



รายงานสหกิจศึกษาฉบับสมบูรณ์

ระบบจัดการงานอัตโนมัติ : การตั้งเวลาอัตโนมัติบนสภาพแวดล้อมพัฒนาและ
สภาพแวดล้อมทดสอบ

Job automation services: Automate development and acceptance
schedule

นายศุภวิชญ์ แสงสุวรรณ

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2562

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อโครงการสหกิจศึกษา	ระบบจัดการงานอัตโนมัติ : การตั้งเวลาอัตโนมัติบนสภาพแวดล้อมพัฒนาและสภาพแวดล้อมทดสอบ	
ชื่อ - สกุล นักศึกษา	นายศุภวิชญ์ แสงสุวรรณ	
คณะ	วิศวกรรมศาสตร์	ภาควิชา วิศวกรรมคอมพิวเตอร์
ชื่อ - สกุล อาจารย์นิเทศ	ผศ.บัณฑิต พัสยา	
ชื่อ - สกุล ผู้นิเทศงาน	นายพีรพันธุ์ พยัคฆมาศ	
สถานประกอบการ	บริษัท เอ็กซ์อนโมบิล คอร์ปอเรชั่น	

บทคัดย่อ

โครงการเล็มนีมีวัตถุประสงค์ที่จะพัฒนาระบบตั้งเวลาใน development และ acceptance ของ Job Automation Services (JAS) ให้สามารถทำงานโดยอัตโนมัติได้ เพื่อลดการใช้ทรัพยากรของทีม Automation Engine ไม่ว่าจะทั้งเวลา และแรงงานในการทำงานประเภทนี้ลง รวมถึงยังช่วยประหยัดเวลาให้กับผู้ใช้งาน

ทางผู้จัดทำจึงได้นำฟังก์ชัน DEV/ACC schedule request ซึ่งมีอยู่แล้วในระบบ มาพัฒนาเพิ่มเติมด้วยการสอบถามปัญหาและความต้องการของผู้ใช้งาน และมีการจัดลำดับความสำคัญของงานให้มีการส่งมอบชิ้นงานที่สร้างผลประโยชน์ให้กับบริษัทได้ก่อน เพื่อให้การพัฒนาโครงการนี้เกิดประโยชน์สูงสุด โดยพัฒนาระบบหลังบ้านบนแพลตฟอร์ม MuleSoft และเว็บไซต์ด้วย Angular

คำสำคัญ :

Job Automation Services (JAS) แอปพลิเคชันสำหรับใช้งาน Automation Engine
Automation Engine (AE) ระบบจัดการตารางเวลา
JSF ตัวเลขระบบข้อมูลใน Automation Engine และ Job Automation Services
ITSM ระบบเก็บ คำร้อง (ticket) กลางของบริษัทเอ็กซ์อนโมบิล เพื่อตรวจสอบการทำงานภายในบริษัท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Co-operative Title	Job automation services: Automate development and acceptance schedule	
Student Intern Name	Mr. Supavitch Sangsuwan	
Faculty	Engineering	Department Computer Engineering
Advisor Name	Asst. Bundit Passaya	
Mentor Name	Mr. Peerapan payakamas	
Company	ExxonMobil Corp.	

ABSTRACT

The purpose of this project is to automate feature development and acceptance schedule request of Job Automation Services (JAS). To reduce the use of resources of the Automation Engine team, whether time or human resources for this type of work and also saving time for our customers.

The producer has therefore introduced the DEV / ACC schedule request function which is already present in the system. Come to develop further by getter problems and requirement of users. And prioritize work to deliver work that have value to the company first. This application implement back end on the MuleSoft platform and website with Angular

Keywords :

Job Automation Services (JAS) application that let customer can use Automation Engine
Automation Engine (AE) scheduling server

JSF is number for identify information in Automation Engine and Job Automation Services

ITSM is a system for manage ticket in ExxonMobil

กิตติกรรมประกาศ

โครงการเล่มนี้เสร็จสมบูรณ์ได้ด้วยความช่วยเหลือจากหลายบุคคลทั้งจากทางตรงและทางอ้อม ซึ่งโครงการเล่มนี้จะสำเร็จไม่ได้หากปราศจากความช่วยเหลือจากบุคคลเหล่านี้

ขอขอบคุณอาจารย์ผู้นิเทศงาน ผศ.บัญญัติ พัสยา ซึ่งเป็นผู้มานิเทศงาน ที่ให้โอกาสทำสหกิจ ศึกษา และเข้ามานิเทศงานถึงบริษัท คอยตรวจเช็คและให้คำแนะนำ ชี้จุดบกพร่องของโครงการ ช่วยให้ โครงการงาน สมบูรณ์ยิ่งขึ้น

ขอขอบคุณ อาจารย์จระศักดิ์ สิทธิกร ซึ่งเป็นผู้ให้คำแนะนำและให้โอกาสในการเข้ามาทำโครงการ สหกิจนี้ ช่วยให้มีโอกาสในการได้ทำโครงการเล่มนี้จนสำเร็จอย่างมีคุณภาพ

ขอขอบคุณ นายพีรพันธุ์ พยัคฆมาศผู้ดูแลและคอยให้คำปรึกษา และเป็นผู้มอบหมายหน้าที่ในการ พัฒนาตนเองในครั้งนี้

ขอขอบคุณบุคลากรทีม Automation Engine ทุกท่านที่คอยให้คำปรึกษา ข้อเสนอแนะในการทำงานตลอด ระยะเวลาที่ได้ร่วมงานกันมา

สุดท้ายนี้ขอขอบคุณ บิดา มารดา ครอบครัว ญาติ พี่น้อง เพื่อน ๆ ตลอดจนผู้เกี่ยวข้องที่ไม่ได้กล่าวถึงทุกท่านที่เป็นกำลังใจ เป็นแรงผลักดัน ให้การสนับสนุน และความช่วยเหลือในการทำโครงการครั้งนี้จนสำเร็จลุล่วงไปได้ด้วยดี

นายศุภวิชญ์ แสงสุวรรณ

สารบัญ

หน้า

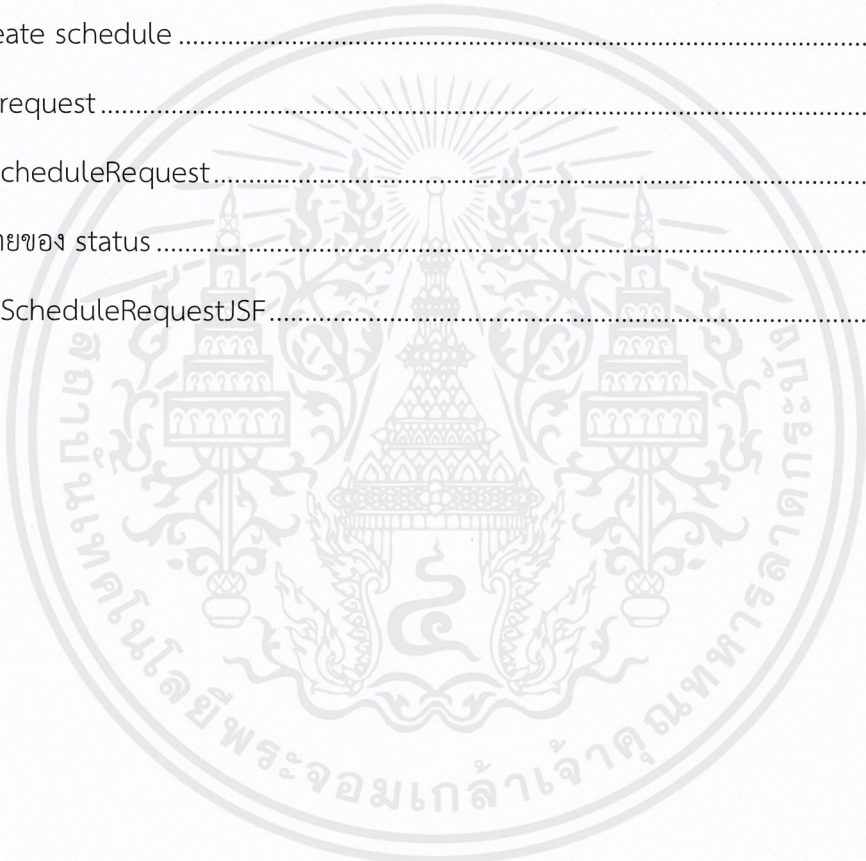
บทคัดย่อ.....	I
ABSTRACT.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VI
สารบัญภาพ.....	VII
บทที่ 1 บทนำ.....	1
1.1 ที่มาและความสำคัญ.....	1
1.2 วัตถุประสงค์ของการทำวิจัย.....	1
1.3 ขอบเขตของการวิจัย.....	1
1.4 วิธีดำเนินการ.....	2
1.5 ประโยชน์ที่คาดว่าจะได้รับ.....	2
บทที่ 2 แนวคิดและทฤษฎีที่เกี่ยวข้อง.....	3
2.1 ทฤษฎีที่เกี่ยวข้อง.....	3
2.1.1 Automation Engine.....	3
2.1.2 Anypoint Platform.....	5
2.1.3 Job Automation Services.....	6
2.1.4 Agile Methodology.....	6
2.1.5 RESTful Web Service.....	7
2.1.6 Git version control.....	9
2.1.7 SQL.....	14
2.1.8 Angular.....	15
บทที่ 3 วิธีการดำเนินงาน.....	16
3.1 สถาปัตยกรรมของระบบ.....	16

สารบัญ (ต่อ)

	หน้า
3.1.1 เทคโนโลยีที่เกี่ยวข้อง	16
3.2 ขั้นตอนการดำเนินงาน	17
3.2.1 เก็บรวบรวมความต้องการ.....	17
3.2.2 ออกแบบโครงสร้างระบบ.....	18
3.2.3 พัฒนา DEV/ACC Schedule Request Form	18
3.2.4 พัฒนาฟังก์ชัน Validate JSF.....	20
3.2.5 พัฒนาฟังก์ชัน Auto create schedule.....	24
3.3 แผนภาพอธิบายโครงสร้างและการทำงานของระบบ	26
3.3.1 Use case diagram	26
3.3.2 Sequential diagram	30
3.3.3 Database schema diagram.....	33
บทที่ 4 ผลการวิจัย	35
4.1 ฟังก์ชัน Submit request.....	35
4.2 ฟังก์ชัน Validate JSF.....	36
4.3 ฟังก์ชัน Auto create schedule	37
บทที่ 5 สรุปผลการวิจัย และข้อเสนอแนะ	39
5.1 สรุปผลการวิจัย	39
5.2 ปัญหาและอุปสรรค และข้อเสนอแนะ.....	39
บรรณานุกรม	40
ภาคผนวก ก.....	41
ภาคผนวก ข.....	44

สารบัญตาราง

ตารางที่	หน้า
3.1 เทคโนโลยีที่เกี่ยวข้อง	16
3.2 ความต้องการของผู้ใช้งาน	17
3.3 validateResult.....	22
3.4 Submit DEV/ACC schedule request	26
3.5 JSF validation report	28
3.6 Auto create schedule	29
3.7 Classify request	29
3.8 devaccScheduleRequest.....	33
3.9 ความหมายของ status	34
3.10 devaccScheduleRequestJSF.....	34



สารบัญภาพ

ภาพที่	หน้า
2.1 Automation Engine architecture.....	3
2.2 ประเภทของ JOB.....	4
2.3 ตัวอย่าง JOB ใน JOB plan.....	4
2.4 Schedule Object.....	5
2.5 AnyPoint Studio.....	5
2.6 Job Automation Services.....	6
2.7 RESTful Web Service.....	8
2.8 ขั้นตอนการทำงานของ Git.....	10
2.9 ขั้นตอนการทำงานของ Git (ต่อ).....	10
3.1 JAS Structure.....	16
3.2 ระบบ Automate DEV/ACC Schedule Request.....	18
3.3 DEV/ACC Schedule Request form permanent schedule.....	18
3.4 DEV/ACC Schedule Request form non-permanent schedule.....	19
3.5 Submit DEV/ACC Schedule Request flowchart.....	20
3.6 แผนผังกระบวนการทำงานระบบตรวจสอบ JSF.....	21
3.7 แผนภาพการทำงานตรวจสอบความสามารถในการตั้งเวลาอัตโนมัติ.....	23
3.8 แผนภาพการทำงานระบบตั้งเวลาอัตโนมัติ.....	24
3.9 แผนภาพการทำงานฟังก์ชัน createSchedule.....	25
3.10 Use case diagram.....	26
3.11 แผนภาพลำดับการทำงานของฟังก์ชัน submit request.....	30
3.12 แผนภาพลำดับการทำงานของฟังก์ชัน validate request.....	31
3.13 แผนภาพลำดับการทำงานของฟังก์ชัน Auto create schedule.....	32
3.14 Database schema diagram.....	33

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญ

ปัจจุบันทีม Automation Engine ของบริษัท เอ็กซอนโมบิล จำกัด มีหน้าที่ในการจัดการตารางเวลาทำงานของ JSF ต่างๆกับฝั่งเอ็กซอนโมบิลไอที (EMIT) และฝั่งธุรกิจ (Business user) เมื่อผู้ใช้งานต้องการจัดการ JSF จะต้องติดต่อเข้ามายังทีม Automation Engine โดยตรงหรือใช้งานแอปพลิเคชัน Job Automation Services (JAS) ซึ่งเป็นระบบ Self-service ที่ให้ผู้ใช้งานสามารถสร้าง แก้ไข หรือลบ JSF ได้เองโดยมีระบบอัตโนมัติทำงานให้ โดยไม่ต้องใช้ทรัพยากรคนในทีม แต่ทว่ายังมีบางฟังก์ชันที่ไม่สามารถทำงานอัตโนมัติได้

เนื่องด้วยความต้องการใช้งานของ Job Automation Services เพิ่มขึ้นทำให้ทีมมีแผนพัฒนาความสามารถในการทำงานอัตโนมัติ โดยการพัฒนาให้บางฟังก์ชันสามารถทำงานอัตโนมัติได้ ซึ่งผู้จัดทำได้รับมอบหมายให้พัฒนาฟังก์ชัน DEV/ACC Schedule Request ซึ่งเป็นฟังก์ชันตั้งค่าเวลาทำงานของ JSF ใน Development และ Acceptance โดยปกติผู้ใช้ต้องส่งคำร้องขอมายังทีมและรอทีมดำเนินการ ซึ่งใช้ทรัพยากรคนของทีมนและสิ้นเปลืองเวลารอของผู้ใช้งาน

ดังนั้นผู้จัดทำจึงได้พัฒนาให้ฟังก์ชัน DEV/ACC Schedule Request ให้สามารถทำงานได้โดยอัตโนมัติและช่วยหาข้อมูลเพื่อลดระยะเวลาทำงานของทีมนสำหรับคำร้องขอที่ไม่สามารถทำงานอัตโนมัติได้ เพื่อลดการใช้ทรัพยากรของทีมนและประหยัดเวลาผู้ใช้งาน ให้สอดคล้องกับแผนพัฒนาความสามารถในการทำงานอัตโนมัติของ Job Automation Services

1.2 วัตถุประสงค์ของการทำวิจัย

1. เพื่อลดปริมาณงานของทีมน Automation Engine
2. เพื่อลดเวลาทำงานของทีมน Automation Engine
3. เพื่อลดเวลารอการดำเนินการของผู้ใช้งาน

1.3 ขอบเขตของการวิจัย

1. แก้ไข DEV/ACC Schedule Request Form ของระบบ Job Automation Services ให้ตรงตามความต้องการของผู้ใช้งาน
2. สามารถจำแนกประเภทของ JSF เพื่อนำไปตั้งค่าทำงานอัตโนมัติได้
3. สามารถเพิ่มข้อมูลลงในคำร้องตั้งเวลาทำงานได้
4. สามารถตั้งค่าเวลาทำงานอัตโนมัติของ JSF แบบไม่ซับซ้อนได้

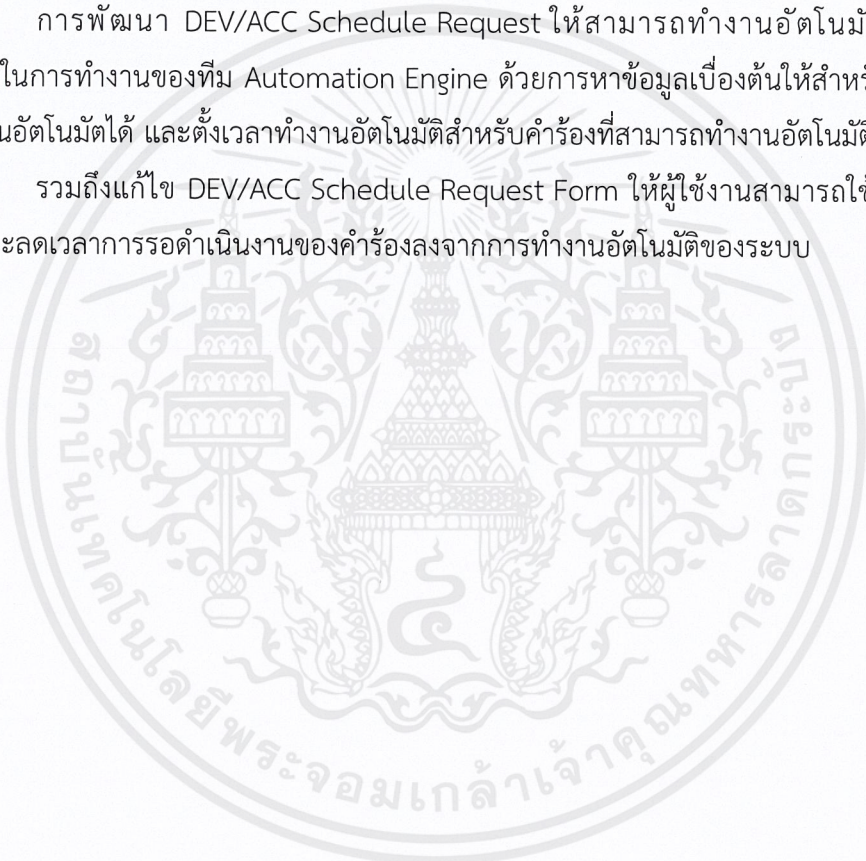
1.4 วิธีดำเนินการ

1. ศึกษาการทำงานของ Automation Engine server
2. ศึกษาการทำงานของ Job Automation Services
3. เก็บรวบรวมความต้องการจากผู้ใช้งาน
4. วิเคราะห์ความต้องการ จัดลำดับความสำคัญ และประเมินความเป็นไปได้ของโครงการ
5. ออกแบบระบบการทำงาน
6. พัฒนาระบบอัตโนมัติ

1.5 ประโยชน์ที่คาดว่าจะได้รับ

การพัฒนา DEV/ACC Schedule Request ให้สามารถทำงานอัตโนมัติได้ จะช่วยประหยัดเวลาในการทำงานของทีม Automation Engine ด้วยการหาข้อมูลเบื้องต้นให้สำหรับคำร้องที่ไม่สามารถทำงานอัตโนมัติได้ และตั้งเวลาทำงานอัตโนมัติสำหรับคำร้องที่สามารถทำงานอัตโนมัติได้

