.9 การใส่รหัสผ่านเพื่อเข้าสู่เซิร์ฟเวอร์ซีดีอี	22
ภาพที่ 2.10 โปรแกรมอโตเมเต็ดไฟร์คอลเดสก์ท็อปกอลเอนท์.....	23
ภาพที่ 2.11 โปรแกรมอินเทอร์เน็ตอินฟอร์เมชันเซอร์วิสเมนเจอร์.....	24

บทที่ 3 วิธีดำเนินการวิจัย

ภาพที่ 3.1 แสดงโครงสร้างทางสถาปัตยกรรมของระบบเอสไอเอส.....	27
ภาพที่ 3.2 แผนภาพยูสเคสของระบบเอสไอเอส.....	29
ภาพที่ 3.3 แผนภาพแอกทิวิตี้แสดงภาพรวมการทำงานของระบบเอสไอเอส.....	30
ภาพที่ 3.4 เฟรมเวิร์คแมทที่เรียลไลซ์	31
ภาพที่ 3.5 การเชื่อมต่อไปยัง TFS เพื่อดาวน์โหลดโปรเจคต์ที่มีโค้ดต้นฉบับของเอสไอเอสทั้งหมด	32
ภาพที่ 3.6 หน้าแรก (Index) ของเว็บไซต์เอสไอเอสแบบเก่า	33
ภาพที่ 3.7 แผนผังเว็บไซต์ของเอสไอเอสแบบเก่า.....	34
ภาพที่ 3.8 หน้าแรก (Index) ของเว็บไซต์เอสไอเอสแบบใหม่	35
ภาพที่ 3.9 แผนผังเว็บไซต์ของเอสไอเอสแบบใหม่.....	35
ภาพที่ 3.10 ปัญหาที่เกิดขึ้นบนเซิร์ฟเวอร์ที่ใช้สำหรับการพัฒนา ส่งผลให้ไม่สามารถใช้งานได้	37
ภาพที่ 3.11 พอร์ตของเว็บไซต์ที่ใช้งานจริง จะเป็นพอร์ตหมายเลข 80	38

ภาพที่ 3.12	พอร์ตของเว็บไซต์ที่ใช้พัฒนา จะเป็นพอร์ตหมายเลข 83.....	38
ภาพที่ 3.13	แสดงอีเมลล์ของทีมอื่นที่ติดต่อเข้ามาเพื่อขอใช้งานแอปพลิเคชันเอสโอเอส.....	39
 บทที่ 4 ผลการวิจัย		
ภาพที่ 4.1	สัญลักษณ์ (Icon) ของโปรแกรมไมโครซอฟต์วิซวลสตูดิโอ.....	40
ภาพที่ 4.2	หน้าต่างภายนอกของโปรแกรมไมโครซอฟต์วิซวลสตูดิโอ	41
ภาพที่ 4.3	แถบ Navbar บริเวณด้านบนของเว็บไซต์.....	41
ภาพที่ 4.4	เมื่อใช้งานหน้าจอที่มีขนาดเล็ก แถบ Navbar จะหายไปและปรากฏปุ่มสามแถบขึ้นมาแทน	41
ภาพที่ 4.5	แถบ SideNav ที่ปรากฏขึ้นหลังจากคลิกที่ปุ่มสามแถบ	42
ภาพที่ 4.6	เปรียบเทียบหน้าจัดการกะพนักงานออนคอล แบบเก่า (ซ้าย) กับแบบใหม่ (ขวา).....	43
ภาพที่ 4.7	เมื่อผู้ใช้ที่ไม่มีสิทธิ์ในการแก้ไขเข้ามาในหน้านี้ จะไม่พบอะไรปรากฏบนหน้าจอเลย	43
ภาพที่ 4.8	หน้าจัดการกะพนักงานออนคอล (On-call Schedule) หลังจากทำการปรับปรุงแก้ไขแล้ว.	44
ภาพที่ 4.9	ปุ่มช่วงเวลาที่ใช้บ่อย (Frequently used time).....	44
ภาพที่ 4.10	เปรียบเทียบหน้าตรวจสอบกะพนักงานออนคอล แบบเก่า (ซ้าย) กับแบบใหม่ (ขวา)	45
ภาพที่ 4.11	หน้าตรวจสอบกะพนักงานออนคอล (Calendar Overview).....	46
ภาพที่ 4.12	หน้าตรวจสอบกะพนักงานออนคอล เมื่อทำการเลือกแสดงกะของบางทีมเท่านั้น	46
ภาพที่ 4.13	เปรียบเทียบหน้ารายงาน แบบเก่า (ซ้าย) กับแบบใหม่ (ขวา)	47
ภาพที่ 4.14	เมื่อคลิกที่กล่องข้อความ from หรือ to จะปรากฏปฏิทินขึ้นมา.....	48
ภาพที่ 4.15	หน้ารายงาน โดยทำการเลือกดูตัวงานของทีมรีเทลย้อนหลังเป็นเวลา 3 เดือน	48
ภาพที่ 4.16	หน้ารายงาน เมื่อทำการคลิกปุ่ม Click to see more details แล้ว	49
ภาพที่ 4.17	เปรียบเทียบหน้าตั้งค่า แบบเก่า (ซ้าย) กับแบบใหม่ (ขวา)	50
ภาพที่ 4.18	เมื่อคลิกที่ปุ่ม Modify จะสามารถแก้ไขรายละเอียดและเพิ่ม-ถอนรายชื่อได้	51
ภาพที่ 4.19	เมื่อคลิกที่ปุ่ม Add More แล้ว จะสามารถเพิ่มรายชื่อพนักงานออนคอลลงไปในทีมได้	51
ภาพที่ 4.20	เครื่องหมายกากบาทที่จะปรากฏบริเวณด้านซ้ายของรูป หลังคลิกที่ปุ่ม Modify	52
ภาพที่ 4.21	หน้าวิธีการใช้งาน (How to use).....	53
ภาพที่ 4.22	ภาพแสดงวิธีการใช้งานเว็บไซต์เอสโอเอส	53
ภาพที่ 4.23	เมื่อคลิกที่ภาพ จะปรากฏเป็นภาพขนาดใหญ่ขึ้นที่หน้าจอ.....	54
ภาพที่ 4.24	หน้าโปรโตคอลการใช้งาน (Protocol)	55

ภาพที่ 4.25 สามารถคลิกที่การ์ด เพื่ออ่านรายละเอียดเพิ่มเติมได้	55
ภาพที่ 4.26 กล่องข้อความ (Modal) แสดงอีเมลล์ของผู้พัฒนา.....	56
ภาพที่ 4.27 หน้าซบมิทพอร์ทอลที่ใช้ในการทดสอบระบบอีเมลล์เท่านั้น.....	57
ภาพที่ 4.28 หลังคลิกปุ่ม Test แล้ว หากส่งสำเร็จ จะปรากฏข้อความว่า Success	58
ภาพที่ 4.29 หลังคลิกปุ่ม Test แล้ว หากส่งไม่สำเร็จ จะปรากฏข้อความว่า Failed	58
ภาพที่ 4.30 เมื่อคลิกที่ไอคอนไมโครซอฟต์เอ็กเซล จะปรากฏกล่องแจ้งเตือนให้เปิดหรือบันทึกไฟล์	59
ภาพที่ 4.31 ปุ่ม Back to top ที่จะปรากฏเมื่อเลื่อนหน้าจอลงมาระดับหนึ่งเท่านั้น	60
ภาพที่ 4.32 ชื่อผู้ใช้ที่ใช้ส่วนต่อประสานโปรแกรมประยุกต์ในการดึงมาแสดงผลบนเว็บไซต์.....	61
ภาพที่ 4.33 การดึงไอดีผู้ใช้ออกมาเพื่อใช้ในการทดสอบการรับ-ส่งอีเมลล์	61
ภาพที่ 4.34 ลิสต์รายการผลิตภัณฑ์ที่ผู้ใช้สามารถเลือกได้.....	62
ภาพที่ 4.35 ตัวอย่างความแตกต่างของตัวงาน 2 ประเภทจากฐานข้อมูล.....	62
ภาพที่ 4.36 เนื้อหาของอีเมลล์ที่ถูกยกระดับ ก่อนที่จะมีการเพิ่มรายละเอียด.....	64
ภาพที่ 4.37 เนื้อหาของอีเมลล์ยกระดับ หลังมีการเพิ่มรายละเอียดลงไปแล้ว.....	65
ภาพที่ 4.38 การเข้าสู่เซิร์ฟเวอร์ที่ใช้สำหรับการพัฒนาด้วยไอดีสิทธิ์พิเศษ.....	66
ภาพที่ 4.39 การระบุพอร์ตในตัวชี้แหล่งในอินเทอร์เน็ตเพื่อเข้าสู่เว็บไซต์เอสไอเอสสำหรับการพัฒนา... ..	66
ภาพที่ 4.40 สัญลักษณ์ของโปรแกรมไมโครซอฟต์เอสคิวแอลเซิร์ฟเวอร์เมนเนจเมนท์	67
ภาพที่ 4.41 หน้าตาภายนอกของโปรแกรมไมโครซอฟต์เอสคิวแอลเซิร์ฟเวอร์เมนเนจเมนท์	67
ภาพที่ 4.42 หน้าต่างแสดงการเชื่อมต่อเข้าสู่ฐานข้อมูล.....	68
ภาพที่ 4.43 เมื่อคลิก New Query แล้ว จะปรากฏหน้าต่างที่ใช้ในการคิวรี่ขึ้นมา.....	69
ภาพที่ 4.44 แสดงอีเมลล์ของทีมนี้อื่นที่ติดต่อมาเพื่อขอใช้งานแอปพลิเคชันเอสไอเอส	70
ภาพที่ 4.45 คำสั่งคิวรี่ที่ใช้ในการเพิ่มทีมใหม่เข้าไปในฐานข้อมูล	70
ภาพที่ 4.46 การเพิ่มพนักงานออนคอลผ่านหน้าตั้งค่าของเว็บไซต์เอสไอเอส.....	71
ภาพที่ 4.47 การเพิ่มผลิตภัณฑ์ใหม่ลงไปในลิสต์รายการของหน้าซบมิทพอร์ทอล.....	71
ภาพที่ 4.48 ตัวงานที่ใช้ในการทดสอบระบบ	72
ภาพที่ 4.49 อีเมลล์ตอบกลับมายังผู้ใช้เมื่อทำการทดสอบระบบ.....	72
ภาพที่ 4.50 อีเมลล์ตอบกลับมายังผู้ใช้เมื่อพนักงานออนคอลตอบกลับอีเมลล์เอสไอเอสแล้ว.....	72

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญ

บริษัท เอ็กซอนโมบิล จำกัด นั้นเป็นองค์กรขนาดใหญ่ด้านพลังงาน มีการประกอบธุรกิจก๊าซธรรมชาติ ปิโตรเลียม และผลิตภัณฑ์ปิโตรเคมีต่าง ๆ อีกทั้งยังเป็นหนึ่งในบริษัทมหาชนที่มีขนาดใหญ่ที่สุดในโลกอีกด้วย บริษัทนี้มีรูปแบบการทำงานแบบครบวงจร เริ่มตั้งแต่การขุดเจาะน้ำมันดิบ นำน้ำมันดิบที่ได้ไปผ่านกระบวนการกลั่นเพื่อให้ได้น้ำมันที่มีส่วนผสมที่เหมาะสมและลงตัวกับประเภทของผลิตภัณฑ์ ก่อนส่งออกจำหน่ายตามสถานีบริการน้ำมันต่าง ๆ ที่ตั้งอยู่ทั่วโลกภายใต้เครื่องหมายการค้า “เอสโซ่” ในประเทศไทย ฮองกง ฯลฯ และภายใต้เครื่องหมายการค้าอื่น ๆ เช่น “โมบิล” ในประเทศนิวซีแลนด์ และอื่น ๆ ในอีกหลากหลายประเทศ ด้วยเหตุนี้ทำให้บริษัทเอ็กซอนโมบิลมีบริษัทในเครืออยู่มากมายเพื่อดูแลในส่วนต่าง ๆ โดยสามารถแบ่งตามประเภทธุรกิจได้ 3 ประเภท อันได้แก่ ส่วนอัพสตรีม (Upstream) ส่วนดาวนสตรีม (Downstream) และส่วนเคมีภัณฑ์ (Chemical) ซึ่งในแต่ละส่วนธุรกิจก็จะมีวิธีการเก็บและจัดการข้อมูลเป็นของตัวเอง แต่จะมีการเก็บข้อมูลไว้ในระบบคอมพิวเตอร์ของบริษัทเหมือนกันทุกส่วน

การเข้าร่วมโครงการสหกิจศึกษากับทางบริษัท เอ็กซอนโมบิล จำกัด ในแผนกเทคโนโลยีสารสนเทศ (ExxonMobil Information Technology) ซึ่งเป็นแผนกที่คอยสนับสนุนธุรกิจหลักของบริษัทด้วยเทคโนโลยีต่าง ๆ ไม่ว่าจะเป็นการใช้งานโปรแกรมต่าง ๆ ให้เกิดประโยชน์สูงสุดแก่บริษัท เช่น โปรแกรมซิสเต็มส์ แอปพลิเคชันส์ แอนด์โปรดักส์ (Systems, Applications and Products) หรือเรียกโดยย่อว่าเอสเอพี (SAP) การประยุกต์ใช้งานคำสั่งพิเศษต่าง ๆ ภายในโปรแกรมเพื่อเพิ่มประสิทธิภาพในการทำงาน เช่น เพาเควรี่ (Power Query) ในโปรแกรมไมโครซอฟต์เอ็กเซล (Microsoft Excel) หรือจะเป็นการพัฒนาแอปพลิเคชันขึ้นเองเพื่อใช้งานภายในบริษัทโดยเฉพาะก็ตาม นอกจากนี้หน้าที่หลักของแผนกเทคโนโลยีสารสนเทศนั้น ยังครอบคลุมไปถึงการทำตั๋วงาน (ticket) ที่ถูกส่งมาจากระบบกลางที่เรียกว่า ไอทีเซอร์วิสเมเนจเมนต์ (IT Service Management) หรือไอทีเอสเอ็ม (ITSM) ตามคำร้องของผู้ใช้อีกด้วย โดยเนื้องานในตั๋วงานนั้นก็แตกต่างกันไปตามแต่ละสายธุรกิจ

ในสายธุรกิจเซลส์แอนด์มาร์เก็ตติ้ง (Sales and Marketing) นั้น ทิมรีเทล (Retail) เป็นหนึ่งในทีมที่อยู่ภายใต้สายธุรกิจเซลส์แอนด์มาร์เก็ตติ้ง โดยจะมีหน้าที่หลักคือดูแลข้อมูลของการขายน้ำมัน ณ สถานี

บริการต่าง ๆ ที่ตั้งอยู่ทั่วโลก โดยการใช้แอปพลิเคชันที่ชื่อว่า ไซท์เดต้าทรานส์เฟอร์ (Site Data Transfer) หรือ เอสดีที (SDT) ในการรวบรวมข้อมูลการขายน้ำมันในแต่ละวัน จากนั้นจึงส่งข้อมูลที่ได้ออกไปยังอีก แอปพลิเคชันหนึ่งที่ชื่อว่า เซ็นทรัลรีเทลอันแอทเทนเด็ดไซท์โอเปอเรชันส์ (Central Retail Unattended Site Operations) หรือเรียกโดยย่อว่า ครูโซ่ (CRUSO) เพื่อให้ครูโซ่ทำการประมวลผลข้อมูลที่ได้รับ จากนั้นจะส่งข้อมูลที่ประมวลผลแล้วไปให้สายธุรกิจอื่นทำการเกิลีย (reconcile) ข้อมูลต่อไป

เนื่องจากข้อมูลที่ได้มาจากเอสดีทีในแต่ละวันนั้นมีปริมาณมากมายมหาศาลมาก และมีการส่งมาในระยะเวลาวันต่อวัน ทำให้ในบางครั้งข้อมูลก็อาจเกิดความผิดพลาดขึ้นได้ เช่น ไม่มีข้อมูลการขายส่งมาเลย หรืออาจส่งมาซ้ำกันเนื่องมาจากปัญหาของเครือข่ายในขณะนั้น ซึ่งจะส่งผลให้ข้อมูลที่ส่งต่อไปประมวลผลในครูโซ่เกิดความผิดพลาดตามไปด้วย ทำให้ในทีมรีเทลจำเป็นต้องมีพนักงานกลุ่มหนึ่งที่คอยดูแลข้อมูลในส่วนนี้ตลอด 24 ชั่วโมงนอกเหนือจากเวลางานปกติ ซึ่งจะเรียกพนักงานเหล่านี้ว่า ออนคอล (On-call) และเนื่องจากข้อมูลที่เอสดีทีรวบรวมมาในแต่ละวันนั้นมีแหล่งที่มาจากทั่วโลก ทำให้ทีมรีเทลมีพนักงานออนคอลคอยประจำการอยู่ใน 2 เขตเวลาดำเนินการ นั่นคือเขตเวลา UTC +07:00 ซึ่งจะประจำการอยู่ในกรุงเทพมหานคร ประเทศไทย และเขตเวลา UTC -03:00 ซึ่งจะประจำการอยู่ในเมืองคูริติบา ประเทศบราซิล

การได้รับมอบหมายให้เป็นพนักงานออนคอลนั้น หมายความว่าจำเป็นต้องไวต่อเหตุฉุกเฉินเป็นอย่างมาก เมื่อได้รับมอบหมายตัวงานมาจากระบบกลาง คือ ระบบไอทีเอสเอ็มแล้วจำเป็นต้องแก้ไขให้ทันท่วงที แต่เนื่องด้วยในวันหยุด พนักงานออนคอลจะไม่ได้ประจำการอยู่หน้าคอมพิวเตอร์ตลอดเวลา ดังเช่นวันทำงานปกติ ดังนั้นจึงมีบางครั้งที่พนักงานออนคอลไม่ได้สังเกตเห็นว่ามีตัวงานถูกมอบหมายมาให้ตน หมายความว่าเหตุฉุกเฉินนั้นอาจไม่ได้รับการแก้ไขได้ทันท่วงที จึงส่งผลให้ธุรกิจของบริษัทเกิดปัญหาขึ้น ก่อให้เกิดผลกระทบในหลายฝ่ายด้วยกัน ดังนั้น ในปีพ.ศ. 2558 ทางทีมรีเทลจึงได้ทำการคิดค้น แอปพลิเคชันใหม่ขึ้นมา เรียกว่า ซิงเกิลออนคอลเซอร์วิส (Single On-Call Service) หรือเอสโอเอส (SOS) โดยเป็นระบบที่จะคอยเตือนพนักงานออนคอลในขณะนั้น ๆ ทุก ๆ 5 นาที เมื่อมีตัวงานที่มีความสำคัญสูง (high priority ticket) ถูกมอบหมายมาให้ตน ผลลัพธ์ของการมีแอปพลิเคชันนี้ทำให้พนักงานออนคอลสามารถทำงานได้อย่างมีประสิทธิภาพมากขึ้น และยังลดเวลาในการทำงานของพนักงานออนคอลได้อีกด้วย เนื่องจากไม่ต้องไปตรวจสอบเนื้องานที่จำเป็นต้องทำในระบบไอทีเอสเอ็มอีกต่อไป

1.2 วัตถุประสงค์ของการวิจัย

- 1.2.1 ลดความซับซ้อนของขั้นตอนการดำเนินงาน
- 1.2.2 พนักงานออนไลน์ได้ทราบถึงเหตุฉุกเฉินเร็วขึ้น
- 1.2.3 พัฒนาภาพลักษณ์ของแอปพลิเคชันให้ไปในทิศทางเดียวกันมากขึ้น
- 1.2.4 หัวหน้างานจะสามารถทราบรายละเอียดของเหตุฉุกเฉินได้ในทันที
- 1.2.5 ผู้ใช้งานทุกฝ่ายสามารถใช้งานแอปพลิเคชันได้
- 1.2.6 ผู้ใช้งานสามารถทราบรายละเอียดของตัวงานที่ถูกรับเรื่องแล้วได้
- 1.2.7 ฝ่ายจัดการตารางงานสามารถเพิ่ม-ลดพนักงานออนไลน์เองได้
- 1.2.8 ผู้พัฒนาสามารถแก้ไขแอปพลิเคชันได้อย่างอิสระ ไม่ต้องพึ่งไอดีกลางอีกต่อไป

1.3 ขอบเขตของการวิจัย

ขอบเขตของงานวิจัยตลอดระยะเวลาการเข้าร่วมโครงการสหกิจศึกษาที่ได้รับมอบหมายให้รับผิดชอบจากบริษัทสามารถแบ่งได้เป็น 4 ส่วน ดังนี้

- 1.3.1 การพัฒนาในส่วนของ front-end
สามารถจำแนกได้เป็น 2 ส่วนย่อย ๆ ดังนี้
 - 1.3.1.1 ส่วนอินเทอร์เฟซ (interface) สำหรับแสดงผลหน้าแอปพลิเคชัน
 - 1.3.1.2 ส่วนสคริปต์ (script) สำหรับเชื่อมต่อและควบคุมการทำงานของแอปพลิเคชัน
- 1.3.2 การพัฒนาในส่วนของ back-end
 - 1.3.2.1 การเพิ่มเติมฟังก์ชันในส่วนเอพีไอ
 - 1.3.2.2 การเพิ่มเติมฟังก์ชันในส่วนควบคุมอีเมลล์
 - 1.3.2.3 การติดตั้งสภาพแวดล้อมสำหรับการพัฒนาขึ้นใหม่

1.3.3 การสนับสนุนแอปพลิเคชันในส่วนของฐานข้อมูล

1.3.4 การสนับสนุนผู้ใช้เมื่อมีทีมอื่นมาติดต่อเพื่อขอใช้งานแอปพลิเคชัน

1.4 วิธีการดำเนินการวิจัย

การวางแผนการทำงานในขั้นตอนต่าง ๆ ในการดำเนินงานวิจัยสามารถจำแนกออกได้ตามแต่ละส่วนของขอบเขตการวิจัยดังที่กล่าวไว้ข้างต้นได้ ดังนี้

1.4.1 การพัฒนาในส่วนของ front-end

การพัฒนาส่วน front-end ถือเป็นส่วนที่มีปริมาณมากที่สุดในงานวิจัยชิ้นนี้ เนื่องจากแอปพลิเคชัน SOS ถูกพัฒนาโดยนักพัฒนาของบริษัท เอ็กซ์อนโมบิล จำกัด ซึ่งทำการพัฒนาแยกส่วนกัน ทำให้ภาพลักษณ์ของแอปพลิเคชันจะถูกแยกออกเป็น 2 ส่วนโดยสิ้นเชิง ดังนั้น เพื่อไม่ให้เกิดความสับสนต่อผู้ใช้งานทั้งพนักงานออนไลน์และผู้ใช้ที่มาสร้างตัวงานในหน้าแอปพลิเคชัน จึงจำเป็นต้องมีการออกแบบภาพลักษณ์ของแอปพลิเคชันใหม่ทั้งหมด แต่จะยังคงใช้กลไกและตรรกะการทำงานเหมือนเดิม ซึ่งการพัฒนาในส่วน front-end จะมีวิธีการดำเนินงานดังนี้

1.4.1.1 สรุปรวบรวมของแอปพลิเคชันว่ามีคุณลักษณะ (feature) ใดบ้าง พร้อมทั้งศึกษาว่าเฟรมเวิร์คใดที่เหมาะสมกับแอปพลิเคชันดังกล่าว

1.4.1.2 พัฒนาหน้าแอปพลิเคชันแต่ละหน้า โดยเริ่มจากหน้าที่มีความสำคัญมากที่สุด ไปยังหน้าที่มีความสำคัญน้อยที่สุด

1.4.1.3 ทดสอบการทำงานของแอปพลิเคชันหลังทำการพัฒนาแล้ว

1.4.1.4 แก้ไขข้อบกพร่องต่าง ๆ ที่เกิดขึ้นระหว่างการพัฒนาแอปพลิเคชัน

1.4.1.5 ทำแบบสอบถามรวบรวมข้อเสนอแนะไปยังพนักงานออนไลน์และผู้ใช้งานแอปพลิเคชัน เพื่อนำข้อเสนอแนะที่ได้ไปปรับปรุงหน้าแอปพลิเคชันต่อไป

1.4.2 การพัฒนาในส่วนของ back-end

เนื่องจากแอปพลิเคชัน SOS นั้นมีการเชื่อมต่อกับฐานข้อมูลและเซิร์ฟเวอร์แลกเปลี่ยนอีเมลล์ (Microsoft Exchange Server) ดังนั้น การพัฒนาในส่วนของ front-end เพียงอย่าง

เดี๋ยวกังไม่สามารถครอบคลุมไปถึงความสามารถในการเชื่อมต่อดังกล่าวได้ จำเป็นต้องมีการพัฒนาในส่วนของ back-end เพื่อทำการเชื่อมต่อแอปพลิเคชันกับส่วนของฐานข้อมูลและเซิร์ฟเวอร์แลกเปลี่ยนอีเมล ซึ่งในส่วนนี้ผู้พัฒนาแอปพลิเคชันได้พัฒนาไว้เมื่อปีพ.ศ. 2558 เป็นส่วนใหญ่แล้ว มีเพียงบางส่วนเท่านั้นที่ต้องพัฒนาเพิ่มเติมเพื่อเพิ่มขีดความสามารถของแอปพลิเคชันให้ดีขึ้นกว่าเดิม โดยขอบเขตการวิจัยส่วนนี้สามารถจัดลำดับการทำงานได้ ดังนี้

1.4.2.1 ศึกษาขั้นตอนการทำงานและตรรกะของแอปพลิเคชัน รวมไปถึงวิเคราะห์ภาษาคอมพิวเตอร์ที่ต้องใช้ และศึกษาโปรแกรมที่เหมาะสมกับภาษาดังกล่าว

1.4.2.2 พัฒนาแอปพลิเคชันตามความต้องการที่ได้รับมอบหมายมาจากหัวหน้างาน

1.4.2.3 ทดสอบแอปพลิเคชันว่าสามารถทำงานได้ตรงตามที่พัฒนาไว้หรือไม่ พร้อมทั้งปรับปรุงข้อผิดพลาดที่เกิดขึ้น

1.4.3 การสนับสนุนแอปพลิเคชันในส่วนของฐานข้อมูล

ดังที่กล่าวไว้ข้างต้นว่าแอปพลิเคชัน SOS นั้นมีการเชื่อมต่อกับฐานข้อมูล ไม่ว่าจะเป็นข้อมูลตัวงาน ข้อมูลพนักงานออนไลน์ ทีมที่พนักงานออนไลน์สังกัดอยู่ ฯลฯ ทุกอย่างล้วนถูกเก็บไว้ในฐานข้อมูลทั้งสิ้น การเรียนรู้ถึงขั้นตอนในการสนับสนุนแอปพลิเคชันในส่วนของฐานข้อมูลจึงเป็นสิ่งจำเป็นในการพัฒนาแอปพลิเคชัน SOS ให้มีประสิทธิภาพมากขึ้น โดยมีลำดับการทำงาน ดังนี้

1.4.3.1 เรียนรู้โครงสร้างของฐานข้อมูลทั้งหมด

1.4.3.2 ศึกษาภาษาที่ใช้ในการจัดการฐานข้อมูล ซึ่งก็คือภาษาเอสคิวแอล (SQL) พร้อมทั้งศึกษาวิธีการใช้โปรแกรมที่ใช้ในการจัดการฐานข้อมูล คือโปรแกรมไมโครซอฟต์เอสคิวแอลเซิร์ฟเวอร์เมนเนจเมนต์สตูดิโอ (Microsoft SQL Server Management Studio)

1.4.3.3 ติดต่อกับผู้ดูแลฐานข้อมูล (Database Administrator) เป็นระยะ ๆ เมื่อเกิดปัญหาขึ้นในฐานข้อมูล หรือเมื่อต้องการสร้างฐานข้อมูลขึ้นมาใหม่

1.4.4 การสนับสนุนผู้ใช้เมื่อมีทีมอื่นมาติดต่อขอใช้งานแอปพลิเคชัน

เนื่องจากแอปพลิเคชัน SOS ถือเป็นวิธีการแก้ปัญหาที่สามารถใช้งานได้กับทุกทีมที่มีพนักงานออนไลน์เป็นของตนเอง ทำให้ในบางครั้งจะมีทีมอื่นที่ไม่เกี่ยวข้องกับสายธุรกิจเซลล์

แอนด์มาร์เก็ตติ้งมาติดต่อขอใช้งานแอปพลิเคชันนี้ด้วย ซึ่งขั้นตอนการดำเนินงานของขอบเขตสุดท้ายของงานวิจัยนี้ มีดังนี้

1.4.4.1 รวบรวมรายชื่อทีมที่ต้องการใช้งาน และรายชื่อพนักงานออนคอลของทีมดังกล่าวทั้งหมด

1.4.4.2 ทำการเพิ่มรายชื่อทีมดังกล่าวเข้าไปในฐานข้อมูล รวมไปถึงแอปพลิเคชันและผลิตภัณฑ์ของทีมดังกล่าวด้วย

1.4.4.3 ทดสอบการทำงานของแอปพลิเคชันกับพนักงานออนคอลของทีมใหม่ โดยการสร้างตัวงานที่มีความสำคัญสูงและระบุผลิตภัณฑ์ของทีมที่เพิ่งทำการเพิ่มเข้าไปในฐานข้อมูล

1.4.4.4 แก้ไขจุดบกพร่องหากมีการพบเจอ เช่น ไม่มีการส่งอีเมลล์ไปหาพนักงานออนคอลทั้งที่มีตัวงานที่มีความสำคัญสูงถูกส่งมา

1.4.4.5 ทำการแจ้งพนักงานออนคอลทุกคน รวมไปถึงหัวหน้างานของทีมใหม่ว่าพร้อมใช้งานแล้วเมื่อผลลัพธ์ของแอปพลิเคชันเป็นที่น่าพึงพอใจ

1.5 ประโยชน์ที่คาดว่าจะได้รับ

ประโยชน์ที่ได้รับในการพัฒนาแอปพลิเคชันที่ได้รับมอบหมายในการเข้าร่วมโครงการสหกิจศึกษากับทางบริษัท เอ็กซอนโมบิล จำกัด สามารถจำแนกได้เป็น 3 ส่วน ดังนี้

1.5.1 ประโยชน์ต่อพนักงานออนคอล

1.5.1.1 สามารถตั้งค่าตารางเวลาเข้าเวรของพนักงานออนคอลได้ง่ายขึ้น

1.5.1.2 เพิ่มเสถียรภาพและประสิทธิภาพของแอปพลิเคชันให้ดีขึ้น

1.5.1.3 ลดระยะเวลาการทำงานของพนักงานออนคอล

1.5.2 ประโยชน์ต่อผู้ใช้แอปพลิเคชัน SOS

1.5.2.1 ใช้งานแอปพลิเคชันได้ง่ายขึ้น เนื่องจากภาพลักษณ์ที่เปลี่ยนไป

1.5.2.2 สามารถทราบได้ว่ามีตัวงานใดที่ถูกสร้างขึ้นในช่วงเวลาที่กำหนดบ้าง

1.5.2.3 สามารถตรวจสอบเวรของพนักงานออนคอลของแต่ละทีมได้อย่างอิสระ

1.5.3 ประโยชน์ต่อผู้วิจัย

1.5.3.1 ได้รับความรู้ในการพัฒนาเว็บไซต์ ทั้งในส่วนของ front-end และ back-end อย่างมหาศาล

1.5.3.2 ได้เรียนรู้ขั้นตอนและวิธีการใช้งานไมโครซอฟต์เอสคิวแอลเซิร์ฟเวอร์สตูดิโอ รวมถึงการใช้งานโปรแกรมรีโมทเดสก์ท็อปกอนเนคชั่น (Remote Desktop Connection) เพื่อเข้าสู่เซิร์ฟเวอร์จำลองที่เป็นที่ตั้งของแอปพลิเคชัน