รวมถึงแก้ไข DEV/ACC Schedule Request Form ให้ผู้ใช้งานสามารถใช้งานได้อย่างสะดวกขึ้น และลดเวลาการรอดำเนินงานของคำร้องลงจากการทำงานอัตโนมัติของระบบ



บทที่ 2

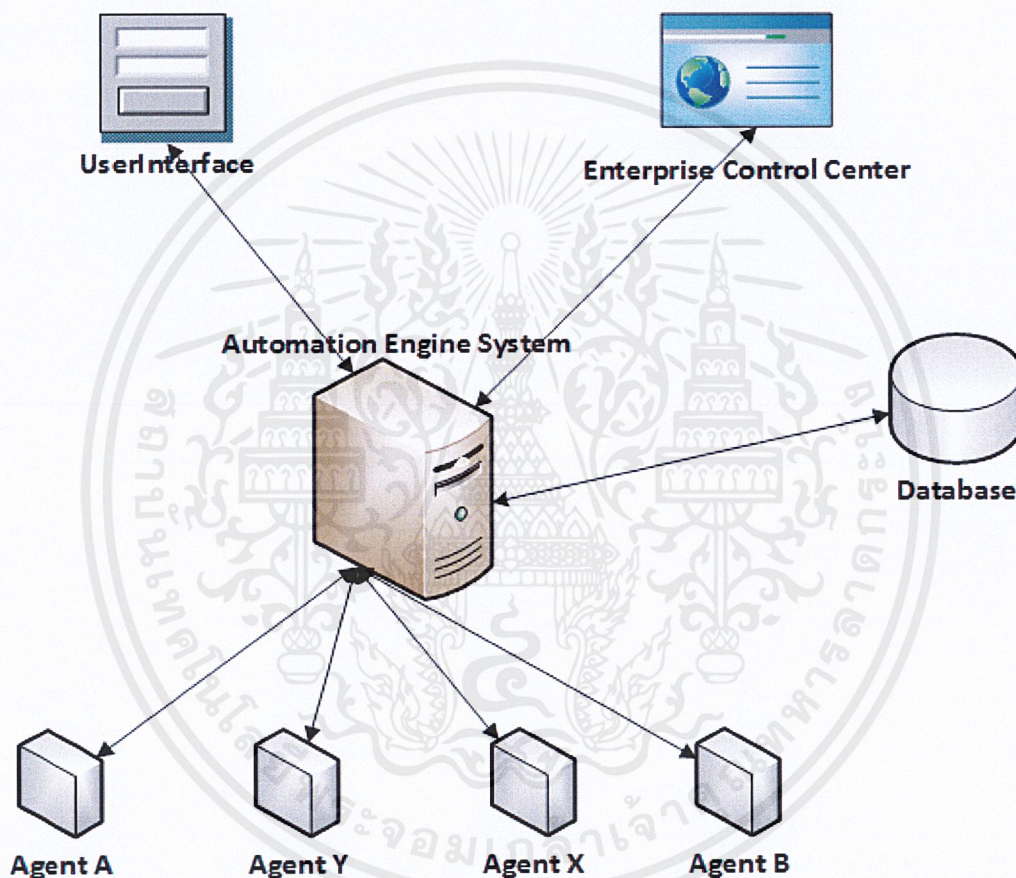
แนวคิดและทฤษฎีที่เกี่ยวข้อง

2.1 ทฤษฎีที่เกี่ยวข้อง

2.1.1 Automation Engine

เป็นเซิร์ฟเวอร์ระบบอัตโนมัติที่ถูกพัฒนาโดย BROADCOM เพื่อเป็นทางเลือกสำหรับบริษัทในการนำ Automation Engine ไปพัฒนาระบบอัตโนมัติ

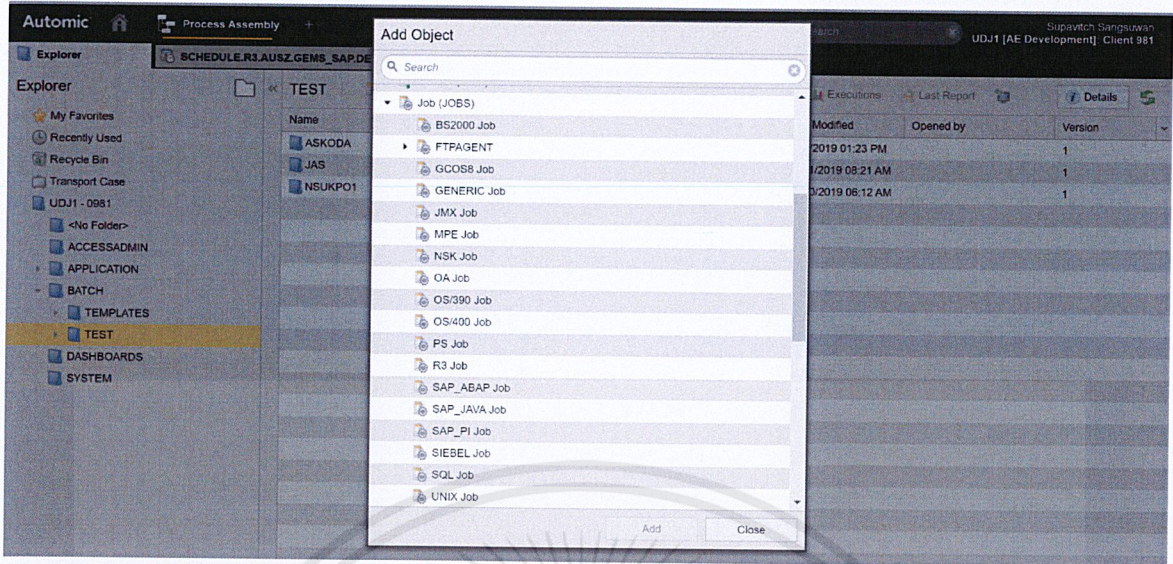
โดย Automation Engine เป็นระบบที่มีลักษณะการทำงานที่มี Automation Engine server เป็นเซิร์ฟเวอร์ประมวลผลกลางและกระจายการทำงานไปยัง Agent อื่นๆ



ภาพที่ 2.1 Automation Engine architecture

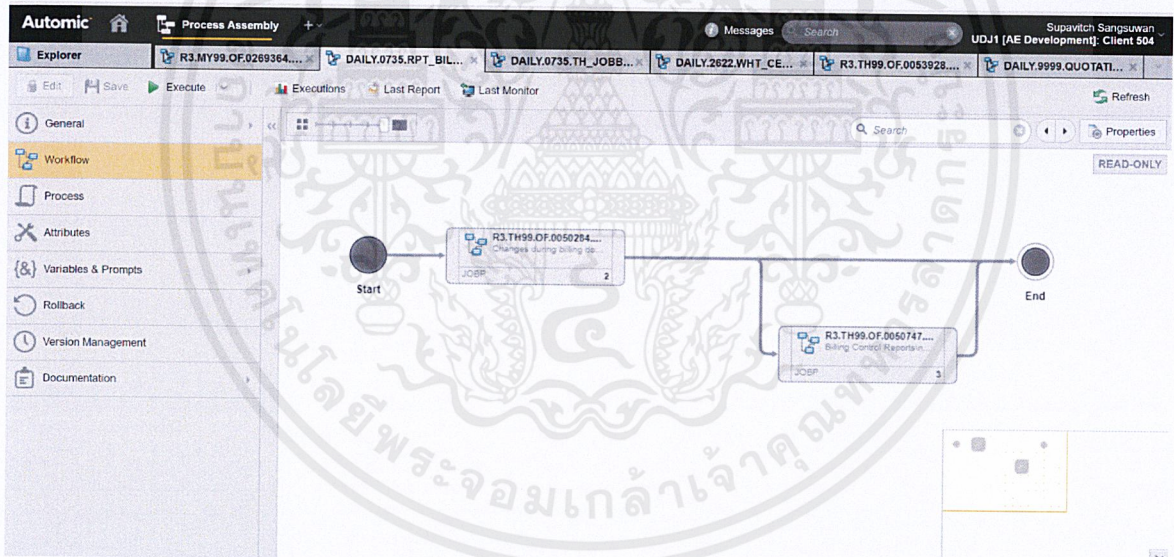
โดย Automation Engine server คอยทำหน้าที่ตั้งค่าการทำงานอัตโนมัติตามเวลาที่กำหนด โดยมีการสร้างคำสั่งที่จะให้ทำงาน หรือที่เรียกว่า JOB ล่วงหน้า โดย JOB นั้นมีหลากหลายประเภท แบ่งตามจุดประสงค์ เช่น JOB สำหรับ unix script , JOB สำหรับ File transfer ,JOB สำหรับส่ง

คำสั่ง SQL หรือ JOB สำหรับติดต่อกับ SAP เป็นต้น



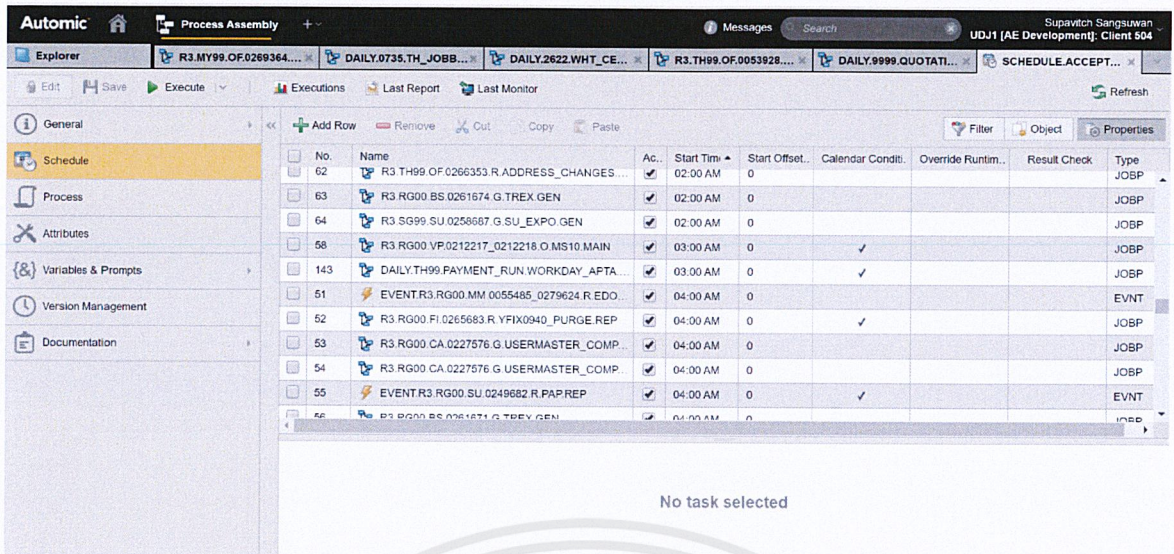
ภาพที่ 2.2 ประเภทของ JOB

ซึ่งการที่จะตั้งเวลาทำงานให้กับ JOB ได้ ต้องนำ JOB ไปเพิ่มใน Schedule เสียก่อน ตั้งเวลา แล้วหลังจากนั้น Automation Engine จะคอยควบคุมการทำงานตามเวลาที่กำหนด



ภาพที่ 2.3 ตัวอย่าง JOB ใน JOB plan

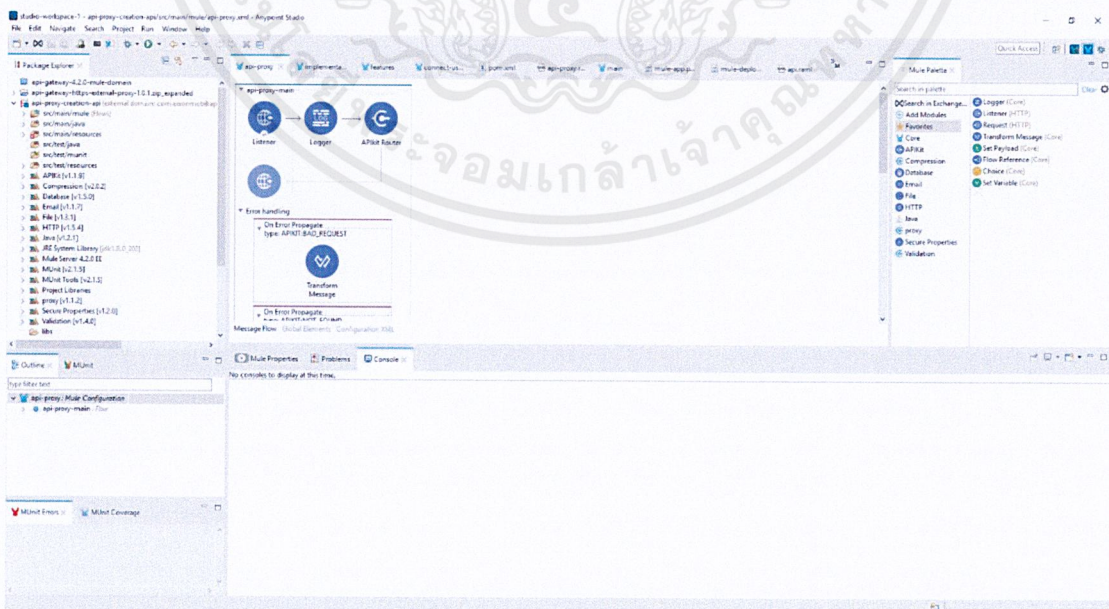
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 2.4 Schedule Object

2.1.2 Anypoint Platform

Anypoint Platform ของ MuleSoft เป็นแพลตฟอร์มการเชื่อมต่อสำหรับ SOA, SaaS และ API MuleSoft มอบความคล่องตัวทางธุรกิจให้กับ บริษัท โดยการเชื่อมต่อแอปพลิเคชันข้อมูลและอุปกรณ์ทั้งภายในบริษัทและในระบบคลาวด์ด้วยวิธีการที่นำโดย API ด้วยการใช้ประโยชน์แพลตฟอร์ม Anypoint บริษัท สามารถสร้างโครงสร้างพื้นฐาน SOA จากระบบเดิมแพลตฟอร์มเพื่อสร้างความคล่องตัวทางธุรกิจ โดยสามารถโยกย้ายโครงสร้างพื้นฐานด้านเทคโนโลยีไปยังคลาวด์สาธารณะหรือส่วนตัวและจัดลำดับความสำคัญของการใช้แอปพลิเคชัน SaaS และเทคโนโลยีคลาวด์อื่น ๆ ซึ่ง Anypoint Platform มีโปรแกรม Anypoint Studio สำหรับการพัฒนาระบบ เพื่อใช้บนแพลตฟอร์ม



ภาพที่ 2.5 AnyPoint Studio

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.3 Job Automation Services

เป็นแอปพลิเคชันที่ทีม Automation Engine พัฒนาขึ้นเพื่อให้ผู้ใช้งานสามารถสร้าง ลบ หรือแก้ไข JOB ใน Automation Engine ง่ายขึ้น รวมถึงลดงานของทีมด้วยการจัดการคำร้องของผู้ใช้งานโดยอัตโนมัติ



ภาพที่ 2.6 Job Automation Services

2.1.4 Agile Methodology

เป็นวิธีบริหารจัดการโปรเจกในรูปแบบที่ยอมรับการเปลี่ยนแปลงได้บ่อย ซึ่งแตกต่างจากวิธี water fall ที่จะต้องดำเนินงานไปตามแผนที่วางไว้ โดย Agile จะเน้นการทำงานเป็นทีมร่วมกับลูกค้าเพื่อที่จะเข้าใจความต้องการได้ง่าย และตอบสนองต่อการเปลี่ยนแปลงได้อย่างรวดเร็ว รวมถึงสามารถส่งมอบผลงานให้ลูกค้าได้ตามความต้องการ โดยมีหลักการ 4 ข้อดังนี้

- Individuals and interactions over processes and tools เน้นการสื่อสารและปฏิสัมพันธ์กันระหว่างคน มากกว่าเครื่องมือต่างๆที่นำมาช่วย
- Working software over comprehensive documentation เน้นทำผลิตภัณฑ์ มากกว่าการทำเอกสาร
- Customer collaboration over contract negotiation เน้นตอบสนองผู้ใช้งาน มากกว่าแค่ทำตามสัญญา
- Responding to change over following a plan เน้นการปรับปรุงพัฒนา มากกว่าการทำตามแผนที่วางเอาไว้

โดยการบริหารจัดการแบบ Agile ที่เน้นความร่วมมือกับลูกค้ามีการแบ่งหน้าที่สำคัญออกเป็น 3 หน้าที่ดังนี้

- Stakeholders ผู้มีส่วนเกี่ยวข้องกับผลิตภัณฑ์, ผู้ใช้งาน, เจ้าของบริษัท
- Product Owner (PO) ผู้ออกแบบผลิตภัณฑ์เพื่อตอบสนอง Stakeholders

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Developer (Dev) ผู้พัฒนาผลิตภัณฑ์ตามที่ออกแบบไว้ให้เกิดขึ้นจริง

ในแต่ละทีมพัฒนาจะต้องมี PO เป็นของทีมตัวเอง (PO 1 คน อาจดูแลหลายทีม ได้ PO ของแต่ละทีมจะต้องมีการคุยเพื่อแลกเปลี่ยนข้อมูลกัน เพื่อลดการทำงานที่ซ้ำซ้อน หรือถ้ามีหลายทีมมาก อาจมีหัวหน้า PO ขึ้นมาอีกคนเพื่อประสานงานระหว่าง PO ทีมต่างๆ ซึ่งผลลัพธ์ที่ควรได้จากการบริการงานแบบ Agile คือ

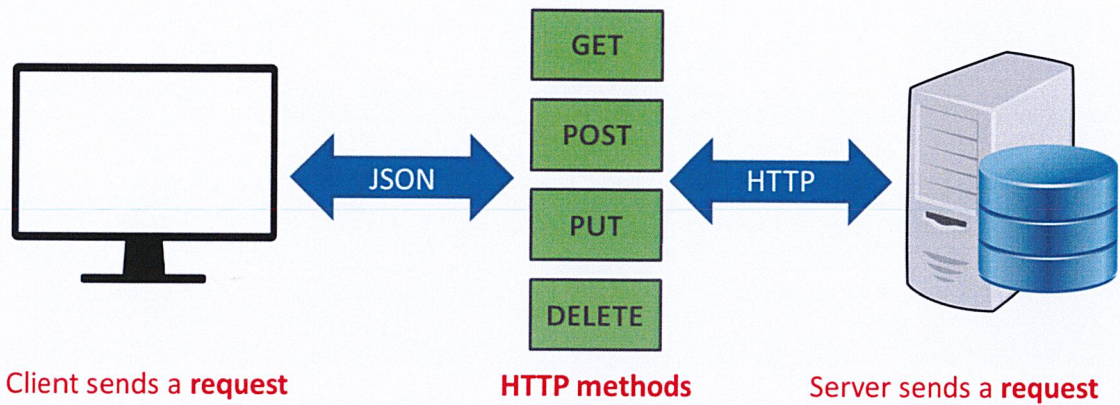
- ลดความเข้าใจผิดพลาด หากคนในทีมเข้าใจงานคนละอย่าง อาจทำให้ผลลัพธ์ของงานไม่ตอบสนองต่อเป้าหมายที่ต้องการ จึงควรสื่อสารกันเรื่อยๆ เพื่อให้ทุกคนเข้าใจตรงกันอยู่เสมอ
- ลดกระบวนการที่ล่าช้า การติดต่อสื่อสารผ่านคนกลางควรเปลี่ยนให้ Dev คุยกับ PO หรือ User โดยตรง และอัปเดตสิ่งที่คุยมาให้ทุกคนในทีมทราบและเข้าใจตรงกัน
- ลดการทำงานที่ซ้ำซ้อนกัน งานบางอย่างอาจมีการทำซ้ำซ้อนกับทีมอื่น งานที่เคยมีคนทำไว้แล้ว ควรสามารถหยิบมาใช้หรือพัฒนาต่อยอดได้ โดยไม่ต้องเริ่มใหม่ทั้งหมด
-