1.5.3.3 มีความเข้าใจในขั้นตอนการทำงานมากขึ้น และสามารถปรับตัวเข้ากับสภาพแวดล้อมการทำงานในชีวิตจริงได้ดีขึ้น

1.5.3.4 ได้เรียนรู้ถึงอุปสรรคต่าง ๆ ที่เกิดขึ้นระหว่างการทำงาน รวมไปถึงวิธีการแก้ปัญหาในรูปแบบต่าง ๆ

1.5.3.5 สามารถนำความรู้ที่ได้รับจากโครงการสหกิจศึกษาครั้งนี้ไปเป็นแนวทางในการเตรียมความพร้อมในด้านต่าง ๆ ไม่ว่าจะเป็นการทำงานหรือศึกษาต่อในอนาคต

บทที่ 2

แนวคิด ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

เนื่องจากงานวิจัยชิ้นนี้เป็นการออกแบบและพัฒนาระบบตอบรับอัตโนมัติ ผู้วิจัยจึงต้องทำการศึกษาค้นคว้าทฤษฎีทางคอมพิวเตอร์ (Computer) ที่เกี่ยวข้องเป็นหลัก ไม่ว่าจะเป็นภาษาคอมพิวเตอร์ (Computer Language) ซอฟต์แวร์คอมพิวเตอร์ (Computer Software) ต่าง ๆ ที่จำเป็นต้องใช้ในการพัฒนาแอปพลิเคชัน ตรรกะการทำงาน (Logic) ของแอปพลิเคชันซึ่งเกิดออนคอลล เซอร์วิส (Single On-call Service) หรือเรียกสั้น ๆ ว่า เอสโอเอส (SOS) รวมไปถึงเฟรมเวิร์ค (Framework) ต่าง ๆ ที่ถูกใช้ในการพัฒนาแอปพลิเคชันเอสโอเอส เพื่อที่จะนำความรู้ที่ได้จากการศึกษาทฤษฎีเหล่านี้ไปใช้ในการพัฒนาแอปพลิเคชันเอสโอเอสให้มีประสิทธิภาพมากขึ้นต่อไป โดยแนวคิด ทฤษฎี และงานวิจัยที่เกี่ยวข้องที่นำมาศึกษาในการทำวิจัยมีรายละเอียดดังต่อไปนี้

2.1 ยูเอ็มแอล (Unified Modeling Language: UML)

ยูเอ็มแอล คือภาษาเชิงสัญลักษณ์ที่ใช้ในการอธิบายรายละเอียด แสดงโครงสร้าง จำลองกระบวนการการสร้างสำหรับใช้ในการสร้างแบบจำลองเชิงวัตถุ ยูเอ็มแอลเป็นภาษามาตรฐานที่มีการใช้งานกันอย่างแพร่หลาย ซึ่งโดยทั่วไปจะใช้ในการสร้างพิมพ์เขียวแก่ระบบงาน แต่ก็สามารถนำแผนภาพแต่ละส่วนมาประกอบเข้าด้วยกันเพื่อวิเคราะห์และออกแบบเพื่อนำไปพัฒนาระบบสารสนเทศต่อไปได้ด้วย




2.1.1 แผนภาพยูสเคส (Use case diagram)

แผนภาพยูสเคส คือแผนภาพที่มีเพื่อแสดงให้เห็นถึงความสามารถทำงานอะไรได้บ้าง (Functionality of system) โดยนอกจากจะแสดงให้เห็นถึงความสามารถของระบบแล้ว ยังสามารถระบุถึงความสัมพันธ์ระหว่างระบบย่อยต่าง ๆ ได้ด้วย

แผนภาพยูสเคสทำให้ผู้พัฒนาระบบสามารถแยกแยะกิจกรรมต่าง ๆ ที่เกิดขึ้นในระบบได้ เนื่องจากรูปภาพที่แสดงในแผนภาพยูสเคสนั้นจะเป็นรูปภาพที่เรียบง่าย ไม่ซับซ้อน และยังทำให้การสื่อสารระหว่างผู้ใช้และผู้พัฒนาระบบเป็นไปได้อย่างง่ายตายอีกด้วย โดยผู้ใช้สามารถทราบจากแผนภาพได้โดยตรงว่าสิ่งที่ต้องการมีอยู่ในระบบหรือไม่

2.1.1.1 สัญลักษณ์ที่ใช้ในแผนภาพยูสเคส

ในการเขียนแผนภาพยูสเคสนั้น จะใช้สัญลักษณ์ต่าง ๆ แทนความหมาย ซึ่งสามารถแสดงได้ดังตารางต่อไปนี้

สัญลักษณ์	ชื่อสัญลักษณ์	ความหมายของสัญลักษณ์
	Actor	ผู้ที่เกี่ยวข้องกับระบบ หรืออาจหมายถึงสิ่งที่อยู่ภายนอกระบบแต่มีการติดต่อกับระบบก็ได้
	Use case	หน้าที่ของระบบที่เกิดจากแอกเตอร์ (Actor) ซึ่งต้องใช้กริยาหรือกริยาวลีในการสื่อความหมาย
	System boundary	เส้นแบ่งขอบเขตการทำงานของระบบ ใช้คั่นระหว่างยูสเคสกับแอกเตอร์ หรืออาจใช้คั่นระหว่างระบบกับระบบย่อยก็ได้
	Connection	เส้นที่เกิดขึ้นจากความสัมพันธ์ระหว่างยูสเคสและแอกเตอร์
	Extend	ความสัมพันธ์ระหว่างยูสเคสที่ขยายความสามารถมาจากยูสเคสหลัก ซึ่งอาจเกิดขึ้นในบางกรณีเท่านั้น
	Include	ความสัมพันธ์จากยูสเคสหลักที่ไปดึงความสามารถของยูสเคสอื่นมาใช้งานร่วมกัน


ตารางที่ 2.1 สัญลักษณ์ที่ใช้ในแผนภาพยูสเคส

2.1.2 แผนภาพกิจกรรม (Activity diagram)

แผนภาพกิจกรรม คือแผนภาพที่ไว้ใช้แสดงกิจกรรมที่เกิดขึ้นในลักษณะของการไหลของงาน (Workflow) ซึ่งจะมีลักษณะคล้ายกับผังงาน (Flowchart)

2.1.2.1 สัญลักษณ์ที่ใช้ในแผนภาพกิจกรรม

ในการเขียนแผนภาพกิจกรรมนั้น จะใช้สัญลักษณ์ต่าง ๆ แทนความหมาย ซึ่งสามารถแสดงได้ดังตารางต่อไปนี้

สัญลักษณ์	ชื่อสัญลักษณ์	ความหมายของสัญลักษณ์
	จุดเริ่มต้น (Start)	จุดเริ่มต้นของแผนภาพกิจกรรม แสดงถึงจุดเริ่มต้นของการไหลของงาน
	จุดสิ้นสุด (End)	จุดสิ้นสุดของแผนภาพกิจกรรม แสดงถึงจุดสุดท้ายที่งานจะสามารถทำงานได้
	กิจกรรม (Activity)	กิจกรรมที่ระบบสามารถทำงานได้
	เส้นทางการไหล (Flow)	เส้นที่เกิดขึ้นจากการไหลของกิจกรรม จะไหลจากจุดเริ่มต้นไปยังจุดสิ้นสุดเสมอ
	กิจกรรมแบบมีเงื่อนไข (Conditional activity)	กิจกรรมที่ระบบสามารถทำงานได้ ซึ่งจะแตกต่างจากกิจกรรมแบบปกติตรงที่จะมีเส้นทางการไหลออกมากกว่าหนึ่งเส้น และจะไหลออกตามเงื่อนไข

ตารางที่ 2.2 สัญลักษณ์ที่ใช้ในแผนภาพกิจกรรม

2.2 ทฤษฎีที่เกี่ยวข้องในด้านภาษาคอมพิวเตอร์

ภาษาคอมพิวเตอร์ถือเป็นองค์ประกอบหลักที่สำคัญที่สุดของงานวิจัยชิ้นนี้เนื่องจากการพัฒนาแอปพลิเคชัน ซึ่งภาษาคอมพิวเตอร์ที่นำมาใช้ในการวิจัยมีดังต่อไปนี้

2.2.1 ภาษาไฮเปอร์เท็กซ์มาร์กอัพ (Hypertext Markup Language)

ภาษาไฮเปอร์เท็กซ์มาร์กอัพ หรือที่รู้จักกันในชื่อ ภาษาเอชทีเอ็มแอล (HTML) เป็นภาษาคอมพิวเตอร์ที่ใช้ในการสร้างเว็บเพจ หรือข้อมูลอื่นที่มีการเรียกดูผ่านเว็บเบราว์เซอร์เป็นหลัก สำหรับแอปพลิเคชันเอสโอเอสที่มีการทำงานผ่านเว็บไซต์เป็นส่วนมาก ภาษาเอชทีเอ็มแอลจึงถือได้ว่าเป็นภาษาที่สำคัญและมีบทบาทมากที่สุดในงานวิจัยชิ้นนี้

2.2.1.1 โครงสร้างของภาษาเอชทีเอ็มแอล

ภาษาเอชทีเอ็มแอล เป็นภาษาประเภทหนึ่งในภาษามาร์กอัป มีการเขียนในลักษณะภายในวงเล็บสามเหลี่ยม (<, >) ครอบข้อความที่เป็นลักษณะ เช่น <p>ทดสอบ</p> หมายความว่า กำหนดให้เบรเซอร์คำนวณข้อความ “ทดสอบ” เป็นลักษณะข้อความย่อหน้าทั่วไป โดยเบรเซอร์จะกำหนดการแสดงผลในรูปแบบมาตรฐานเท่านั้น หากผู้พัฒนาต้องการกำหนดลักษณะเฉพาะ จะสามารถทำได้โดยใช้แคสเคดดิ้งสไตล์ชีท (Cascading Style Sheet) หรือซีเอสเอส (CSS) ซึ่งจะกล่าวถึงในหัวข้อถัดไป

โดยปกติแล้ว ภาษาเอชทีเอ็มแอลจะมีการทำงานที่ละคำสั่งภายในวงเล็บสามเหลี่ยม โดยจะนับจากวงเล็บสามเหลี่ยมที่ครอบอยู่นอกสุดไปยังที่ครอบอยู่ในสุด หากระบบพบว่ามียวงเล็บสามเหลี่ยมที่ถูกครอบซ้อนกัน จะเกิด error ขึ้น ทำให้ข้อความดังกล่าวไม่มีการแสดงผลออกมายังหน้าเว็บไซต์ อาทิเช่น <p>ทดสอบ</p> จะเห็นว่าจากตัวอย่างดังกล่าว วงเล็บสามเหลี่ยมที่ควรจะอยู่นอกสุดคือ <p> แต่วงเล็บสามเหลี่ยมที่ใช้ปิดแท็กข้อความนั้นกลับกลายเป็น แทน ผลลัพธ์ของชุดคำสั่งดังกล่าวจะไม่มีการแสดงผลใด ๆ บนเบรเซอร์ เนื่องจากเกิด error ขึ้นนั่นเอง

2.2.2 แคสเคดดิ้งสไตล์ชีท (Cascading Style Sheet)

แคสเคดดิ้งสไตล์ชีท หรือซีเอสเอส (CSS) คือภาษาคอมพิวเตอร์ภาษาหนึ่งในกลุ่มภาษาสไตล์ชีท (Style Sheet) ใช้ในการจัดวางรูปแบบและโครงสร้างของหน้าเว็บไซต์ที่เขียนจากภาษาเอชทีเอ็มแอล ซึ่งมีประโยชน์มากในการทำให้หน้าเว็บไซต์สามารถแสดงผลได้ตามที่ออกแบบไว้

2.2.3 ภาษาจาวาสคริปต์ (JavaScript Language)

ภาษาจาวาสคริปต์ เป็นภาษาสคริปต์เชิงวัตถุ (Object-oriented) ที่ถูกคิดค้นมาเพื่อเขียนโปรแกรมและควบคุมการทำงานของฟังก์ชันต่าง ๆ บนเว็บไซต์โดยเฉพาะ สามารถทำงานข้ามแพลตฟอร์มได้ สามารถทำงานกับภาษาเอชทีเอ็มแอลและซีเอสเอสได้ และยังสามารถควบคุมได้ทั้งฝั่งไคลเอนท์ (Client) และฝั่งเซิร์ฟเวอร์ (Server) ได้ด้วย

เนื่องด้วยส่วน front-end ของแอปพลิเคชันเอสโอเอสเอ็นี่จะทำงานโดยการระบุที่อยู่ (URL) ของเอพีไอ (API) ที่ชี้ไปยัง back-end ส่วนต่าง ๆ ดังนั้น ภาษาจาวาสคริปต์จึงเป็นหนึ่งในอีกภาษาที่มีความสำคัญสำหรับการพัฒนางานวิจัยชิ้นนี้ โดยเฟรมเวิร์คจาวาสคริปต์ที่ใช้ในการพัฒนาจะเป็นเฟรมเวิร์คเจควรี่ (jQuery) เป็นส่วนมาก ซึ่งจะกล่าวถึงในโอกาสต่อไป

2.2.4 ภาษาซีชาร์ป (C#)

ภาษาซีชาร์ป เป็นภาษาโปรแกรมหลายโมเดล (multi-paradigm programming language) มีรูปแบบการเขียนที่หลากหลาย เช่น การเขียนโปรแกรมเชิงคำสั่ง (imperative) การเขียนโปรแกรมเชิงประกาศ (declarative) การเขียนโปรแกรมเชิงฟังก์ชัน (functional) การเขียนโปรแกรมเชิงวัตถุ (object-oriented) และการเขียนโปรแกรมเชิงส่วนประกอบ (component-oriented) เป็นต้น ภาษาซีชาร์ปถูกพัฒนาโดยบริษัท ไมโครซอฟต์ เพื่อใช้งานบน ไมโครซอฟต์ดอตเน็ตเฟรมเวิร์ค (Microsoft .NET framework) โดยมีรากฐานมาจาก ภาษาซีพลัสพลัส (C++)

2.2.5 ภาษาเอสคิวแอล (SQL)

ภาษาควีรีเชิงโครงสร้าง (Structured Query Language) หรือเอสคิวแอล (SQL) หรือบางท่านอาจออกเสียงว่า ซีควอล หรือ ซีควล เป็นภาษาที่ใช้เขียนโปรแกรมเพื่อจัดการกับฐานข้อมูลโดยเฉพาะ ซึ่งจะมีโครงสร้างแตกต่างกับภาษาอื่น ๆ ที่กล่าวมาข้างต้นค่อนข้างชัดเจน และยังสามารถทำงานได้กับฐานข้อมูลทุกชนิดอีกด้วย จึงทำให้เอสคิวแอลเป็นภาษาควีรีที่เป็นที่นิยมมากที่สุดในการจัดการกับฐานข้อมูลทั้งหลาย โดยจะสามารถแบ่งการทำงานของเอสคิวแอลได้ 4 ประเภท ดังนี้

2.2.5.1 Select query ใช้สำหรับเลือกข้อมูลที่ต้องการแสดงผล

2.2.5.2 Update query ใช้สำหรับแก้ไขข้อมูล

2.2.5.3 Insert query ใช้สำหรับการเพิ่มข้อมูล

2.2.5.4 Delete query ใช้สำหรับการลบข้อมูล

2.3 ทฤษฎีกระบวนการออกแบบ (Design process)

การออกแบบ (Design) เป็นหนึ่งในขั้นตอนที่สำคัญของการสร้างสรรค์สิ่งใหม่ ๆ ไม่ว่าจะเป็นข้าวของเครื่องใช้ในชีวิตรจริง อุปกรณ์อันทันสมัย หรือโปรแกรมที่แสนซับซ้อนต่าง ๆ ล้วนผ่านขั้นตอนการออกแบบมาก่อนทั้งสิ้น การออกแบบที่ดีจะสามารถช่วยลดขั้นตอนการทำงานขั้นต่อไป คือ ขั้นพัฒนาหรือลงมือทำได้อย่างมหาศาล ส่งผลให้สามารถลดค่าใช้จ่ายและเวลาในขั้นตอนดังกล่าวไปได้ นอกจากนี้ การออกแบบยังอาจช่วยเปิดประสบการณ์ใหม่ ๆ หรือมุมมองใหม่ ๆ ให้แก่ทั้งผู้ใช้งานและผู้พัฒนาอีกด้วย

2.3.1 การออกแบบโดยคำนึงถึงผู้ใช้เป็นศูนย์กลาง (User-Centered Design)



ภาพที่ 2.1 ภาพรวมของยูซีดี

การออกแบบโดยคำนึงถึงผู้ใช้เป็นศูนย์กลาง หรือยูซีดี (UCD) เป็นแนวคิดเชิงออกแบบที่คำนึงถึงผู้ใช้เป็นหลัก มีกระบวนการหลัก ๆ ทั้งหมด 4 ขั้นตอน ซึ่งการนำแนวคิดยูซีดีมาใช้ในการออกแบบงานนั้นไม่จำเป็นจะต้องเรียงลำดับตายตัว แต่ละขั้นตอนจะสามารถทำสลับไปมาได้ อย่างอิสระ โดยแนวคิดยูซีดีนั้นมีขั้นตอนต่าง ๆ ดังต่อไปนี้

2.3.1.1 ขั้นตอนการเรียนรู้ (Learn)

ในขั้นตอนแรกนี้จะเป็นขั้นตอนการรวบรวมข้อมูลต่าง ๆ ที่จำเป็น รวมไปถึงการทำความเข้าใจในสิ่งต่าง ๆ ไม่ว่าจะเป็นมุมมองของผู้ใช้ เป้าหมายของธุรกิจ ซึ่งสิ่งเหล่านี้สามารถนำไปสู่การออกแบบที่สามารถแก้ปัญหาได้ตรงจุดได้ จากนั้นจะนำข้อมูลที่ได้ไปทำการตีความเพื่อสร้างวิธีแก้ปัญหาที่ดีที่สุดต่อไป

2.3.1.2 ขั้นตอนการสำรวจ (Explore)

ขั้นต่อมาคือกระบวนการสำรวจโดยการระดมความคิดจินตนาการในรูปแบบต่าง ๆ เพื่อค้นหาทางออกที่ดีที่สุด ซึ่งอาจเป็นไปได้ทั้งการคิดขึ้นมาเอง คิดต่อจากความคิดของผู้อื่น หรือการคิดนอกกรอบแบบสุดโต่งก็ได้ เมื่อระดมจินตนาการจนออกมาเป็นแนวคิด (concept) แล้ว จึงทำการไตร่ตรองถึงความเป็นไปได้ของแนวคิดแต่ละรูปแบบ โดยอาจเป็นการรับฟังข้อเสนอแนะต่าง ๆ เพื่อค้นหาความคิดที่ดีที่สุดต่อไป

2.3.1.3 ขั้นตอนการออกแบบและสร้าง (Design & Build)

ในขั้นนี้จะเป็นการออกแบบโดยใช้ข้อมูลที่รวบรวมมาได้ทั้งหมดเป็นแหล่งอ้างอิง ขั้นตอนนี้เป็นขั้นตอนที่ต้องพึงระวังมากที่สุด เนื่องจากการออกแบบที่ผิดพลาด แม้เพียงนิดเดียวก็อาจนำไปสู่ผลลัพธ์ที่ไม่ได้ที่ดีที่สุดได้ ดังนั้น ยิ่งใช้เวลาในส่วนนี้มากเท่าใด จะยิ่งทำให้ลดเวลาในขั้นตอนต่อไปได้มากเท่านั้น เมื่อทำการออกแบบเสร็จแล้ว จึงเข้าสู่กระบวนการสร้างโดยใช้แนวคิดที่ออกแบบมา ในการสร้างนั้นควรที่จะทำการทดสอบให้บ่อยเข้าไว้ และคอยรับฟังข้อเสนอแนะต่าง ๆ ที่ได้รับมาในระหว่างการสร้าง

2.3.1.4 ขั้นตอนการพัฒนา (Improve)

ขั้นตอนสุดท้ายของยูซีดีคือขั้นตอนการพัฒนา เมื่อทำการสร้างเสร็จแล้วแน่นอนว่าไม่มีงานชิ้นใดในโลกที่สมบูรณ์แบบ 100% ปัญหาบางอย่างนั้นจะค้นพบทางแก้ที่แท้จริงหลังจากการปล่อยวิธีแก้ขั้นแรกไปแล้วเท่านั้น ซึ่งในส่วนการพัฒนาจำเป็นที่จะต้องคอยติดต่อกับผู้ใช้อยู่ตลอดเวลา เพื่อรับฟังข้อเสนอแนะ คำแนะนำต่าง ๆ ที่จะประโยชน์ต่อการพัฒนาวิธีการแก้ปัญหาที่แท้จริงได้

2.3.2 แนวคิดการพัฒนาแบบอไจล์ (Agile concept)

แนวคิดแบบอไจล์ คือแนวคิดการพัฒนาซอฟต์แวร์ (Software Methodology) ที่มีลักษณะการพัฒนาแบบทำซ้ำ (iteration) คือ แบ่งส่วนย่อย ๆ แล้วค่อย ๆ พัฒนาต่อมาเรื่อย ๆ ในแต่ละกระบวนการนั้นจะต้องเกิดซอฟต์แวร์ที่ใช้งานได้จริงในแต่ละช่วงของการพัฒนา แนวคิดแบบอไจล์นั้นได้เกิดขึ้นมาเพราะการพัฒนาซอฟต์แวร์ในรูปแบบเดิมนั้นมีข้อเสียมากมาย โดยเฉพาะเรื่องความพึงพอใจต่อลูกค้า เวลาในการพัฒนา และงบประมาณ ดังนั้น แนวคิดแบบ

อโง์จ้งเป็นแนวคืดที่ถูกรังขึ้นโดยมีจุดประสงค์หลักคือเพื่อตอบสนองต่อการเปลี่ยนแปลงความต้องการของลูกค้ำ และช่วยในเรื่องการทำงานร่วมกับลูกค้ำ

การพัฒนาซอฟต์แวร์โดยใช้แนวคืดอโง์จ้ จะสามารถช่วยลดความเสี่ยงที่โครงการจะล้มเหลว (project failure) ได้อย่างมหาศาล เนื่องจากลูกค้ำหรือผู้ใช้งานจะสามารถมองเห็นผลิตภัณฑ์ (product) ได้เป็นระยะ ๆ ทำให้สามารถปรับเปลี่ยนแก้ไขผลิตภัณฑ์ได้ตามความต้องการจนกว่าผลิตภัณฑ์สุดท้าย (final product) จะเสร็จสิ้น ไม่เหมือนกับแนวคืดวอเตอร์พอล (waterfall) ซึ่งเป็นแนวคืดการพัฒนาซอฟต์แวร์แบบเก่า ที่ลูกค้ำหรือผู้ใช้งานจะสามารถมองเห็นได้เพียงแค่ผลิตภัณฑ์ที่สำเร็จแล้วเท่านั้น หากออกมาไม่ตรงตามความต้องการของลูกค้ำก็จำเป็นต้องรื้อและพัฒนาใหม่ตั้งแต่ต้น

2.3.3 ทฤษฎีการออกแบบประสบการณ์การใช้งานของผู้ใช้ (User Experience) และส่วนต่อประสานกับผู้ใช้ (User Interface) ด้วยเมทที่เรียลตี้ไซน์ (Material Design)



ภาพที่ 2.2 ตัวอย่างเมทที่เรียลตี้ไซน์

เมทที่เรียลตี้ไซน์ เป็นภาษาการออกแบบ (Design Language) ที่ถูกคิดค้นและพัฒนาโดยบริษัท Google เปิดตัวครั้งแรกในงาน Google I/O ประจำปีค.ศ. 2014 มีแนวคืดมาจากกระดาษที่ใช้กันในชีวิตประจำวัน โดยจะมุ่งเน้นไปที่การออกแบบที่เรียบง่าย แต่ละส่วนมีความหมายในตัว สามารถแยกองค์ประกอบได้อย่างชัดเจน การเคลื่อนไหวที่ลื่นไหล และมุมมองที่มีมิติ

เดิมแล้วแมททีเรียลดีไซน์นั้นเป็นแนวทางการออกแบบที่เน้นไปยังการพัฒนาแอปพลิเคชันในระบบปฏิบัติการแอนดรอยด์ (Android Operating System) เป็นหลัก แต่ด้วยแนวทางที่ดูเรียบง่ายและสวยงาม จึงมีผู้พัฒนาหลายท่านนำแนวคิดนี้มาประยุกต์ใช้ในการพัฒนาเว็บไซต์มากมายในปัจจุบัน

2.4 ทฤษฎีที่เกี่ยวข้องในด้านซอฟต์แวร์คอมพิวเตอร์

2.4.1 โปรแกรมไมโครซอฟต์วิซวลสตูดิโอ (Microsoft Visual Studio)



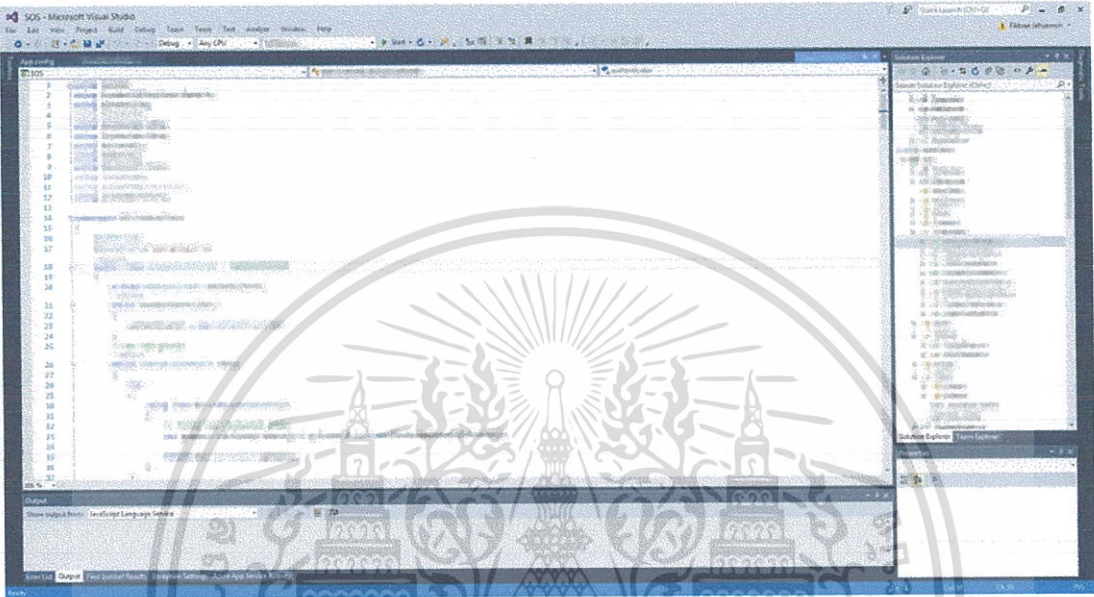
ภาพที่ 2.3 สัญลักษณ์ของโปรแกรมไมโครซอฟต์วิซวลสตูดิโอ

โปรแกรมไมโครซอฟต์วิซวลสตูดิโอ เป็นโปรแกรมที่ใช้สำหรับสร้าง หรือพัฒนาโปรแกรมเว็บไซต์ เว็บแอปพลิเคชัน เว็บเซอร์วิส เพื่อใช้งานบนระบบปฏิบัติการวินโดวส์ (Windows) สมาร์ทโฟน (Smartphone) และเว็บเบราว์เซอร์ (Web browser) ซึ่งจะรองรับภาษาโปรแกรมได้หลากหลายมาก อาทิเช่น ภาษาที่ถูกแนบมากับตัวโปรแกรมอยู่แล้ว คือ C, C#, C++, VB.NET นอกจากนี้ยังสนับสนุนภาษาอื่นโดยการติดตั้งส่วนเสริมอีกมาก เช่น Python, Ruby, Node.js ฯลฯ และยังสามารถสนับสนุนภาษา XML, HTML, JavaScript, CSS อีกด้วย ในอดีตวิซวลสตูดิโอเคยสนับสนุนภาษาจาวา (Java) ด้วย แต่ในเวอร์ชันปัจจุบันไม่มีการสนับสนุนแล้ว

คุณลักษณะของไมโครซอฟต์วิซวลสตูดิโอนั้นเหมาะสำหรับนักพัฒนาที่ค่อนข้างมีความชำนาญในการเขียนโปรแกรมอยู่แล้ว เนื่องจากตัวโปรแกรมมีเครื่องมือ (Tools) ให้เลือกใช้มากมายมหาศาล รวมไปถึงส่วนขยาย (Add-ins) ที่มีให้เลือกใช้มากมายเช่นกัน ถึงแม้ว่าจะช่วยให้การพัฒนาโปรแกรมเป็นไปได้ง่ายขึ้น แต่ด้วยจำนวนที่ค่อนข้างเยอะจึงอาจนักพัฒนามือใหม่อาจเกิดการงุนงงเมื่อใช้งานครั้งแรกได้ ดังนั้นถึงแม้จะไม่ได้มีความชำนาญในการเขียนโปรแกรมมาก แต่นักพัฒนาที่ต้องการลองใช้งานโปรแกรมไมโครซอฟต์วิซวลสตูดิโอก็ควรที่จะมีความรู้

พื้นฐานในการเขียนโปรแกรมมาก่อนบ้าง โดยวิซวลสตูดิโอนั้นมีคุณลักษณะ (features) ต่าง ๆ ดังนี้

2.4.1.1 ส่วนแก้ไขโค้ด (Code editor)



ภาพที่ 2.4 ส่วนแก้ไขโค้ดของวิซวลสตูดิโอ

ส่วนแก้ไขโค้ดนั้นถือเป็นคุณลักษณะหลักของวิซวลสตูดิโอ ซึ่งจะสนับสนุนการเติมสีโครงสร้างภาษาสำหรับภาษาที่สนับสนุนโดยอัตโนมัติ (Syntax highlighting) การเพิ่มเติมโค้ดคำสั่งให้โดยอัตโนมัติด้วยอินเทลลิเซนส์ (Code completion using IntelliSense) สำหรับตัวแปร ฟังก์ชัน เมธอด ลูป นอกจากนี้ยังสนับสนุนการสร้างจุดค้น (bookmark) สำหรับการนำทางที่รวดเร็ว และการย่อโค้ด (collapsing) เมื่อฟังก์ชันมีขนาดใหญ่อีกด้วย

2.4.1.2 ส่วนตรวจสอบจุดบกพร่อง (Debugger)