2.1.5 RESTful Web Service

Web Service ที่ใช้ REST architectural style เป็นที่รู้จักกันในชื่อ RESTful web services (RWS). ซึ่งอนุญาตให้ระบบ Request และเข้าถึง Resource บนเว็บโดยใช้ชุดคำสั่งที่กำหนดเอาไว้ล่วงหน้า โดยที่การโต้ตอบของระบบที่ใช้ REST จะอยู่บนพื้นฐานของ Hypertext Transfer Protocol (HTTP). Request จะส่งคำขอไปยัง URI ที่กำหนด และนำเอา response กลับมาเป็น Payload ในแบบ HTML, XML, JSON หรือ format อื่นๆ

โดย RESTful API มี HTTP Method ที่สำคัญดังนี้

- GET เป็น method สำหรับร้องขอข้อมูลจาก resource และจำไม่มี side effect . เช่น /companies/34/employees จะตอบสนองเป็นรายการ employee ของ company 34
- POST เป็น method สำหรับสร้างข้อมูลใหม่ใน resource ซึ่งจะส่งข้อมูลใน body. เช่น /companies/34/employees จะสร้าง employee ใหม่ของ company 34 โดยที่ POST เป็น non-idempotent หมายความว่าหลายๆ request จะมีผลที่แตกต่างกัน
- PUT เป็น method สำหรับอัปเดตข้อมูลของข้อมูลที่มีอยู่แล้วใน resource หรือสร้างใหม่ เช่น /companies/34/employees/john จะอัปเดตหรือสร้างข้อมูลของ john ใน resource employees ของ company 34 โดยที่ PUT เป็น idempotent หมายความว่าหลายๆ request จะมีผลเดียว
- DELETE เป็น method สำหรับลบข้อมูลที่มีอยู่ใน resource เช่น /companies/34/employees/john จะลบข้อมูล john ใน resource employees ของ company 34



ภาพที่ 2.7 RESTful Web Service

และเมื่อ client สร้าง request ไปยัง server ผ่าน API ควรจะรู้ผลลัพท์กลับ ไม่ว่าจะการส่ง request นั้น จะ Pass, Fail หรือ การ request นั้นผิด โดยที่ HTTP status จะเป็นรหัสมาตรฐานที่บอกสถานะของการ request ซึ่งจะมีหลากหลายในแต่ละสถานการณ์ ซึ่ง ตัว Server เองควรจะ return status code ให้ถูกต้อง ซึ่งต่อไปนี้จะเห็น HTTP status code ที่สำคัญโดยแบ่งออกเป็นหมวดดังนี้:

1. 2xx (หมวด Success) เป็น status code ที่บอกว่าการ request นั้นได้รับแล้วและกระทำตาม method สำเร็จโดย Server
 - 1.1. 200 Ok เป็นมาตรฐานของ HTTP response นั้น Success สำหรับ GET, PUT หรือ POST
 - 1.2. 201 Create เป็น response สำหรับข้อมูลใหม่ได้ถูกสร้างขึ้น ใช้สำหรับ POST
 - 1.3. 204 No Content เป็น response สำหรับ request ที่ดำเนินการ Success แต่ไม่ได้ return ข้อมูลกลับ
2. 3xx (หมวด Redirection)
 - 2.1. 304 Not Modified เป็น status code ที่บอกว่า client ได้รับการ response แล้วอยู่ใน cache และไม่จำเป็นต้องส่งผ่านข้อมูลเดิมอีกครั้ง
3. 4xx (หมวด Client error) โดยที่ status code เหล่านี้จะบอก client ว่าสิ่งที่ request มานั้น Failed
 - 3.1. 400 Bad Request บอกว่า request ที่ส่งมาโดย client นั้นไม่ถูกดำเนินการ และ Server ไม่เข้าใจว่า request เกี่ยวกับอะไร
 - 3.2. 401 Unauthorized บอกว่า client ไม่ได้รับอนุญาตในการเข้าถึง Resource และควรจะ request ใหม่ด้วย credential
 - 3.3. 403 Forbidden บ่งบอกว่า request นั้นถูกต้องและ client ได้รับการอนุญาต แต่ Client ไม่ได้รับการอนุญาตให้เข้าถึง Resource หรือหน้าเพจด้วยเหตุผลบางประการ เช่น บางครั้ง Client ที่ได้รับอนุญาต ไม่อนุญาตให้เข้าถึงระบบไฟล์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 3.4. 404 Not Found บ่งบอกว่า resource ที่ request มานั้น ไม่ว่างใช้งานตอนนี้
- 3.5. 405 Gone บ่งบอกว่า resource ที่ต้องการนั้นไม่มีอยู่แล้ว หรืออาจจะย้ายไปที่อื่น
4. 5xx (หมวด Server error)
 - 4.1. 500 Internal Server Error บ่งบอกว่าการ request นั้นถูกต้อง แต่ server มีความสับสนและจะบริการด้วยเงื่อนไขที่คาดการณ์ไม่ได้
 - 4.2. 503 Service Unavailable บ่งบอกว่า server ใช้งานไม่ได้ หรือไม่ว่างที่จะรับและดำเนินการ request โดยส่วนใหญ่แล้ว server อยู่ในช่วงบำรุงรักษา

2.1.6 Git version control

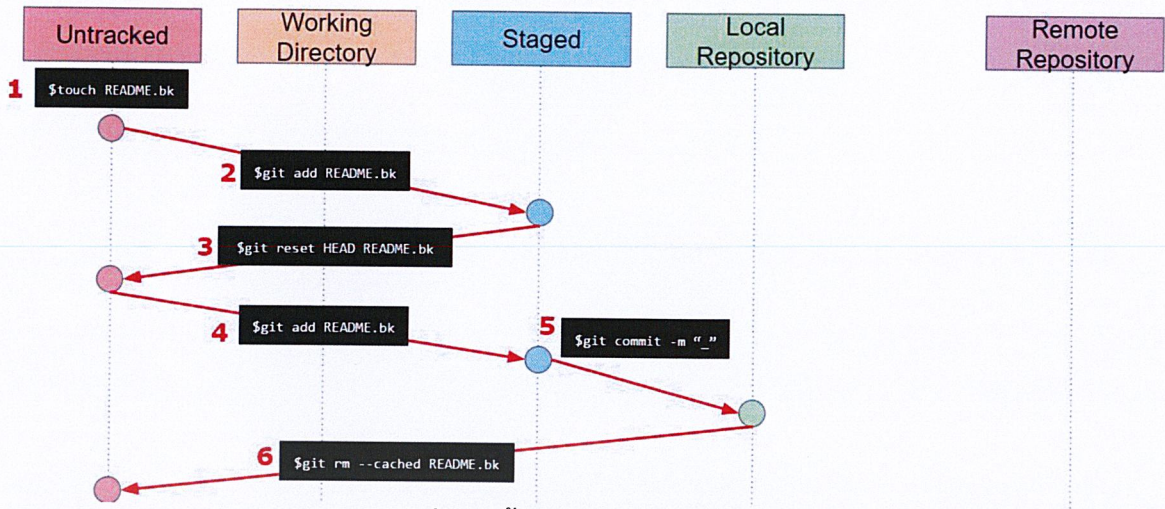
Git คือ Version Control แบบ Distributed ตัวหนึ่ง เป็นระบบที่ใช้จัดเก็บและควบคุมการเปลี่ยนแปลงที่เกิดขึ้นกับไฟล์ชนิดใดก็ได้ ไม่ว่าจะเป็น Text File หรือ Binary File

ซึ่งประโยชน์ของการใช้ Git คือ Track version ของ Source Code ย้อนกลับได้ เมื่อจัดเก็บไฟล์เข้าไปในระบบของ Git จะเรียกว่า Git Repository ซึ่งเก็บสำรองข้อมูลและการเปลี่ยนแปลงของ Source Code ทำให้สามารถย้อนกลับไปเวอร์ชันใดๆ ก่อนหน้า และดูรายละเอียดการเปลี่ยนแปลงของแต่ละเวอร์ชันได้ นอกจากนี้ยังสามารถดูได้ว่าใครเป็นคนแก้ไข รวมทั้งยังช่วยในการพัฒนาซอฟต์แวร์เป็นทีม Git สามารถเก็บบันทึกการเปลี่ยนแปลงของ Source Code เวอร์ชันล่าสุดไว้ที่ Local Repository ซึ่งสามารถทำงานได้โดยไม่ต้องต่อกับอินเทอร์เน็ต และเมื่อต้อง Update การเปลี่ยนแปลงของ Source Code เวอร์ชันล่าสุดให้กับทีมก็สามารถที่จะ Push ขึ้นไปเก็บที่ Remote Repository(Git Hosting) และเพื่อนร่วมทีมก็สามารถ Pull เวอร์ชันล่าสุดนั้นมารวม(Auto Merge) ที่เครื่องของเขาเอง ทำให้ Source Code ที่พัฒนาร่วมกันกับคนภายในทีมเป็นเวอร์ชันล่าสุดเสมอ

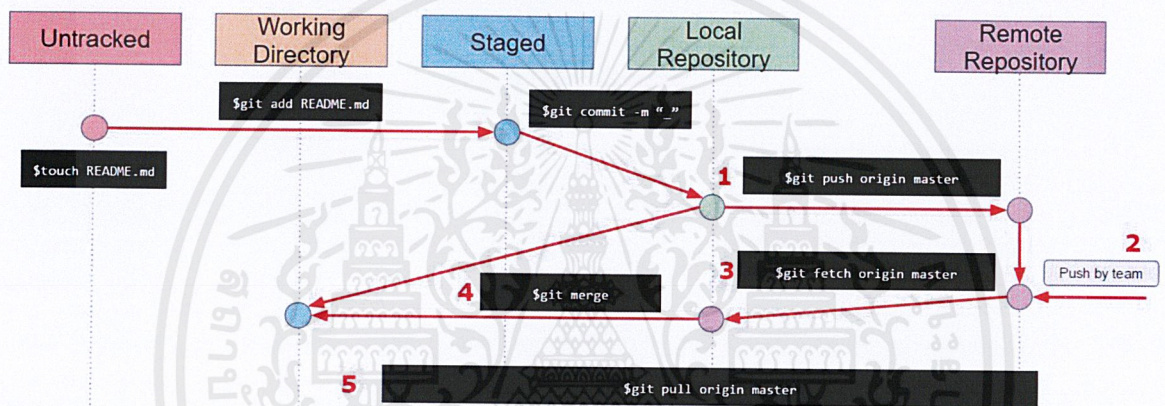
โดย Git มีสถานะของ source code ที่เก็บอยู่ในระบบดังนี้

- Untracked เป็นสถานะที่ Source Code ถูกเพิ่มเข้ามาใหม่และยังไม่ได้ถูกเก็บไว้ในระบบของ Git
- Working Directory เป็นสถานะที่กำลังมีการเปลี่ยนแปลงหรือแก้ไข Source Code หรืออาจจะเรียกสถานะนี้ว่า Modified
- Staged เป็นสถานะที่ Source Code กำลังเตรียมที่จะ Commit เพื่อยืนยันการเปลี่ยนแปลงก่อนที่จะเก็บลงในสถานะ Local Repository
- Local Repository เป็นสถานะที่มีการเก็บบันทึกข้อมูลการเปลี่ยนแปลงของ Source Code ลงไปที่ Git Repository ที่เป็น Local (ที่เครื่องตัวเอง)
- Remote Repository เป็นสถานะที่มีการเก็บบันทึกข้อมูลการเปลี่ยนแปลงของ Source Code ลงไปที่ Git Repository ที่เป็น Hosting (ที่เครื่องเซิร์ฟเวอร์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 2.8 ขั้นตอนการทำงานของ Git



ภาพที่ 2.9 ขั้นตอนการทำงานของ Git (ต่อ)

โดยที่การใช้งาน Git จะต้องป้อนคำสั่งผ่าน command line หรือ shell ซึ่งมีคำสั่งพื้นฐานดังต่อไปนี้

- Git Config เป็นคำสั่งที่ใช้แสดงและกำหนดข้อมูลของผู้ใช้เพื่อระบุตัวตน และคุณสมบัติอื่นๆ

```
$git config --global --list #แสดงคุณสมบัติของ Git ทั้งหมด
$git config --list #แสดงคุณสมบัติของ Git เฉพาะ Repository นั้น
```

การกำหนดชื่อและอีเมลของผู้ใช้

```
$git config --global user.name "Your Name" #กำหนดชื่อผู้ใช้
$git config --global user.email "example@email.com" #กำหนดอีเมลของผู้ใช้
$git config --global --list #ตรวจสอบอีกครั้งหลังจากกำหนดค่าเสร็จแล้ว
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Git Init เป็นคำสั่งที่ใช้สร้างระบบของ Git ขึ้นมาภายใต้โฟลเดอร์หรือ Path นั้น โดยจะสร้างโฟลเดอร์ .git ขึ้นมาเพื่อใช้เก็บ สำรองข้อมูล การเปลี่ยนแปลงและคุณสมบัติอื่นๆ ของ Git

```
$git init
```

- Git Status เป็นคำสั่งที่ใช้ตรวจสอบสถานะของ Source Code ในระบบของ Git ซึ่งจะแสดงสถานะดังที่ได้อธิบายข้างต้นไปแล้ว

```
$git status
```

- Git Add เป็นคำสั่งที่ใช้เพิ่มการเปลี่ยนแปลงของ Source Code เข้าไปที่สถานะ Staged

```
$git add <file_name>
```

```
$git add README.md
```

```
#เพิ่มไฟล์ชื่อ README.md เข้าไปที่สถานะ Staged
```

```
$git add .
```

```
#ใช้ในกรณีที่มีหลายๆ ไฟล์และต้องการเพิ่มเข้าไปทั้งหมด
```

- Git Commit เป็นคำสั่งที่ใช้ยืนยัน Source Code ที่อยู่ในสถานะ Staged เข้าไปเก็บไว้ที่ Local Repository

```
$git commit -m "message"
```

```
#ยืนยันการเปลี่ยนแปลงพร้อมข้อความ
```

```
$git commit -am "message"
```

```
#เพิ่มการเปลี่ยนแปลงและยืนยันพร้อมข้อความ
```

```
$git commit
```

```
#เพิ่มข้อความในโปรแกรม vi
```

```
#ยืนยันการเปลี่ยนแปลงพร้อมข้อความและ merge ลงใน commit ล่าสุด
```

```
$git commit --amend -m "message"
```

- Git Log เป็นคำสั่งที่ใช้แสดงประวัติของ Commit ที่เก็บไว้ใน Repository

```
$git log
```

```
$git log --oneline
```

```
$git log --oneline --decorate
```

```
$git log --oneline --decorate --graph
```

```
$git log --stat #Diff from log
```

```
$git log --grep="Message" #Log by message
```

```
$git log --after="2017-2-14" #Log in date
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Git Branch เป็นคำสั่งที่ใช้ในแสดงและแต่งกิ่งสาขาในการพัฒนา ซึ่งทำให้การพัฒนาซอฟต์แวร์มีความยืดหยุ่นมากขึ้น

```
$git branch
$git branch --all
$git branch develop          #สร้าง branch ชื่อ develop
$git branch --delete develop #ลบ branch ชื่อ develop
#ส่งการเปลี่ยนแปลง branch develop ไปยัง Remote ที่ชื่อ origin
$git push origin develop
#ส่งการเปลี่ยนแปลงลบ branch develop ไปยัง Remote ที่ชื่อ origin
$git push --delete origin develop
$git log --before="2017-2-14" #Log in date
$git log --author=pakin      #Log by user
```

- Git Checkout เป็นคำสั่งที่ใช้ในการสลับ Working Directory ไปยัง Branch หรือ Commit ที่เราระบุ คำสั่งนี้ยังสามารถใช้งานได้ในอีกหลายๆ แบบ

```
#ย้ายการทำงานไปที่ Branch หรือ commit_id ที่ระบุ
$git checkout <branch name, commit id>
#สร้าง branch ชื่อ test และทำการสลับการทำงานมาที่ Branch นี้
$git checkout -b test
#ยกเลิกการเปลี่ยนแปลงของไฟล์ใน Working Directory
$git checkout -- <file name>
#เลือกแค่บางไฟล์จาก Branch อื่น เข้ามา Merge กับ Working Directory ที่กำลังทำงาน
$git checkout <branch name> <file name>
#คำสั่งนี้จะเหมือนคำสั่งด้านบนแต่จะมีโหมดตอบโต้กับผู้ใช้ในการเลือกสถานะของไฟล์ที่ระบุ
$git checkout --patch <branch name> <file name>
```

- Git Reset เป็นคำสั่งที่ใช้ย้อนกลับไปเวอร์ชันใดๆ ก่อนหน้า โดยระบุ Branch หรือ Commit Id (SH-1 แบบย่อของ Commit 7 ตัว เช่น 4bcb295) ซึ่งมี Option ที่สำคัญ 3 ตัวดังนี้
 - soft ย้อนการเปลี่ยนแปลง และคงสถานะการเปลี่ยนแปลงของ Source Code ไว้ที่สถานะ Staged

- mixed ย้อนการเปลี่ยนแปลง และคงสถานะการเปลี่ยนแปลงของ Source Code ไว้ที่สถานะ Working Directory หรือ Modified
- hard ย้อนการเปลี่ยนแปลงแบบลบทิ้งการเปลี่ยนแปลงก่อนหน้าทั้งหมด คำสั่งนี้อันตราย เพราะมันจะทำให้ประวัติของ Commit ที่เก็บไว้ใน Repository หายไป จึงยังไม่เหมาะกับมือใหม่

```
$git reset --soft 4bcb295 #ย้อนกลับไป Commit id 4bcb295
$git reset --mixed develop #ย้อนกลับไป Branch develop
```

- Git Merge เป็นคำสั่งที่ใช้ในการรวม Branch หรือ Commit ทั้งสองเข้าด้วยกัน ตัวอย่างเราจะอยู่ที่ Branch Master และต้องการ Merge Branch Feature เข้ามาทำงานร่วมด้วย การ Merge แบบ No Fast Forward จะเรียกอีกอย่างหนึ่งว่า 3-Way Merge

```
#รวม branch master กับ branch feature แบบ no fast forward
$git merge --no-ff feature
#รวม branch master กับ branch feature แบบ fast forward
$git merge feature
```

- Git Remote [เริ่มต้นทำงานกับ Git Hosting (Remote Repository)] เพิ่ม URL ของ Remote Repository เข้าไปยังคุณสมบัติของ Git โดยชื่อว่า origin ส่วนใหญ่จะเป็นชื่อ Default ที่หลายๆคนเข้าใจตรงกัน แต่เราก็สามารถตั้งชื่ออื่นๆ ได้

```
$git remote add origin <URL> #เพิ่ม Remote Repository ชื่อ origin
$git remote add origin https://github.com/NewGame0/Android_HelloWorld.git
#เพิ่ม Remote Repository ใหม่ชื่อ origin
$git remote set-url origin <New URL>
$git remote -v #แสดง Remote Repository
$git config --list #แสดงคุณสมบัติต่างๆของ Git ซึ่งจะมี Remote Repository แสดงออกมาด้วย
```