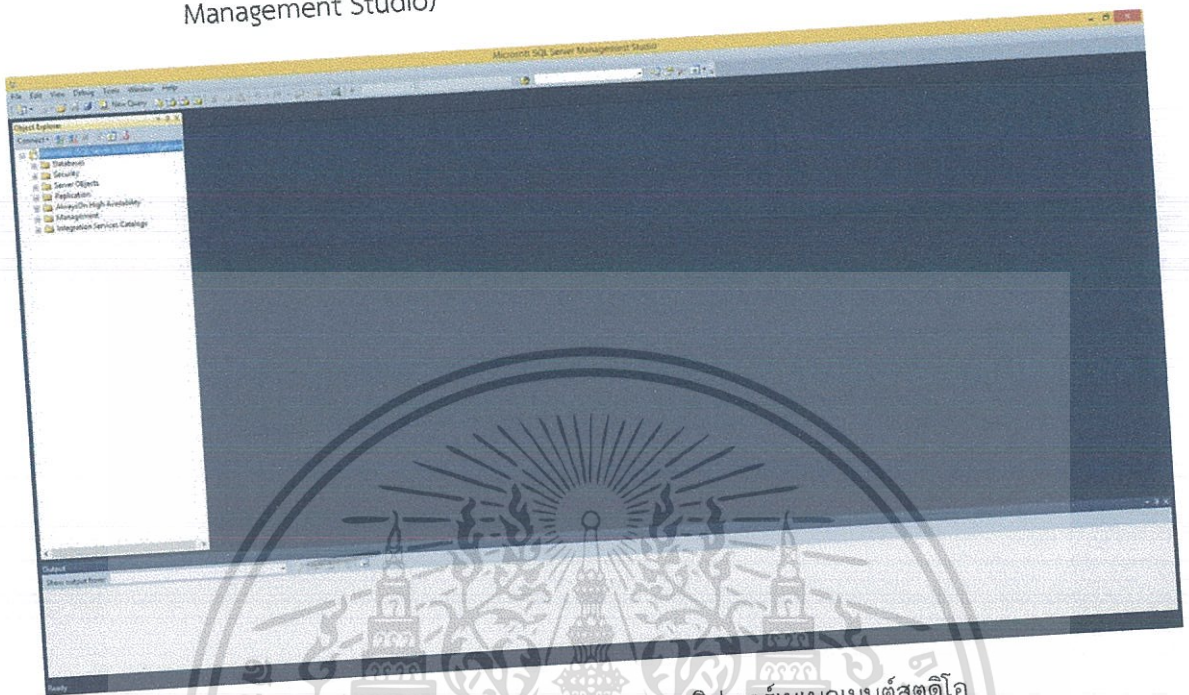
ส่วนตรวจสอบจุดบกพร่องของวิซวลสตูดิโอนั้นสนับสนุนการทำงานทั้งในระดับการตรวจสอบจุดบกพร่องทั่วไป (Source-level debugger) ซึ่งใช้ในการหาจุดบกพร่อง

ของโค้ดภาษาคอมไพเลอร์ทั่วไป และการตรวจสอบจุดบกพร่องในระดับเครื่อง (machine-level debugger) ซึ่งจะใช้ในการหาจุดบกพร่องของภาษาคอมไพเลอร์ในระดับภาษาเครื่อง และยังสามารถตั้งจุดหยุดอัตโนมัติเมื่อโปรแกรมทำงานถึงจุดที่ตั้งไว้ (breakpoint) และจุดสังเกตการณ์ (watch) ซึ่งจะคอยจับตามูลค่าของตัวแปรในขณะที่โปรแกรมกำลังทำงานได้ด้วย

2.4.1.3 Team Foundation Server

ทีมฟาวนด์ชันเซิร์ฟเวอร์หรือทีเอฟเอส คือบริการเสริมของไมโครซอฟต์ที่ช่วยในการจัดการโค้ดต้นฉบับ (Source code management) การควบคุมเวอร์ชัน (Version control) การจัดการโปรเจกต์ (Project management) ซึ่งรองรับทั้งแบบอโอฟไลต์ และแบบวอเทอร์ฟอล การทดสอบระบบ การควบคุมการปล่อยโค้ด (Release management) และอื่น ๆ อีกมากมาย ซึ่งทีเอฟเอสจะช่วยอำนวยความสะดวกในการพัฒนาโปรแกรมที่ให้นักพัฒนาหลายคนได้เป็นอย่างดี เนื่องจากมีการทำงานอยู่บนคลาวด์ (Cloud) และมีการควบคุมเวอร์ชันที่ชัดเจน

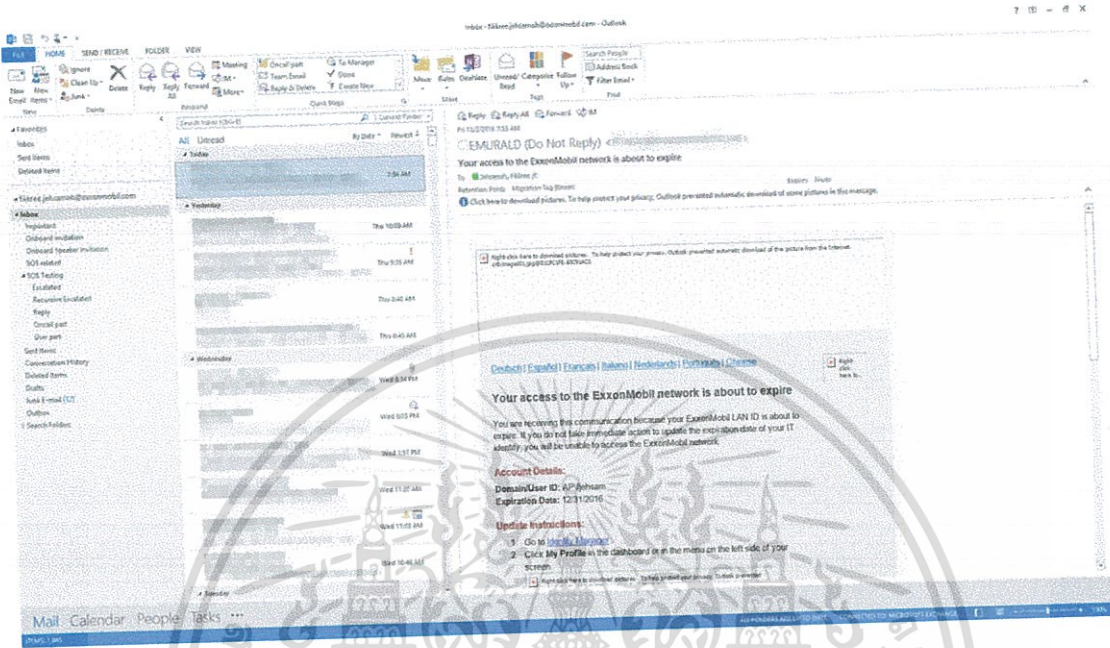
2.4.2 โปรแกรมไมโครซอฟต์เอสคิวแอลเซิร์ฟเวอร์เมนเนจเมนต์สตูดิโอ (Microsoft SQL Server Management Studio)



ภาพที่ 2.5 โปรแกรมไมโครซอฟต์เอสคิวแอลเซิร์ฟเวอร์เมนเนจเมนต์สตูดิโอ

โปรแกรมไมโครซอฟต์เอสคิวแอลเซิร์ฟเวอร์เมนเนจเมนต์สตูดิโอ เป็นโปรแกรมจัดการและควบคุมฐานข้อมูล (Database) รวมไปถึงองค์ประกอบต่าง ๆ ที่อยู่ในไมโครซอฟต์เอสคิวแอลเซิร์ฟเวอร์ (Microsoft SQL Server) ทั้งหมด ตัวโปรแกรมสนับสนุนการใช้งานทั้งส่วนแก้ไขสคริปต์ (Script editor) และเครื่องมือต่าง ๆ ที่อำนวยความสะดวกในการจัดการทั้งเซิร์ฟเวอร์และฐานข้อมูล

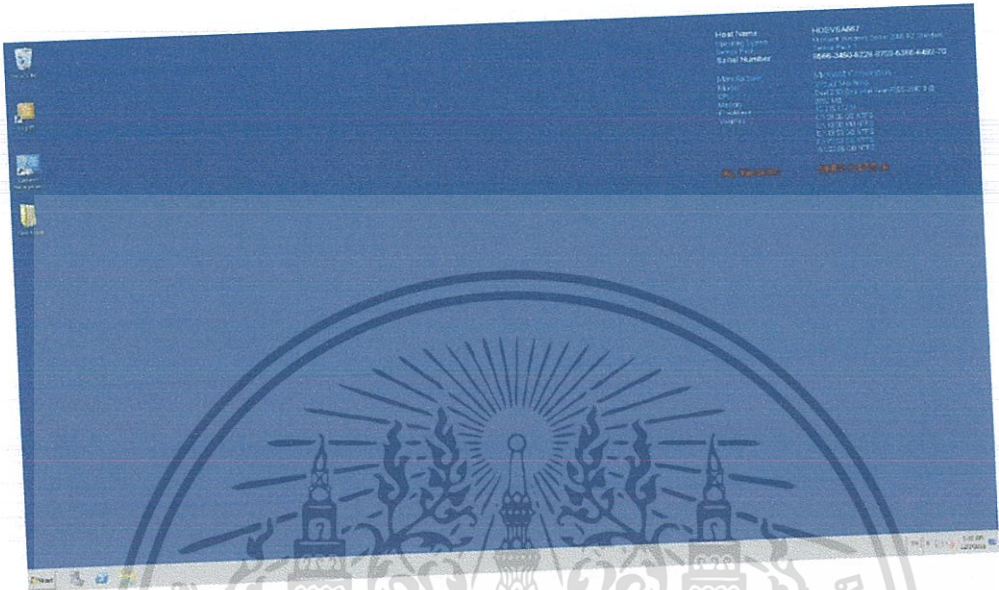
2.4.3 โปรแกรมไมโครซอฟต์เอ้าท์ลุค (Microsoft Outlook)



ภาพที่ 2.6 โปรแกรมไมโครซอฟต์เอ้าท์ลุค

โปรแกรมไมโครซอฟต์เอ้าท์ลุค เป็นอีเมลล์ไคลเอนท์ (Email client) ที่ถูกพัฒนาโดยบริษัทไมโครซอฟต์ จัดเป็นโปรแกรมหนึ่งในกลุ่มโปรแกรมไมโครซอฟต์ออฟฟิศ (Microsoft Office) ปัจจุบันเป็นโปรแกรมจัดการอีเมลล์ที่ได้รับความนิยมที่สุดในระบบสำนักงานทั่วโลก

2.4.4 สภาพแวดล้อมการพัฒนาสำหรับนักพัฒนาทั่วไป (Common Developer Environment)

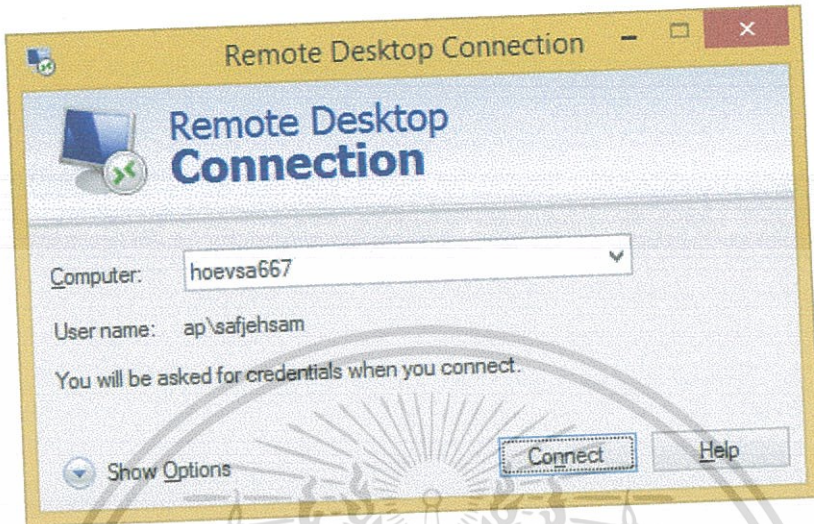


ภาพที่ 2.7 เซิร์ฟเวอร์สภาพแวดล้อมสำหรับนักพัฒนา

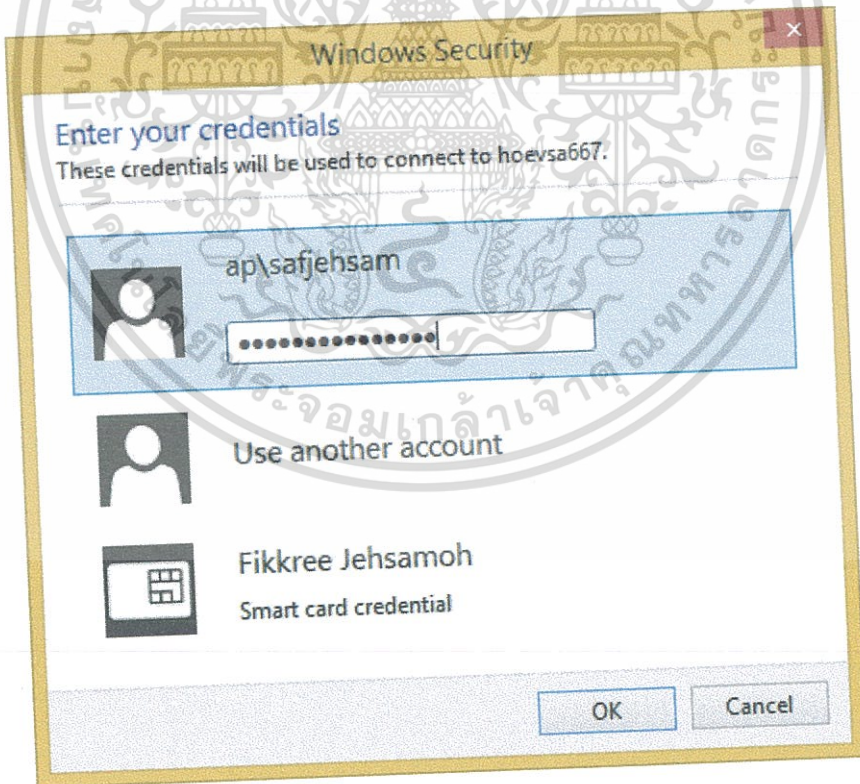
สภาพแวดล้อมสำหรับนักพัฒนาทั่วไป หรือเรียกโดยย่อว่า ซีดีอี (CDE) เป็นเซิร์ฟเวอร์จำลองที่ทำงานบนระบบปฏิบัติการไมโครซอฟต์วินโดวส์เซิร์ฟเวอร์ (Microsoft Windows Server) โดยซีดีอีนั้นคือเซิร์ฟเวอร์พิเศษที่มีเฉพาะพนักงานที่มีตำแหน่งเป็นนักพัฒนา (developer) ของบริษัท เอ็กซอนโมบิล จำกัดเท่านั้นที่สามารถใช้งานได้ โดยหน้าที่หลักของซีดีอีคือ ทำหน้าที่เป็นเซิร์ฟเวอร์จำลองให้ทดสอบความสามารถของโปรแกรมที่เขียนไป ว่าสามารถทำงานได้อย่างถูกต้องตามความต้องการหรือไม่ เมื่อผลการทดสอบเป็นที่น่าพอใจแล้ว จึงจะนำโค้ดดังกล่าวไปวางไว้บนเซิร์ฟเวอร์ผลิตภัณฑ์ (Production server) เพื่อให้ลูกค้าใช้งานต่อไป

เนื่องจากเซิร์ฟเวอร์ทั้งหมดของซีดีอีถูกตั้งอยู่ที่ประเทศสหรัฐอเมริกา ดังนั้น การใช้งานซีดีอีนั้นจึงทำการเข้าใช้งานผ่านโปรแกรมอีกตัวหนึ่งที่เรียกว่า รีโมทเดสก์ท็อปปคอนเนคชัน (Remote Desktop Connection) หรือเรียกอีกชื่อหนึ่งว่า อาร์ดีพี (RDP) โดยจะต้องทำการใส่ไอดีที่ทำการอนุญาตแล้วก่อน ซึ่งจะเรียกว่า ไอดีสิทธิพิเศษ (Privilege ID) จึงสามารถเข้าสู่เซิร์ฟ

เวอร์ซีดีอีได้ ซึ่งโดยปกติตามนโยบายของบริษัทแล้ว ไอดีที่สามารถเข้าสู่เซิร์ฟเวอร์ซีดีอีได้จะต้องเป็นไอดีสิทธิพิเศษเท่านั้น ไอดีปกติจะไม่สามารถใช้งานเซิร์ฟเวอร์ซีดีอีได้เลย

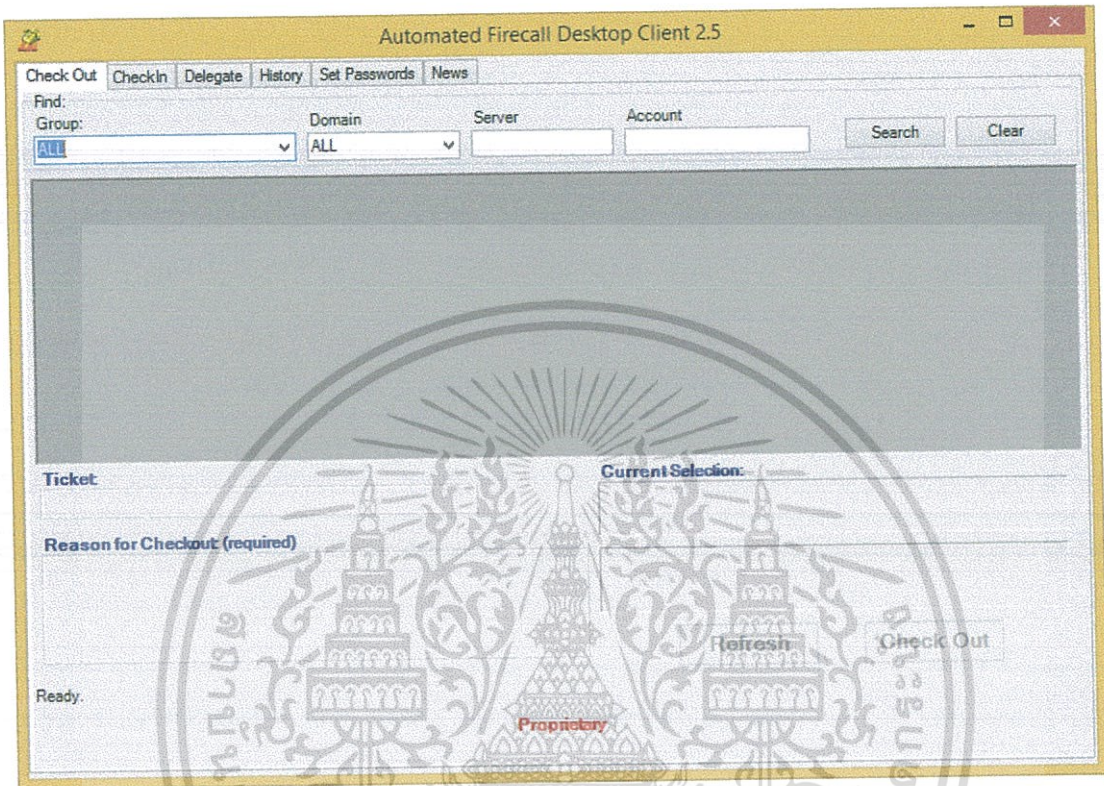


ภาพที่ 2.8 การเชื่อมต่อโปรแกรมรีโมทเดสก์ทอปคอนเน็คชันด้วยไอดีพิเศษ



ภาพที่ 2.9 การใส่รหัสผ่านเพื่อเข้าสู่เซิร์ฟเวอร์ซีดีอี

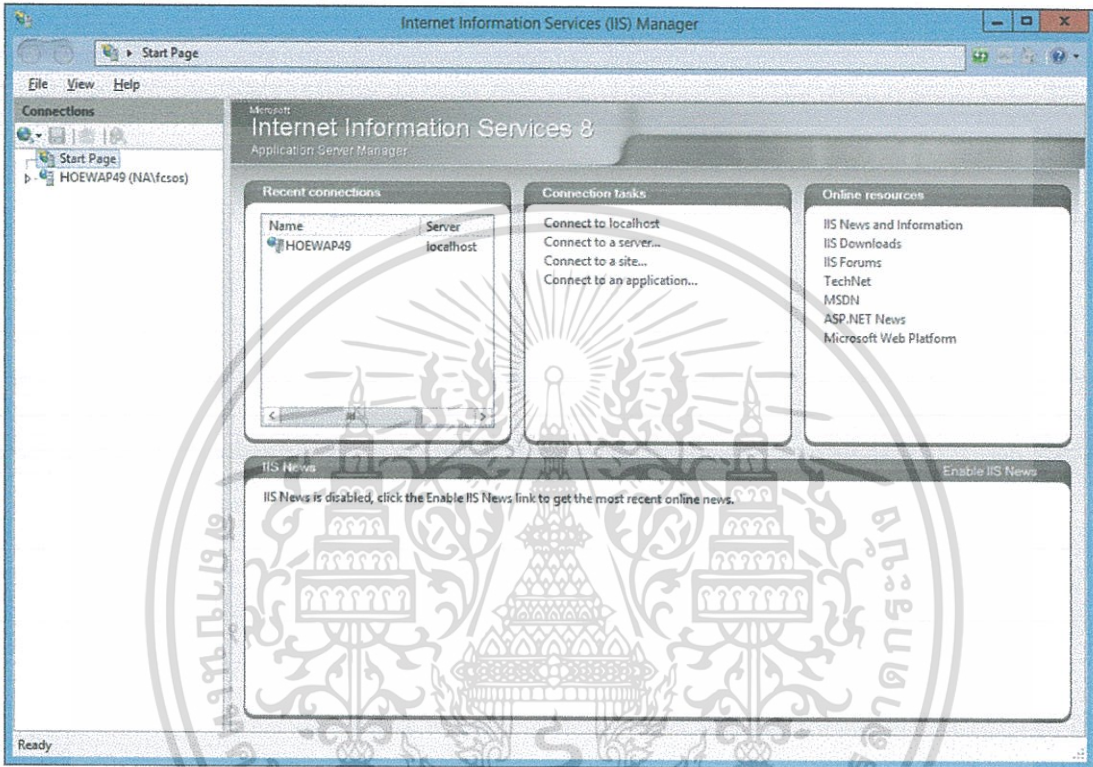
2.3.5 โปรแกรมอัตโนมัติเช็คเอาท์ไฟรคอลลิสต์ที่อปโคลเอนท์ (Automated Firecall Desktop Client)



ภาพที่ 2.10 โปรแกรมอัตโนมัติเช็คเอาท์ไฟรคอลลิสต์ที่อปโคลเอนท์

นอกจากไอดีพิเศษที่ใช้ในการเข้าสู่เซิร์ฟเวอร์ซีดีไอแล้ว บริษัทเอ็กซ์คอนโมบิลยังมีไอดีอีกประเภทหนึ่งซึ่งจะใช้กับเซิร์ฟเวอร์ที่มีการเก็บรักษาสิ่งที่เป็นความลับเท่านั้น เช่น เซิร์ฟเวอร์ผลิตภัณฑ์ที่เป็นที่อยู่ของแอปพลิเคชันที่มีการใช้งานตลอดเวลา โดยจะเรียกไอดีประเภทนี้ว่า ไฟรคอลลิสต์ (Firecall ID) โดยปกติแล้ว การอาร์ดีทีเข้าสู่เซิร์ฟเวอร์ต่าง ๆ นั้นผู้ใช้จะเป็นคนกรอกไอดีและรหัสผ่านด้วยตัวเอง แต่สำหรับไอดีที่เป็นไฟรคอลลิสต์นั้น จำเป็นต้องทำการเช็คเอาท์ (Check out) ไอดีและบอกเหตุผลที่ต้องเช็คเอาท์เสียก่อน จึงจะสามารถนำไอดีไปใช้งานได้ ซึ่งทุกครั้งที่มีการเช็คเอาท์เกิดขึ้น ระบบจะทำการสุ่มรหัสผ่านให้ โดยรหัสผ่านนี้จะมีอายุการใช้งานตลอดจนกว่าจะทำการเช็คอิน (Check in) ไฟรคอลลิสต์ ซึ่งจำเป็นต้องทำทุกครั้งเมื่อใช้งานไฟรคอลลิสต์เสร็จ เนื่องจากเป็นนโยบายของบริษัทที่ไม่ต้องการให้ใครก็ตามเข้าไปเปลี่ยนแปลงการทำงานของแอปพลิเคชันบ่อย ๆ นั่นเอง

2.3.6 โปรแกรมอินเทอร์เน็ตอินฟอร์เมชันเซอร์วิสเมนเนเจอร์ (Internet Information Services Manager)



ภาพที่ 2.11 โปรแกรมอินเทอร์เน็ตอินฟอร์เมชันเซอร์วิสเมนเนเจอร์

โปรแกรมอินเทอร์เน็ตอินฟอร์เมชันเซอร์วิสเมนเนเจอร์ หรือเรียกโดยย่อว่า ไอไอเอส (IIS) เป็นโปรแกรมที่ทำหน้าที่เป็น Web Server ให้แก่เว็บแอปพลิเคชันต่าง ๆ ซึ่งจะทำให้เซิร์ฟเวอร์จำลองทั้งหลายสามารถให้บริการเว็บไซต์ และสามารถเรียกดูข้อมูลผ่านเว็บเบราว์เซอร์ได้ โดยไอไอเอสนั้นสามารถทำงานและอ่านภาษาโปรแกรมได้หลากหลาย ไม่ว่าจะเป็น ASP.NET, PHP, JHP

2.5 ทฤษฎีที่เกี่ยวข้องในด้านเฟรมเวิร์ค

การพัฒนาเว็บไซต์ในส่วนที่ซับซ้อนมาก ๆ เช่น จาวาสคริปต์นั้นจำเป็นต้องใช้ความรู้ความสามารถในระดับหนึ่ง ยิ่งเว็บไซต์ที่ต้องการมีความซับซ้อนสูง สคริปต์ที่ต้องพัฒนาก็คงต้องยิ่งเขียนยากตามไปด้วย ดังนั้น การเลือกใช้เฟรมเวิร์คในการช่วยพัฒนาเว็บไซต์จึงเป็นอีกหนึ่งทางเลือกของนักพัฒนาที่ต้องการประหยัดเวลาในการพัฒนาเว็บไซต์ สำหรับแอปพลิเคชันเอสโอเอสเอ็นั้นก็มีการใช้เฟรมเวิร์คเพื่อช่วยในการพัฒนาเว็บไซต์จำนวนหนึ่งเช่นกัน ซึ่งสามารถแสดงได้ดังนี้

2.5.1 เจควีรี่ (jQuery)

เจควีรี่ คือไลบรารีจาวาสคริปต์ที่ถูกพัฒนาโดย John Resig ถูกปล่อยให้ดาวน์โหลดครั้งแรกในเดือนมกราคม ปีค.ศ. 2006 โดยมีวัตถุประสงค์หลักคือถูกออกแบบมาเพื่อให้การเขียนจาวาสคริปต์มีความสะดวกและง่ายขึ้น และสามารถทำให้เว็บไซต์ทำงานได้ตรงตามความต้องการในทุก ๆ เบราเซอร์ ในปัจจุบันเจควีรี่ถือเป็นไลบรารีจาวาสคริปต์ที่นิยมใช้งานมากที่สุดในโลก โดยมีอัตราการติดตั้งสูงถึง 65% จากทั้งหมด 10 ล้านเว็บไซต์ที่มีการเยี่ยมชมมากที่สุด

2.5.2 แมททีเรียลไลซ์ (Materialize)

แมททีเรียลไลซ์ เป็นเฟรมเวิร์คที่ถูกพัฒนาขึ้นเพื่อนักพัฒนาเว็บไซต์ที่มีเป้าหมายคือต้องการให้เว็บไซต์ของตนถูกออกแบบโดยอ้างอิงแมททีเรียลดีไซน์เป็นหลัก ภายในเฟรมเวิร์คจะประกอบด้วยส่วนประกอบพื้นฐานของเว็บไซต์ เช่น ปุ่ม (button) สัญลักษณ์ (icon) φόρม (form) ฯลฯ นอกจากนี้ยังมีส่วนของสคริปต์ที่ทำให้เว็บไซต์ใช้งานได้ง่ายตามแนวคิดของแมททีเรียลดีไซน์อีกด้วย ซึ่งเฟรมเวิร์คแมททีเรียลไลซ์นี้จะทำให้เว็บไซต์ใช้งานง่าย สะดวก รวดเร็ว และสนับสนุนการใช้งานบนโทรศัพท์มือถืออย่างเต็มรูปแบบ

2.5.3 ฟูลคาเล็นดาร์ (fullcalendar)

ฟูลคาเล็นดาร์ คือเฟรมเวิร์คจาวาสคริปต์ที่ช่วยในส่วนการจัดการเกี่ยวกับปฏิทิน มีความสามารถในการเพิ่ม-ลบตารางเวลาได้ และยังสามารถตั้งค่าเกี่ยวกับปฏิทินและตารางเวลาได้อย่างอิสระ

2.5.4 เอเอสพีดีทอเน็ต (ASP.NET)

เอเอสพีดีทอเน็ต คือเฟรมเวิร์คเว็บแอปพลิเคชันที่ทำงานบนฝั่งเซิร์ฟเวอร์ ถูกพัฒนาโดยบริษัทไมโครซอฟต์ โดยมีวัตถุประสงค์คือเพื่อให้นักพัฒนาสามารถสร้างเว็บไซต์ที่มีความยืดหยุ่น (Dynamic web page) เช่น เว็บแอปพลิเคชัน เว็บเซอร์วิส ได้อย่างอิสระและง่ายดาย เอเอสพีดีทอเน็ตเวอร์ชันแรกสุด คือเวอร์ชัน 1.0 ถูกปล่อยให้ดาวน์โหลดครั้งแรกในเดือนมกราคม ปี ค.ศ. 2002 ในปัจจุบันนี้จะเป็นเวอร์ชัน 4.6 ซึ่งถือว่าเป็นเวอร์ชันที่มีความเสถียรที่สุด

2.6 ทฤษฎีที่เกี่ยวข้องอื่นๆ

2.6.1 ไมโครซอฟต์เอ็กซ์เชนจ์เซิร์ฟเวอร์ (Microsoft Exchange Server)

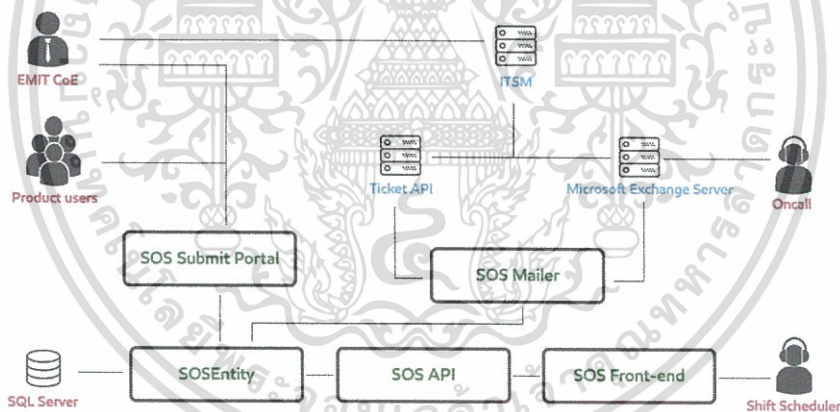
ไมโครซอฟต์เอ็กซ์เชนจ์เซิร์ฟเวอร์ คือเซิร์ฟเวอร์สำหรับจัดการอีเมลล์และปฏิทินที่ถูกพัฒนาโดยบริษัทไมโครซอฟต์ ซึ่งเป็นเซิร์ฟเวอร์หลักที่ใช้ในการรับ-ส่งอีเมลล์ต่าง ๆ ทั่วทั้งเครือข่ายของบริษัท สำหรับบริษัทที่มีขนาดใหญ่ขึ้นนั้นจะอาจจะมีเซิร์ฟเวอร์เป็นของตัวเอง แต่สำหรับบริษัทขนาดเล็กส่วนมาก ไมโครซอฟต์จะเป็นคนเปิดเซิร์ฟเวอร์ให้โดยจะต้องจ่ายเป็นค่าสมาชิกแก่ไมโครซอฟต์