- Git Push เป็นคำสั่งที่ใช้ส่งการเปลี่ยนแปลงของ Source Code ที่เก็บอยู่บน Local Repository ขึ้นไปยัง Remote Repository

```
#ส่งการเปลี่ยนแปลง Branch master ไปยัง Remote ที่ชื่อ origin
$git push origin master
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Git Fetch เป็นคำสั่งที่ใช้รับการเปลี่ยนแปลงของ Source Code ล่าสุดที่อยู่บน Remote Repository ลงมายัง Local Repository แต่ยังไม่ได้ทำการรวม Source Code (Merge)

#รับการเปลี่ยนแปลงทุก Branch จาก Remote Repository

```
$git fetch --all
```

#รับการเปลี่ยนแปลง Branch master จาก Remote Repository ที่ชื่อ origin

```
$git fetch origin master
```

- Git Pull [fetch + merge] เป็นคำสั่งที่ใช้รับการเปลี่ยนแปลงของ Source Code ล่าสุดที่อยู่บน Remote Repository ลงมายัง Local Repository และทำการ Auto Merge

```
$git pull <remote> <branch>
```

```
$git pull origin master
```

- Git Clone เป็นคำสั่งที่ใช้ดึงประวัติทั้งหมดบน Remote Repository ของเพื่อนร่วมทีม ของคนอื่นหรือของเราเองที่มีอยู่แล้วบน Git Hosting มาที่เครื่องของเรา คำสั่งนี้จะคล้ายๆ Git Init ที่ใช้สร้างระบบ Git ขึ้นมาตอนเริ่มต้น แต่เราจะได้ประวัติเดิมของ Repository มาด้วย ทำให้เราเริ่มพัฒนาต่อจากตรงจุดนี้ได้เลย

```
$git clone https://github.com/NewGame0/Android_HelloWorld.git
```

2.1.7 SQL

SQL ย่อมาจาก structured query language คือภาษาที่ใช้ในการเขียนโปรแกรม เพื่อจัดการกับฐานข้อมูลโดยเฉพาะ เป็นภาษามาตรฐานบนระบบฐานข้อมูลเชิงสัมพันธ์และเป็นระบบเปิด (open system) หมายถึงเราสามารถใส่คำสั่ง sql กับฐานข้อมูลชนิดใดก็ได้ และ คำสั่งงานเดียวกันเมื่อส่งงานผ่าน ระบบฐานข้อมูลที่แตกต่างกันจะได้ ผลลัพธ์เหมือนกัน ทำให้เราสามารถเลือกใช้ฐานข้อมูล ชนิดใดก็ได้โดยไม่ติดขัดกับฐานข้อมูลใดฐานข้อมูลหนึ่ง นอกจากนี้แล้ว SQL ยังเป็นชื่อโปรแกรมฐานข้อมูล ซึ่งโปรแกรม SQL เป็นโปรแกรมฐานข้อมูลที่มีโครงสร้างของภาษาที่เข้าใจง่าย ไม่ซับซ้อน มีประสิทธิภาพการทำงานสูง สามารถทำงานที่ซับซ้อนได้โดยใช้คำสั่งเพียงไม่กี่คำสั่ง โปรแกรม SQL จึงเหมาะที่จะใช้กับระบบฐานข้อมูลเชิงสัมพันธ์ และเป็นภาษาหนึ่ง ซึ่งแบ่งการทำงานได้เป็น 4 ประเภท ดังนี้

1. Select query ใช้สำหรับดึงข้อมูลที่ต้องการ
2. Update query ใช้สำหรับแก้ไขข้อมูล
3. Insert query ใช้สำหรับการเพิ่มข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. Delete query ใช้สำหรับลบข้อมูลออกไป

ปัจจุบันมีซอฟต์แวร์ระบบจัดการฐานข้อมูล (DBMS) ที่สนับสนุนการใช้คำสั่ง SQL เช่น Oracle, DB2, MS-SQL, MS-Access นอกจากนี้ภาษา SQL ถูกนำมาใช้เขียนร่วมกับโปรแกรมภาษาต่างๆ เช่น ภาษา C/C++, Visual Basic และ Java ซึ่งประโยชน์ของภาษา SQL มีดังนี้

1. สร้างฐานข้อมูลและ ตาราง
2. สนับสนุนการจัดการฐานข้อมูล ซึ่งประกอบด้วย การเพิ่ม การปรับปรุง และการลบข้อมูล
3. สนับสนุนการเรียกใช้หรือ ค้นหาข้อมูล

ประเภทของคำสั่งภาษา SQL ที่สามารถจับกลุ่มเป็นกลุ่มใหญ่ได้ 3 กลุ่มได้แก่

1. ภาษานิยามข้อมูล(Data Definition Language : DDL) เป็นคำสั่งที่ใช้ในการสร้างฐานข้อมูล กำหนดโครงสร้างข้อมูลว่ามี Attribute ใด ชนิดของข้อมูล รวมทั้งการเปลี่ยนแปลงตาราง และการสร้างดัชนี คำสั่ง : CREATE,DROP,ALTER
2. ภาษาจัดการข้อมูล (Data Manipulation Language :DML) เป็นคำสั่งที่ใช้ในการเรียกใช้ เพิ่ม ลบ และเปลี่ยนแปลงข้อมูลในตาราง คำสั่ง : SELECT,INSERT,UPDATE,DELETE
3. ภาษาควบคุมข้อมูล (Data Control Language : DCL) เป็นคำสั่งที่ใช้ในการกำหนดสิทธิการอนุญาต หรือ ยกเลิก การเข้าถึงฐานข้อมูล เพื่อป้องกันความปลอดภัยของฐานข้อมูล คำสั่ง : GRANT,REVOKE

2.1.8 Angular

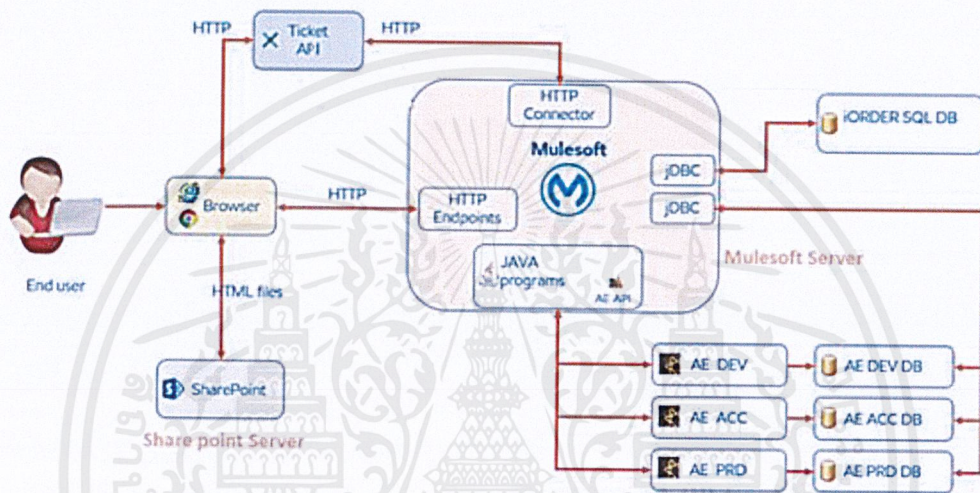
Angular เป็น Front-end JavaScript Framework ที่ทำงานบนฝั่ง Client ที่เรานำไปสร้าง Reactive Single Page Applications (SPA) ซึ่งก็คือทุกๆหน้าจะถูกโหลดมารวมอยู่ในหน้าเดียว การคลิกเปลี่ยนหน้าหรือการคลิกปุ่มต่างๆ จะทำให้เรารู้สึกเหมือนเป็น Desktop Application ที่ไม่มีการโหลดเปลี่ยนหน้า Angular เป็น Model-View-Controller (MCV) และยังเป็น Model-View-ViewModel (MVVM) อีกด้วย มีการเชื่อมการทำงานระหว่าง JavaScript เข้ากับ DOM Element ของ HTML ใช้การทำงาน client-side template สามารถสร้าง template ไปใส่ไว้ในที่ที่เรากำหนดไว้ได้ และเป็น 2-way data binding เพื่อ sync Model กับ View

บทที่ 3

วิธีการดำเนินงาน

การพัฒนาฟังก์ชัน DEV/ACC Schedule Request ซึ่งเดิมมีอยู่แล้วบนแอปพลิเคชัน Job Automation Services หรือ JAS แต่ผู้จัดทำต้องเพิ่มเติมความสามารถในการตั้งเวลาโดยอัตโนมัติได้ ซึ่งในบทนี้จะกล่าวถึงขั้นตอนการดำเนินงานทั้งหมด ตั้งแต่การรวบรวมความต้องการของระบบ การออกแบบ และการพัฒนาฟังก์ชันต่างๆ

3.1 สถาปัตยกรรมของระบบ



ภาพที่ 3.1 JAS Structure

3.1.1 เทคโนโลยีที่เกี่ยวข้อง

เราสามารถจำแนกเทคโนโลยีที่ใช้งานในโครงการนี้ได้ ด้วยการแบ่งตามองค์ประกอบของระบบ ได้ดังตารางต่อไปนี้

ตารางที่ 3.1 เทคโนโลยีที่เกี่ยวข้อง

องค์ประกอบระบบ	เทคโนโลยี
JAS front end	<ul style="list-style-type: none"> - Angular - Share point - RESTful API - HTML5, CSS, JavaScript
JAS back end	<ul style="list-style-type: none"> - MuleSoft - RESTful API - AE API - JAVA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	- SQL
--	-------

3.2 ขั้นตอนการดำเนินงาน

การพัฒนาฟังก์ชัน DEV/ACC Schedule Request แบ่งออกเป็น 2 ส่วนหลักได้แก่ การพัฒนาเว็บไซต์ DEV/ACC Schedule Request Form และการพัฒนาระบบตั้งเวลาอัตโนมัติ ซึ่งประกอบไปด้วย การตรวจสอบ JSF และการสร้างตารางเวลา

3.2.1 เก็บรวบรวมความต้องการ

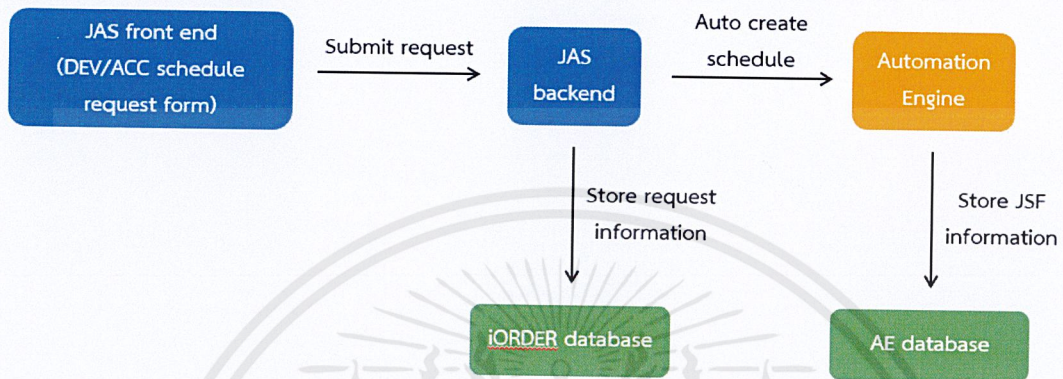
ผู้จัดทำได้ทำการเก็บรวบรวมปัญหาปัจจุบัน รวมถึงความต้องการของผู้ใช้งานระบบ JAS และปัญหาหรือความต้องการของ AE analyst จากนั้นนำมาเรียงลำดับความสำคัญ แล้วเลือกนำมาพัฒนาด้วยข้อจำกัดเรื่องเวลา และทรัพยากร ทำให้ไม่สามารถทำตามความต้องการทั้งหมดได้ ซึ่งสรุปมาได้ดังตารางต่อไปนี้

ตารางที่ 3.2 ความต้องการของผู้ใช้งาน

ลำดับ	ความต้องการของผู้ใช้งาน	เลือกพัฒนา
1	ระบบต้องสามารถแยกประเภทว่า JSF ไหนควร Automate หรือ Manual	✓
2	ระบบต้องสามารถใส่ข้อมูล Time Zone และ Client Number ของ JSF ลงใน Ticket ได้	✓
3	ระบบต้องสามารถ Automate Schedule ที่ไม่มี Dependency และไม่มี Schedule อยู่แล้วได้	✓
4	DEV/ACC schedule request form ต้องสามารถเลือก Time Zone ได้	✓
5	ระบบต้องสามารถส่ง Request ไปยัง JAS Dispatcher ได้	✓
6	ระบบต้องสามารถแจ้งเตือนความคืบหน้าของการทำงานผ่าน Ticket API ได้	✓
7	ระบบต้องสามารถแจ้งเตือนความคืบหน้าของการทำงานผ่าน Email ได้	✗
8	ระบบต้องสามารถแสดงผลกระทบ และแนะนำการแก้ปัญหาเมื่อ JSF เกิดความคลุมเครือได้ผ่าน Email	✗
9	ระบบต้องสามารถให้ผู้ใช้เลือกขั้นตอนต่อไปเมื่อ JSF เกิดความคลุมเครือได้ผ่าน Email	✗
10	ระบบต้องสามารถส่ง Request ไปยัง JAS Dispatcher ได้	✗
11	ระบบต้องสามารถแสดงรายการ DEV/ACC schedule request ของผู้ใช้ได้	✗

3.2.2 ออกแบบโครงสร้างระบบ

ระบบ Automate DEV/ACC Schedule Request นั้นมีการแบ่งการทำงานออกเป็น 3 ส่วนหลักๆได้แก่ Submit request, Validate JSF และ Auto create schedule ซึ่งแต่ละส่วนจะมีการทำงานที่แยกจากกัน แต่จะมีการเก็บข้อมูลลงในฐานข้อมูลเพื่อส่งต่อข้อมูลให้กัน โดยสามารถอธิบายการเชื่อมโยงของแต่ละส่วนได้ด้วยภาพต่อไปนี้



ภาพที่ 3.2 ระบบ Automate DEV/ACC Schedule Request

3.2.3 พัฒนา DEV/ACC Schedule Request Form

ผู้จัดทำได้เพิ่มช่องเลือก Time zone และแก้ไขชุดข้อมูลที่ส่งเข้าเซิร์ฟเวอร์ เพื่อให้ตรงตามความต้องการของลูกค้า

The screenshot shows the 'DEV/ACC Schedule Request Form' in the JAS system. The form includes the following fields and options:

- JSF list ***: A text input field containing the value 'e.g. 123456,234567,345678'.
- Environment ***: A dropdown menu with 'DEV' and 'ACC' options.
- Permanent schedule ***: A dropdown menu with 'Yes' and 'No' options.
- Special requirement**: A large text area for additional notes.
- Characters left: 1000**: A label indicating the remaining character count for the special requirement field.
- Submit**: A green button at the bottom right of the form.

ภาพที่ 3.3 DEV/ACC Schedule Request form permanent schedule

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

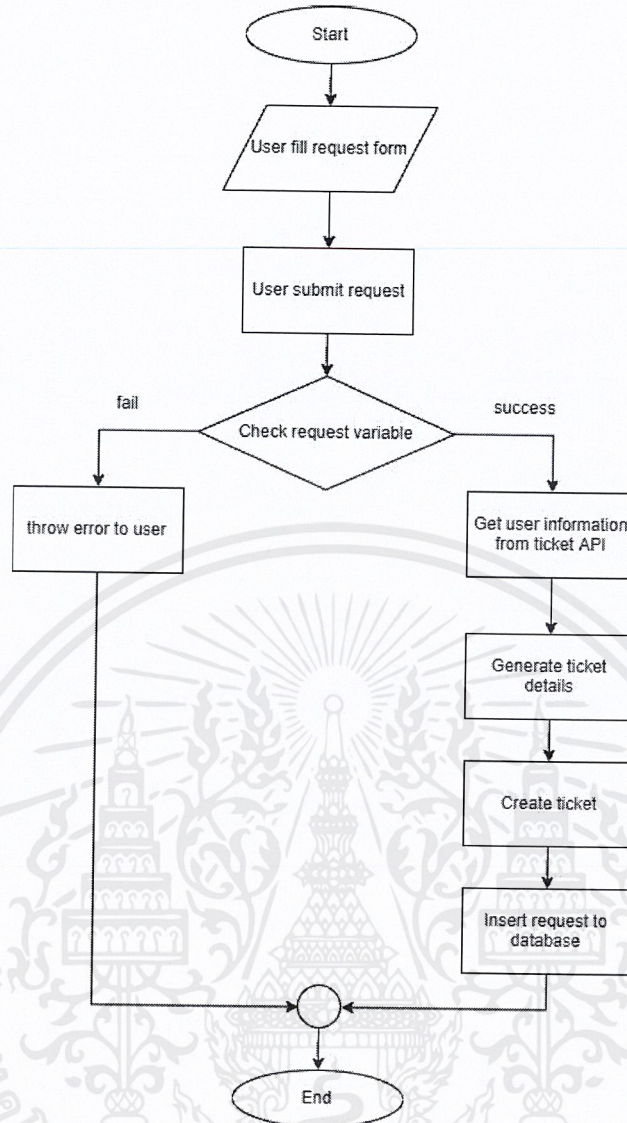
The screenshot shows a web-based form titled "DEV/ACC Schedule Request Form". At the top, there is a search bar and navigation links for "Admin", "AE", "My Request", "Create JSF", and "Utilities". The form itself has several sections:

- JSF list:** A text input field with a placeholder "e.g. 123456,234567,345678".
- Environment:** Two radio buttons, "DEV" and "ACC".
- Permanent schedule:** Two radio buttons, "Yes" and "No".
- Timezone:** A dropdown menu with options: "UTC", "CST", "SGT", "CEI", and "Asia/Bangkok (UTC+7)".
- Start date and time (UTC):** A date and time input field showing "30 10 2019, 17:00".