บทที่ 3

วิธีดำเนินการวิจัย

การศึกษาวิจัยในครั้งนี้เป็นการศึกษาวิจัยเชิงพัฒนา มีการศึกษาถึงปัจจัยต่าง ๆ ที่ส่งผลกระทบต่อกระบวนการทำงานของระบบ โดยการพัฒนาเว็บแอปพลิเคชันซึ่งเกิดออนคอลเซอร์วิส (Single On-call Service) หรือเอสโอเอส (SOS) นั้นจะแบ่งออกเป็น 4 ส่วน คือ การพัฒนาในส่วน front-end การพัฒนาในส่วน back-end การสนับสนุนแอปพลิเคชันในส่วนของฐานข้อมูล และการสนับสนุนผู้ใช้เมื่อมีทีมอื่นมาติดต่อขอใช้งานแอปพลิเคชัน โดยความสัมพันธ์ของสิ่งต่าง ๆ ในระบบจะถูกแสดงออกมาในรูปแบบของโครงสร้างทางสถาปัตยกรรมของระบบ (System Architecture) ดังนี้

3.1 โครงสร้างทางสถาปัตยกรรมของระบบ



ภาพที่ 3.1 แสดงโครงสร้างทางสถาปัตยกรรมของระบบเอสโอเอส

จากภาพที่ 3.1 แสดงถึงโครงสร้างทางสถาปัตยกรรมทั้งหมดของระบบเอสโอเอส โครงสร้างดังกล่าวแสดงให้เห็นถึงการเชื่อมต่อในส่วนต่าง ๆ ของระบบ โดยกระบวนการทำงานจะเริ่มต้นเมื่อผู้ใช้ (User) ในที่นี้อาจเป็นไปได้ทั้งผู้ใช้ที่เป็นพนักงานผู้เชี่ยวชาญในแผนกไอทีของบริษัท เอ็กชอนโมบิล จำกัด (EMIT Center of Expertise: EMIT CoE) หรือผู้ใช้ทั่วไปที่ไม่ใช่พนักงานของแผนกไอที ทำการสร้างตั๋วงานที่มีความสำคัญสูง (high priority ticket) ขึ้นมาในระบบ ไม่ว่าจะเป็นการสร้างผ่านระบบไอทีเซอร์วิส

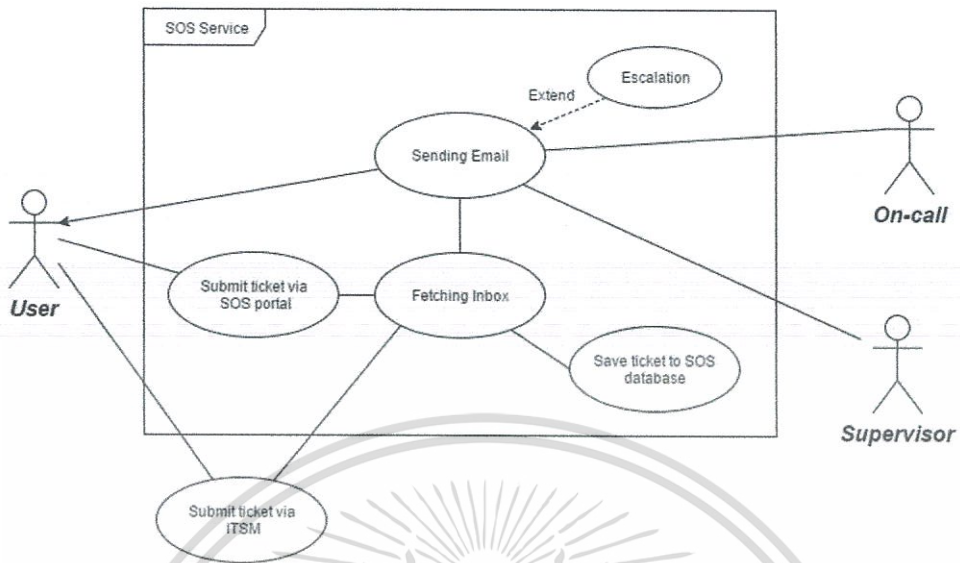
เมเนจเมนต์ (IT Service Management: ITSM) หรือสร้างผ่านเอสโอเอสซัพไมท์พอร์ทัล (SOS Submit Portal) ก็ตาม

เมื่อมีตัวงานถูกสร้างขึ้นมาแล้ว ระบบเอสโอเอสในส่วนควบคุมอีเมลล์ (SOS Mailer) จะทำการตรวจสอบกล่องจดหมายของตัวเองทุก ๆ 5 วินาทีว่ามีอีเมลล์แจ้งเตือนเข้ามาหรือไม่ โดยอีเมลล์นี้จะถูกส่งมาจากระบบไอทีเซอร์วิสเมเนจเมนต์ ถ้าหากพบว่ามี ระบบในส่วนควบคุมฐานข้อมูล (SOS Entity) ก็จะทำกรตรวจสอบเนื้อหาของตัวงาน แล้วบันทึกตัวงานลงฐานข้อมูลของเอสโอเอส จากนั้นจึงทำการสร้างอีเมลล์เอสโอเอสส่งไปแจ้งเตือนพนักงานออนคอลในเวลานั้นต่อไป

สำหรับการกำหนดเวลาทำงานของพนักงานออนคอล จะถูกกระทำโดยพนักงานจัดตารางเวลา (Shift Scheduler) ประจำแต่ละทีม โดยพนักงานจัดตารางเวลาจะทำการจัดเวรให้พนักงานออนคอลผ่าน front-end ของระบบเอสโอเอส

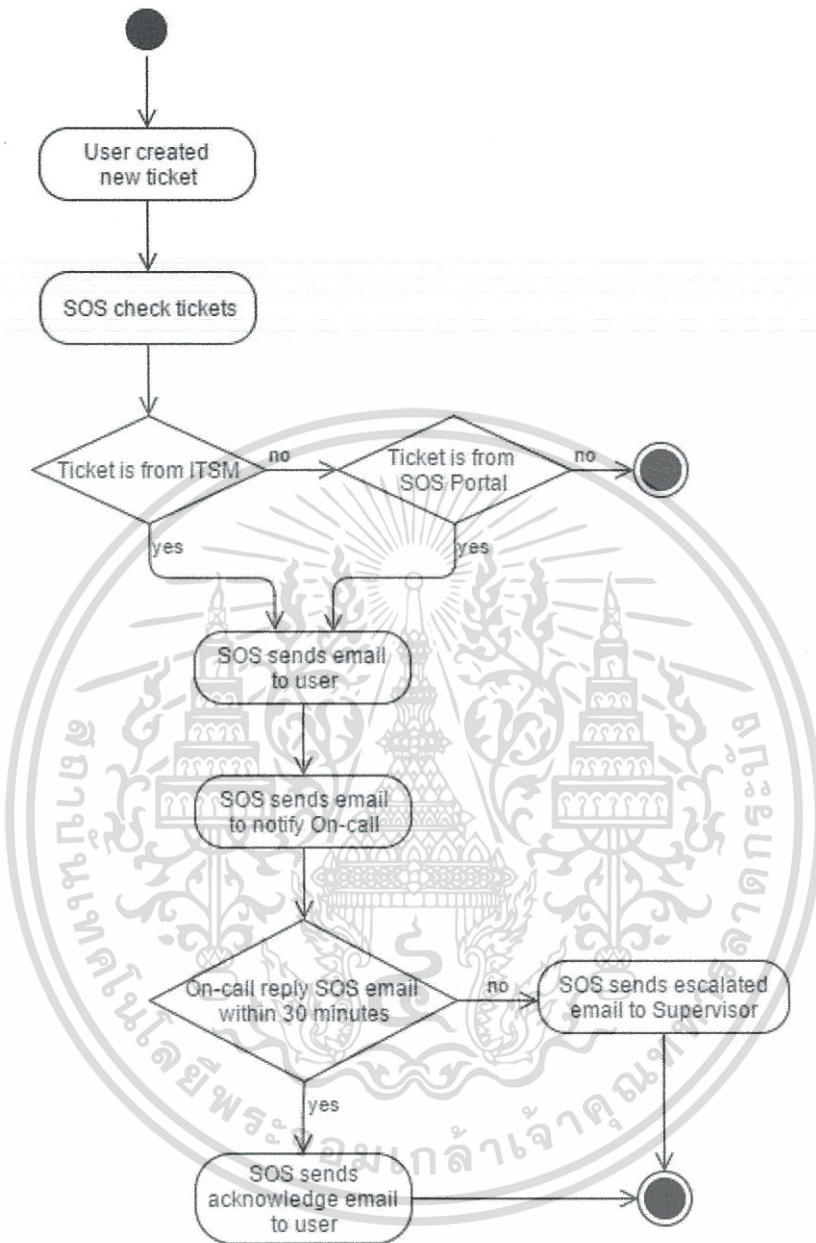
3.2 กระบวนการทำงานของระบบ

กระบวนการทำงานของงานวิจัยชิ้นนี้ถูกแบ่งออกเป็น 4 ขอบเขตหลัก ซึ่งแต่ละขอบเขตนั้นก็จะมีลักษณะที่แตกต่างกันออกไปตามหน้าที่การทำงาน แต่ทุกขอบเขตจะมีการทำงานร่วมกันทั้งสิ้น โดยมีส่วน back-end เป็นส่วนที่เปรียบเสมือนเป็นใจกลางของระบบนี้ ทุก ๆ ส่วนในส่วนของ back-end ทั้งแอปพลิเคชันโปรแกรมมิ่งอินเตอร์เฟส (Application Programming Interface) หรือเอพีไอ (API) ที่ใช้สำหรับการแสดงผลในหน้า front-end ส่วนที่คอยควบคุมการรับ-ส่งอีเมลล์ (SOS Mailer) และส่วนที่ใช้ติดต่อกับฐานข้อมูล (SOS Entity) ล้วนเป็นส่วนที่สำคัญที่สุดที่ทำให้แอปพลิเคชันเอสโอเอสนี้ทำงานได้อย่างราบรื่นตามที่ตั้งใจไว้



ภาพที่ 3.2 แผนภาพยูสเคสของระบบเอสโอเอส

จากภาพที่ 3.2 แสดงแผนภาพยูสเคสของระบบทั้งหมด โดยประกอบด้วย 3 ส่วน คือฝั่งผู้ใช้งานระบบ (User) ฝั่งพนักงานออนคอล และฝั่งหัวหน้างาน (Supervisor) ฝั่งผู้ใช้งานนั้นคือการใช้งานระบบทั่วไป ซึ่งสามารถสร้างตั๋วงานขึ้นมาได้ทั้งจากหน้าเว็บพอร์ทัล และจากระบบไอทีเอสเอ็ม



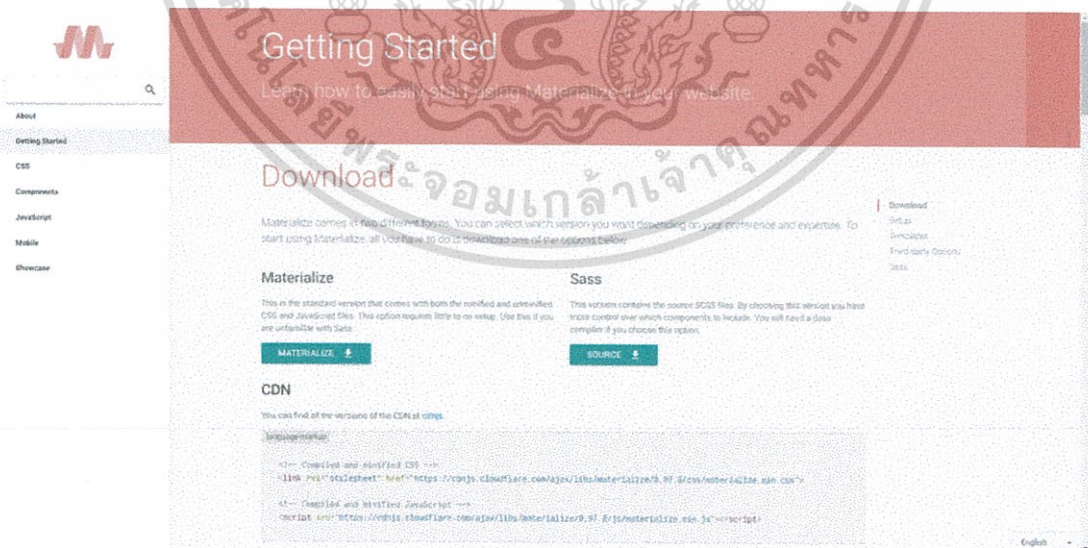
ภาพที่ 3.3 แผนภาพแอกทिवิตีที่แสดงภาพรวมการทำงานของระบบเอสโอเอส

ภาพที่ 3.3 แสดงถึงแผนภาพแอกทिवิตีของระบบเอสโอเอส โดยจะทำการสร้างตั๋วงาน (Submit ticket) ขึ้นมาในระบบ จากนั้นระบบเอสโอเอสจะคอยตรวจกล่องจดหมายทุก ๆ 5 วินาที หากพบอีเมลที่เกี่ยวข้องกับตั๋วงานแล้ว จะทำการเก็บข้อมูลตั๋วงานดังกล่าวลงฐานข้อมูล แล้วส่งอีเมลแจ้งเตือนพนักงานออนคอลทุก ๆ 5 นาที พร้อม ๆ กับแจ้งเตือนผู้ใช้งานระบบได้ทำการแจ้งพนักงานออนคอลให้

ทราบแล้ว และหากว่าเวลาผ่านไป 30 นาทีโดยที่พนักงานออนไลน์ยังไม่ทำการตอบกลับอีเมลดังกล่าว ระบบจะทำการ “ยกระดับ (Escalate)” อีเมลล์โดยการแจ้งเตือนไปที่หัวหน้างานของพนักงานออนไลน์นั้น ๆ แทน โดยขั้นตอนและวิธีการพัฒนาในแต่ละส่วนของชิ้นงานสามารถอธิบายได้ ดังนี้

3.2.1 การพัฒนาในส่วนของ front-end

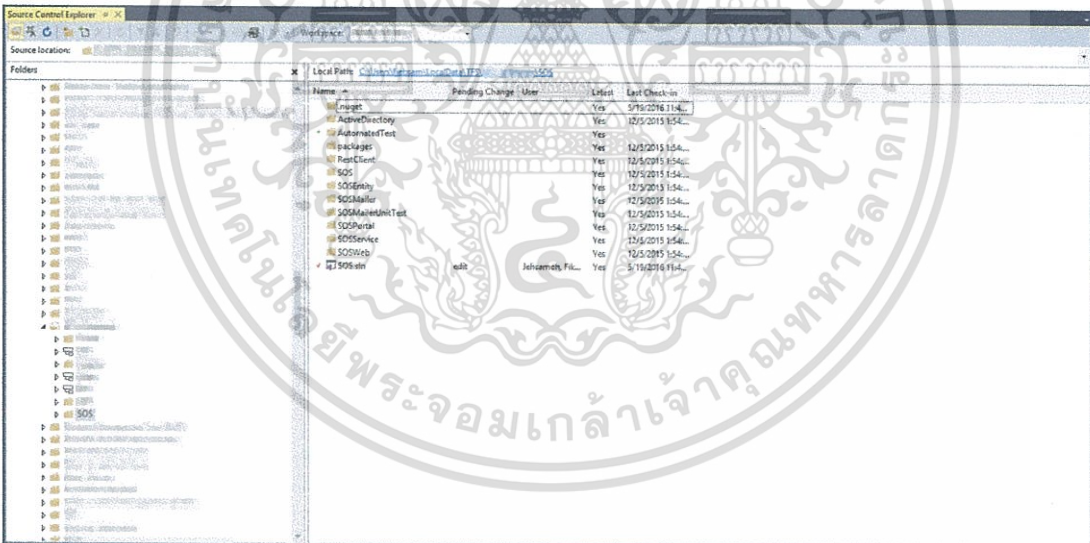
กระบวนการทำงานในส่วนนี้มีปริมาณงานมากที่สุดใน 4 ขอบเขต เนื่องจากจำเป็นต้องทำการออกแบบหน้าเว็บไซต์ใหม่หมดทั้งแอปพลิเคชัน ดังนั้น กระบวนการพัฒนาในส่วนนี้จึงเริ่มจากการศึกษาภาพรวมและส่วนประกอบต่าง ๆ ของแอปพลิเคชันทั้งหมด และทำการวิเคราะห์ว่าการใช้เฟรมเวิร์คใดที่เหมาะสมกับแอปพลิเคชันตัวนี้ ซึ่งหลังจากทดสอบด้วยเฟรมเวิร์คหลากหลายรูปแบบแล้วพบว่า เฟรมเวิร์คแมททีเรียลไลซ์ (Materialize) เป็นเฟรมเวิร์คที่เหมาะสมที่สุดในการนำมาพัฒนาแอปพลิเคชันนี้ เนื่องจากตัวแอปพลิเคชันนั้นต้องการเน้นไปที่ความเร็วและเรียบง่ายในการใช้งานให้มากที่สุด และองค์ประกอบต่าง ๆ ในตัวแอปพลิเคชันนั้นไม่ได้ซับซ้อนมากนัก การใช้เฟรมเวิร์คแมททีเรียลไลซ์ ซึ่งถือเป็นเฟรมเวิร์คขนาดเล็ก เหมาะกับเว็บไซต์ที่ต้องการความเร็ว เรียบง่าย ไม่ซับซ้อน จึงตอบโจทย์ความต้องการในจุดนี้ได้ดีที่สุด ดังแสดงให้เห็นในภาพที่ 3.4



ภาพที่ 3.4 เฟรมเวิร์คแมททีเรียลไลซ์

เมื่อทำการเลือกเฟรมเวิร์คได้แล้ว ขั้นตอนต่อไปคือการพัฒนาส่วน front-end ใหม่ทั้งหมดโดยใช้เฟรมเวิร์คใหม่ ซึ่งในส่วนนี้แม้จะเป็นขั้นตอนที่ไม่ยากนัก แต่เป็นขั้นตอนที่ใช้เวลานาน เนื่องจากเดิมแล้วตัวแอปพลิเคชันนั้นใช้เฟรมเวิร์คบุทสเตรป (Bootstrap) การแปลงเว็บไซต์จากเฟรมเวิร์คบุทสเตรปมาเป็นเฟรมเวิร์คแมทที่เรียลไทม์ แน่นอนว่าจะต้องเกิดข้อผิดพลาดขึ้นบางประการอย่างแน่นอน เนื่องมาจากการใช้คำสงวนหรือคีย์เวิร์ด (Keyword) บางคำที่ไม่เหมือนกันระหว่างสองเฟรมเวิร์ค จึงทำให้จำเป็นต้องพัฒนาบางส่วนของ front-end เช่น ปฏิทินและรายชื่อพนักงานออนคอล ขึ้นมาใหม่ทั้งหมด

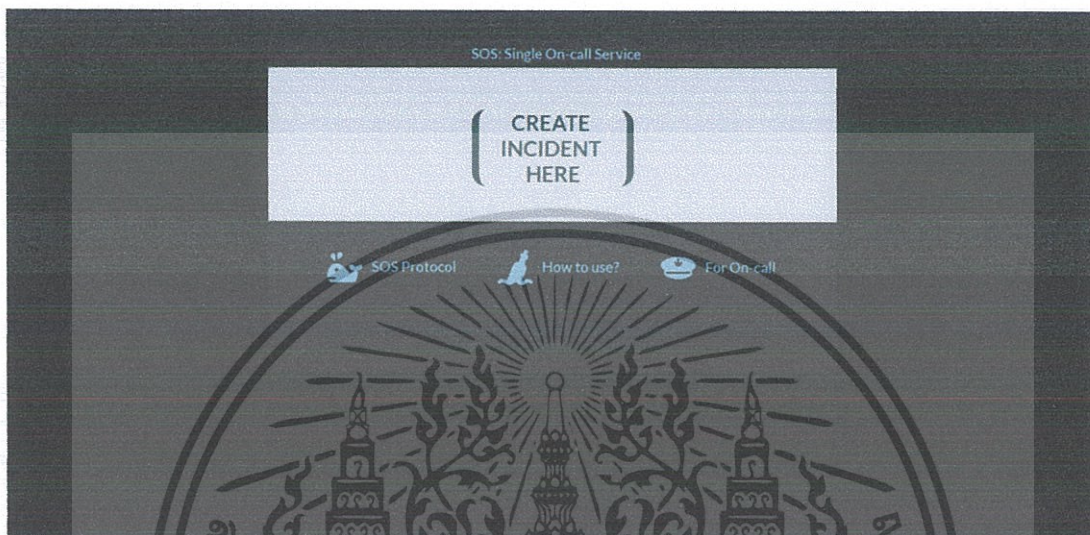
การพัฒนาในส่วน front-end นั้นจะทำทั้งหมด 7 หน้าด้วยกัน โดยจะเริ่มพัฒนาจากการดาวน์โหลดโค้ดต้นฉบับ (Source code) มาจากทีมฟาว์นเดชันเซิร์ฟเวอร์ (Team Foundation Server: TFS) ซึ่งเป็นบริการเสริมของไมโครซอฟต์ที่มีไว้จัดการและจัดเก็บโค้ดต้นฉบับออนไลน์ ภาพที่ 3.5 แสดงให้เห็นถึงเส้นทางที่เก็บไฟล์โค้ดต้นฉบับของเอสโอเอส



ภาพที่ 3.5 การเชื่อมต่อไปยัง TFS เพื่อดาวน์โหลดโปรเจกต์ที่มีโค้ดต้นฉบับของเอสโอเอสทั้งหมด

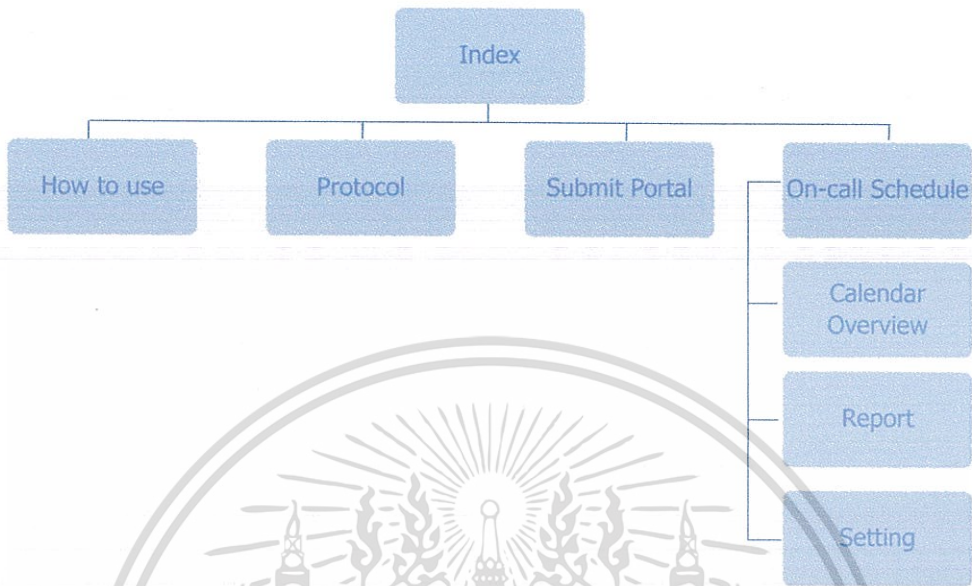
เมื่อดาวน์โหลดโค้ดต้นฉบับมาไว้ในเครื่องของตัวเองแล้ว ขั้นตอนต่อไปคือการพัฒนาเว็บไซต์โดยเริ่มจากหน้าที่มีความสำคัญมากที่สุด คือหน้า On-call Schedule ซึ่งเป็นหน้าที่มีไว้จัดการกะเวลาของพนักงานออนคอลทุกทีม ไล่ไปจนถึงหน้าที่มีความสำคัญน้อยที่สุด คือการ

ออกแบบหน้า Manual และหน้า Protocol ซึ่งเป็นหน้าที่สอนวิธีการใช้งานแอปพลิเคชันเอสโอเอส และหน้าที่บอกถึงโปรโตคอลการทำงานของแอปพลิเคชันเอสโอเอสตามลำดับ



ภาพที่ 3.6 หน้าแรก (Index) ของเว็บไซต์เอสโอเอสแบบเก่า

ภาพที่ 3.6 แสดงให้เห็นถึงหน้าแรก (Index) ของเว็บไซต์เอสโอเอสแบบเก่า ซึ่งสามารถเชื่อมต่อไปยังหน้าต่าง ๆ ของระบบอื่น ๆ ของเอสโอเอสได้ เมื่อผู้ใช้งานต้องการสร้างตัวงานขึ้น ผู้ใช้สามารถคลิกไปที่ปุ่ม “CREATE INCIDENT HERE” ที่ปรากฏอยู่ที่กลางหน้าจอได้ จากนั้นผู้ใช้งานจะถูกนำทางไปยังหน้าซบมิทพอร์ทอล (Submit Portal) ทว่า หากผู้ใช้งานต้องการที่จะทราบว่าพนักงานออนคอลของทีมต่าง ๆ ในเวลานี้คือใคร ผู้ใช้จะไม่สามารถเข้าไปยังหน้าตรวจสอบกะของพนักงานออนคอลได้ทันที แต่จะต้องกลับไปยังหน้าแรกของเว็บไซต์อีกครั้ง จากนั้นจึงทำการคลิกไปที่ปุ่ม “For On-call” ผู้ใช้จะถูกนำทางไปยังหน้าที่ใช้จัดการกะของพนักงานออนคอล จากนั้นผู้ใช้งานจะต้องทำการคลิกที่ปุ่ม “On-call Schedule” บริเวณแถบด้านบนของเว็บไซต์ จึงจะปรากฏปุ่ม “Calendar Overview” ขึ้นมาเป็น drop-down list โดยหน้าดังกล่าวจะเป็นหน้าที่แสดงข้อมูลกะของพนักงานออนคอลของทุกทีมที่มีการจัดตารางเวลาในขณะนั้นได้ ซึ่งขั้นตอนทั้งหมดที่กล่าวมานั้นถือเป็นขั้นตอนที่ค่อนข้างลำบาก และมอบประสบการณ์การใช้งานที่ไม่ดีแก่ผู้ใช้ โดยแผนผังเว็บไซต์ของเอสโอเอสแบบเก่านั้นจะสามารถสรุปได้ดังรูปข้างล่าง

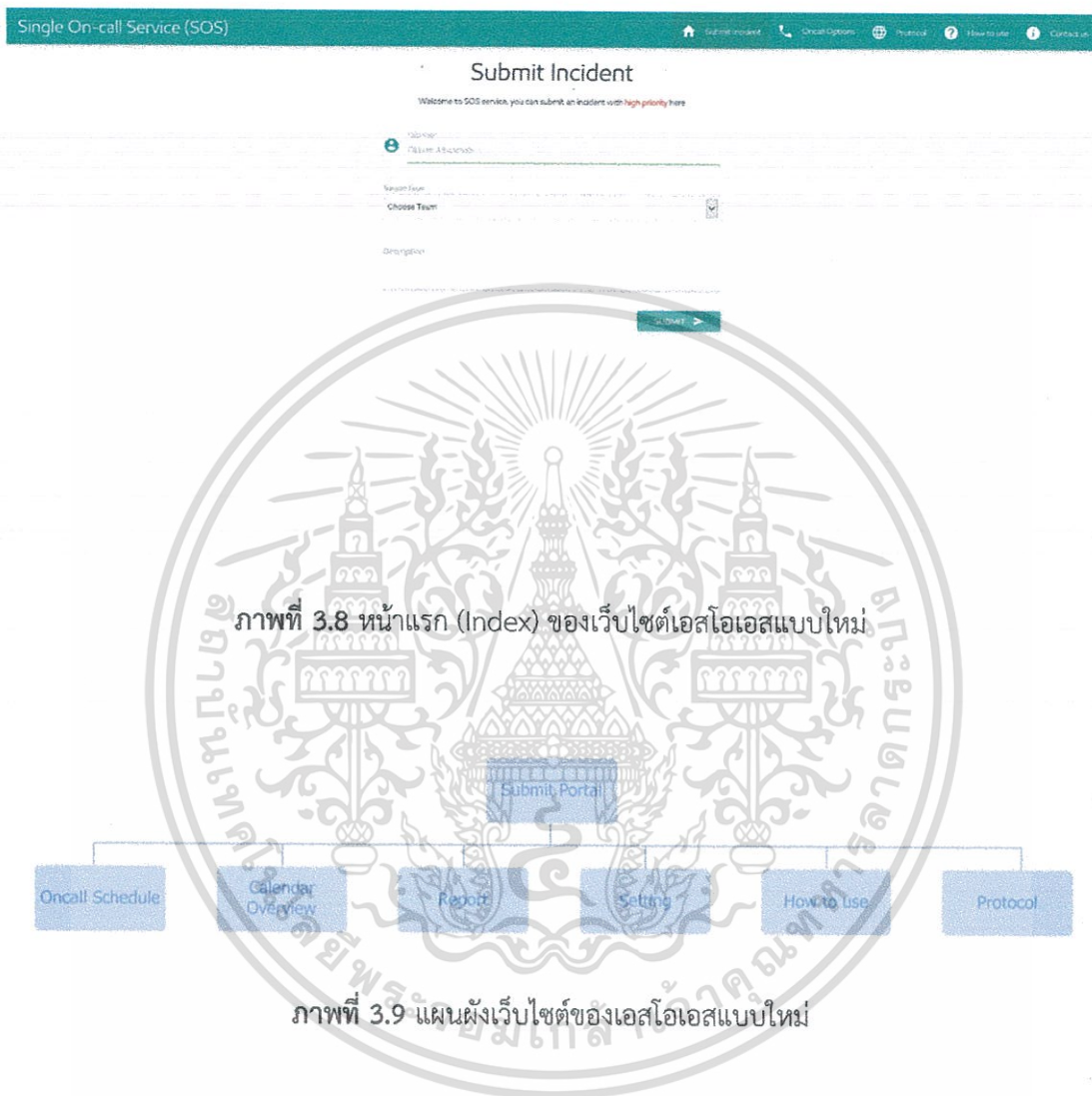


ภาพที่ 3.7 แผนผังเว็บไซต์ของเอสโอเอสแบบเก่า

จากผังงานของเว็บไซต์ดังแสดงในภาพที่ 3.7 แสดงให้เห็นว่า แต่ละหน้าของเอสโอเอส นั้นไม่มีการเชื่อมต่อถึงกันเลยยกเว้นในส่วนของพนักงานออนคอล 4 หน้า นั่นคือ หน้าจัดการกะของพนักงานออนคอล (On-call Schedule) หน้าตรวจสอบกะของพนักงานออนคอล (Calendar Overview) หน้ารายงาน (Report) และหน้าตั้งค่า (Setting) หนึ่งในขั้นตอนการปรับปรุงเว็บไซต์ให้มีประสิทธิภาพมากขึ้นคือการขจัดความลำบากในจุดนี้ออกไปให้หมดสิ้น

เมื่อทำการศึกษาโครงสร้างของหน้าเว็บไซต์ทั้งหมดแล้ว พบว่าสิ่งที่เว็บไซต์นี้ยังขาดไปคือ ความยืดหยุ่น (flexibility) และความคล่องแคล่ว (agility) ในการใช้งาน เนื่องจากเป้าหมายของเว็บไซต์เอสโอเอสแต่เดิมคือการสร้างและแจ้งเหตุของตัวงานที่มีความสำคัญสูง (high priority ticket) ดังนั้น การทำเว็บไซต์ให้ใช้งานได้อย่างรวดเร็ว จึงควรเป็นเป้าหมายหลักในการปรับปรุงเว็บไซต์นี้เพื่อให้มีประสิทธิภาพมากขึ้น การเปลี่ยนแปลงแรกที่เกิดขึ้นกับเว็บไซต์เอสโอเอสคือ การนำหน้าแรก (index) ดังแสดงในภาพที่ 3.6 ออก และแทนที่ด้วยหน้าซัมมิทพอร์ทอลเป็นหน้าแรก ดังภาพที่ 3.8 แทน ซึ่งจะช่วยให้ผู้ใช้งานสามารถทำการสร้างตัวงานได้เร็วขึ้น ซึ่งถือว่าตรงตามวัตถุประสงค์หลักของเว็บไซต์เอสโอเอส นอกจากนี้ยังทำให้ทุกหน้าของเอสโอเอสมี