- End date and time (UTC):** A date and time input field showing "30 10 2019, 17:05".
- Special requirement:** A large text area for additional notes.
- Characters left:** A small indicator showing "1000".
- Submit:** A green button at the bottom right.

ภาพที่ 3.4 DEV/ACC Schedule Request form non-permanent schedule

และหลังจากผู้ใช้งานกดปุ่ม Submit เว็บไซต์จะส่งข้อมูลไปยังระบบซึ่งมีกระบวนการทำงานดังแผนผังกระบวนการทำงานต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

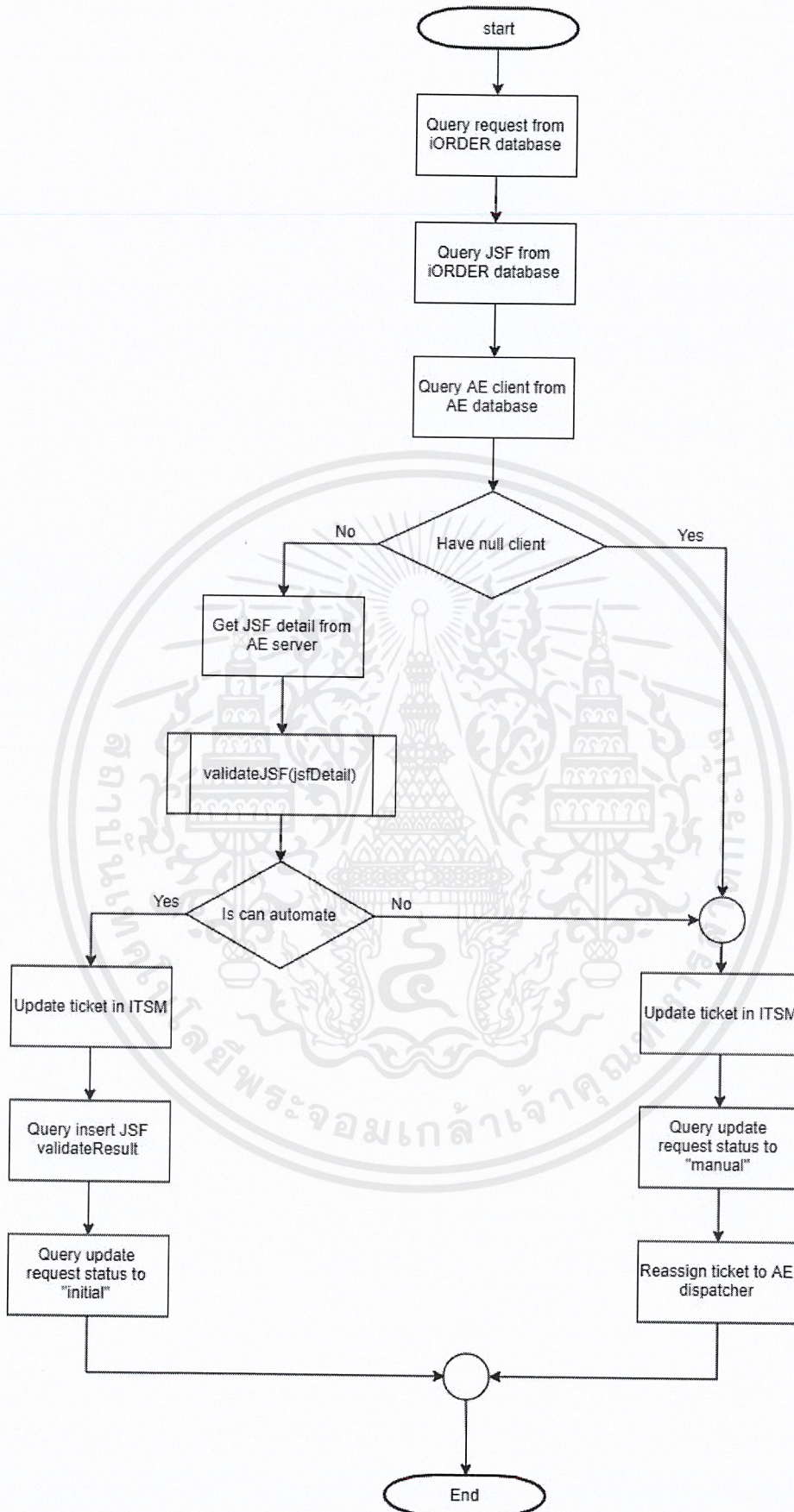


ภาพที่ 3.5 Submit DEV/ACC Schedule Request flowchart

3.2.4 พัฒนาฟังก์ชัน Validate JSF

เป็นส่วนที่ถูกพัฒนาขึ้นมาใหม่ใน MuleSoft เพื่อที่สามารถตรวจสอบว่า JSF ที่อยู่ในคำร่อนั้นสามารถนำไปตั้งเวลาอัตโนมัติได้หรือไม่ โดยต้องทำการรวบรวมข้อมูลจาก iORDER database, AE database, Automation Engine และ ITSM จากนั้นจะเก็บข้อมูลที่ได้ลงในฐานข้อมูลเพื่อนำไปใช้งานในขั้นตอนตั้งเวลาอัตโนมัติต่อไป โดยขั้นตอนการทำงานสามารถอธิบายได้ด้วยแผนผังกระบวนการทำงานต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 3.6 กระบวนการทำงานระบบตรวจสอบ JSF

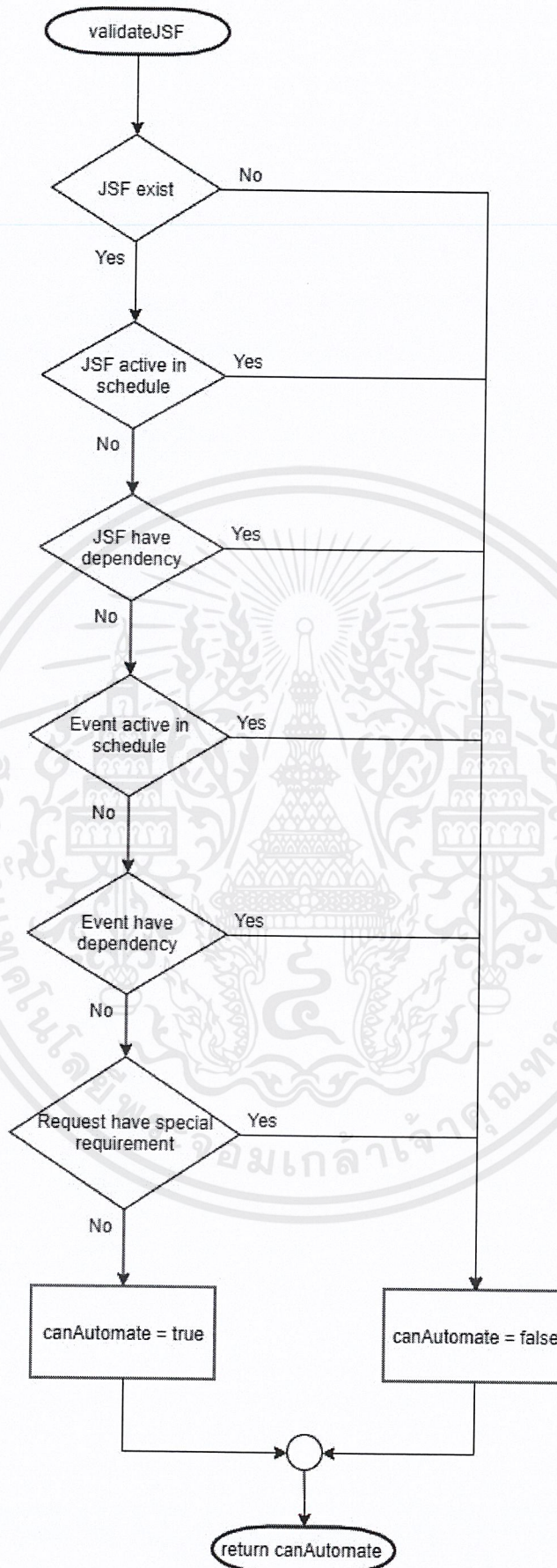
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยฟังก์ชัน validateJSF() นั้นถูกเขียนด้วยภาษา Java เพื่อเก็บรวบรวมข้อมูลของ JSF และนำมาเก็บในตัวแปร validationResult เพื่อใช้ในการวิเคราะห์ต่อไป โดยข้อมูลที่ต้องเก็บรวบรวมมีดังตารางต่อไปนี้

ตารางที่ 3.3 validationResult

ลำดับ	ชื่อข้อมูล	ประเภทข้อมูล
1	JSFExist	Boolean
2	dependency	Boolean
3	JobInSchedule	Boolean
4	JobActiveInSchedule	Boolean
5	events	ArrayList<String>
6	eventDependency	Boolean
7	eventInSchedule	Boolean
8	eventActiveInSchedule	Boolean
9	requestSpecialRequirement	Boolean
10	JSFSpecialRequirement	Boolean

หลังจากทำการเก็บรวบรวมข้อมูลเสร็จสิ้นจะต้องนำข้อมูลดังกล่าวมาตรวจสอบเพื่อสรุปผลว่า JSF ดังกล่าวสามารถตั้งเวลาอัตโนมัติได้หรือไม่ โดยมีขั้นตอนดังแผนภาพการทำงานต่อไปนี้

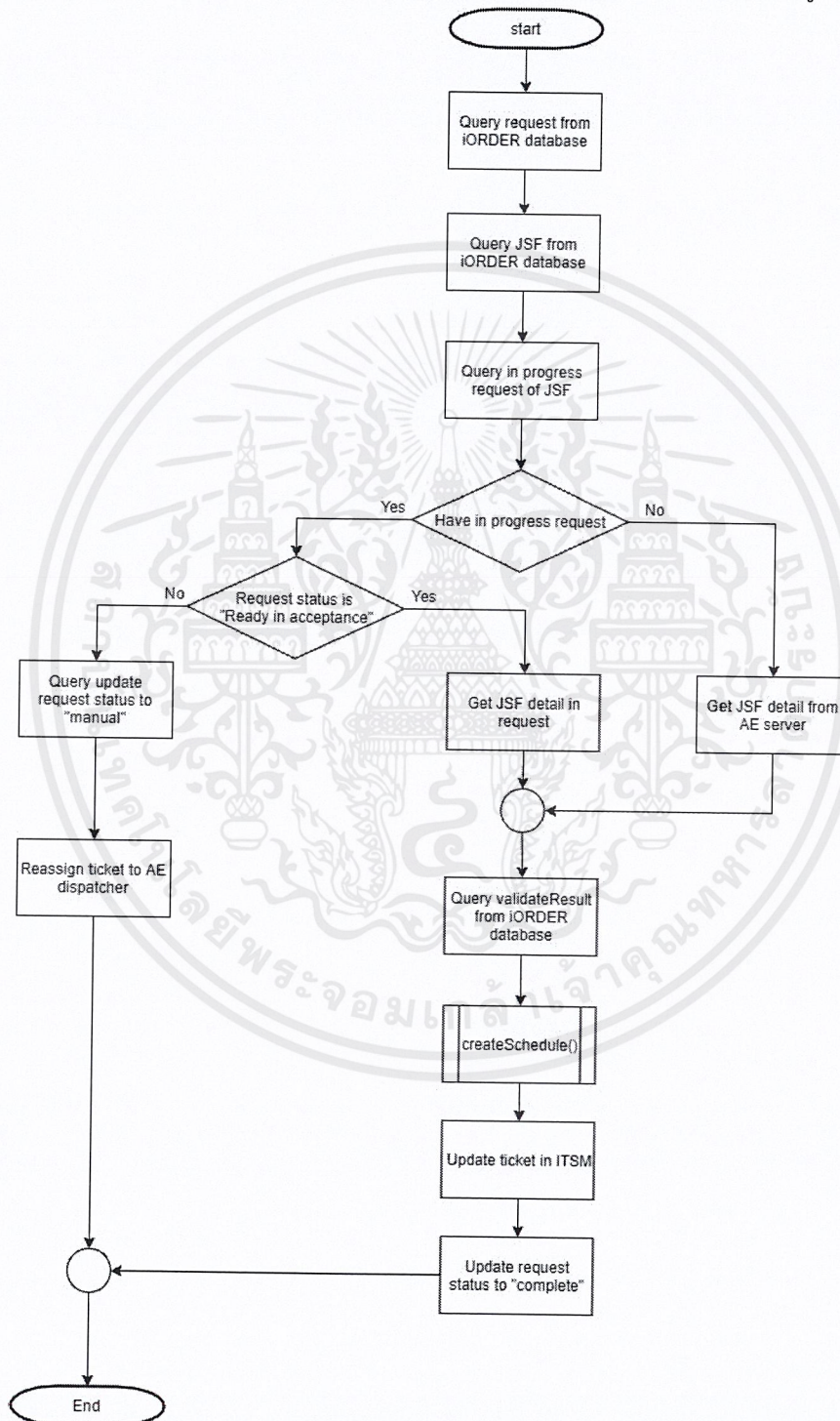


ภาพที่ 3.7 การทำงานตรวจสอบความสามารถในการตั้งเวลาอัตโนมัติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

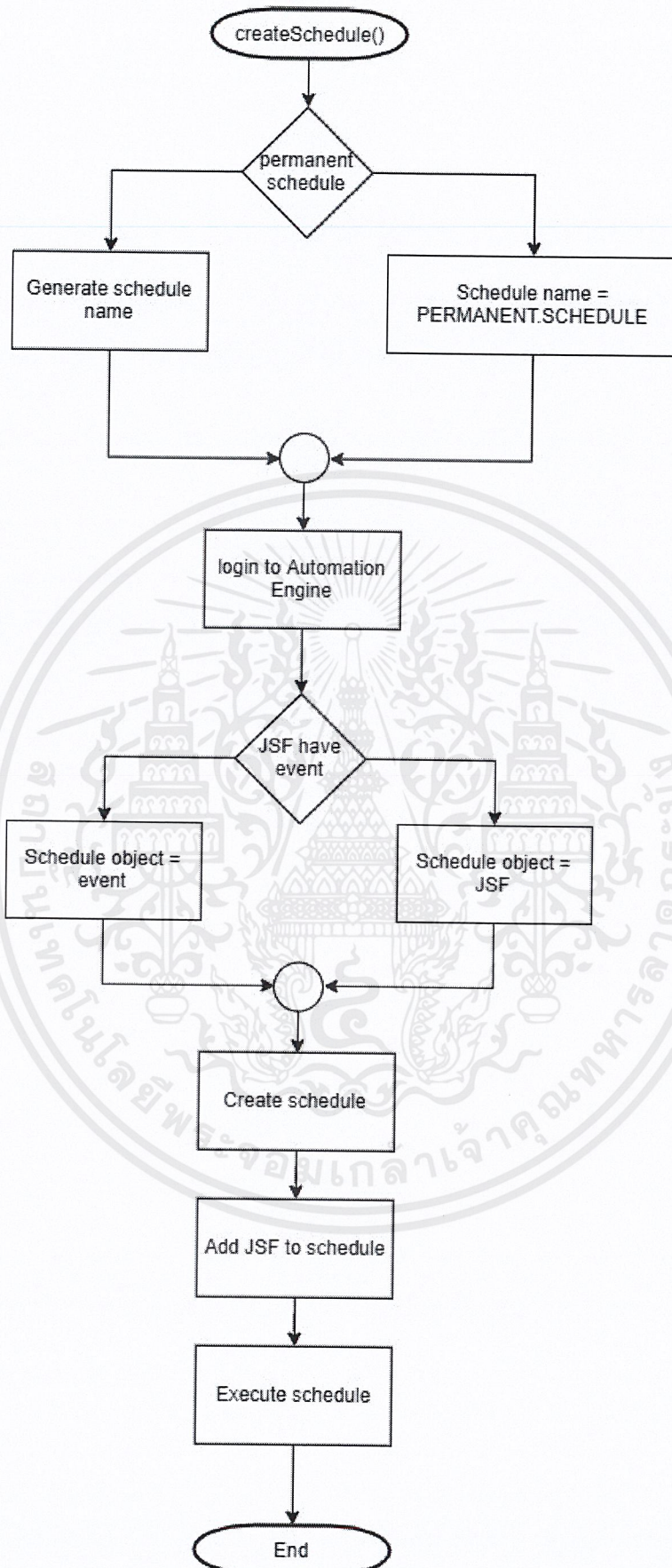
3.2.5 พัฒนาฟังก์ชัน Auto create schedule

หลังจากตรวจสอบความสามารถในการตั้งเวลาอัตโนมัติของคำร้องแล้ว ระบบจะนำคำร้องที่ตรวจสอบแล้วว่าสามารถตั้งเวลาอัตโนมัติได้มา มาเริ่มการตั้งเวลาอัตโนมัติในฟังก์ชัน createSchedule() ที่พัฒนาโดยภาษา JAVA ซึ่งโดยจะทำหน้าที่ตรวจสอบข้อมูล JSF, สร้างตารางเวลา, เพิ่ม JSF ลงในตารางเวลา, สั่งตารางเวลาทำงาน เพื่อให้การตั้งเวลาอัตโนมัติเสร็จสมบูรณ์



ภาพที่ 3.8 การทำงานระบบตั้งเวลาอัตโนมัติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



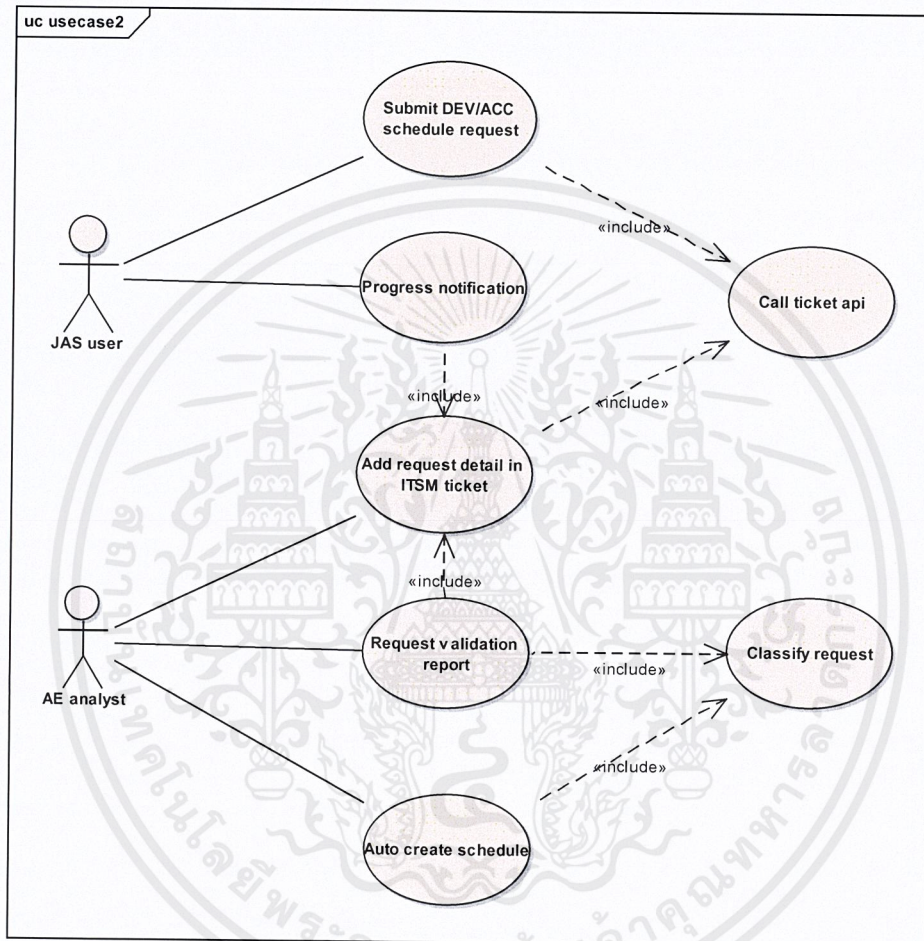
ภาพที่ 3.9 การทำงานฟังก์ชัน createSchedule

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 แผนภาพอธิบายโครงสร้างและการทำงานของระบบ

3.3.1 Use case diagram

โครงสร้างของระบบ Automate DEV/ACC Schedule Request แสดงอยู่ในรูปของแผนภาพยูสเคส (Use case Diagram) เพื่อแสดงหน้าที่ที่ระบบจะต้องกระทำทั้งหมด แผนภาพยูสเคสนี้เกิดจากการเก็บรวบรวมความต้องการของผู้ใช้ (Requirement) ซึ่งเว็บแอปพลิเคชันจะประกอบด้วยยูสเคส ดังนี้



ภาพที่ 3.10 Use case diagram

ตารางที่ 3.4 Submit DEV/ACC schedule request

Use case Title: Submit DEV/ACC schedule request
Primary Actor: JAS user
Brief Description: สำหรับ Submit DEV/ACC schedule request ทำหน้าที่รับข้อมูลจากผู้ใช้งาน ผ่านฟอร์มในรูปแบบที่กำหนด ซึ่งประกอบด้วย JSF number, Environment, Permanent schedule, Start date, End date และ Special requirement จากนั้นจะนำข้อมูลที่ได้ออกไปเก็บลงในฐานข้อมูลรวมถึง ส่งข้อมูลบางส่วนไปยัง ITSM และสร้าง Ticket เพื่อจะได้เก็บข้อมูลไว้ในฐานข้อมูลกลางของบริษัท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Related Use Case: -
<p>Main Success Scenario:</p> <ol style="list-style-type: none"> 1. ผู้ใช้งานคลิกเมนู Utilities เพื่อเลือก DEV/ACC Schedule Request Form 2. ผู้ใช้งานกรอก JSF number ลงในช่อง JSF list 3. ผู้ใช้งานเลือก Environment ที่จะให้ Schedule ระหว่าง DEV และ ACC 4. ผู้ใช้งานเลือก Permanent Schedule ระหว่าง YES และ NO 5. กรณีผู้ใช้งานเลือก Permanent Schedule เป็น No <ul style="list-style-type: none"> - ผู้ใช้งานเลือก Time Zone ซึ่งประกอบด้วย UTC, CST, SGT, CET และเวลา Local ของเครื่องผู้ใช้งาน - ผู้ใช้งานเลือก Start date and time โดยจะมี Moment selector ให้เลือกเวลาเริ่มจาก ปี > เดือน > วัน > ชั่วโมง > นาที - ผู้ใช้งานเลือก End date and time โดยจะมี Moment selector ให้เลือกเวลาเริ่มจาก ปี > เดือน > วัน > ชั่วโมง > นาที 6. กรณีผู้ใช้งานเลือก Permanent Schedule เป็น Yes จะขึ้นช่องกรอกข้อมูล Time zone, Start date and time และ End date and time 7. ผู้ใช้งานสามารถเลือกที่กรอก Special requirement เพิ่มเติมหรือไม่ก็ได้ 8. ผู้ใช้งานกด Submit เพื่อส่งข้อมูล 9. เว็บไซต์แสดงข้อความยืนยันการส่งข้อมูลสำเร็จ และแสดงเลข Ticket number ของระบบ ITSM
<p>Extension:</p> <ol style="list-style-type: none"> 1. ผู้ใช้งานกรอก JSF number ผิดรูปแบบที่กำหนด ช่องกรอกข้อมูลจะเปลี่ยนเป็นสีแดง และมีความเตือน 2. เมื่อผู้ใช้งานกรอกข้อมูลไม่ครบเมื่อกด Submit จะขึ้นกรอกสีแดงที่ช่องที่ยังไม่ได้กรอกข้อมูล 3. เมื่อการส่งข้อมูลผิดพลาด จะแสดงข้อความว่า Failed to submit request

ตารางที่ 3.5 Progress notification

Use case Title: Progress notification
Primary Actor: JAS user
<p>Brief Description:</p> <p>สำหรับ Get progress notification ทำหน้าที่แจ้งเตือนผู้ใช้งานผ่านทาง ITSM เมื่อคำร้องของผู้ใช้งานส่งมาเกิดการเปลี่ยนแปลงสถานะเช่น คำร้องถูกจัดการโดยอัตโนมัติแล้ว จะมีการเพิ่มข้อมูลไปในคำร้อง และ ITSM จะส่งอีเมลไปหาเจ้าของคำร้องอัตโนมัติ</p>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Related Use Case: Auto create schedule
Main Success Scenario: <ol style="list-style-type: none"> 1. คำร้องมีการเปลี่ยนแปลงสถานะ 2. ระบบรวบรวมข้อมูลของคำร้อง 3. ระบบส่งข้อมูลให้ ITSM 4. ITSM ส่งอีเมลหาเจ้าของคำร้อง
Extensions: -

ตารางที่ 3.4 Add request detail in ITSM ticket

Use case Title: Add request detail in ITSM ticket
Primary Actor: JAS user, AE analyst
Brief Description: <p>สำหรับ Add request detail in ITSM ticket ทำหน้าที่ส่งข้อมูลที่ต้องการเพิ่มใน work details ของ ticket โดยใช้ ticket API ของ ITSM</p>
Related Use Case: Progress notification, Request validation report
Main Success Scenario: <ol style="list-style-type: none"> 1. ระบบรับข้อมูลที่ต้องการเพิ่มลงใน work details 2. ส่งข้อมูลให้ ITSM ผ่าน ticket API
Extensions: <ol style="list-style-type: none"> 1. ถ้าระบบ Ticket API ไม่สามารถใช้งานได้จะส่ง error กลับไปหาผู้ใช้งาน

ตารางที่ 3.5 JSF validation report

Use case Title: JSF validation report
Primary Actor: AE analyst
Brief Description: <p>สำหรับ JSF validation report ทำหน้าที่นำข้อมูลของ JSF ที่ผ่านการวิเคราะห์แล้ว มา จัดเรียงเป็นรูปแบบ แล้วนำไปเพิ่มลงใน work details เพื่อให้ AE analyst สามารถนำข้อมูลไปใช้งานได้</p>
Related Use Case: Add request detail in ITSM ticket, Classify request
Main Success Scenario: <ol style="list-style-type: none"> 1. รับผลการวิเคราะห์ JSF 2. นำข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Extensions:

1. ถ้าระบบ Ticket API ไม่สามารถใช้งานได้จะส่ง error กลับไปหาผู้ใช้งาน

ตารางที่ 3.6 Auto create schedule