การเชื่อมต่อกันทั้งหมด ไม่ต้องกลับไปหน้าแรกเพื่อที่จะไปยังหน้าอื่น ๆ อีก โดยแผนผังเว็บไซต์ของเอสโอเอสแบบใหม่สามารถแสดงให้เห็นได้ตามภาพที่ 3.9



ภาพที่ 3.8 หน้าแรก (Index) ของเว็บไซต์เอสโอเอสแบบใหม่

ภาพที่ 3.9 แผนผังเว็บไซต์ของเอสโอเอสแบบใหม่

3.2.2 การพัฒนาในส่วนของ back-end

อย่างที่กล่าวไว้ข้างต้นว่าส่วน back-end นั้นเปรียบเสมือนเป็นใจกลางของระบบเอสโอเอส ดังนั้น การศึกษาโครงสร้างส่วน back-end ของระบบเอสโอเอสนี้จึงเป็นสิ่งสำคัญมาก หากพัฒนาพลาดก็อาจทำให้ระบบเกิดการดำเนินงานผิดพลาดได้ ซึ่งอาจส่งผลกระทบต่อระบบหลักอย่างมหาศาล

เมื่อทำการศึกษาโครงสร้างส่วน back-end ของระบบเอสโอเอสเสร็จแล้ว ขั้นตอนถัดมา คือการเริ่มพัฒนาระบบตามตารางงานที่ได้รับมอบหมาย ซึ่งสามารถแจกแจงได้ ดังนี้

3.2.2.1 การเพิ่มเติมฟังก์ชันในส่วนของเอพีไอ

เอพีไอ หรือแอปพลิเคชันโปรแกรมมิ่งอินเตอร์เฟซ (Application Programming Interface) คือช่องทางการเชื่อมต่อที่ทำให้เว็บไซต์สามารถเชื่อมต่อกับ แอปพลิเคชันได้ ในส่วนนี้ได้มีการเพิ่มเติมรายละเอียดต่าง ๆ ที่ต้องนำมาแสดงผลใน หน้ารายละเอียดตั๋วงาน (Ticket Report) ทั้งหมด นั่นคือต้องทำการเขียนโปรแกรมเพื่อ รับค่ามาจากฐานข้อมูลเพิ่มเติม นอกจากนี้ยังต้องเพิ่มเติมฟังก์ชันในส่วนของการรองรับ ผลิตภัณฑ์ใหม่ ๆ อันเป็นผลมาจากการที่มีทีมอื่นมาติดต่อขอใช้งานแอปพลิเคชันนั่นเอง

3.2.2.2 การเพิ่มเติมฟังก์ชันในส่วนควบคุมอีเมลล์

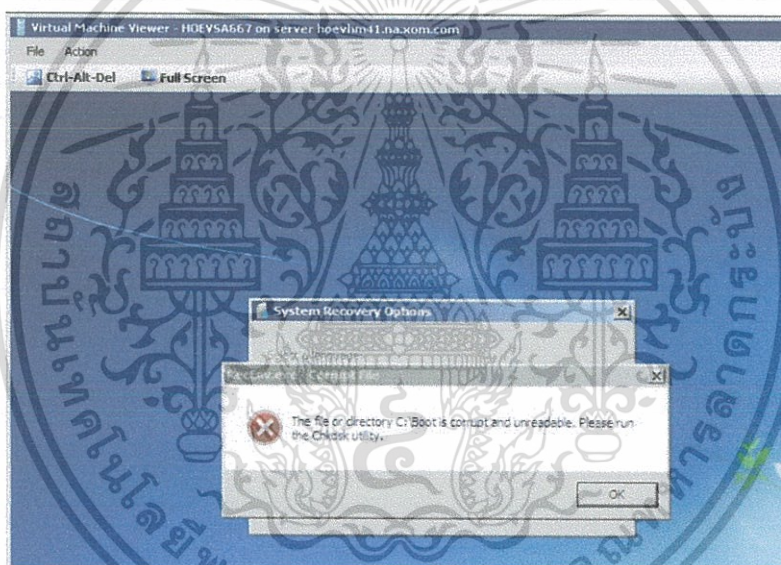
ส่วนควบคุมอีเมลล์ของแอปพลิเคชันเอสโอเอส จะเป็นโปรเจกต์ย่อยในโปรเจกต์ใหญ่ที่เรียกว่า เอสโอเอสเมลเลอร์ (SOS Mailer) จะมีไฟล์โค้ดที่เป็นตัวควบคุมหลักคือ SOSMailer.cs ซึ่งเป็นไฟล์ที่บรรจุฟังก์ชันที่ไว้สำหรับควบคุมการรับ-ส่งอีเมลล์ ทั้งหมด และ OutboundController.cs ซึ่งเป็นไฟล์ที่ไว้ควบคุมรายละเอียดเนื้อหาของอีเมลล์ที่ส่งออกไปต่าง ๆ ทั้งอีเมลล์ที่ส่งถึงผู้ใช้ อีเมลล์ที่ส่งถึงพนักงานออนคอลล และ อีเมลล์ที่ส่งถึงหัวหน้างาน ในตอนแรกนั้น รายละเอียดของอีเมลล์ที่ถูกยกระดับให้ หัวหน้างานจะมีบอกเพียงชื่อของพนักงานออนคอลลในขณะนั้นเท่านั้น หัวหน้างานจึงไม่อาจทราบได้เลยว่าตัวงานดังกล่าวมีเนื้อหาอะไรบ้าง ในจุดนี้จึงต้องทำการเพิ่ม รายละเอียดของเนื้อหาในอีเมลล์ที่ถูกยกระดับด้วย เพื่อให้หัวหน้างานจะได้ทราบถึง รายละเอียดของตัวงาน

3.2.2.3 การติดตั้งสิ่งแวดล้อมสำหรับการพัฒนาขึ้นมา

ในการใช้งานเอพีไอนั้น ไม่สามารถที่จะเรียกใช้งานตรง ๆ เหมือนกับการเขียนเว็บไซต์ด้วยภาษาเอชทีเอ็มแอลปกติได้ จำเป็นต้องเรียกใช้งานผ่านเซิร์ฟเวอร์จำลองขึ้นมา และเนื่องจากคอมพิวเตอร์ส่วนตัวของพนักงานบริษัท เอ็กซอนโมบิล จำกัด ทุกเครื่องไม่สามารถที่จะทำให้เครื่องตัวเองเป็นเจ้าบ้าน (Host) ในการสร้างเซิร์ฟเวอร์จำลองได้ ดังนั้น ทางออกหนึ่งของนักพัฒนาในบริษัทนี้คือการใช้เซิร์ฟเวอร์จำลองของบริษัทที่

ตั้งอยู่ที่เมืองดัลลาสและเมืองฮูสตัน ประเทศสหรัฐอเมริกา จากนั้นจึงใช้โปรแกรมรีโมทเดสก์ท็อปคอนเนคชัน (Remote Desktop Connection) เชื่อมต่อเข้าไปควบคุมการทำงาน

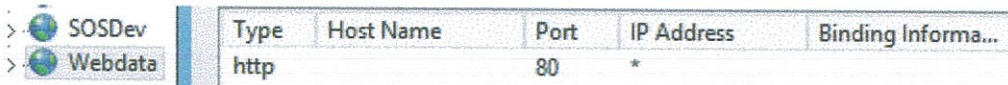
ในระหว่างการพัฒนาเอพีไอให้กับแอปพลิเคชันเอสโอเอสนั้น ได้เกิดเหตุการณ์ที่ไม่คาดคิดขึ้น คือ เซิร์ฟเวอร์จำลองที่ใช้ในการทดสอบระบบเกิดความเสียหายขึ้น ผลลัพธ์ที่ตามมาคือไฟล์ทุกอย่างที่อยู่บนเซิร์ฟเวอร์ดังกล่าวเกิดสูญหาย และไม่สามารถกู้คืนขึ้นมาได้ วิธีแก้ปัญหาในส่วนนี้คือ ทำการติดตั้งสิ่งแวดล้อมสำหรับการพัฒนาขึ้นมาใหม่ทั้งหมด ซึ่งรวมไปถึงฐานข้อมูลที่ใช้ในการทดสอบระบบก่อนนำออกไปใช้งานจริงด้วย



ภาพที่ 3.10 ปัญหาที่เกิดขึ้นบนเซิร์ฟเวอร์ที่ใช้สำหรับการพัฒนา ส่งผลให้ไม่สามารถใช้งานได้

ภาพที่ 3.10 แสดงให้เห็นถึงข้อผิดพลาด (Error) ที่เกิดขึ้นในเซิร์ฟเวอร์พัฒนาจากการที่เซิร์ฟเวอร์พัฒนาเกิดปัญหาขึ้นจนไม่สามารถใช้งานได้ ทำให้จำเป็นต้องหาเซิร์ฟเวอร์สำหรับการพัฒนาใหม่ ซึ่งหลังจากที่ทีมริเทิลได้ทำการปรึกษาหารือกันแล้ว ได้ข้อสรุปว่า จะทำการติดตั้งสภาพแวดล้อมสำหรับการพัฒนาบนเซิร์ฟเวอร์เดียวกับเซิร์ฟเวอร์ผลิตภัณท์โดยตรง แต่จะทำการแยกโฟลเดอร์ (folder) ที่เก็บไว้ และแยกที่อยู่ของเอพีไอตามพอร์ต (port) ซึ่งพอร์ตปกติของเว็บไซต์ที่ใช้งานจริง จะใช้พอร์ตหมายเลข 80

ตามภาพที่ 3.11 ส่วนพอร์ตของเว็บไซต์ที่ใช้สำหรับการพัฒนา จะใช้พอร์ตหมายเลข 83
ตามภาพที่ 3.12



Type	Host Name	Port	IP Address	Binding Informa...
http		80	*	

ภาพที่ 3.11 พอร์ตของเว็บไซต์ที่ใช้งานจริง จะเป็นพอร์ตหมายเลข 80



Type	Host Name	Port	IP Address	Binding Informa...
http		83	*	

ภาพที่ 3.12 พอร์ตของเว็บไซต์ที่ใช้พัฒนา จะเป็นพอร์ตหมายเลข 83

ข้อดีของการติดตั้งสภาพแวดล้อมสำหรับพัฒนาบนเซิร์ฟเวอร์ผลิตภัณฑ์โดยตรง คือ มีการบำรุงรักษาง่าย และไม่ต้องพะวงว่าเซิร์ฟเวอร์จะเสียหาย เนื่องจากหากเซิร์ฟเวอร์ผลิตภัณฑ์เกิดความเสียหายขึ้น หมายความว่าตัวเว็บไซต์ที่ใช้งานจริงย่อมต้องได้รับผลกระทบ ซึ่งส่งผลให้ทีมผู้ดูแลเซิร์ฟเวอร์ (Server Operation) ต้องรีบเข้ามาดูแลจัดการอย่างเร่งด่วน ทว่าหากเป็นเซิร์ฟเวอร์ของนักพัฒนา เมื่อเกิดความเสียหายขึ้น ทีมผู้ดูแลเซิร์ฟเวอร์อาจมองว่าไม่ใช่เรื่องใหญ่เนื่องจากเว็บไซต์ที่ใช้งานจริงไม่ได้รับผลกระทบ ทำให้การดูแลจัดการเซิร์ฟเวอร์ดำเนินไปได้อย่างล่าช้า

3.2.3 การสนับสนุนแอปพลิเคชันในส่วนของฐานข้อมูล

ฐานข้อมูลของเอสไอเอส เป็นอีกหนึ่งองค์ประกอบที่สำคัญไม่น้อยไปกว่าส่วน back-end เนื่องจากในฐานข้อมูลนี้มีการบรรจุข้อมูลทุกอย่างที่จำเป็นในการประมวลผล เช่น รายชื่อทีมต่างๆ รายชื่อพนักงานออนไลน์ อีเมลล์ของลูกค้า รายละเอียดตั๋วงาน วันและเวลาที่ตั๋วงานถูกสร้างขึ้นมา ทุกอย่างล้วนถูกเก็บไว้ในฐานข้อมูลทั้งสิ้น การสนับสนุนในส่วนนี้ส่วนมากจะเป็นการตรวจสอบตั๋วงานว่ามีตั๋วงานใดบ้างที่ไม่ถูกประมวลผลอันเนื่องมาจากความผิดพลาดของเครือข่าย

ระบบ หรือเหตุผลอื่น ๆ นอกจากนี้ เมื่อมีทีมอื่นมาติดต่อขอใช้งานแอปพลิเคชัน ก็จำเป็นที่จะต้องเพิ่มข้อมูลต่าง ๆ ของทีมใหม่เข้าไปในฐานข้อมูลด้วย

3.2.4 การสนับสนุนผู้ใช้เมื่อมีทีมอื่นมาติดต่อขอใช้งานแอปพลิเคชัน

แนวคิดระบบการทำงานของเอสโอเอสนั้นไม่ได้ถูกจำกัดเพียงแค่ว่าทีมมีเทเลเพียงทีมเดียว แต่แนวคิดนี้สามารถนำไปประยุกต์ใช้กับทุกทีมที่มีพนักงานออนคอลเป็นของตนเองได้ ดังนั้น จึงเป็นเรื่องปกติที่จะมีทีมอื่นติดต่อมาเพื่อขอใช้งานแอปพลิเคชันด้วย ดังแสดงให้เห็นในอีเมลล์ตามภาพที่ 3.13



ภาพที่ 3.13 แสดงอีเมลล์ของทีมอื่นที่ติดต่อเข้ามาเพื่อขอใช้งานแอปพลิเคชันเอสโอเอส

โดยในส่วนนี้จะเป็นการติดต่อกับพนักงานออนคอลของทีมอื่นเป็นหลัก ซึ่งโดยส่วนมากจะเป็นการสอบถามข้อมูลที่จำเป็น เช่น ชื่อทีม รายชื่อพนักงานออนคอลทั้งหมด รายชื่อผลิตภัณฑ์ที่ต้องการให้มีในระบบ จากนั้นจะทำการเพิ่มข้อมูลต่าง ๆ ลงไปฐานข้อมูล และสุดท้ายจะเป็นการทดสอบระบบโดยการทดลองสร้างตัวงานขึ้นมา แล้วระบุทีมที่ต้องการทดสอบลงไป เมื่อทดสอบระบบเสร็จสิ้นและผลลัพธ์เป็นไปตามที่ต้องการแล้ว จึงทำการแจ้งพนักงานออนคอลทุกคนในทีมดังกล่าวว่าระบบพร้อมใช้งานแล้วต่อไป

บทที่ 4

ผลการวิจัย

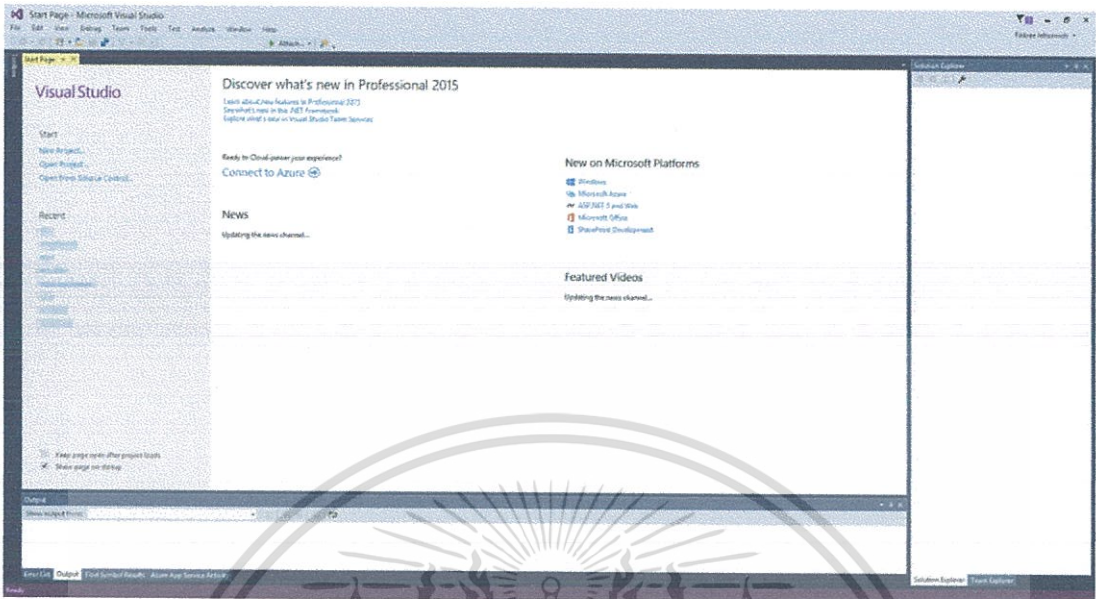
จากการทำวิจัยตลอดจนสิ้นสุดโครงการสหกิจศึกษา นี้ ผลการวิจัยที่ได้สามารถแสดงให้เห็นถึง ส่วนต่อประสานผู้ใช้ (Interface) ของชิ้นงาน ขั้นตอนการทำงานต่าง ๆ ของระบบ การแก้ไขจุดผิดพลาด (bug) ต่าง ๆ ในระบบ การเชื่อมต่อเซิร์ฟเวอร์สำหรับทดลองระบบเข้าสู่ฐานข้อมูลใหม่ รายละเอียดต่าง ๆ ที่ถูกเพิ่มขึ้นในอีเมลที่ถูกละระดับ (escalating email) โดยผลการวิจัยสามารถแบ่งได้เป็น 4 หัวข้อ ดังต่อไปนี้

4.1 การพัฒนาในส่วนของ front-end

การพัฒนาเว็บไซต์เอสโอเอสในส่วน front-end จะใช้โปรแกรมไมโครซอฟต์วิซวลสตูดิโอ (Microsoft Visual Studio) ในการพัฒนาเป็นหลัก สัญลักษณ์และหน้าตาของโปรแกรมสามารถแสดงให้เห็นได้ในภาพที่ 4.1 และ 4.2 สำหรับในส่วนของการทดสอบนั้น จะใช้เว็บเบราว์เซอร์ 2 ตัวในการทดสอบ นั่นคือกูเกิ้ลโครม (Google Chrome) และไมโครซอฟต์อินเทอร์เน็ตเอ็กซ์พลอเรอร์ (Microsoft Internet Explorer) เนื่องจากสคริปต์ (script) บางคำสั่งของบริษัท เอ็กซอนโมบิล จำกัด นั้น ถูกจำกัดการใช้งานให้ เปิดดูได้เพียงเบราว์เซอร์อินเทอร์เน็ตเอ็กซ์พลอเรอร์เท่านั้น แต่ด้วยประสิทธิภาพของกูเกิ้ลโครมที่ทำงานได้ ไวกว่าอินเทอร์เน็ตเอ็กซ์พลอเรอร์ ทำให้สำหรับบางหน้าที่ไม่จำเป็นต้องใช้สคริปต์ในการทดสอบ จะใช้กูเกิ้ลโครมในการทดสอบเพื่อความรวดเร็ว แต่หากมีหน้าที่ต้องใช้สคริปต์ จึงจะใช้อินเทอร์เน็ตเอ็กซ์พลอเรอร์ในการทดสอบแทน



ภาพที่ 4.1 สัญลักษณ์ (Icon) ของโปรแกรมไมโครซอฟต์วิซวลสตูดิโอ



ภาพที่ 4.2 หน้าตาภายนอกของโปรแกรมไมโครซอฟต์วิซวลสตูดิโอ

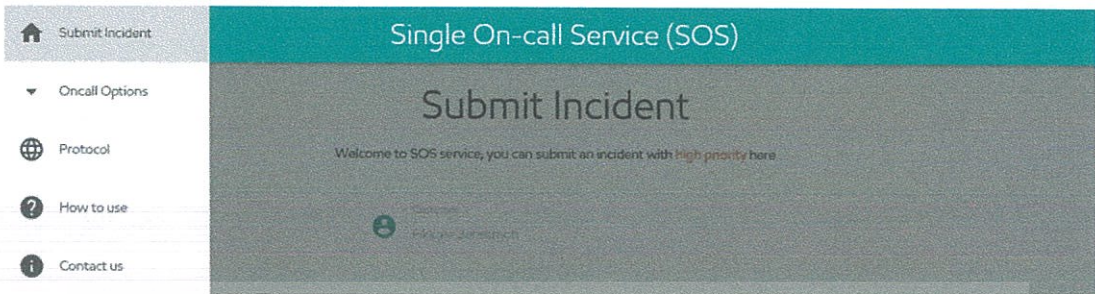
การพัฒนาหน้า front-end แต่ละหน้า ในทุก ๆ หน้าจะมีสิ่งหนึ่งที่เหมือนกันหมด คือ ส่วนบริเวณแถบด้านบนของเว็บไซต์ เรียกว่า Navigation bar หรือเรียกโดยย่อว่า Navbar ดังแสดงให้เห็นในภาพที่ 4.3 ซึ่งจะเป็นแถบที่มีลิ้งค์ (Link) ไปยังหน้าต่าง ๆ ของเว็บไซต์ และแถบบริเวณด้านซ้ายของเว็บไซต์ เรียกว่า Side bar หรือ SideNav ซึ่งจะปรากฏขึ้นก็ต่อเมื่อนำจอที่ใช้งานเว็บไซต์มีขนาดเล็กกว่าขนาดขั้นต่ำที่กำหนดไว้ โดยเป็นแถบที่มีการทำงานเหมือนกับ Navbar ทุกประการ ภาพที่ 4.4 และ 4.5 แสดงถึงลักษณะของแถบ SideNav ที่ปรากฏขึ้นบนหน้าจอของผู้ใช้งาน



ภาพที่ 4.3 แถบ Navbar บริเวณด้านบนของเว็บไซต์



ภาพที่ 4.4 เมื่อใช้งานหน้าจอที่มีขนาดเล็ก แถบ Navbar จะหายไปและปรากฏปุ่มสามแถบขึ้นมาแทน



ภาพที่ 4.5 แถบ SideNav ที่ปรากฏขึ้นหลังจากคลิกที่ปุ่มสามแถบ

นอกจากนี้ ในทุก ๆ หน้า ลิงค์ของหน้าที่ผู้ใช้กำลังใช้งานอยู่นั้น ทั้งแถบ Navbar และแถบ SideNav จะมีการทำเครื่องหมายไว้โดยจะเปลี่ยนสีพื้นหลังให้เป็นโทนเข้มขึ้นเล็กน้อย เพื่อเป็นจุดสังเกตให้แก่ผู้ใช้งานที่กำลังใช้งานอยู่ในหน้าใดของเว็บไซต์ ดังภาพที่ 4.3 และ 4.5 จะพบว่า บริเวณลิงค์ของหน้าสร้างตั๋วงาน (Submit Incident) นั้นเป็นสีเขียวเข้ม และสีเทา แทนที่จะเป็นสีเขียว และสีขาวตามลำดับ เนื่องจากเป็นจุดสังเกตให้ผู้ใช้ทราบว่ากำลังใช้งานอยู่ในหน้าสร้างตั๋วงานนั่นเอง

สำหรับส่วนอื่น ๆ ที่แตกต่างกันของแต่ละหน้านั้น จะสามารถแบ่งการพัฒนาออกตามขอบเขตได้ 2 ประเภท คือ การพัฒนาในส่วนอินเทอร์เฟซ (interface) สำหรับแสดงผลหน้าเว็บไซต์ และการพัฒนาในส่วนสคริปต์ (script) ที่ใช้สำหรับควบคุมการทำงานต่าง ๆ ของเว็บไซต์ โดยจะสามารถแจกแจงรายละเอียดได้ ดังนี้

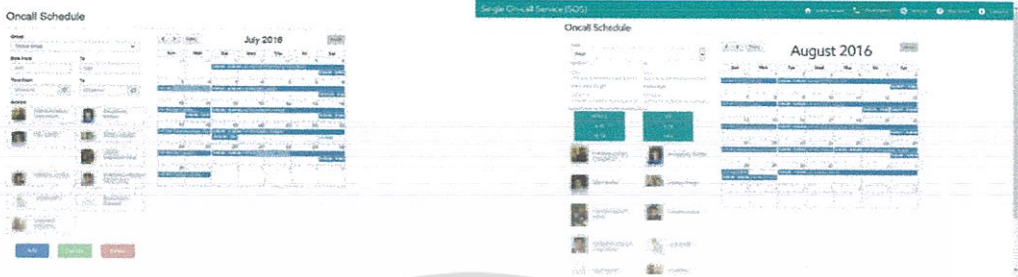
4.1.1 การพัฒนาในส่วนอินเทอร์เฟซ (interface)

อินเทอร์เฟซของเว็บไซต์นั้น เป็นส่วนที่มีสำคัญมากเป็นอันดับต้น ๆ ของเว็บไซต์ หากเว็บไซต์มีการออกแบบอินเทอร์เฟซที่ดี มีการแยกองค์ประกอบชัดเจน ผู้ใช้ก็สามารถใช้งานเว็บไซต์ได้โดยง่าย ซึ่งการพัฒนาในส่วนนี้สามารถแบ่งได้เป็น 7 ส่วน ดังนี้

4.1.1.1 หน้าจัดการกะพนักงานออนคอล (On-call Schedule)

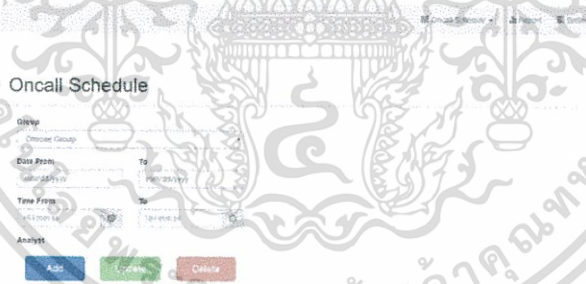
หน้าจัดการกะพนักงานออนคอลนั้นเป็นหน้าที่มีความสำคัญที่สุดในส่วน front-end เนื่องจากเป็นหน้าที่ผู้จัดกะ (Shift Scheduler) จะต้องมาคอยจัดกะให้พนักงานออนคอลทุกสัปดาห์ จากภาพที่ 4.6 แสดงให้เห็นถึงความแตกต่างระหว่างหน้าจัดการกะ

พนักงานออนคอลของเว็บไซต์เอสโอเอสแบบเก่า เมื่อเปรียบเทียบกับเว็บไซต์เอสโอเอสแบบใหม่



ภาพที่ 4.6 เปรียบเทียบหน้าจัดการกะพนักงานออนคอล แบบเก่า (ซ้าย) กับแบบใหม่ (ขวา)

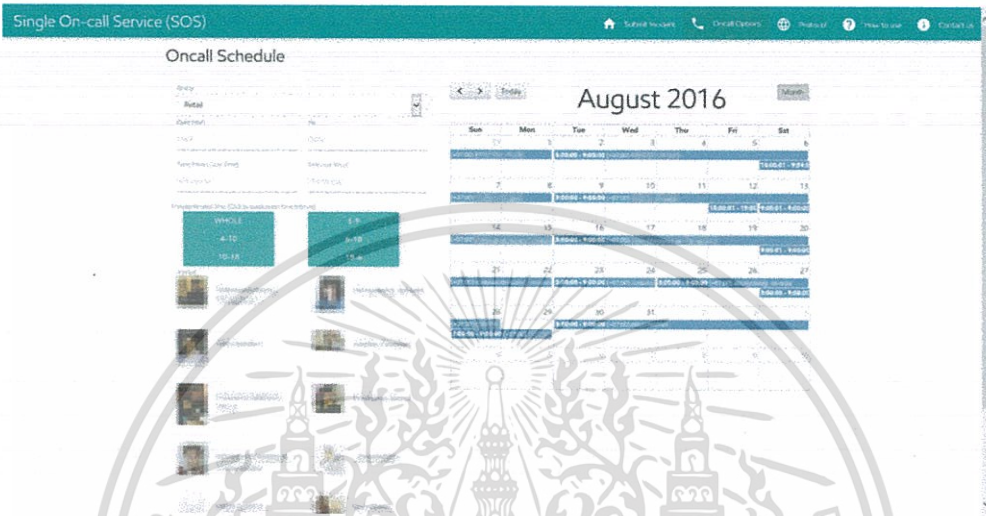
เดิมทีแล้ว ในหน้าจัดการกะพนักงานออนคอล มีจุดมุ่งหมายเพียงเพื่อให้ผู้จัดการกะเข้ามาจัดการกะเวลาอย่างเดียวนั้น เมื่อผู้ใช้อื่นทั่วไปที่ไม่ได้รับอนุญาตให้จัดการกะเข้ามาในหน้านี้ จะไม่สามารถมองเห็นอะไรได้เลยดังภาพที่ 4.7



ภาพที่ 4.7 เมื่อผู้ใช้ที่ไม่มีสิทธิ์ในการแก้ไขเข้ามาในหน้านี้ จะไม่พบอะไรปรากฏบนหน้าจอเลย

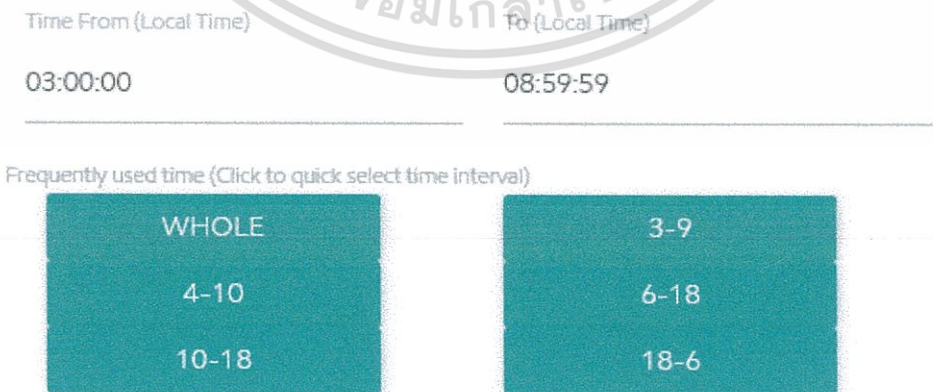
โครงข่ายอินเทอร์เน็ตของบริษัท เอ็กซอนโมบิล จำกัด นั้น ถูกจำกัดให้ใช้งานได้เฉพาะพนักงานของบริษัท เอ็กซอนโมบิล จำกัด อยู่แล้ว ดังนั้น การป้องกันไม่ให้ผู้ใช้ทั่วไปมองเห็นหน้านี้จึงเป็นสิ่งที่ไม่จำเป็น ดังนั้น สิ่งแรกที่ต้องทำการแก้ไขในหน้าจัดการ

กะพนักงานออนคอล คือทำให้ผู้ใช้ทุกคนสามารถมองเห็นหน้านี้ได้อย่างปกติ ดังภาพที่ 4.8



ภาพที่ 4.8 หน้าจัดการกะพนักงานออนคอล (On-call Schedule) หลังจากทำการปรับปรุงแก้ไขแล้ว

นอกจากนี้ยังมีการเพิ่มเติมองค์ประกอบคือ การทำปุ่ม “ช่วงเวลาที่ใช้บ่อย (frequently used time) สำหรับผู้จัดกะ เพื่อลดภาระที่เกิดขึ้นของผู้จัดกะเมื่อต้องทำการจัดกะในสัปดาห์ต่อ ๆ ไป ดังแสดงให้เห็นในภาพที่ 4.9



ภาพที่ 4.9 ปุ่มช่วงเวลาที่ใช้บ่อย (Frequently used time)

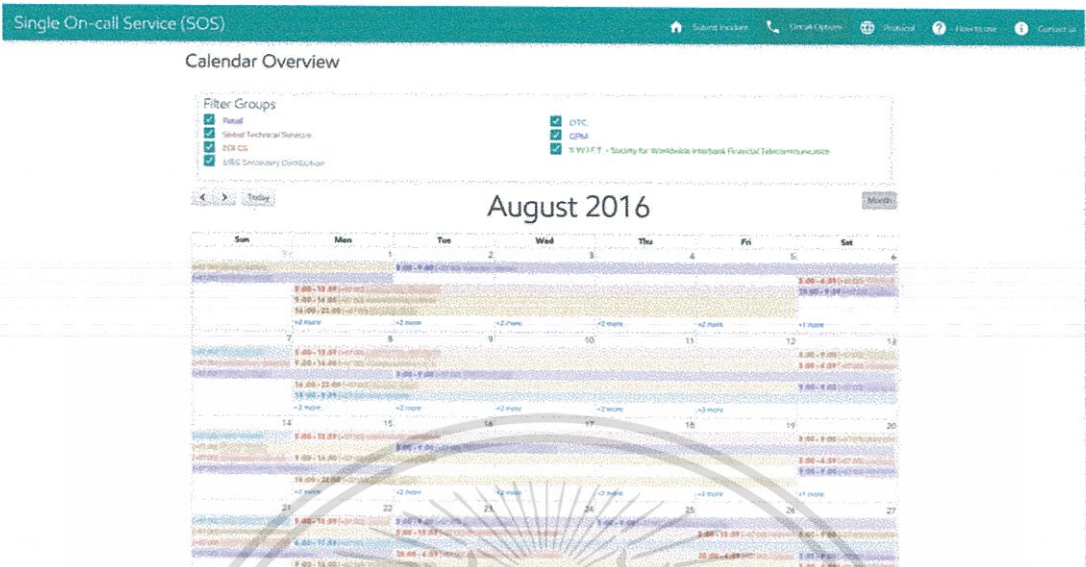
4.1.1.2 หน้าตรวจสอบกะพนักงานออนคอล (Calendar Overview)

ในหน้านี้จะแสดงกะของพนักงานออนคอล โดยจะแสดงเพียงกะของพนักงานออนคอลในทีมที่ได้ทำเครื่องหมายถูก (Check) ไว้ จากภาพที่ 4.10 แสดงให้เห็นถึงความแตกต่างระหว่างหน้าตรวจสอบกะพนักงานออนคอลของเว็บไซต์เอสไอเอสแบบเก่า เมื่อเปรียบเทียบกับเว็บไซต์เอสไอเอสแบบใหม่

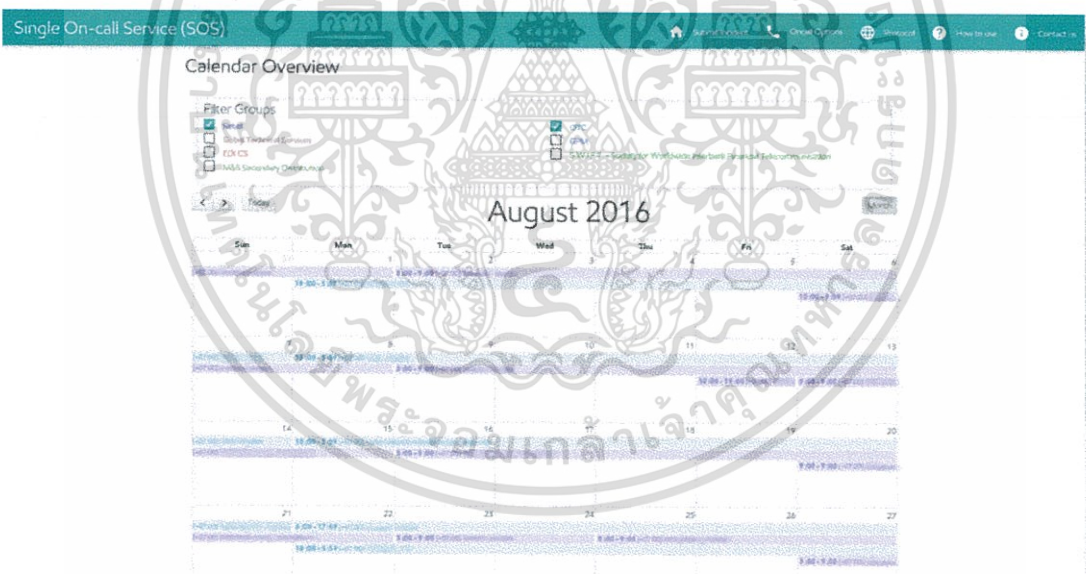


ภาพที่ 4.10 เปรียบเทียบหน้าตรวจสอบกะพนักงานออนคอล แบบเก่า (ซ้าย) กับแบบใหม่ (ขวา)

โดยปกติแล้วเมื่อเข้าสู่หน้านี้ครั้งแรก จะทำการเลือกทุกทีมไว้โดยอัตโนมัติ ดังแสดงให้เห็นในภาพที่ 4.11 ทว่า ผู้ใช้สามารถเลือกเครื่องหมายถูกเพื่อแสดงเฉพาะกะเวลาของพนักงานออนคอลของทีมที่ต้องการดูได้ ดังแสดงให้เห็นในภาพที่ 4.12



ภาพที่ 4.11 หน้าตรวจสอบกะพนักงานออนคอล (Calendar Overview)



ภาพที่ 4.12 หน้าตรวจสอบกะพนักงานออนคอล เมื่อทำการเลือกแสดงกะของบางทีมเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.1.3 หน้ารายงาน (Report)

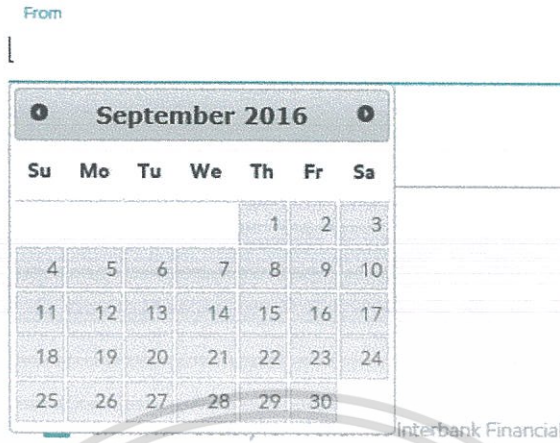
ในหน้ารายงานนี้จะเป็นหน้าที่แสดงตัวงานทั้งหมดที่ระบบเอสโอเอสได้ทำการส่งไป จากภาพที่ 4.13 แสดงให้เห็นถึงความแตกต่างระหว่างหน้ารายงานของเว็บไซต์เอสโอเอสแบบเก่า เมื่อเปรียบเทียบกับเว็บไซต์เอสโอเอสแบบใหม่



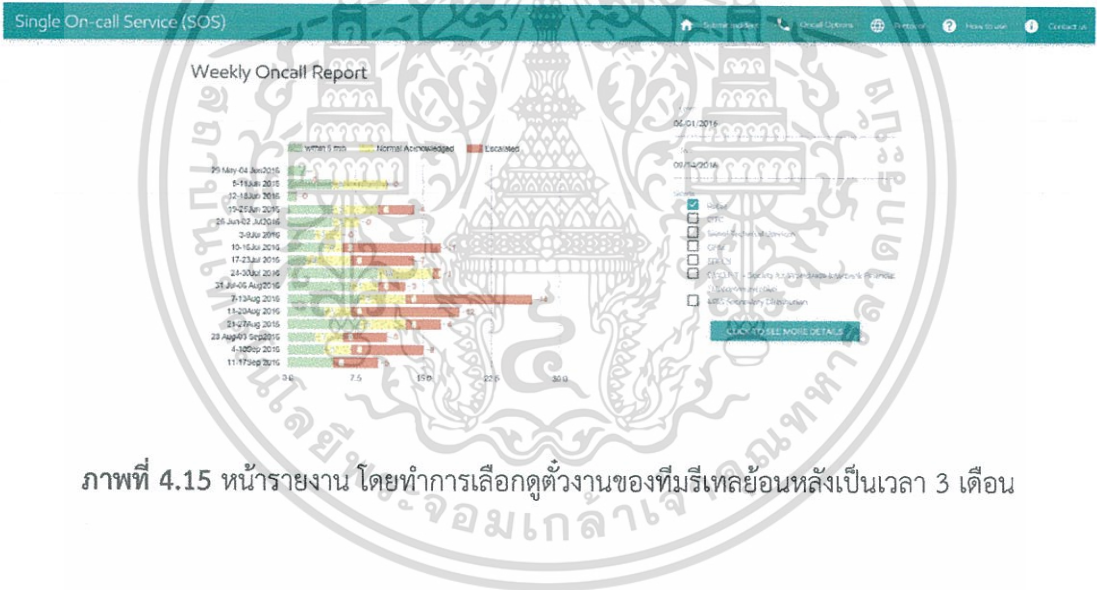
ภาพที่ 4.13 เปรียบเทียบหน้ารายงาน แบบเก่า (ซ้าย) กับแบบใหม่ (ขวา)

โดยในหน้ารายงานนั้นจะแบ่งตัวงานที่แสดงในหน้าจออกเป็น 3 ประเภท ได้แก่ ตัวงานที่พนักงานออนคอลทำการตอบรับภายใน 5 นาที จะแสดงเป็นสีเขียว ตัวงานที่พนักงานออนคอลทำการตอบรับภายใน 30 นาที จะแสดงเป็นสีเหลือง และตัวงานที่พนักงานออนคอลทำการตอบรับหลัง 30 นาที หรือไม่มีการตอบรับเลย หมายความว่า ตัวงานนี้ถูกยกระดับ (escalate) ไปสู่หัวหน้างาน (supervisor) ของแต่ละทีมแล้ว จะแสดงเป็นสีแดง

นอกจากนี้ ผู้ใช้ยังสามารถทำการเลือกดูตัวงานเฉพาะทีมที่ต้องการดูได้ โดยทำการเลือกเครื่องหมายถูก (check) แบบเดียวกับในหน้าตรวจสอบกะพนักงานออนคอล ซึ่งในหน้ารายงานจะปรากฏเพียงตัวงานของทีมที่มีการทำเครื่องหมายถูกเท่านั้น และผู้ใช้ยังสามารถเลือกช่วงเวลาของตัวงานที่จะดูได้อย่างอิสระ โดยเมื่อคลิกไปที่กล่องข้อความ from และ to จะปรากฏหน้าต่างปฏิทินให้เลือกวันที่ ดังแสดงให้เห็นในภาพที่ 4.14 ซึ่งหน้ารายงานจะแสดงตัวงานในช่วงเวลาวันที่ที่ผู้ใช้เลือกไว้ทั้งหมด ดังแสดงให้เห็นในภาพที่ 4.15 แต่หากไม่มีการเลือกวันที่ไว้ หน้ารายงานจะแสดงผลเพียง 3 สัปดาห์ล่าสุดจนถึงวันที่ปัจจุบันเท่านั้น



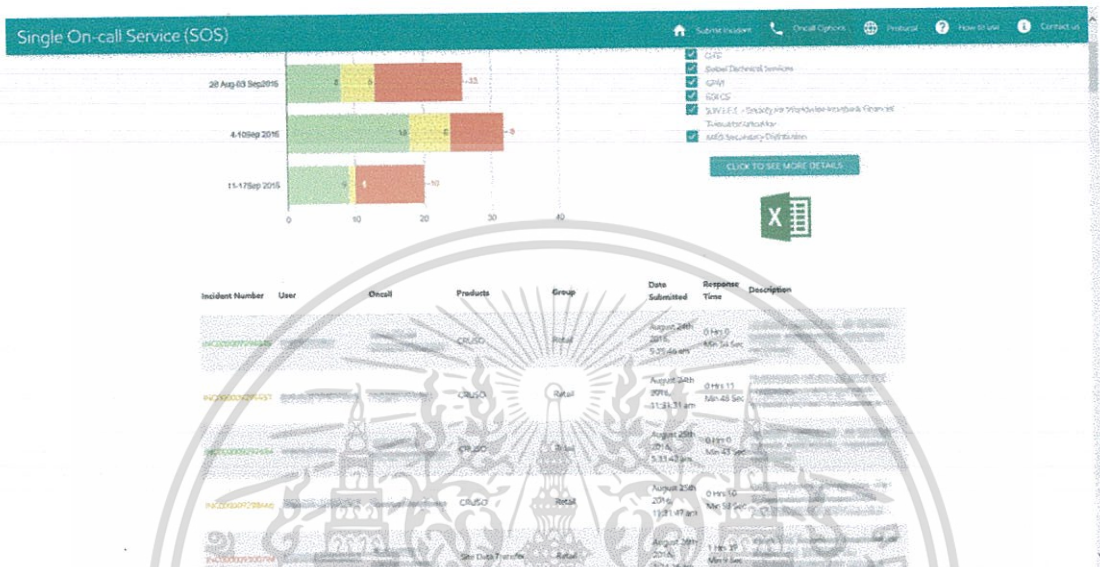
ภาพที่ 4.14 เมื่อคลิกที่กล่องข้อความ from หรือ to จะปรากฏปฏิทินขึ้นมา



ภาพที่ 4.15 หน้ารายงาน โดยทำการเลือกดูตัวงานของทีมที่เรลย่อนหลังเป็นเวลา 3 เดือน

โดยปกติแล้ว ในหน้ารายงานจะแสดงเพียงจำนวนตัวงานในช่วงเวลาหนึ่งเท่านั้น หากต้องการทราบถึงรายละเอียดของตัวงาน ผู้ใช้สามารถคลิกที่ปุ่ม Click to see more details เพื่อดูรายละเอียดเพิ่มเติมของตัวงานได้ ดังแสดงให้เห็นในภาพที่ 4.16 โดยในรายละเอียดของตัวงานที่แสดงจะประกอบไปด้วยหมายเลขเหตุฉุกเฉิน (Incident Number) ผู้ใช้ (User) พนักงานออนคอลที่รับเรื่อง (On-call) ผลิตภัณฑ์หรือแอปพลิเคชันที่มีปัญหา (Products) ทีมที่รับผิดชอบ (Group) วันที่สร้างตัวงาน (Date

submitted) เวลาที่ใช้ในการตอบอีเมลล์เอสโอเอส (Response time) และรายละเอียดของตัวงาน (Description)



ภาพที่ 4.16 หน้ารายงาน เมื่อทำการคลิกปุ่ม Click to see more details แล้ว

4.1.1.4 หน้าตั้งค่า (Setting)

นอกจากการจัดการกะพนักงานออนคอลแล้ว การตั้งค่าพนักงานออนคอลเองก็เป็นอีกหนึ่งปัจจัยหลักที่ขาดไม่ได้สำหรับระบบนี้ หนึ่งในหัวข้อการพัฒนาคือการทำให้หน้าตั้งค่าสามารถใช้งานได้อย่างที่ควรจะเป็น ภาพที่ 4.17 แสดงให้เห็นถึงความแตกต่างระหว่างหน้าตั้งค่าของเว็บไซต์เอสโอเอสแบบเก่า เมื่อเปรียบเทียบกับเว็บไซต์เอสโอเอสแบบใหม่



ภาพที่ 4.17 เปรียบเทียบหน้าตั้งค่า แบบเก่า (ซ้าย) กับแบบใหม่ (ขวา)

จุดประสงค์หลักของหน้าตั้งค่าในตอนแรกนั้น คือการให้พนักงานที่มีสิทธิ์ในการเพิ่ม-ถอดรายชื่อพนักงานออนไลน์ หรือพนักงานที่เป็นผู้ดูแล (Admin) ของแต่ละทีม สามารถทำการเพิ่ม-ถอดรายชื่อพนักงานออนไลน์ได้เลยโดยไม่ต้องไปทำการเพิ่ม-ถอดรายชื่อในฐานข้อมูล การพัฒนาให้เว็บไซต์สามารถทำงานได้บรรลุตามวัตถุประสงค์ จึงเป็นสิ่งสำคัญที่สุดในการพัฒนาหน้าตั้งค่านี้

พนักงานผู้ดูแลประจำทีมจะสามารถทำการเพิ่ม-ถอดรายชื่อพนักงานออนไลน์ออกจากทีมได้ โดยการคลิกเลือกทีมจากรายชื่อ แล้วคลิกที่ปุ่ม Modify เมื่อคลิกแล้ว จะสามารถเพิ่ม-ถอดพนักงาน แก้ไขอีเมลล์สำรอง และแก้ไขการเข้าถึงต่าง ๆ ของพนักงานแต่ละคน คือ การมีสิทธิ์ของผู้ดูแล (Admin) การมีสิทธิ์ได้รับอีเมลล์ยกระดับ (Level 2) การมีสิทธิ์ได้รับอีเมลล์ระหว่างช่วงเวลาทำงาน (Work hour receiver) และการมีสิทธิ์ได้รับอีเมลล์ยืนยัน (Acknowledgement) ได้ ภาพที่ 4.18 แสดงให้เห็นถึงหน้าตาของส่วนประสานผู้ใช้หลังจากทำการคลิกที่ปุ่ม Modify แล้ว

Setting

Support Group: M&S Secondary Distribution

SUBMIT CANCEL

Domain	Lan ID	Name	Email address	Shared Mailbox	Alternate Email	
01	01	01	01	01	01	<input checked="" type="checkbox"/> Admin <input checked="" type="checkbox"/> Level 2 <input type="checkbox"/> Work Hour Receiver <input type="checkbox"/> Acknowledgement
02	02	02	02	02	02	<input type="checkbox"/> Admin <input type="checkbox"/> Level 2 <input type="checkbox"/> Work Hour Receiver <input type="checkbox"/> Acknowledgement
03	03	03	03	03	03	<input type="checkbox"/> Admin <input type="checkbox"/> Level 2 <input type="checkbox"/> Work Hour Receiver <input type="checkbox"/> Acknowledgement
04	04	04	04	04	04	<input type="checkbox"/> Admin <input type="checkbox"/> Level 2 <input type="checkbox"/> Work Hour Receiver <input type="checkbox"/> Acknowledgement

+ ADD MORE

ภาพที่ 4.18 เมื่อคลิกที่ปุ่ม Modify จะสามารถแก้ไขรายละเอียดและเพิ่ม-ถอนรายชื่อได้

เมื่อพนักงานผู้ดูแลต้องการเพิ่มรายชื่อพนักงานออนคอลเข้าไปในทีม พนักงานผู้ดูแลสามารถคลิกที่ปุ่ม Add More ซึ่งจะเป็นการสร้างช่องรายละเอียดใหม่ขึ้นมา ดังแสดงให้เห็นในภาพที่ 4.19 จากนั้นพนักงานผู้ดูแลจะต้องกรอกโดเมน (Domain) และไอดีที่ใช้ภายใน (Lan ID) ของพนักงานออนคอลคนนั้น แล้วทำการคลิกที่ปุ่ม Submit สีเขียวที่จะปรากฏขึ้นเฉพาะหลังคลิกปุ่ม Modify แล้วเท่านั้นเพื่อทำการบันทึกชื่อพนักงานออนคอลคนใหม่ลงในฐานข้อมูลโดยตรง หากพนักงานผู้ดูแลไม่ต้องการบันทึกสามารถคลิกที่ปุ่ม Cancel สีแดงเพื่อยกเลิกการแก้ไขได้

Domain	Lan ID	Name	Email address	Shared Mailbox	Alternate Email	
01	01	01	01	01	01	<input type="checkbox"/> Admin <input type="checkbox"/> Level 2 <input type="checkbox"/> Work Hour Receiver <input type="checkbox"/> Acknowledgement

ภาพที่ 4.19 เมื่อคลิกที่ปุ่ม Add More แล้ว จะสามารถเพิ่มรายชื่อพนักงานออนคอลลงไปใหม่ได้

เมื่อพนักงานผู้ดูแลต้องการลบรายชื่อพนักงานออนคอลออกจากทีม พนักงานผู้ดูแลสามารถทำได้โดยการคลิกที่เครื่องหมายกากบาทที่จะปรากฏบริเวณด้านซ้ายของรูปพนักงานออนคอลหลังจากคลิกที่ปุ่ม Modify แล้วเท่านั้น ดังแสดงให้เห็นในภาพที่ 4.20 เมื่อคลิกแล้วรายชื่อพนักงานออนคอลคนดังกล่าวจะถูกลบออกจากรายชื่อข้อมูล

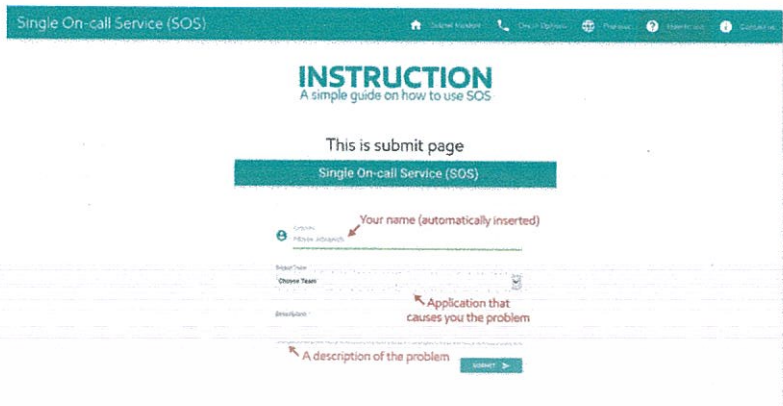


ภาพที่ 4.20 เครื่องหมายกากบาทที่จะปรากฏบริเวณด้านซ้ายของรูป หลังคลิกที่ปุ่ม Modify

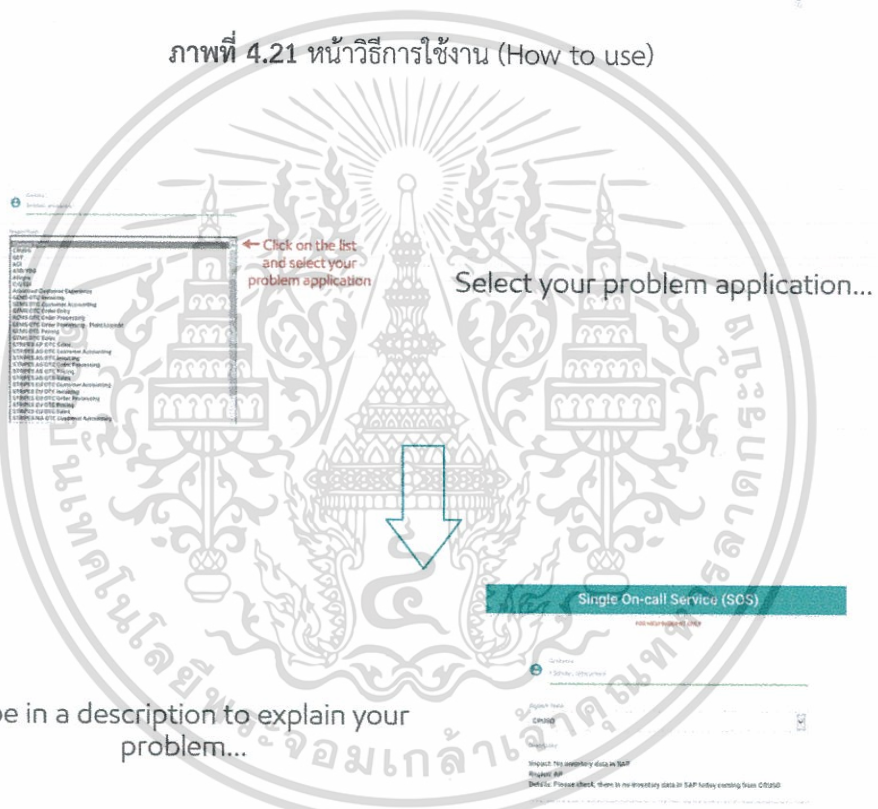
4.1.1.5 หน้าวิธีการใช้งาน (How to use)

สำหรับผู้ใช้ที่เข้ามาใช้งานในเว็บไซต์เป็นครั้งแรก อาจเกิดความงุนงงในการใช้งานได้ การสร้างหน้าเว็บไซต์ที่ไว้สอนวิธีการใช้งานแก่ผู้ใช้งานใหม่ต่างหากอีกหนึ่งหน้า จึงเป็นหนึ่งในวิธีที่สามารถแก้ปัญหาในจุดนี้ได้

เมื่อผู้ใช้ทำการเลื่อนลงมาตามหน้าเว็บไซต์แล้ว จะพบกับภาพที่แสดงวิธีการใช้งานขั้นตอนต่อไปออกมาเรื่อย ๆ จนจบขั้นตอน ตัวอย่างของวิธีการใช้งานเว็บไซต์จะแสดงให้เห็นในภาพที่ 4.21 และ 4.22



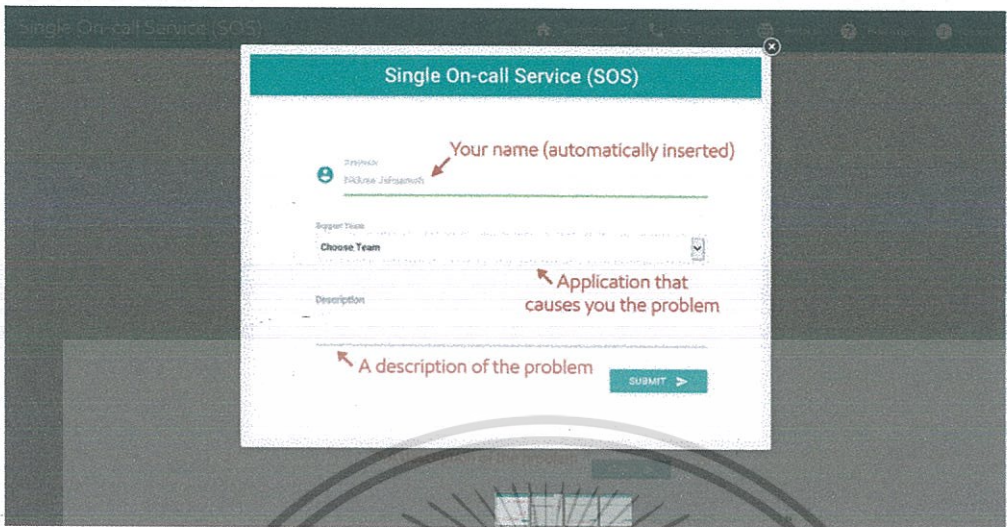
ภาพที่ 4.21 หน้าวิธีการใช้งาน (How to use)



Type in a description to explain your problem...

ภาพที่ 4.22 ภาพแสดงวิธีการใช้งานเว็บไซต์เอสโอเอส

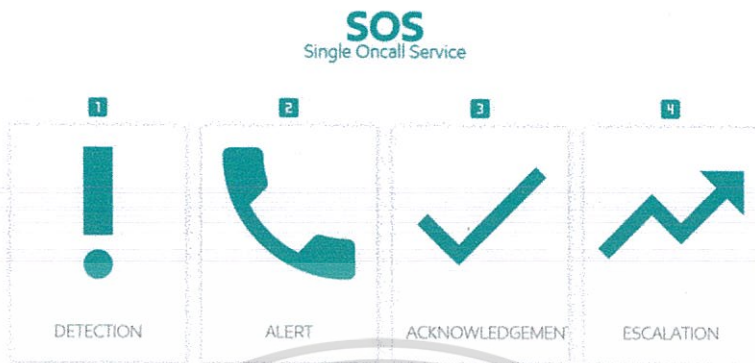
นอกจากนี้ผู้ใ้ยังสามารถขยายภาพให้มีขนาดใหญ่ขึ้นได้โดยคลิกที่รูปภาพ จะปรากฏเป็นรูปภาพขนาดใหญ่ขึ้นที่หน้าจอ ซึ่งสามารถเลือกภาพที่ต้องการดูได้จากแถบนำทางบริเวณด้านล่าง ดังแสดงให้เห็นในภาพที่ 4.23



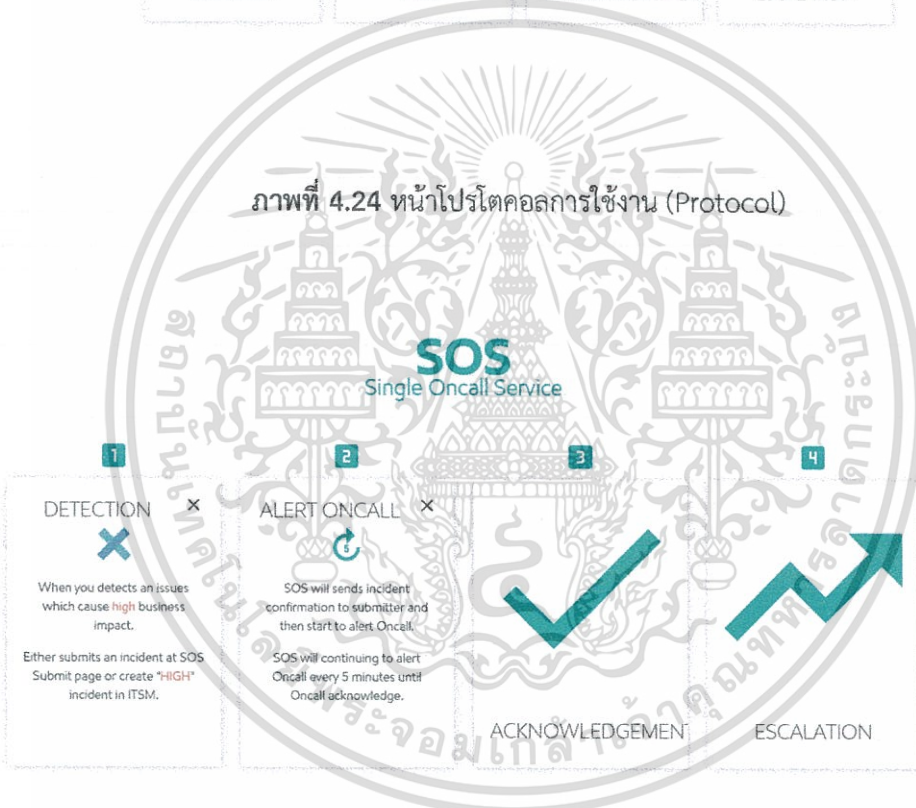
ภาพที่ 4.23 เมื่อคลิกที่ภาพ จะปรากฏเป็นภาพขนาดใหญ่ขึ้นที่หน้าจอ

4.1.1.6 หน้าโปรโตคอลการใช้งาน (Protocol)

การอธิบายถึงขั้นตอนการทำงานของระบบเอสโอเอสให้ผู้ใช้งานได้ทราบเองก็ถือเป็นอีกหนึ่งปัจจัยสำคัญ โดยขั้นตอนการทำงานของระบบนั้นสามารถพบได้ในหน้าโปรโตคอลนี้ ซึ่งผู้ใช้งานสามารถคลิกไปที่การ์ด (Card) ต่าง ๆ เพื่ออ่านรายละเอียดของแต่ละขั้นตอนได้ ภาพที่ 4.24 และ 4.25 แสดงให้เห็นถึงตัวอย่างการใช้งานหน้าโปรโตคอล



ภาพที่ 4.24 หน้าโปรโตคอลการใช้งาน (Protocol)



ภาพที่ 4.25 สามารถคลิกที่การ์ด เพื่ออ่านรายละเอียดเพิ่มเติมได้

4.1.1.7 รายละเอียดอื่น ๆ

หากผู้ใช้ต้องการที่จะติดต่อกับทางผู้พัฒนาโดยตรง ผู้ใช้สามารถคลิกที่ปุ่ม Contact us บริเวณแถบ Navbar และในแถบ Sidebar ของเว็บไซต์ได้ ซึ่งจะแสดงเป็น

กล่องข้อความที่มีอีเมลล์ของผู้พัฒนาอยู่ ผู้ใช้สามารถทำการส่งอีเมลล์มายังอีเมลล์นี้เพื่อติดต่อกับผู้พัฒนาได้โดยตรง ดังแสดงให้เห็นในภาพที่ 4.26