Use case Title: Auto create schedule
Primary Actor: AE analyst
Brief Description: สำหรับ Auto create schedule ทำหน้าที่สร้างตารางเวลา และตั้งเวลาทำงานอัตโนมัติ โดยจะมีการนำข้อมูล JSF ที่ผ่านการวิเคราะห์แล้วมาประมวลผลว่าต้องตั้งเวลาทำงานอัตโนมัติอย่างไร รวมถึงดำเนินการด้วย
Related Use Case: Classify request
Main Success Scenario: <ol style="list-style-type: none"> 1. รับผลการวิเคราะห์ JSF 2. ตรวจสอบข้อมูลของ JSF หาวิธีการตั้งเวลาที่เหมาะสม 3. สร้างตารางเวลา 4. เพิ่ม JSF ลงในตารางเวลา 5. ตั้งเวลาทำงานอัตโนมัติ
Extensions: <ol style="list-style-type: none"> 1. เมื่อข้อมูลของ JSF ไม่สามารถตั้งเวลาทำงานอัตโนมัติได้ จะยุติการทำงาน

ตารางที่ 3.7 Classify request

Use case Title: Classify request
Primary Actor:
Brief Description: สำหรับ Classify request ทำหน้าที่รวบรวมข้อมูลทั้งหมดของ JSF จาก Automation Engine, JAS, ITSM, iORDER database และ AE database เพื่อตรวจสอบว่า JSF นั้นๆสามารถนำไปตั้งเวลาอัตโนมัติได้หรือไม่
Related Use Case: Classify request
Main Success Scenario: <ol style="list-style-type: none"> 1. รับรายการ JSF จากคำร้องใน ITSM 2. ค้นหาข้อมูลจาก Automation Engine, JAS, ITSM, iORDER database และ AE database 3. วิเคราะห์ข้อมูลเพื่อแบ่งประเภท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. เก็บผลลัพธ์ลงฐานข้อมูล

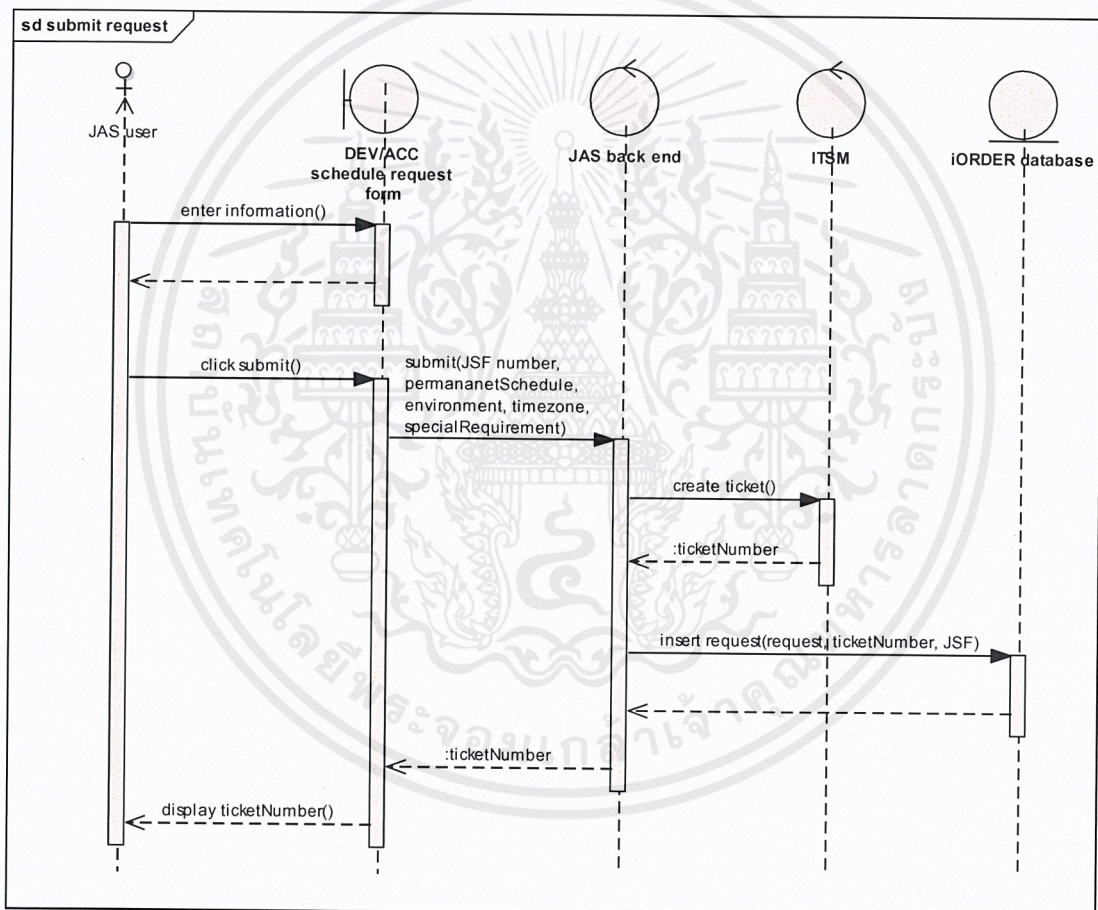
Extensions:

1. เมื่อ JSF ไม่มีอยู่จริง จะหยุดการทำงาน

3.3.2 Sequential diagram

ซึ่งขั้นตอนการทำงานทั้งหมดของระบบทั้งสามขั้นตอนคือ Submit request, validate JSF และ auto create schedule สามารถนำมาอธิบายลำดับการทำงานและการ ปฏิสัมพันธ์ (Sequence of Interactions) ระหว่างอ็อบเจกต์ ตามลำดับของเวลาจากก่อนไปหลัง ซึ่งผู้จัดทำได้แบ่ง Sequential diagram ตามการทำงานดังนี้

1. ฟังก์ชัน Submit request



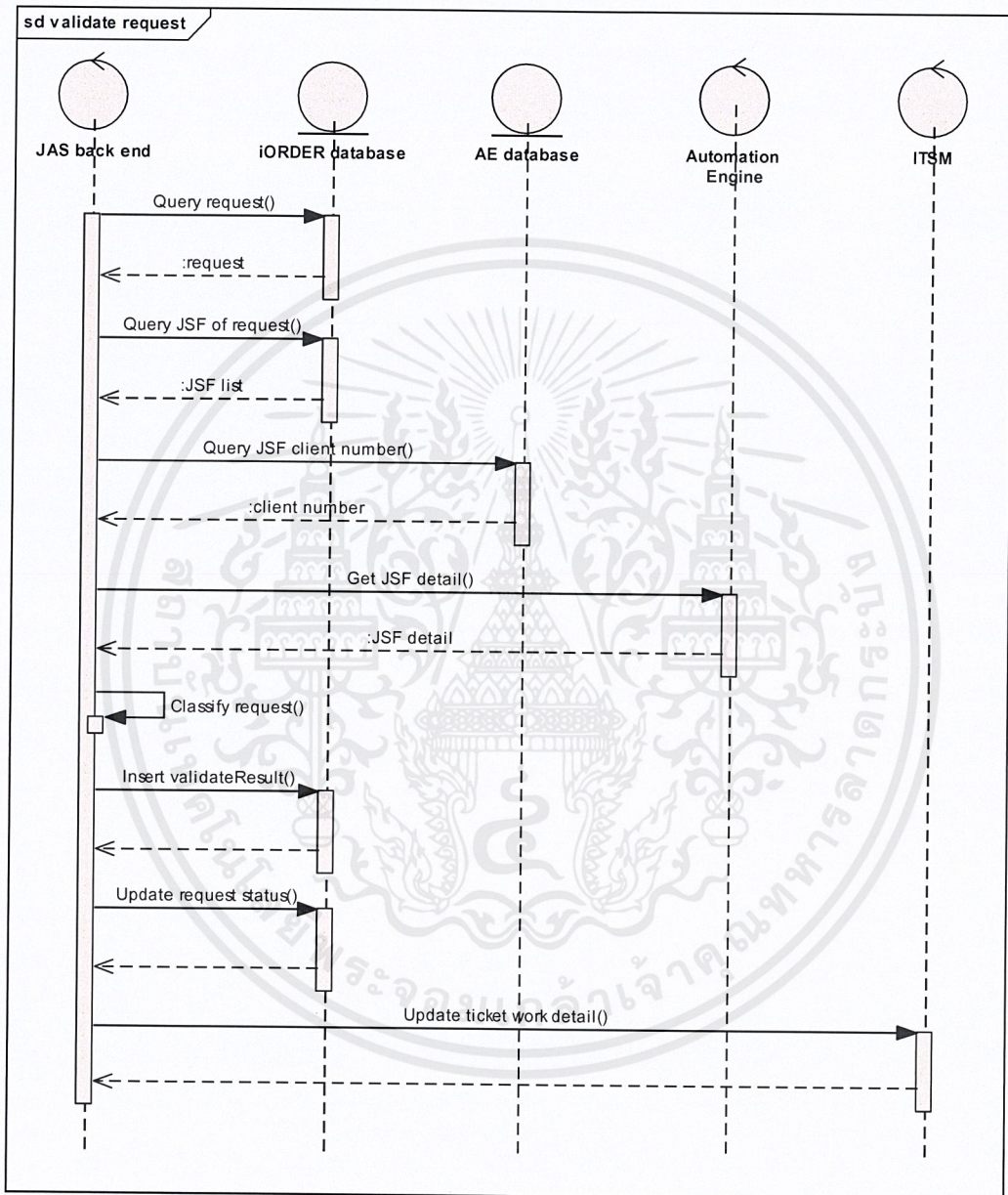
ภาพที่ 3.11 ฟังก์ชัน submit request

จากภาพที่ 3.8 แสดงแผนภาพการทำงานลำดับเหตุการณ์ของฟังก์ชัน submit request ซึ่งเริ่มจากผู้ใช้งานกรอกข้อมูลต่างๆลงยังหน้าเว็บไซต์ DEV/ACC schedule request form ซึ่งประกอบไปด้วย JSF number, environment, permanent schedule, time zone, schedule time และ special requirement หลังจากนั้นผู้ใช้งานกดปุ่ม Submit เพื่อส่งข้อมูลที่กรอกเข้ามาয়ระบบ โดย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบจะทำการ ส่งข้อมูลดังกล่าวไปยัง ITSM ผ่าน ticket API เพื่อสร้าง ticket ไว้สำหรับติดตามการทำงานและเป็นข้อมูลส่วนกลางของบริษัท เมื่อการสร้าง ticket เสร็จสิ้น ITSM จะคืนค่า ticket number กลับมาเพื่อใช้อ้างอิงถึง ticket นั้นๆ ต่อมาระบบจะนำข้อมูลที่ผู้ใช้งานส่งเข้ามาพร้อมกับข้อมูล ticket number เพิ่มเข้าไปใน iORDER database เพื่อเก็บข้อมูลไว้ใช้งานในลำดับต่อไป

2. ฟังก์ชัน Validate JSF



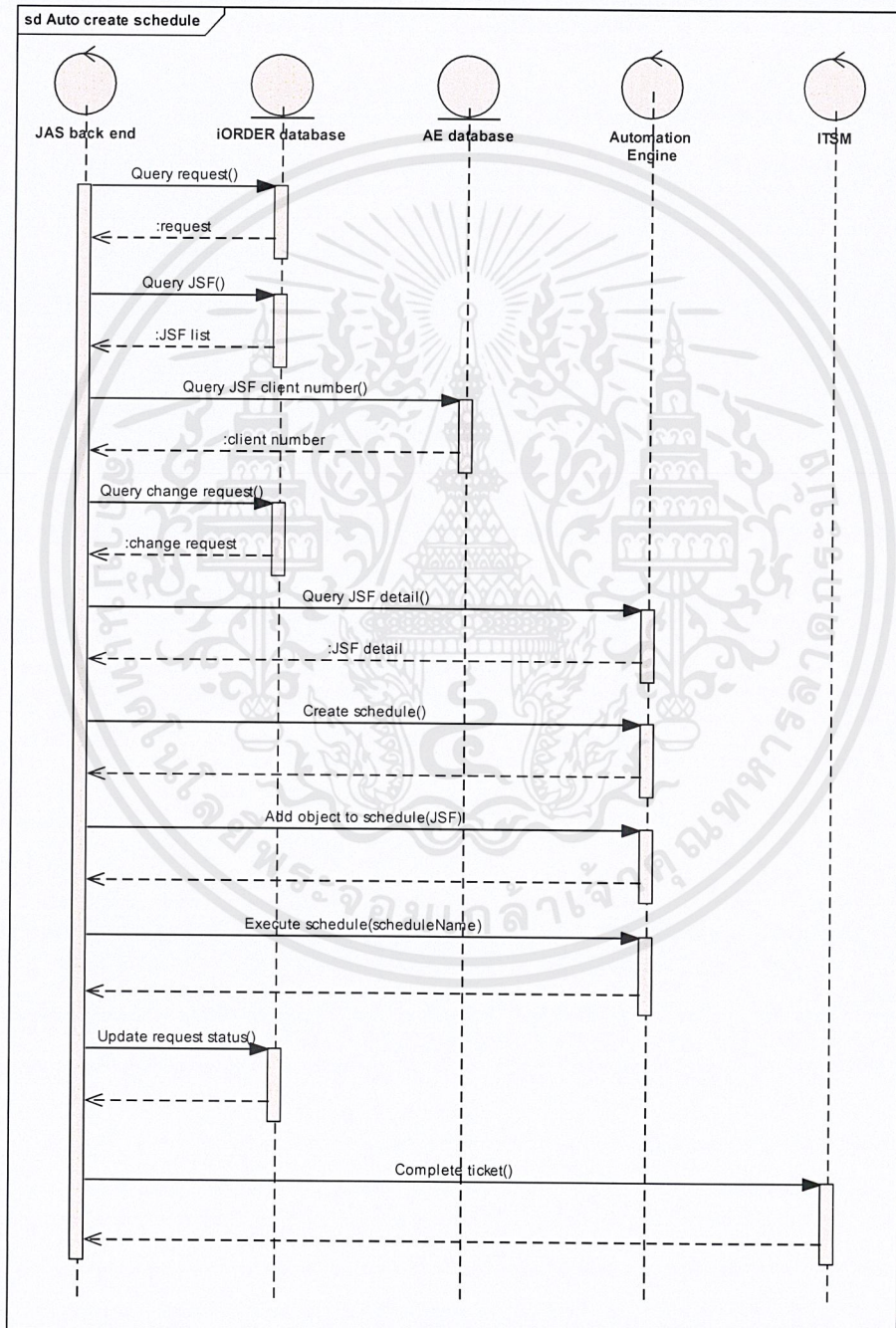
ภาพที่ 3.12 ฟังก์ชัน validate request

จากภาพที่ 3.9 แสดงแผนภาพการทำงานลำดับเหตุการณ์ของฟังก์ชัน validate request โดยเริ่มหลังจากที่ผู้ใช้งานได้ Submit request เข้ามายังระบบในหัวข้อก่อนหน้า ซึ่งระบบจะมีการตรวจสอบคำร้องทุกๆ 5 นาทีว่ามีคำร้องใหม่หรือไม่ ถ้าตรวจพบคำร้องใหม่จึงจะเริ่มการทำงาน validate request โดยจะดึงข้อมูลคำร้องมาอ่านค่า JSF จากนั้นจะนำเลข JSF ไปค้นหา client ใน AE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

database เพื่อให้ทราบว่า JSF นี้อยู่ใน AE server ไหน จากนั้นจะเข้าไปค้นหาข้อมูลของ JSF ใน AE server นั้นๆ แล้วนำข้อมูลทั้งหมดมาวิเคราะห์เพื่อแยกแยะว่า JSF สามารถนำไปตั้งเวลาอัตโนมัติได้หรือไม่ เมื่อวิเคราะห์เสร็จสิ้นแล้วจะนำผลลัพธ์บันทึกลงฐานข้อมูลเพื่อใช้ในขั้นตอนต่อไป รวมถึงบันทึกลง ticket ใน ITSM เพื่อให้สามารถติดตามได้ง่าย และเปลี่ยนสถานะของคำร้องในฐานข้อมูลจาก submitted เป็น initial ในกรณีที่สามารถตั้งเวลาอัตโนมัติได้ แต่ถ้าไม่สามารถตั้งเวลาอัตโนมัติได้จะเปลี่ยนสถานะจาก submitted เป็น manual

3. ฟังก์ชัน Auto create schedule



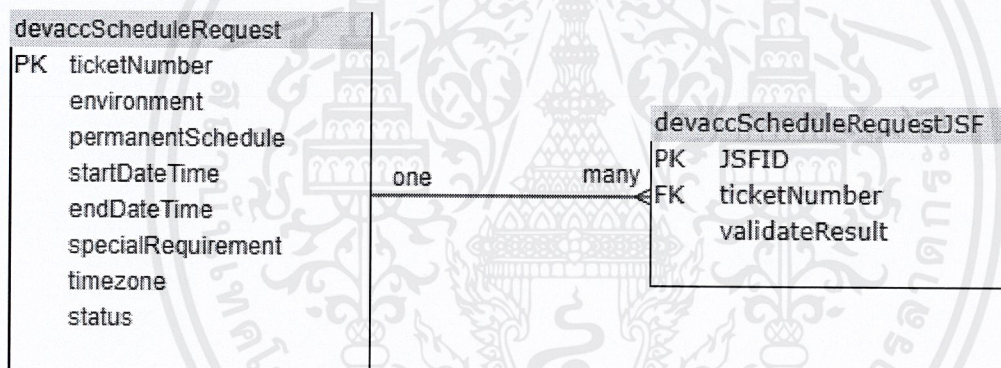
ภาพที่ 3.13 ฟังก์ชัน Auto create schedule

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากภาพที่ 3.10 แสดงแผนภาพการทำงานลำดับเหตุการณ์ของฟังก์ชัน Auto create schedule ซึ่งจะตรวจสอบว่ามีคำร้องที่ผ่านการตรวจสอบแล้วว่าสามารถตั้งเวลาอัตโนมัติได้ทุกๆ 30 นาที เพื่อนำมาตั้งเวลาอัตโนมัติ โดยเริ่มจากการรวบรวมข้อมูลของ JSF จาก iORDER database, AE database และ Automation Engine เพื่อตัดสินใจว่าจะตั้งเวลาอัตโนมัติด้วยการตั้งค่าใด หลังจากนั้นจะทำการสร้างตารางเวลาลงใน Automation Engine ที่ JSF นั้นๆ อยู่แล้วจึงเพิ่ม JSF ที่ต้องการตั้งเวลาลงในตารางเวลานั้น ตามด้วยการสั่งเริ่มทำงานตารางเวลาตามข้อมูล ซึ่งนับว่าเป็นการเสร็จสิ้นขั้นตอนการตั้งเวลาอัตโนมัติ จากนั้นจึงแก้ไขสถานะคำร้องในฐานข้อมูลจาก initial เป็น complete และถ้าเกิดปัญหาระหว่างการตั้งเวลาอัตโนมัติจะแก้ไขสถานะคำร้องจาก initial เป็น manual และส่งคำร้องให้ AE analyst ดำเนินการต่อ และสุดท้ายก็คือการเปลี่ยนสถานะของ ticket ใน ITSM เป็น complete เพื่อแสดงว่าคำร้องของผู้ใช้งานได้ถูกจัดการแล้ว

3.3.3 Database schema diagram

ในส่วนของการออกแบบรายละเอียดฐานข้อมูล ผู้จัดทำได้นำเอา Schema Diagram มาใช้ในการออกแบบและอธิบายโครงสร้างของฐานข้อมูลทั้งหมด



ภาพที่ 3.14 Database schema diagram

จากภาพ แสดงถึงโครงสร้างฐานข้อมูลทั้งหมดของส่วน Automatic DEV/ACC schedule request ซึ่งในแต่ละส่วนมีรายละเอียดดังนี้

ตารางที่ 3.8 devaccScheduleRequest

Field	Key	Data type	Description	Example data
ticketNumber	PK	Varchar(20)	เลขของ Ticket ใน ITSM	WO0000015312728
environment		Varchar(3)	Environment ของ Schedule	DEV, ACC
permananetSchedule		Varchar(3)	บ่งบอกว่า Request นี้ต้องการให้ schedule เป็น Permanent หรือไม่	Yes, No

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

startDateTime		Datetime	เวลาเริ่มต้น Schedule	2019-10-30 03:40:00.000
endDateTime		Datetime	เวลาสิ้นสุด Schedule	2019-10-30 03:45:00.000
specialRequirement		Text	ข้อมูลเพิ่มเติมจากผู้ใช้งาน	"test in dev"
timezone		Varchar(20)	Time zone	UTC
status		Varchar(20)	สถานะของ Request	submitted, initial, manual, complete

โดยใน devaccScheduleRequest จะมีข้อมูล status ซึ่งจะหมายถึงสถานะของคำร้องนั้นๆ โดยสถานะทั้งหมด 4 สถานะ ดังตารางต่อไปนี้

ตารางที่ 3.9 ความหมายของ status

Status	Meaning
submitted	สถานะเมื่อ request ถูกสร้าง
initial	สถานะที่ request เสร็จสิ้นการตรวจสอบว่าสามารถ automate ได้หรือไม่
manual	Request ที่ไม่สามารถ automate ไม่ว่าจะมาจากการตรวจสอบ หรือการเกิด error ในขั้นตอนการทำงานใดๆก็ตาม
Complete	Request ที่ auto create schedule เสร็จเรียบร้อย

ตารางที่ 3.10 devaccScheduleRequestJSF

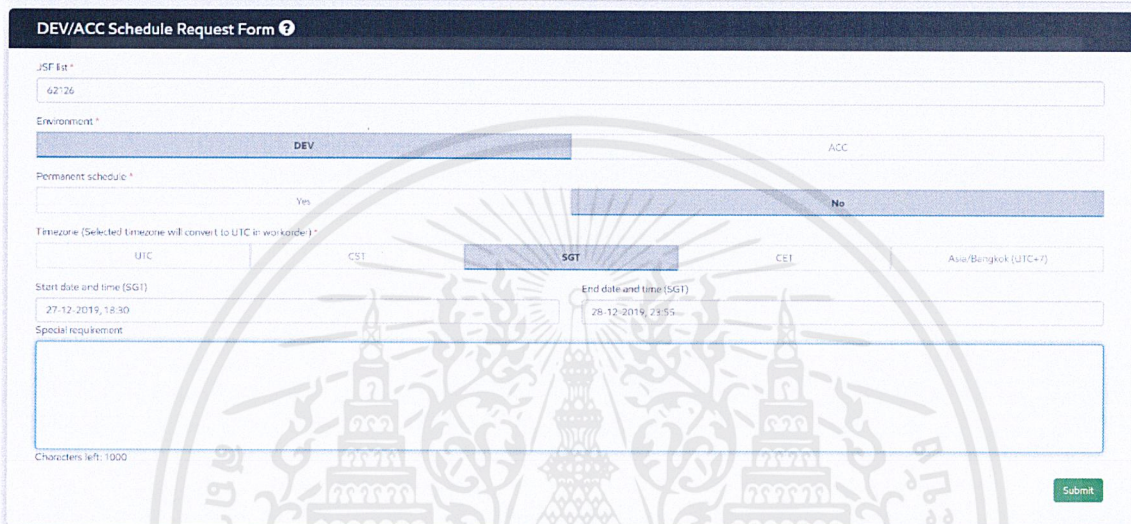
Field	Key	Data type	Description
JSFID	PK	Int(10)	เลขของ JSF
ticketNumber	FK	Varchar(20)	เลขของ Ticket ใน ITSM
validateResult		Text	ผลลัพธ์ของการตรวจสอบแต่ละ JSF เก็บในรูปแบบข้อมูล JSON

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4 ผลการวิจัย

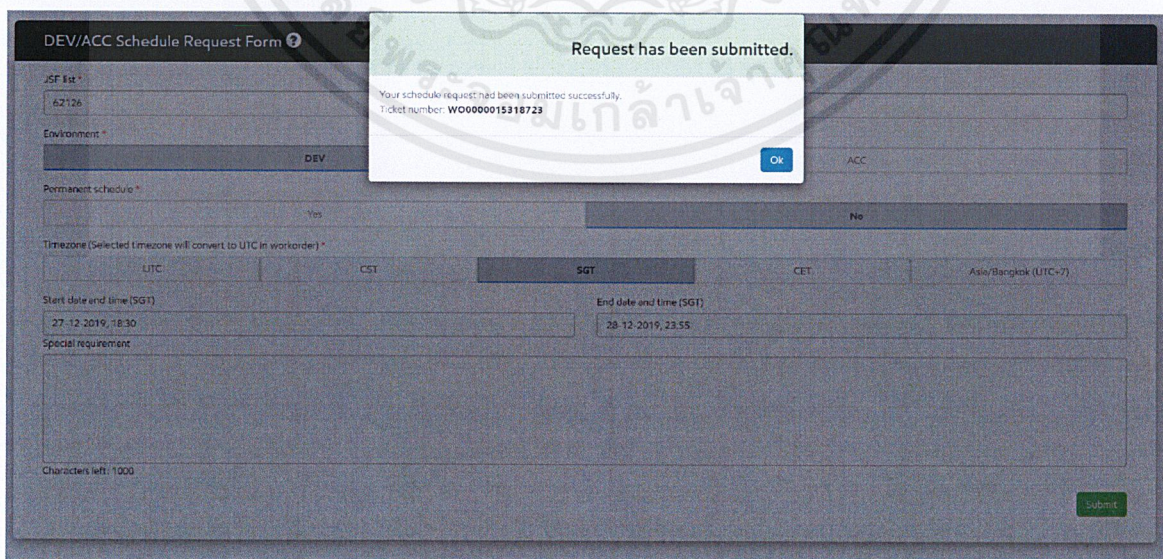
ผลลัพธ์ของโครงการนี้คือการใช้งานที่สามารถส่งคำร้องการตั้งเวลาเข้าสู่ระบบได้ แล้วระบบสามารถแยกแยะคำร้องที่ตั้งเวลาอัตโนมัติได้ และทำการตั้งเวลาอัตโนมัติเพื่อลดการทำงานของ AE analyst ลงรวมถึงลดเวลารอของผู้ส่งคำร้อง ซึ่งบ่งชี้จะเป็นการแสดงผลภาพของขั้นตอนต่างๆในการทำงาน ตั้งแต่การส่งคำร้อง จนถึงการตั้งเวลาอัตโนมัติ

4.1 ฟังก์ชัน Submit request



ภาพที่ 4.1 DEV/ACC schedule request form

จากภาพที่ 4.1 คือภาพแสดงถึงหน้าเว็บไซต์ DEV/ACC schedule request form ที่ผู้ใช้งานได้มีการกรอกข้อมูล JSF เท่ากับ 62126 Environment เท่ากับ DEV และตัวแปรอื่นๆอีกดังรูป