ภาพที่ 4.26 กล่องข้อความ (Modal) แสดงอีเมลล์ของผู้พัฒนา

4.1.2 การพัฒนาในส่วนสคริปต์ (script)

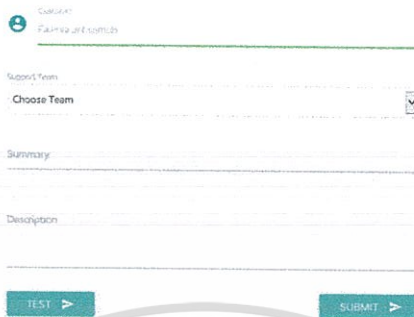
นอกจากส่วนต่อประสานผู้ใช้แล้ว สคริปต์เองก็ถือเป็นอีกหนึ่งองค์ประกอบที่ทำให้เว็บไซต์ทำงานได้ถูกต้องตรงวัตถุประสงค์ ซึ่งสามารถแยกส่วนของการพัฒนาได้ 3 ส่วน ดังนี้

4.1.2.1 หน้าซั่มมิทพอร์ทอล (Submit Portal)

สคริปต์ที่มีการพัฒนาในหน้านี้ ไม่ถือเป็นส่วนหนึ่งของการพัฒนาเว็บไซต์เอสโอเอส แต่เป็นสคริปต์ที่ช่วยลดระยะเวลาการทดสอบในส่วนของ back-end ได้อย่างมหาศาล คือสคริปต์ที่ใช้ทดสอบอีเมลล์ ซึ่งสคริปต์นี้จะทำการสร้างตัวงานขึ้นจากข้อความที่กรอกไปในกล่องรายละเอียด แล้วส่งไปที่ระบบไอทีเอสเอ็มโดยอัตโนมัติ โดยระบบไอทีเอสเอ็มนี้จะไม่ใช่ระบบจริงที่ใช้งานกันในบริษัท แต่จะเป็นระบบทดสอบที่ถูกสร้างขึ้นมาเพื่อใช้ทดสอบแอปพลิเคชันต่าง ๆ โดยเฉพาะ ซึ่งจะเรียกระบบไอทีเอสเอ็มนี้ว่า ไอทีเอสเอ็มแอกเซพแทนซ์ (ITSM Acceptance)

Submit Incident

Welcome to SOS service, you can submit an incident with **high priority** here



Comment
Please write comment

Support Team
Choose Team

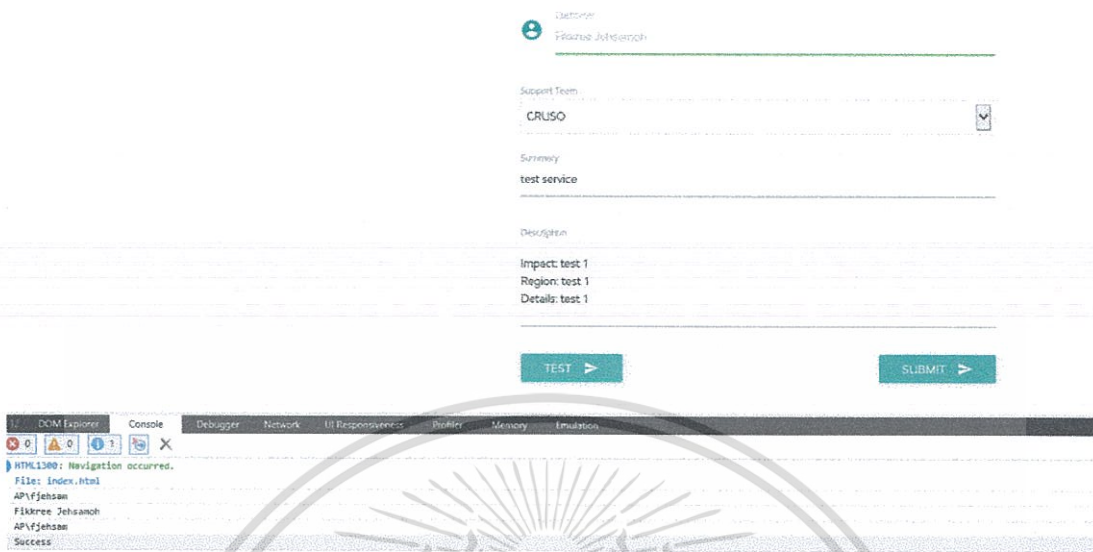
Summary

Description

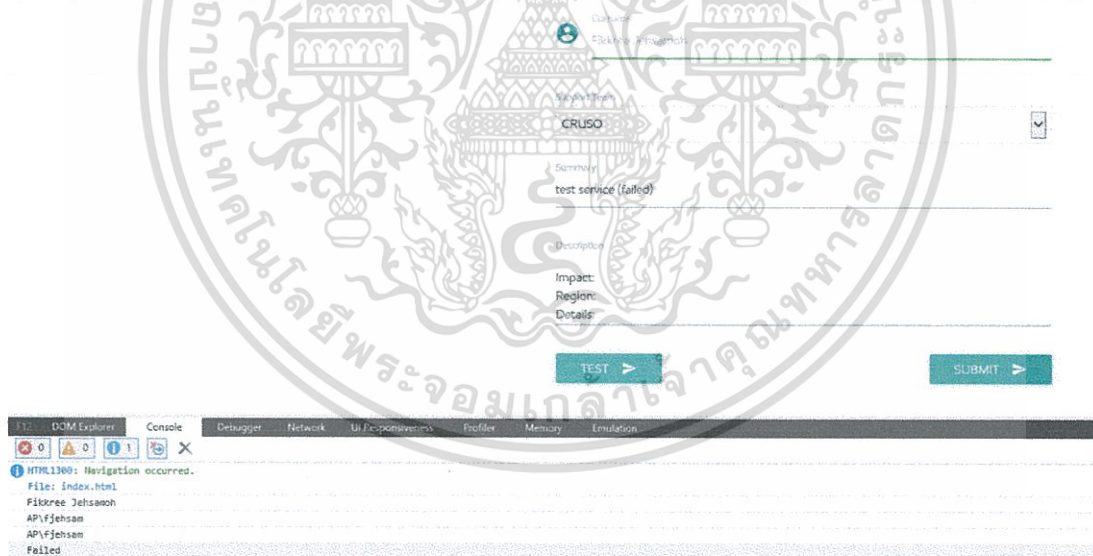
TEST > SUBMIT >

ภาพที่ 4.27 หน้าซับมิทพอร์ทอลที่ใช้ในการทดสอบระบบอีเมลล์เท่านั้น

ภาพที่ 4.27 แสดงให้เห็นถึงหน้าซับมิทพอร์ทอลที่ใช้ในการทดสอบระบบอีเมลล์เท่านั้น ขั้นตอนการทำงานในหน้านี้คือ เมื่อกรอกรายละเอียดต่าง ๆ ลงไปในช่อง Summary และ Description แล้ว จึงทำการทดสอบด้วยการคลิกไปที่ปุ่ม Test ด้านซ้ายของหน้าจอ ต่อมาจะทำการตรวจสอบว่ารายละเอียดต่าง ๆ ที่ได้กรอกไปนั้นส่งสำเร็จหรือไม่ โดยการกดปุ่ม F12 บนคีย์บอร์ดเพื่อเปิดส่วนคอนโซลของเว็บเบราว์เซอร์ขึ้นมา หากส่งสำเร็จจะปรากฏข้อความระบุว่า Success ดังแสดงให้เห็นในภาพที่ 4.28 แต่หากส่งไม่สำเร็จ จะปรากฏข้อความว่า Failed แทน ดังแสดงให้เห็นในภาพที่ 4.29 จากนั้นจึงทำการรออีเมลล์ที่ส่งกลับมาจากระบบเพื่อทดสอบว่าระบบเอสโอเอสสามารถใช้งานได้จริง และข้อความที่ระบุในอีเมลล์นั้นตรงตามที่ต้องการแสดงให้เห็น



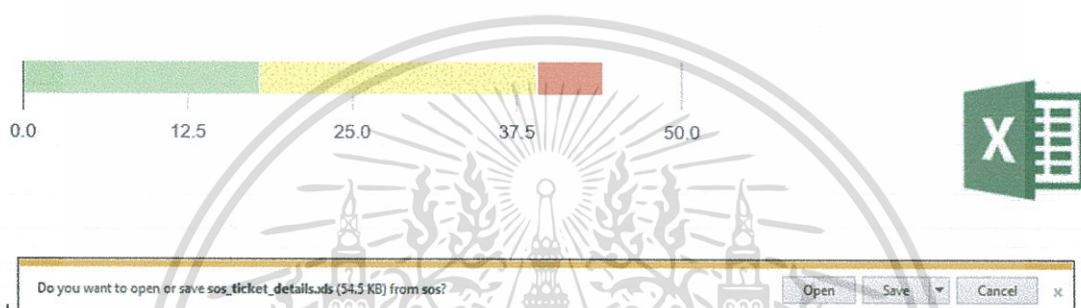
ภาพที่ 4.28 หลังคลิกปุ่ม Test แล้ว หากส่งสำเร็จ จะปรากฏข้อความว่า Success



ภาพที่ 4.29 หลังคลิกปุ่ม Test แล้ว หากส่งไม่สำเร็จ จะปรากฏข้อความว่า Failed

4.1.2.2 หน้ารายงาน (Report)

สคริปต์ที่มีการใช้ในหน้ารายงาน คือ สคริปต์ภาษา ASP .NET ซึ่งช่วยในการแปลงตารางข้อมูลตัวงานทั้งหมดที่แสดงในเวลานั้นเป็นไฟล์สกุล .xls เพื่อใช้เปิดในโปรแกรมไมโครซอฟต์เอ็กเซล ซึ่งมีประโยชน์เมื่อผู้ใช้ต้องการรายละเอียดตัวงานต่าง ๆ ในช่วงเวลาไปใช้อ้างอิงในกรณีอื่น ๆ ต่อไป ภาพที่ 4.30 แสดงให้เห็นกล่องแจ้งเตือนที่ปรากฏขึ้นเมื่อผู้ใช้ทำการดาวน์โหลดตารางข้อมูลตัวงานมาไว้ในเครื่องของตน



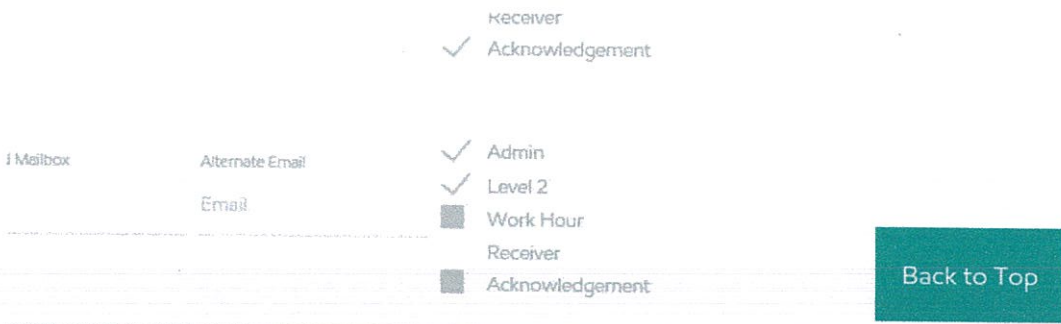
ภาพที่ 4.30 เมื่อคลิกที่ไอคอนไมโครซอฟต์เอ็กเซล จะปรากฏกล่องแจ้งเตือนให้เปิดหรือบันทึกไฟล์

4.1.2.3 หน้าตั้งค่า (Setting)

ในเว็บไซต์เอสโอเอสแบบเก่า นั้น หน้าตั้งค่าเป็นหน้าที่ไม่สามารถใช้งานใด ๆ ได้เลย ทำได้เพียงแสดงรายชื่อของพนักงานออนคอลในแต่ละทีมเท่านั้น การพัฒนาหน้าตั้งค่าให้สามารถใช้งานได้ตามวัตถุประสงค์จึงเป็นหนึ่งในปัจจัยสำคัญในงานวิจัยชิ้นนี้

ในส่วนของฟังก์ชันต่าง ๆ ของเว็บไซต์ เช่น การเพิ่ม-ถอดรายชื่อพนักงานออนคอล หรือการเปลี่ยนแปลงสิทธิ์ต่าง ๆ ของพนักงานออนคอลแต่ละคนนั้น ถูกมอบหมายให้พนักงานอีกท่านที่เป็นนักพัฒนาของบริษัท เอ็กซอนโมบิล จำกัด รับผิดชอบการพัฒนาสคริปต์สำหรับหน้าตั้งค่าให้ใช้งานได้ตามวัตถุประสงค์

สำหรับสคริปต์เพิ่มเติมที่มีการใช้ในหน้านี้นั้น คือปุ่ม Back to top ที่จะปรากฏบนหน้าจอเมื่อมีการเลื่อนหน้าจอลงมาถึงระดับหนึ่งเท่านั้นดังแสดงให้เห็นในภาพที่ 4.31 เมื่อผู้ใช้คลิกที่ปุ่มนี้แล้ว จะเป็นการเลื่อนหน้าจอขึ้นไปยังส่วนบนสุดในทันที



ภาพที่ 4.31 ปุ่ม Back to top ที่จะปรากฏเมื่อเลื่อนหน้าจอลงมาระดับหนึ่งเท่านั้น

4.2 การพัฒนาในส่วนของ back-end

เนื่องจากระบบเอสโอเอสนั้นเป็นระบบที่ค่อนข้างมีเสถียรภาพคืออยู่แล้ว การพัฒนาในส่วนของ back-end จึงเป็นเพียงการเพิ่มเติมระบบเพียงเล็กน้อยเท่านั้น โดยการพัฒนาในส่วนของ back-end จะใช้โปรแกรมไมโครซอฟต์วิซวลสตูดิโอเป็นหลักเช่นเดียวกับในส่วนของ front-end ซึ่งสามารถแบ่งย่อยได้ 3 หัวข้อ ดังต่อไปนี้

4.2.1 การเพิ่มเติมฟังก์ชันในส่วนต่อประสานโปรแกรมประยุกต์ (Application Programming Interface: API)

ดังที่กล่าวไว้ในบทที่ 2 ว่า การทำงานของระบบ front-end ของเอสโอเอสนั้น จะเป็นการระบุตัวชี้แหล่งในอินเทอร์เน็ต (Uniform Resource Locator: URL) ของส่วนต่อประสานโปรแกรมประยุกต์ที่ไปยังส่วนต่าง ๆ ของ back-end ดังนั้น การเพิ่มเติมรายละเอียดต่าง ๆ ใน front-end ในบางส่วนที่ส่วนต่อประสานโปรแกรมประยุกต์ไม่ได้เข้ามาให้ จึงต้องทำการเพิ่มเติมโค้ดคำสั่งในส่วนต่อประสานโปรแกรมประยุกต์ขึ้นมาใหม่ ซึ่งสามารถแยกย่อยตามไฟล์ได้ ดังนี้

4.2.1.1 การเพิ่มเติมฟังก์ชันใน ITSMController.cs

ไฟล์ ITSMController.cs คือส่วนต่อประสานโปรแกรมประยุกต์ที่มีไว้ควบคุมข้อมูลเบื้องต้นของผู้ใช้ซึ่งจะทำการดึงมาจากแอคทีฟไดเรกทอรี (Active Directory) ซึ่งเดิมแล้วจะเป็นการดึงมาเพียงชื่อผู้ใช้เพื่อทำการแสดงในส่วนของหน้าซัมมิทพอร์ทอล ภาพที่ 4.32 แสดงให้เห็นถึงชื่อผู้ใช้ที่ถูกส่วนต่อประสานโปรแกรมประยุกต์ดึงมาใช้งาน เพื่อแสดงผลในหน้าเว็บไซต์



Customer

Fikkree Jehsamoh

ภาพที่ 4.32 ชื่อผู้ใช้ที่ใช้ส่วนต่อประสานโปรแกรมประยุกต์ในการดึงมาแสดงผลบนเว็บไซต์

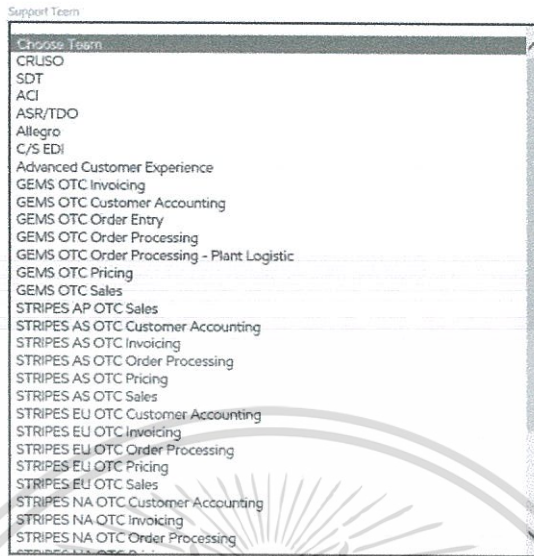
ฟังก์ชันที่ถูกเพิ่มเติมในหน้านี้ คือฟังก์ชันที่มีไว้เพื่อดึงไอดีของผู้ใช้ (Lan ID) ออกมาจากแอคทีฟไดเรกทอรี ซึ่งส่วนต่อประสานโปรแกรมประยุกต์ในส่วนนี้จะถูกใช้ทดสอบการรับ-ส่งอีเมลล์ของระบบเอสโอเอส ดังแสดงให้เห็นในภาพที่ 4.33



ภาพที่ 4.33 การดึงไอดีผู้ใช้ออกมาเพื่อใช้ในการทดสอบการรับ-ส่งอีเมลล์

4.2.1.2 การเพิ่มเติมฟังก์ชันใน SOSTicketsController.cs

ในไฟล์ SOSTicketsController.cs นั้น จะเป็นส่วนที่ไว้ควบคุมตัวงานที่ถูกส่งมาจากหน้าซบมีทพอร์ทอลเท่านั้น ซึ่งจะทำงานโดยการรับค่ามาจากลิสต์รายการที่ผู้ใช้ทำการเลือกตั้งแสดงให้เห็นในภาพที่ 4.34 จากนั้นเมื่อผู้ใช้คลิกที่ปุ่ม Submit เว็บไซต์จะส่งค่าไปยังส่วนต่อประสานโปรแกรมประยุกต์ด้วยคำสั่ง HTTP Post ตามที่เขียนไว้ จากนั้นส่วนต่อประสานโปรแกรมประยุกต์จะทำการประมวลผลแล้วส่งอีเมลล์เอสโอเอสไปยังผู้ใช้ต่อไป



ภาพที่ 4.34 ลิสต์รายการผลิตภัณฑ์ที่ผู้ใช้สามารถเลือกได้

การสร้างตัวงานในหน้าซัพพอร์ตทอลของเว็บไซต์เอสโอเอสโดยตรงนั้น จะแตกต่างจากตัวงานที่ถูกสร้างจากระบบโอทีเอสเอ็ม โดยตัวงานของระบบโอทีเอสเอ็ม นั้นจะสามารถติดตามสถานะและตรวจสอบรายละเอียดจากในเว็บไซต์ของโอทีเอสเอ็ม ได้ และจะมีหมายเลขตัวงานขึ้นต้นด้วย INC แล้วตามด้วยเลข 12 หลัก เช่น INC000001234567 แต่ตัวงานของระบบเอสโอเอสนั้นจะเป็นตัวงานชั่วคราวเท่านั้น ซึ่งไม่สามารถติดตามสถานะได้ สามารถตรวจสอบรายละเอียดได้จาก 3 ทาง คือ จาก อีเมลล์ที่ส่งไปยังพนักงานออนคอล จากฐานข้อมูล และจากหน้ารายงาน ไม่สามารถติดตามสถานะได้ และจะมีหมายเลขตัวงานขึ้นต้นด้วย SOS แล้วตามด้วยเลข 12 หลัก เช่น SOS000000000123 ความแตกต่างนี้สามารถแสดงให้เห็นได้ในภาพที่ 4.35

FirstName	minutes wait	IncidentNumber	Name	Summary
	2	INC000009715766	Site Data Transfer	The service: "ExxonMobil Sdt Engine - ...
	14	SOS000000001318	CRUSO	Impact: NZ CRUSO Region: AP new zealand Details:...

ภาพที่ 4.35 ตัวอย่างความแตกต่างของตัวงาน 2 ประเภทจากฐานข้อมูล

การที่ส่วนต่อประสานโปรแกรมประยุกต์จะทำการประมวลผลค่าที่ได้รับมาได้ จำเป็นต้องเขียนโค้ดคำสั่งในส่วนของส่วนต่อประสานโปรแกรมประยุกต์ให้รองรับกับผลิตภัณฑ์ในลิสต์รายการเสียก่อน เมื่อมีผลิตภัณฑ์ใหม่ถูกเพิ่มเข้าไปในลิสต์รายการ ก็ต้องไปทำการเพิ่มโค้ดคำสั่งในส่วนของส่วนต่อประสานโปรแกรมประยุกต์ด้วย จึงจะสามารถใช้งานได้ตามปกติ

4.2.1.3 การเพิ่มเติมฟังก์ชันใน TicketsController.cs

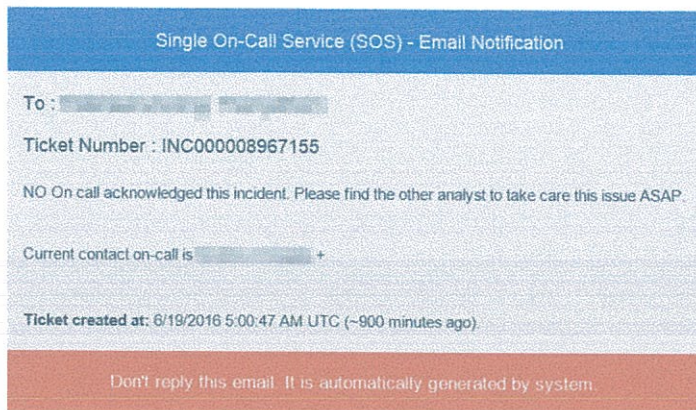
TicketsController.cs จะเป็นส่วนต่อประสานโปรแกรมประยุกต์ที่ใช้ในการดึงรายละเอียดต่าง ๆ ของตั๋วงานมาจากฐานข้อมูล แล้วนำไปแสดงผลในหน้าเว็บไซต์ การสร้างตารางแสดงรายละเอียดตั๋วงานในหน้ารายงานเองก็เป็นส่วนหนึ่งที่ใช้ส่วนต่อประสานโปรแกรมประยุกต์ตัวนี้ ซึ่งในตอนแรกนั้น สิ่งที่สามารถดึงมาจากฐานข้อมูลได้มีเพียงชื่อทีมที่เกี่ยวข้อง ชื่อผลิตภัณฑ์ที่เกี่ยวข้อง และเวลาที่ตั๋วงานถูกสร้างขึ้นเท่านั้น ซึ่งข้อมูลเพียงสามอย่างไม่เพียงพอต่อการแสดงผลรายละเอียดตั๋วงาน จึงจำเป็นต้องทำการเขียนโค้ดคำสั่งเพิ่มเติมเพื่อดึงข้อมูลอื่น ๆ ของตั๋วงาน เช่น ชื่อของผู้ใช้ ชื่อพนักงานออนไลน์ ฯลฯ ออกมาจากฐานข้อมูลเพิ่มจากเดิม

4.2.2 การเพิ่มเติมฟังก์ชันในส่วนควบคุมอีเมลล์ (SOS Mailer)

ส่วนควบคุมอีเมลล์ (SOS Mailer) นั้น เป็นส่วนที่ใช้ควบคุมการรับ-ส่งอีเมลล์ทุกประเภท รวมไปถึงรายละเอียดเนื้อหาของอีเมลล์ด้วย โดยจะมีไฟล์โค้ดคำสั่งหลัก 2 ไฟล์ด้วยกัน ได้แก่ OutboundController.cs ซึ่งเป็นส่วนที่ไว้คอยควบคุมรายละเอียดต่าง ๆ ของอีเมลล์ และ SOSMailer.cs ซึ่งเป็นส่วนที่ไว้คอยควบคุมการรับ-ส่งอีเมลล์ โดยสามารถแจกแจงรายละเอียดการเพิ่มเติมฟังก์ชันได้ ดังนี้

4.2.2.1 การเพิ่มเติมฟังก์ชันใน OutboundController.cs

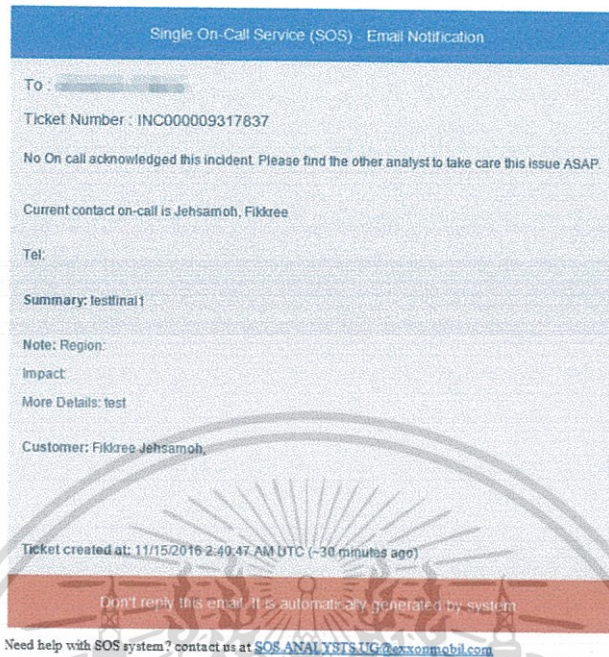
OutboundController.cs คือส่วนที่ไว้ควบคุมรายละเอียดของอีเมลล์ คือ เนื้อหาของอีเมลล์ที่ถูกส่งไปหาทั้งผู้ใช้และพนักงานออนไลน์



Need help with SOS system? contact us at SOS_ANALYSTS.UG@exxonmobil.com

ภาพที่ 4.36 เนื้อหาของอีเมลที่ถูกยกระดับ ก่อนที่จะมีการเพิ่มรายละเอียด

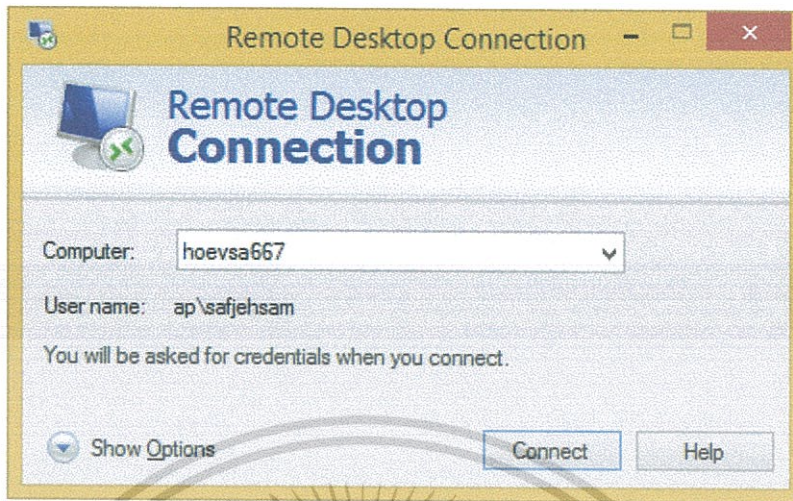
จากภาพที่ 4.36 จะเห็นได้ว่า อีเมลที่ถูกยกระดับไปหาหัวหน้างานนั้น ไม่มีรายละเอียดของตัวงานใด ๆ ระบุอยู่เลย การพัฒนาในส่วนนี้คือ การเพิ่มรายละเอียดของตัวงานลงไปในอีเมลที่ถูกยกระดับ เพื่อให้หัวหน้างานสามารถทราบถึงรายละเอียดของตัวงานดังกล่าวได้ ซึ่งเนื้อหาอีเมลที่ถูกเพิ่มรายละเอียดแล้วจะแสดงให้เห็นได้ในภาพที่ 4.37



ภาพที่ 4.37 เนื้อหาของอีเมลล์ระดับ หลังมีการเพิ่มรายละเอียดลงไปแล้ว

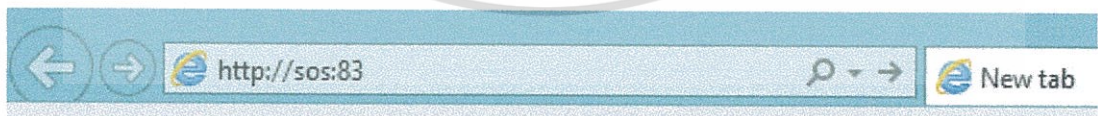
4.2.3 การติดตั้งสภาพแวดล้อมสำหรับการพัฒนา (Developer environment) ขึ้นใหม่

ในปกติแล้ว การพัฒนาเว็บไซต์เอสโอเอสในส่วนของ back-end จะใช้สภาพแวดล้อมสำหรับการพัฒนาเป็นหลัก ไม่เหมือนกับการพัฒนาในส่วน front-end ที่สามารถใช้คอมพิวเตอร์ส่วนตัวพัฒนาได้โดยตรง โดยการเข้าสู่เซิร์ฟเวอร์สำหรับการพัฒนานั้น จะทำการเชื่อมต่อเข้าไปด้วยไอดีสิทธิ์พิเศษ (Privilege ID) ผ่านโปรแกรมรีโมทเดสก์ท็อปคอนเนคชัน (Remote Desktop Connection) ดังแสดงให้เห็นในภาพที่ 4.38



ภาพที่ 4.38 การเข้าสู่เซิร์ฟเวอร์ที่ใช้สำหรับการพัฒนาด้วยไอทีสิทธิ์พิเศษ

ทว่า ระหว่างการพัฒนานั้น เซิร์ฟเวอร์ที่ใช้สำหรับการพัฒนาเกิดปัญหาขึ้น ส่งผลให้ทุกอย่างที่ถูกเก็บไว้อยู่บนเซิร์ฟเวอร์ดังกล่าวถูกลบไป ทำให้ต้องทำการติดตั้งสภาพแวดล้อมสำหรับการพัฒนาขึ้นมาใหม่เพื่อทดแทนเซิร์ฟเวอร์เดิมที่เกิดปัญหาขึ้นจนไม่สามารถใช้งานได้ ซึ่งจากการปรึกษาของพนักงานในทีมรีเทลทั้งหมดแล้ว ได้ข้อสรุปว่า จะทำการติดตั้งจะทำการติดตั้งสภาพแวดล้อมสำหรับการพัฒนาบนเซิร์ฟเวอร์เดียวกับเซิร์ฟเวอร์ผลิตภัณฑ์ (Production Server) โดยตรง แต่จะทำการแยกโฟลเดอร์ (folder) ที่เก็บไว้ และแยกที่อยู่ของส่วนต่อประสานโปรแกรมประยุกต์ตามพอร์ต (port) โดยจะสามารถเรียกใช้งานเว็บไซต์เอสไอเอสสำหรับการพัฒนาได้โดยการระบุพอร์ตลงในตัวชี้แหล่งในอินเทอร์เน็ตโดยตรง ดังแสดงให้เห็นในภาพที่ 4.39



ภาพที่ 4.39 การระบุพอร์ตในตัวชี้แหล่งในอินเทอร์เน็ตเพื่อเข้าสู่เว็บไซต์เอสไอเอสสำหรับการพัฒนา

4.3 การสนับสนุนแอปพลิเคชันในส่วนของฐานข้อมูล

ข้อมูลทั้งหมดของแอปพลิเคชันเอสไอเอสนั้นถูกเก็บไว้ในฐานข้อมูล ดังนั้นในการพัฒนาจึงต้องเข้าไปคอยจัดการเกี่ยวกับฐานข้อมูลตลอดเวลา โดยการสนับสนุนในส่วนนี้นั้นจะใช้โปรแกรมไมโครซอฟต์เอสคิวแอลเซิร์ฟเวอร์เมนเนจเมนต์ (Microsoft SQL Server Management) เป็นหลัก สัญลักษณ์และหน้าต่างของโปรแกรมไมโครซอฟต์เอสคิวแอลเซิร์ฟเวอร์เมนเนจเมนต์สามารถแสดงให้เห็นได้ในภาพที่ 4.40 และ 4.41

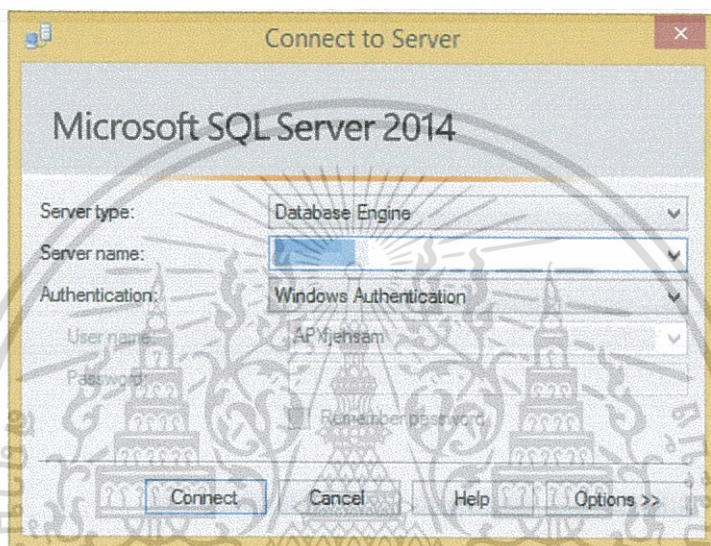


ภาพที่ 4.40 สัญลักษณ์ของโปรแกรมไมโครซอฟต์เอสคิวแอลเซิร์ฟเวอร์เมนเนจเมนต์



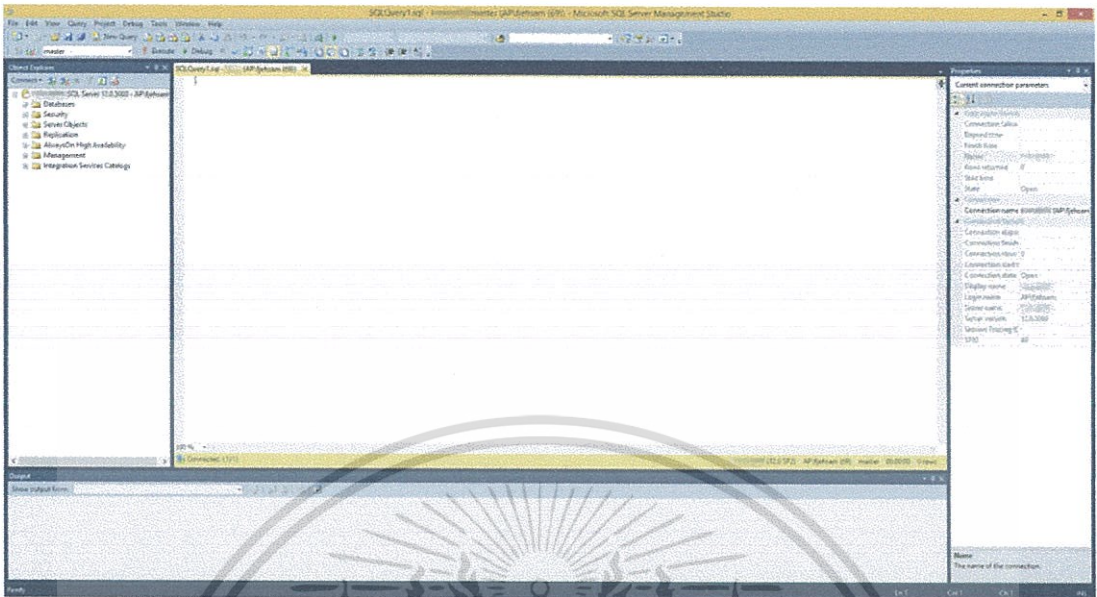
ภาพที่ 4.41 หน้าตาภายนอกของโปรแกรมไมโครซอฟต์เอสคิวแอลเซิร์ฟเวอร์เมนเนจเมนต์

เมื่อเข้าสู่โปรแกรมไมโครซอฟต์เอสคิวแอลเซิร์ฟเวอร์เมนเนจเม้นท์ จะปรากฏหน้าต่างการเชื่อมต่อเข้าสู่ฐานข้อมูลให้เลือก ดังแสดงให้เห็นในภาพที่ 4.42 สำหรับแอปพลิเคชันเอสโอเอสเอ็นี่จะใช้ฐานข้อมูล 2 แหล่งด้วยกัน คือฐานข้อมูลหลักที่ใช้สำหรับแอปพลิเคชันจริง และฐานข้อมูลที่ใช้สำหรับพัฒนาและทดสอบระบบเท่านั้น



ภาพที่ 4.42 หน้าต่างแสดงการเชื่อมต่อเข้าสู่ฐานข้อมูล

เมื่อทำการเชื่อมต่อเข้าสู่ฐานข้อมูลแล้ว (ทั้งฐานข้อมูลของแอปพลิเคชันจริงและฐานข้อมูลที่ใช้สำหรับการพัฒนา) ต่อมาคือการสร้างไฟล์ที่ใช้ในการคิวรี (query) ข้อมูล โดยการคลิกที่ปุ่ม New Query หรือกดปุ่ม Ctrl+N บนคีย์บอร์ด จากนั้นจะปรากฏหน้าต่างสำหรับการเขียนสคริปต์เอสคิวแอลที่ใช้ในการคิวรีข้อมูล ดังภาพที่ 4.43



ภาพที่ 4.43 เมื่อคลิก New Query แล้ว จะปรากฏหน้าต่างที่ใช้ในการควิรี่ขึ้นมา

การสนับสนุนแอปพลิเคชันเอสโอเอสในส่วนฐานข้อมูลนั้น มีทั้งการใช้คำสั่งควิรี่ข้อมูลเพื่อดูรายละเอียดต่าง ๆ ในฐานข้อมูล การเพิ่มหรือลบข้อมูลออกจากฐานข้อมูล และการแก้ไขข้อมูล ซึ่งโดยปกติแล้ว อดีผู้ใช้ทั่วไปนั้นจะไม่มีสิทธิ์ในการแก้ไข เพิ่มหรือลบข้อมูลจากฐานข้อมูลของแอปพลิเคชันจริง การจะแก้ไขข้อมูลในฐานข้อมูลดังกล่าวนั้น จำเป็นต้องเข้าสู่เซิร์ฟเวอร์ผลิตภัณฑ์ด้วยไอดีสิทธิ์พิเศษที่เรียกว่า ไฟร์คอลไอดี (Firecall ID) จากนั้นทำการเปิดโปรแกรมไมโครซอฟต์เอสคิวแอลเซิร์ฟเวอร์เมนเนจเมนท์จากในเซิร์ฟเวอร์ผลิตภัณฑ์ จึงจะมีสิทธิ์ในการแก้ไข เพิ่มและลบข้อมูลจากฐานข้อมูลดังกล่าวได้

4.4 การสนับสนุนผู้ใช้เมื่อมีทีมอื่นมาติดต่อขอใช้งานแอปพลิเคชัน

แนวคิดระบบการทำงานของแอปพลิเคชันเอสโอเอสนั้น สามารถนำไปใช้ได้กับทุกทีมที่มีพนักงานออนไลน์อยู่ ไม่จำกัดว่าจะต้องเป็นทีมรีเทลเท่านั้นที่สามารถใช้งานได้ ดังนั้นจึงเป็นเรื่องปกติที่จะมีทีมอื่นที่มีพนักงานออนไลน์ติดต่อมาเพื่อขอใช้งานแอปพลิเคชันเอสโอเอส ซึ่งในฐานะผู้พัฒนาแล้ว การให้การสนับสนุนผู้ใช้ที่เป็นทีมอื่นให้ดีที่สุดย่อมเป็นเรื่องปกติที่ต้องทำ ภาพที่ 4.44 แสดงให้เห็นถึงอีเมลที่ทีมอื่นติดต่อเข้ามาเพื่อขอใช้งานแอปพลิเคชัน

Reply Reply All Forward IM
Thu 8/18/2016 8:16 PM

new app to be included to SOS

To [redacted] Jehsanoh, Fikree, C; [redacted]

Cc [redacted]

Retention Policy All Other 13 Months (1 year, 1 month)

Expires 9/17/2017

You replied to this message on 8/19/2016 10:54 AM.

Hello – can one of you please help me including SWIFT to the SOS application. I'd like to use the on-call schedule function you made it available. Thanks

RUN WebMethods Supervisor
Applications – Technical Services – Application Integration & Scheduling

WEBMETHODS HOTLINE: [redacted]

ExxonMobil IT Global Services Company
Rua Visconde de Nacar, 1440 – Ed. Século XXI 8ºandar - WS822E
Curitiba, Paraná – 80410-201
Tel [redacted]
Mobile [redacted]

ภาพที่ 4.44 แสดงอีเมลของทีมอื่นที่ติดต่อมาเพื่อขอใช้งานแอปพลิเคชันเอสโอเอส

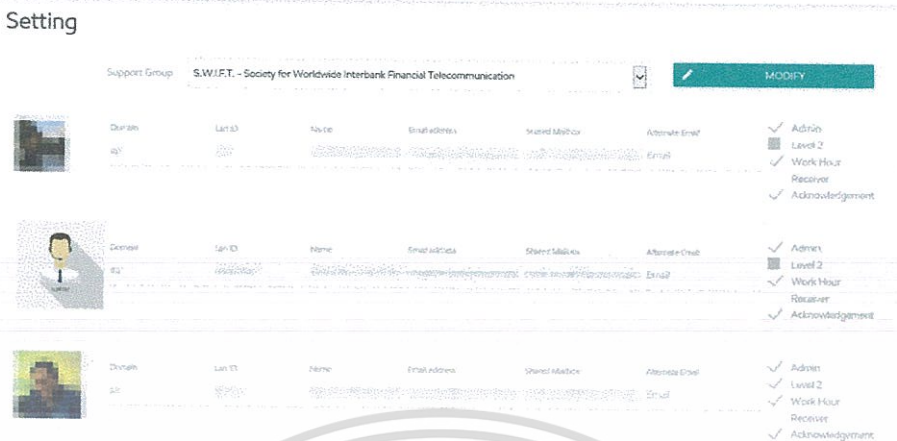
ขั้นตอนแรกคือการเพิ่มทีมที่ติดต่อเข้าไปในฐานข้อมูล เพื่อให้ระบบเอสโอเอสสามารถตรวจจับได้เมื่อพบตัวงานที่เกี่ยวข้องกับทีมดังกล่าว โดยการเขียนคำสั่งเอสคิวแอลสคริปต์เพื่อเพิ่มทีมใหม่ลงในฐานข้อมูล ตามภาพที่ 4.45

INSERT INTO SOS..Groups(GroupID, Name, Urgency) VALUES (7, 'S.W.I.F.T. - Society for Worldwide Interbank Financial Telecommunication', '2-High');

GroupID	Name	Urgency
1	Retail	2-High
2	OTC	2-High
3	Siebel Technical Services	2-High
4	Disbanded	2-High
5	GPM	2-High
6	EDI CS	2-High
7	S.W.I.F.T. - Society for Worldwide Interbank Fin...	2-High

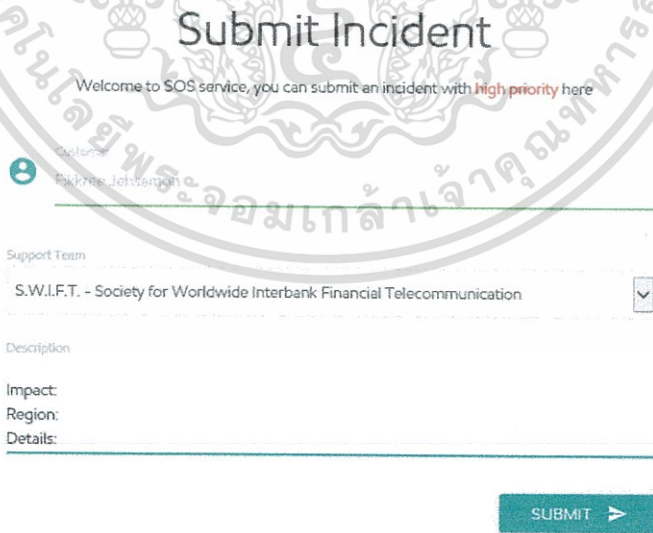
ภาพที่ 4.45 คำสั่งคิวรีที่ใช้ในการเพิ่มทีมใหม่เข้าไปในฐานข้อมูล

เมื่อเพิ่มกลุ่มใหม่ลงในฐานข้อมูลแล้ว ต่อมาคือการเพิ่มรายชื่อพนักงานออนคอลของทีมดังกล่าวลงในระบบ ซึ่งในส่วนนี้สามารถทำได้ทั้งจากการเขียนคำสั่งคิวรีในฐานข้อมูลโดยตรง หรือจะทำการเพิ่มจากในหน้าตั้งค่าก็ได้ ภาพที่ 4.46 แสดงให้เห็นถึงการเพิ่มรายชื่อพนักงานออนคอลของทีมใหม่ผ่านหน้าตั้งค่าของเว็บไซต์เอสโอเอส



ภาพที่ 4.46 การเพิ่มพนักงานออนไลน์ผ่านหน้าตั้งค่าของเว็บไซต์เอสโอเอส

หลังจากนั้นจะเป็นการเพิ่มผลิตภัณฑ์ลงไปในลิสต์รายการในหน้าซัพพอร์ตทอลของเอสโอเอส และเพิ่มโค้ดคำสั่งลงไปในส่วนต่อประสานโปรแกรมประยุกต์ SOSTicketsController.cs เพื่อให้ระบบเอสโอเอสสามารถส่งอีเมลล์ไปหาพนักงานออนไลน์ได้เมื่อมีผู้ใช้สร้างตัวงานที่เกี่ยวกับผลิตภัณฑ์ดังกล่าวขึ้นมา ภาพที่ 4.47 แสดงให้เห็นถึงผลิตภัณฑ์ของทีมใหม่ที่ปรากฏในหน้าซัพพอร์ตทอลหลังจากได้ทำการเพิ่มลงไปในส่วนต่อประสานโปรแกรมประยุกต์แล้ว

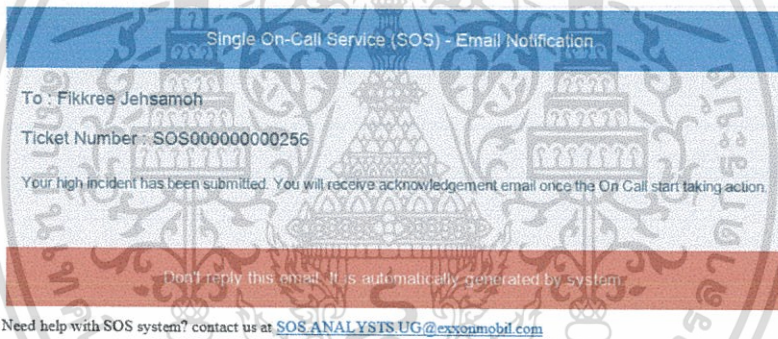


ภาพที่ 4.47 การเพิ่มผลิตภัณฑ์ใหม่ลงไปในลิสต์รายการของหน้าซัพพอร์ตทอล

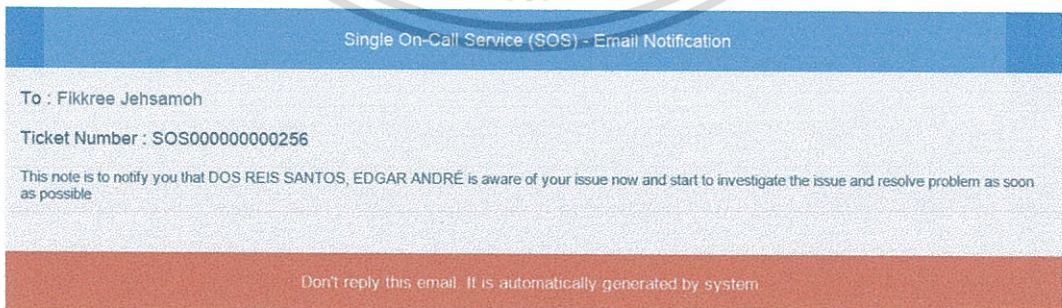
ต่อมาคือการทดสอบระบบกับพนักงานออนไลน์ในทีมดังกล่าว เพื่อเป็นการยืนยันให้มั่นใจว่าระบบเอสโอเอสสามารถทำงานได้ตามวัตถุประสงค์ โดยมีขั้นตอนคือ สร้างตัวงานขึ้นมาในหน้าซบมิทพอร์ทอลและคลิกไปที่ปุ่ม Submit จากนั้นจึงทำการติดต่อกับพนักงานออนไลน์ในทีมดังกล่าวโดยตรงเพื่อตรวจสอบว่าได้รับอีเมลล์เอสโอเอสหรือไม่ ภาพที่ 4.48 แสดงให้เห็นถึงตัวงานที่ใช้ในการทดสอบระบบ ภาพที่ 4.49 และ 4.50 แสดงให้เห็นถึงอีเมลล์ตอบกลับไปยังผู้ใช้ที่ถูกส่งมาหลังจากทำการทดสอบระบบแล้ว

Customer	Oncall	minutes wait	Incident Number	Team	Summary
397 Fikkree Jehsamoh	EDGAR ANDRÉ	11	SOS000000000256	S.W.I.F.T. - Society for Worldwide Interb...	Impact: testing email Region: Details:

ภาพที่ 4.48 ตัวงานที่ใช้ในการทดสอบระบบ



ภาพที่ 4.49 อีเมลล์ตอบกลับมายังผู้ใช้เมื่อทำการทดสอบระบบ



Need help with SOS system? contact us at SOS_ANALYSTS_UG@exxonmobil.com

ภาพที่ 4.50 อีเมลล์ตอบกลับมายังผู้ใช้เมื่อพนักงานออนไลน์ตอบกลับอีเมลล์เอสโอเอสแล้ว

เมื่อระบบทำงานได้ตรงตามวัตถุประสงค์แล้ว ขั้นตอนสุดท้ายคือการแจ้งแก่พนักงานออลทุกคนในทีมดังกล่าวว่าระบบพร้อมใช้งานแล้ว จากนั้นจึงเป็นการสนับสนุนในส่วนอื่น ๆ ต่อไป เช่น การช่วยเหลือเมื่อเกิดปัญหาในการใช้งาน หรือการแก้ไขปัญหาดังกล่าวต่าง ๆ ที่พบเจอและถูกรายงานมา



บทที่ 5

สรุปผลการวิจัยและข้อเสนอแนะ

5.1 สรุปผลการวิจัย

การออกแบบและพัฒนาแอปพลิเคชันซิงเกิลออนคอลเซอร์วิส (Single On-call Service) หรือ เอสโอเอส (SOS) นั้น ส่งผลให้ผู้ใช้งาน ทั้งผู้ใช้งานผ่านเว็บไซต์ และผู้ใช้งานผ่านระบบไอทีเอสเอ็ม สามารถใช้งานได้ง่ายดายและสะดวกขึ้น อีกทั้งยังเพิ่มความรวดเร็วในการใช้งานจากหน้าเว็บไซต์ ตรงตามวัตถุประสงค์ของแอปพลิเคชันที่มุ่งเน้นไปยังการตอบรับเหตุฉุกเฉินที่อาจส่งผลกระทบต่อธุรกิจหลักของบริษัทให้ไวที่สุด และยังสามารถทราบถึงรายละเอียดต่าง ๆ ของตัวงานที่ผ่านมาได้อย่างสะดวก นอกจากนี้ พนักงานออนคอลยังสามารถจัดการระยะเวลาได้รวดเร็ว และสะดวกขึ้น อีกทั้งยังสามารถเพิ่มและถอดชื่อพนักงานออนคอลได้จากหน้าเว็บไซต์โดยตรง ไม่ต้องทำการเพิ่มในฐานข้อมูลด้วยตัวเอง ในส่วนของทางฝั่งหัวหน้างานเองก็สามารถรับรู้ถึงรายละเอียดของตัวงานได้จากอีเมลล์ยกระดับโดยตรง สุดท้ายคือ พนักงานทีมรีเทลที่ทำการสนับสนุนแอปพลิเคชันเอสโอเอสคนอื่น ๆ ก็สามารถทำการพัฒนาได้ง่ายขึ้น เนื่องจากไม่ต้องโยกไปมาระหว่าง 2 เซิร์ฟเวอร์ คือ เซิร์ฟเวอร์ผลิตภัณฑ์ และเซิร์ฟเวอร์พัฒนาอีกต่อไป

5.2 ปัญหาที่พบในระหว่างการดำเนินงานวิจัย

- ขาดความรู้ในด้านภาษาคอมพิวเตอร์ที่ใช้ในการพัฒนา รวมไปถึงความรู้ในด้านเฟรมเวิร์คต่าง ๆ ทำให้ต้องเสียเวลาในการศึกษาเพิ่มเติม
- ปัญหาที่เกิดขึ้นเกี่ยวกับเซิร์ฟเวอร์ได้รับการแก้ไขล่าช้า เนื่องจากผู้ดูแลอาศัยอยู่คนละเขตเวลา
- เมื่อพัฒนาเสร็จแล้ว ผลลัพธ์ที่ได้ไม่ตรงกับที่คาดการณ์ไว้
-

5.3 แนวทางการแก้ไขปัญหา

- ศึกษาหาความรู้เกี่ยวกับภาษาคอมพิวเตอร์ที่เกี่ยวข้องเพิ่มเติม พร้อมทั้งทดลองทำแบบฝึกต่าง ๆ ควบคู่กันไปด้วย

- พยายามตรวจสอบโค้ดอีกครั้ง เพื่อหาจุดที่ทำให้ตรรกะการคำนวณผิดพลาดและทำการแก้ไข
- ปรับปรุงโค้ดที่มีอยู่แล้วโดยการทดลองวิธีใหม่ ๆ เพื่อลดเวลาการทำงานในส่วน front-end ลง
- ติดต่อประสานงานกับผู้ที่เกี่ยวข้องกับเซิร์ฟเวอร์ รวมไปถึงผู้ที่ช่วยทดสอบระบบเป็นระยะ ๆ

5.4 แนวทางการพัฒนาต่อและนำไปใช้

- พัฒนาต่อยอดจากความต้องการเพิ่มเติมของผู้ใช้ เช่น สนับสนุนการใช้งานผ่านโทรศัพท์
- พัฒนาให้หน้าเว็บไซต์ทำงานได้เร็วขึ้น โดยการเปลี่ยนแปลงขั้นตอนวิธี (algorithm) ในการพัฒนาใหม่
- นำไปต่อยอดเป็นการเตือนผ่านโทรศัพท์ แทนการเตือนผ่านอีเมลล์ได้



เอกสารอ้างอิง

[1] ASP.NET. (19 มกราคม 2550) เข้าถึงได้จาก:

<https://en.wikipedia.org/wiki/ASP.NET>

[2] User-centered design. (12 กรกฎาคม 2549) เข้าถึงได้จาก:

https://en.wikipedia.org/wiki/User-centered_design

[3] jQuery. (25 กรกฎาคม 2550) เข้าถึงได้จาก:

<https://en.wikipedia.org/wiki/JQuery>

[4] Microsoft Exchange Server. (9 มิถุนายน 2548) เข้าถึงได้จาก:

https://en.wikipedia.org/wiki/Microsoft_Exchange_Server

[5] Alvin Wang, Alan Chang, Alex Mark, Kevin Louie. Materialize CSS. (25 มิถุนายน 2557)

เข้าถึงได้จาก: <http://materializecss.com/>

[6] Adam Shaw. Fullcalendar.io. (26 เมษายน 2552) เข้าถึงได้จาก:

<https://github.com/fullcalendar/fullcalendar>

[7] Kirill Fakhroutdinov. The Unified Modeling Language. (8 กันยายน 2552) เข้าถึงได้จาก:

<http://www.uml-diagrams.org/>

[8] Agile software development. (2 มีนาคม 2548) เข้าถึงได้จาก:

https://en.wikipedia.org/wiki/Agile_software_development



ภาคผนวก





สหกิจศึกษา ปีการศึกษา 2559

การออกแบบและพัฒนาระบบการเตือนและการตอบรับ ของปัญหาที่ส่งผลกระทบต่อธุรกิจหลักของบริษัท Escalation Process Design and Improvement

ExxonMobil

Energy lives here

จัดทำโดย: นายพิภกร เจ๊ะสาม๊ะ

อาจารย์ที่ปรึกษา: คร.พิภุฑแก้ว กังคิสาณนท์

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

บทคัดย่อ

ในทุก ๆ บริษัท การดูแลธุรกิจหลักให้ดำเนินไปอย่างราบรื่นนับถือเป็นเรื่องที่สำคัญมาก สำหรับบริษัท เช็คซอน โมบิล เมื่อเกิดเหตุการณ์ที่ส่งผลกระทบต่อธุรกิจหลักของบริษัท ผู้ใช้จะทำการสร้าง ticket ขึ้นมาในระบบ IT Service Management เพื่อให้ผู้ที่เกี่ยวข้องทั้งหมดสามารถเข้ามาแก้ไข โดย ticket นั้นก็มีความสำคัญต่างกัน ซึ่ง ticket มีความสำคัญสูงมากเท่าใด พนักงานที่เกี่ยวข้องก็ต้องรีบแก้ไขให้ทันวันถัดมาเท่านั้น ในวันทำงานปกติ พนักงานส่วนมากจะคอยดู ticket อยู่ตลอดเวลา ทว่าในวันหยุดสุดสัปดาห์นั้น พนักงานไม่ได้คอยดู ticket ตลอดเวลาเหมือนในวันทำงาน ดังนั้นหากมี ticket ความสำคัญสูงถูกส่งมาในวันหยุดสุดสัปดาห์ เหตุการณ์ดังกล่าวอาจไม่ได้รับการแก้ไขให้ทันเวลาที่ จึงอาจส่งผลกระทบต่อธุรกิจหลักของบริษัทได้ ดังนั้น ทีมที่ดูแล (Retail) ของบริษัท เช็คซอน โมบิล จึงได้สร้างแพลตฟอร์มขึ้นขึ้นมาหนึ่งตัว คือซิงเกิลออกคอลเซอร์วิส (Single On-call Service) หรือเอสไอเอส (SOS) ซึ่งมีกระบวนการทำงานคือ เมื่อ SOS ตรวจสอบว่ามี ticket ที่มีความสำคัญสูงถูกส่งมาในระบบ SOS จะทำการส่ง Email ไปแจ้งเตือนพนักงานที่เกี่ยวข้องกับ ticket ดังกล่าว ซึ่งพนักงานคนเหล่านี้ว่า พนักงานออนไลน์ (on-call) โดยจะคอยส่ง Email ขึ้นไปเรื่อยๆ จนกว่าพนักงานออนไลน์จะทำการตอบ Email ดังกล่าว ซึ่งแพลตฟอร์มตัวนี้จะทำให้พนักงานออนไลน์สามารถทราบ และจัดการเหตุการณ์ได้ในทันทีแม้จะเป็นวันหยุดก็ตาม รวมถึงยังมีแจ้งเตือนการใส่แผนการล่วงหน้าดังกล่าวให้มีประสิทธิภาพมากขึ้น จึงจะส่งผลให้ประสิทธิภาพการแจ้งเตือนดีขึ้นไม่ให้เกิดเหตุการณ์งานออนไลน์

ตัวอย่างผลงาน

The image displays several screenshots of the SOS (Single On-call Service) system. It shows a dashboard with a grid of tickets, a detailed view of a specific ticket with its history and assigned staff, and a notification email template. The interface is clean and professional, using a blue and white color scheme.

โครงสร้างแอปพลิเคชัน

The diagram illustrates the system architecture. It starts with a user submitting a ticket through a 'SOS Submit Portal'. This data is processed by 'SOS Mailer' and 'SOS Mail' components, which interact with a 'SOS Entity' database. The system then triggers an 'SOS Email' notification, which is sent back to the user via an 'SOS Email' service.

แพลตฟอร์ม SOS ถูกควบคุมโดยคอนโทรลเลอร์ต่างๆ ที่ถูกเขียนขึ้นจากภาษา C# ด้วยเครื่องมือ ASP.NET ซึ่งจะแบ่งการทำงานออกเป็นส่วน ๆ ดังต่อไปนี้

- > SOS Mailer ทำหน้าที่ควบคุมการรับ - ส่ง Email ทั้งหมด รวมไปถึงเนื้อหาที่แสดงใน Email ที่ส่งไปด้วย
- > SOS Entity ทำหน้าที่คอยจัดการส่วนที่เกี่ยวกับ database ทั้งหมด เช่น การบันทึก ticket หรือการเก็บรายชื่อของพนักงานออนไลน์
- > ส่วนประกอบอื่นๆ อีกมากมาย

การทำงานของระบบ

```

graph TD
    Start([User submit new ticket]) --> Mailer[SOS Mailer]
    Mailer --> Mail[SOS Mail]
    Mail --> Entity[SOS Entity]
    Entity --> Email[SOS Email]
    Email --> User([User receive SOS email])
  
```

The flowchart details the process: 1. User submits a new ticket. 2. The SOS Mailer processes the ticket. 3. The SOS Mail component handles the email generation. 4. The SOS Entity database stores the ticket information. 5. The SOS Email component sends the notification. 6. The user receives the SOS email.

การทำงานของแพลตฟอร์ม SOS จะเริ่มเมื่อผู้ใช้ทำการสร้าง ticket ขึ้นมาแล้วส่งไปในระบบ IT Service Management หรืออาจสร้างโดยตรงผ่าน SOS Submit Portal ก็ได้ จากนั้นระบบจะทำการบันทึก ticket ลงไปใน database แล้วทำการส่ง Email แจ้งผู้ใช้ หรือส่ง Email ไปแจ้งเตือนพนักงานออนไลน์เกี่ยวกับ ticket จากนั้นระบบจะเริ่มทำการประมวลผล และจะส่ง Email ไปแจ้งเตือนพนักงานออนไลน์ต่อไปเรื่อยๆ ทุก ๆ 5 นาที หากถึงเวลา 30 นาทีแล้วพนักงานออนไลน์ยังไม่ทำการตอบ Email ของ SOS ระบบจะทำการส่ง Email ไปยังหัวหน้างานเพื่อแจ้งว่าพนักงานออนไลน์ไม่มีการตอบรับ Email เมื่อพนักงานออนไลน์ทำการตอบ Email แล้ว ระบบจะทำการส่ง Email แจ้งไปยังผู้ใช้อีกครั้งว่าพนักงานออนไลน์ทำการรับทราบเกี่ยวกับ ticket แล้ว

กิตติกรรมประกาศ

ขอขอบคุณบริษัท เช็คซอน โมบิล จำกัด รวมไปถึงทีมงานในทีมรีเทล (Retail) ของบริษัทส่งตรงสู่ทุกท่าน ที่ได้ให้ความอนุเคราะห์ในการเข้าร่วมโครงการนักศึกษาฝึกงานของบริษัท และโครงการสหกิจศึกษา ประจำปี.ศ. 2559 และขอขอบคุณอาจารย์ประจำสาขาวิชาวิศวกรรมสารสนเทศ ภาควิชาวิศวกรรมคอมพิวเตอร์ที่ให้ความช่วยเหลือ ตลอดจนคำแนะนำและคำปรึกษาตลอดการดำเนินงานวิจัยสำเร็จจะไปได้ด้วยดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้