ภาพที่ 4.2 Submit request

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจากผู้ใช้งานกดปุ่ม Submit ด้านขวาล่าง จะมีการส่งข้อมูลไปยังระบบเพื่อสร้าง ticket ใน ITSM ให้แล้วจากนั้นจะแสดง ticket number ออกทางเว็บไซต์เพื่อให้ผู้ใช้งานสามารถรู้เลข ticket ประจำคำร้องของตนเองได้

[Back](#) [Home](#) [REG000025836452](#) [WO0000015318723](#)

[JAS-General request] DEV/ACC schedule request

On Saturday, Dec 14 2019 at 08:51 PM

LOW PRIORITY TAG

[Notes](#)

[Work Details](#)

[Approvals](#)

[Tasks](#)

[Relationships](#)

Last Modified:
Dec 14 2019 at 08:51 PM

Last Modified By:
Not Defined (remedy application service)

Notes

*** BASE AE change request form created by JAS ***

[Edit Notes](#)

JSF : 62126

Details of Changes : Schedule.JSF

Environment : DEV

Permanent Schedule : No

Schedule Period (UTC) : 2019-12-27 10:30:00 - 2019-12-28 15:55:00

ภาพที่ 4.3 ticket ใน ITSM

4.2 ฟังก์ชัน Validate JSF

โดยในฟังก์ชัน Validate JSF นี้จะตรวจสอบ JSF ของคำร้องและสร้าง report ขึ้นมา จากนั้นจะนำมาใส่ไว้ใน work details ของ ticket นั้นๆ เพื่อให้ AE analyst นำไปใช้งานได้ในอนาคต

[Notes](#)

[Work Details](#)

[Approvals](#)

[Tasks](#)

[Relationships](#)

Last Modified:
Dec 14 2019 at 08:56 PM

Last Modified by:
AE - Automation Development (nawtopand)

Work Details (1)

[New Work Detail](#)

JAS automation DEV/ACC schedule log :
Start collecting JSF information

>Validate jsf client process
62126 : 981

>Validate jsf details process
JSFID : 62126

Dependency : false
JOBP in schedule : true
JOBP activate in schedule : true

Events : []
Event in common : false
Event in schedule : false
Event active in schedule : false
JSF have special requirement : false
Request have special requirement : false

This request can't automate create schedule by JAS
Ticket will reassigned to dispatcher

[Read Less](#)

AE-Automation Development

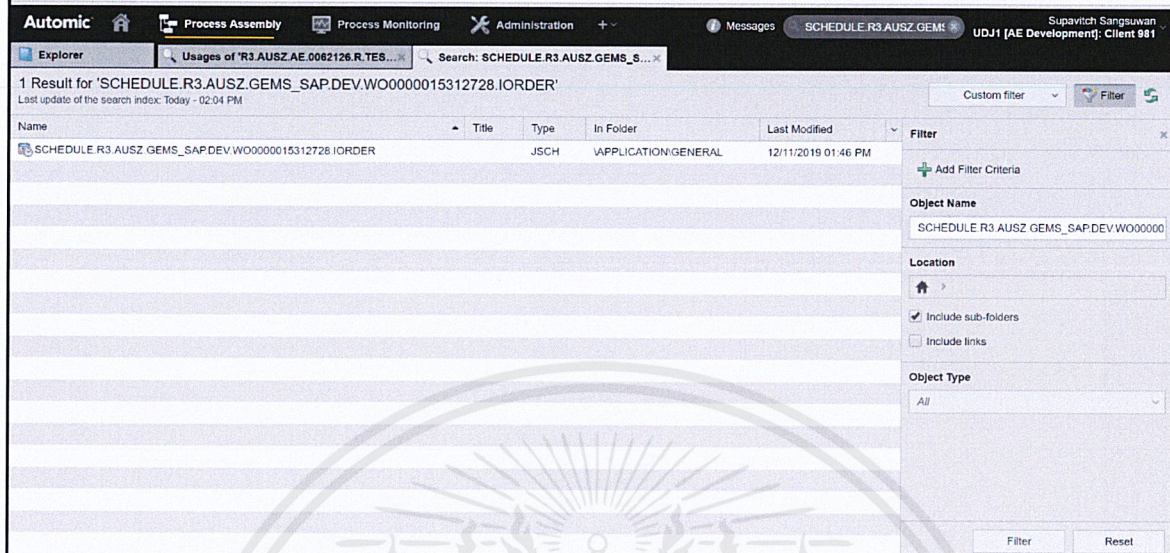
Dec 14 at 08:56 PM

ภาพที่ 4.4 validate JSF report

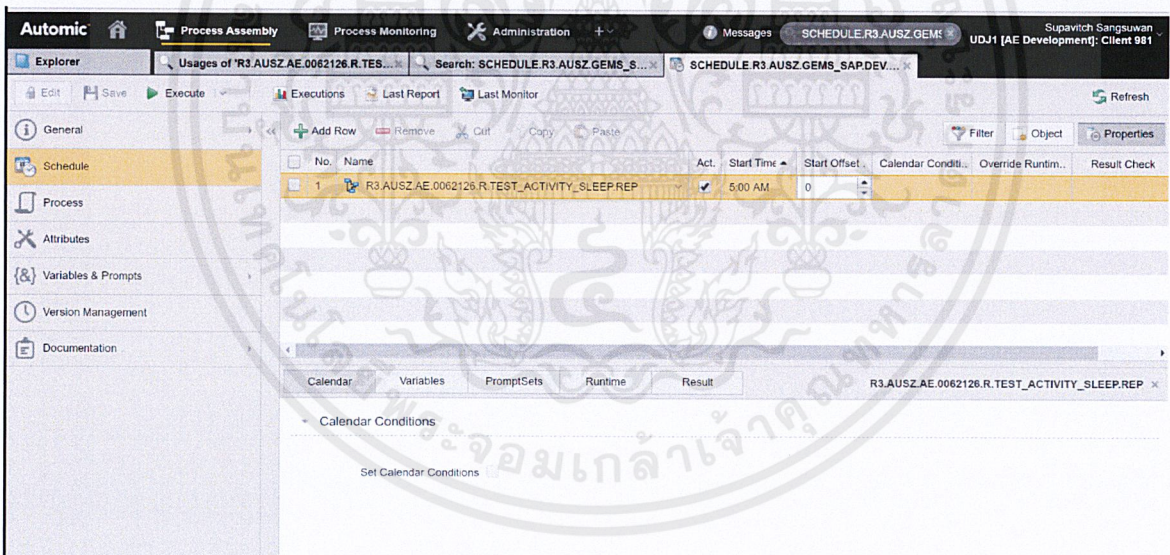
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 ฟังก์ชัน Auto create schedule

เป็นขั้นตอนที่จะสร้างตารางเวลาขึ้นมาจากนั้นนำ JSF ไปใส่ในตารางเวลาแล้วจึงเริ่มการทำงาน ซึ่งผลลัพธ์ของการพัฒนาแสดงให้เห็นว่า ระบบสามารถตั้งเวลาอัตโนมัติได้



ภาพที่ 4.5 Schedule ที่ถูกสร้างโดยระบบ



ภาพที่ 4.6 JSF ใน Schedule

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

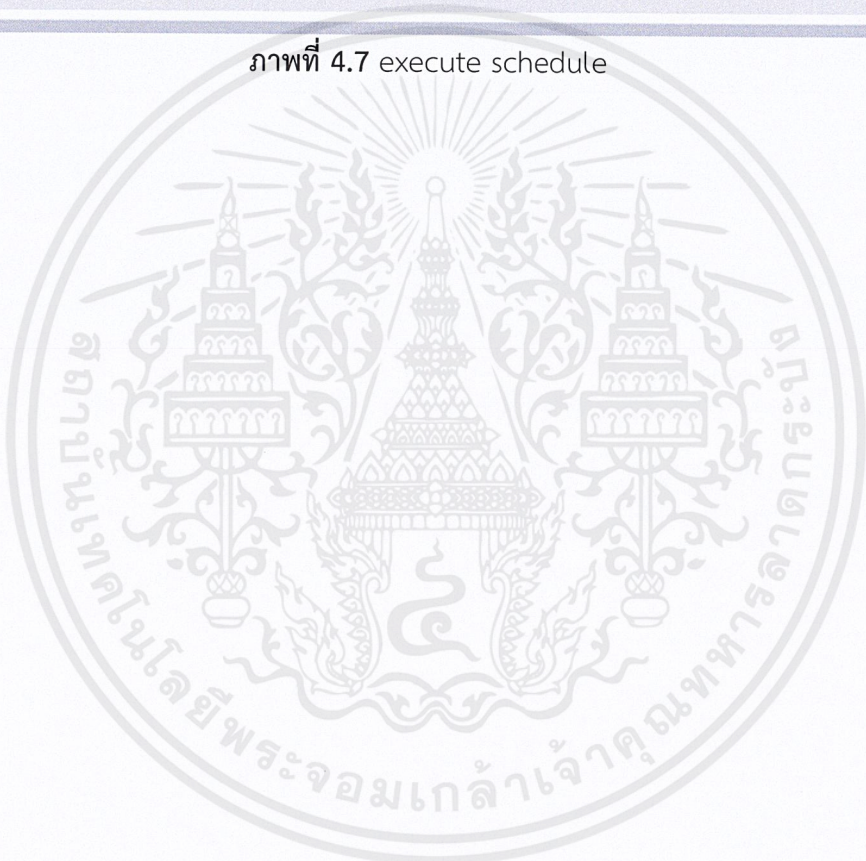
Automic Process Assembly Process Monitoring Messages SCHEDULE.R3.AUSZ.GEM UDJ1 [AE Development]; Client 981 Supavitch Sangsuan

Custom filter Executions of SCHEDULE.R3.AUSZ.GE...

Executions of SCHEDULE.R3.AUSZ.GEMS_SAP.DEV.WO0000015312728.IORDER (5)
Last Update: 12/14/2019 02:08 PM Details

Name	Type	Platform	RunID	Parent	Status	Activation Time	Start Time	End Time
▼ SCHEDULE.R3.AUSZ.GEMS_SAP.DEV.WO0000015312728.IORDER	JSCH		24017041		Waiting for Queue slot	12/14/2019 02:07 PM	12/14/2019 02:07 PM	
No tasks available				24017041				
▶ SCHEDULE.R3.AUSZ.GEMS_SAP.DEV.WO0000015312728.IORDER	JSCH		24017040		ENDED_OK - ended normally	12/14/2019 02:07 PM		12/14/2019 02:07 PM
▶ SCHEDULE.R3.AUSZ.GEMS_SAP.DEV.WO0000015312728.IORDER	JSCH		23710478		ENDED_OK - ended normally	12/11/2019 01:46 PM	12/11/2019 01:46 PM	12/12/2019 02:42 AM
▶ SCHEDULE.R3.AUSZ.GEMS_SAP.DEV.WO0000015312728.IORDER	JSCH		23709654		ENDED_OK - ended normally	12/11/2019 01:46 PM		12/11/2019 01:46 PM
▶ SCHEDULE.R3.AUSZ.GEMS_SAP.DEV.WO0000015312728.IORDER	JSCH		23679841		ENDED_OK - ended normally	12/11/2019 08:25 AM		12/11/2019 08:26 AM

ภาพที่ 4.7 execute schedule



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลการวิจัย และข้อเสนอแนะ

5.1 สรุปผลการวิจัย

ระบบ Automate development and acceptance schedule สามารถแยกแยะ JSF , เพิ่ม validation report ลงใน ticket ได้ และสามารถตั้งเวลาอัตโนมัติได้ ซึ่งนับว่าตรงตามวัตถุประสงค์ของโครงการและตรงตามความต้องการของผู้ใช้งาน ปัจจุบันมีบางฟังก์ชันที่ถูกนำไปใช้งานจริง คือ Submit request และ Validate JSF ซึ่งฟังก์ชัน Auto create schedule ยังอยู่ในขั้นตอนการตรวจสอบก่อนนำไปใช้งานจริง ซึ่งในการทดสอบพบว่าผลลัพธ์สามารถทำงานได้อย่างถูกต้องแล้ว ซึ่งการทำงานของระบบที่สามารถตั้งเวลาอัตโนมัติได้นั้น ซึ่งจะช่วยประหยัดเวลาให้กับผู้ใช้งานและ AE analyst โดยจากการประเมินแล้ว ค่าร้องประมาณ 40% จะสามารถถูกระบบตั้งเวลาอัตโนมัติให้ได้

5.2 ปัญหาและอุปสรรค และข้อเสนอแนะ

ปัญหาและอุปสรรคสำคัญของระบบคือความซับซ้อนของ JSF ที่มีได้หลากหลายรูปแบบ และอาจมีการเกี่ยวเนื่องกับ JSF อื่นๆ ซึ่งทำให้การทำงานอัตโนมัติเกิดความผิดพลาดได้ ซึ่งผู้จัดทำได้แก้ไขปัญหาด้วยการวิเคราะห์ JSF ก่อนเพื่อให้เห็นถึงความเป็นไปได้ในการตั้งเวลาอัตโนมัติ โดยปัญหานี้จะสามารถแก้ไขได้ในอนาคตเมื่อมีการเข้าใจถึงรูปแบบและความสัมพันธ์ต่างๆของ JSF มากยิ่งขึ้น หรือการจัดระเบียบ JSF ด้วยมาตรฐานที่สามารถนำมาพัฒนาระบบอัตโนมัติได้ง่ายยิ่งขึ้น

บรรณานุกรม

1. Agile คืออะไร เริ่มใช้งานอย่างไร. Thanyavuth Akarasomcheep. Retrieved December 14, 2019 from <https://medium.com/fastwork-engineering/agile-คืออะไร-เริ่มใช้งานอย่างไร-ab749306d96e>
2. MuleSoft Retrieved December 14,2018 from <https://www.mulesoft.com/>
3. What is RESTful API?. Fermesk Rashidi. Retrieved December 14,2018 from <https://www.doprax.com/content/What-is-restful-API%3F/>
4. แนวออกแบบ RESTful API — วิธีปฏิบัติที่ดี. Phayao Boonon. Retrieved December 14,2018 from <https://medium.com/@phayao/แนวออกแบบ-restful-api-วิธีปฏิบัติที่ดี-c320d806e30b>
5. Git คืออะไร ... Git is your friend. Pakin Phuhinkong. Retrieved December 14,2018 from <https://medium.com/@pakin/git--คืออะไร-git-is-your-friend-c609c5f8efea>
6. SQL คืออะไร. Retrieved December 14,2018 from <https://www.mindphp.com/คู่มือ/73-คืออะไร/2088-sql-คืออะไร.html>
7. Automation Engine System Architecture. Retrieved December 14,2018 from <https://docs.automic.com/documentation/webhelp/english/ARA/11.2/AE/11.2/All%20Guides/Content/ucaseh.htm>

ภาคผนวก ก

ในภาคผนวก ก นี้จะเป็นรูปภาพ และข้อมูลเพิ่มเติมเกี่ยวกับโครงการ Job Automation Services : Auto development and acceptance schedule โดยเริ่มจากภาพขั้นตอนการเก็บรวบรวมความต้องการ ที่ผู้จัดทำได้สัมภาษณ์ผู้ใช้งานระบบ JAS และ AE analyst



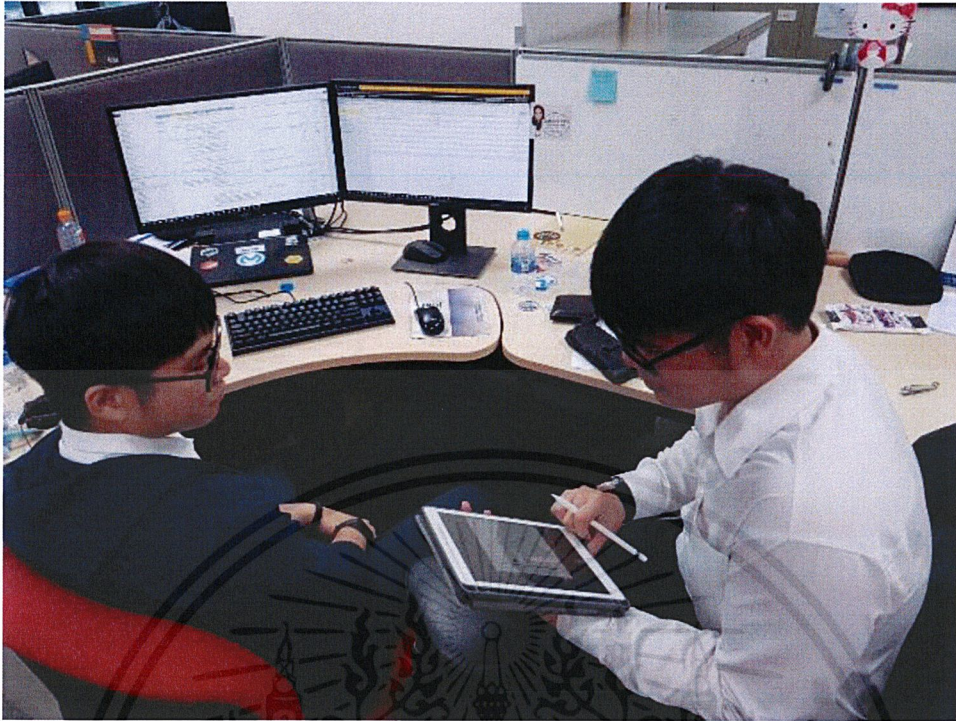
ภาพ การเก็บข้อมูลจากผู้ใช้งานระบบ JAS



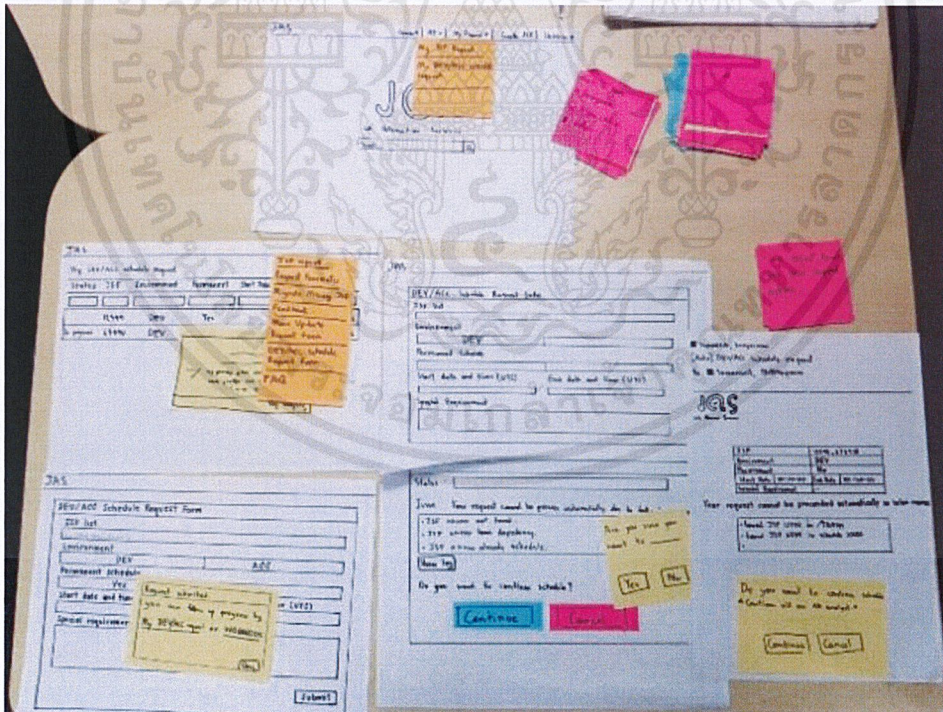
ภาพ การเก็บข้อมูลจากผู้ใช้งานระบบ JAS (ต่อ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(ต่อ)



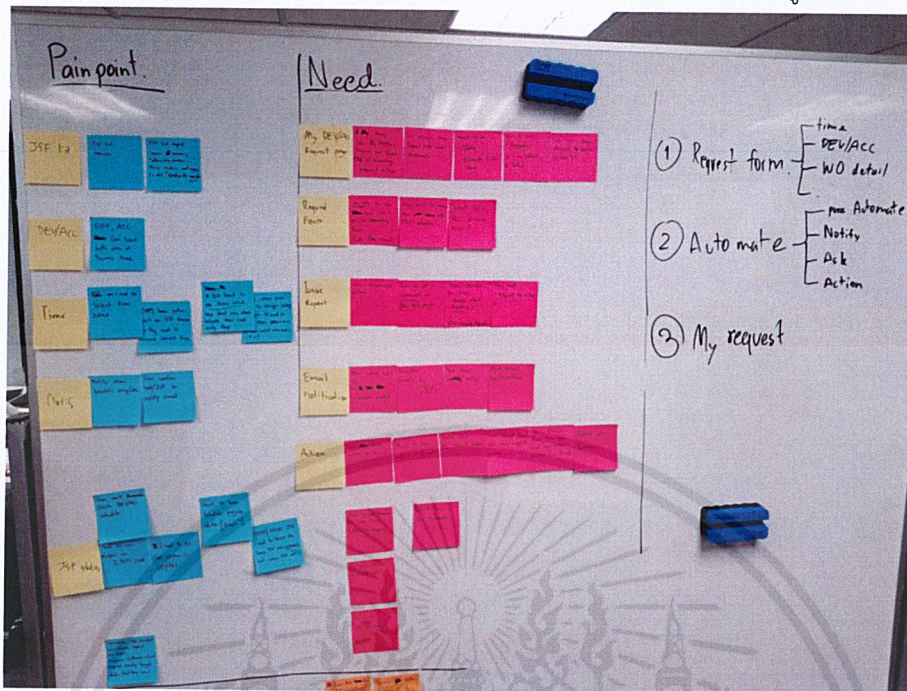
ภาพ การเก็บข้อมูลจาก AE analyst



ภาพ Mockup UI

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และได้มีการนำความต้องการมาจัดหมวดหมู่และเรียงลำดับความสำคัญ เพื่อใช้ในการเลือกฟังก์ชันที่จะพัฒนาตามลำดับ รวมถึงการออกแบบ User Journey เพื่อใช้ในการอธิบายกับผู้ใช้งานด้วย



ภาพ ความต้องการของผู้ใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

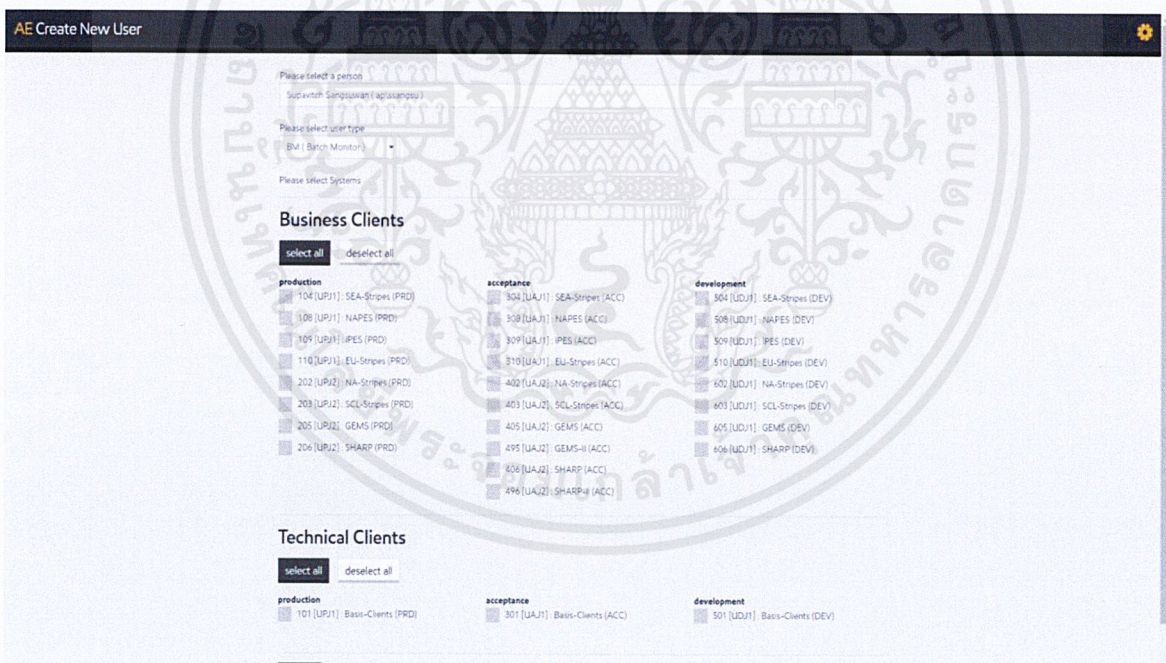
ภาคผนวก ข

ในภาคผนวก ข นี้จะกล่าวถึงโครงการอื่นที่ผู้จัดทำได้พัฒนาในระหว่างฝึกงานอยู่ที่บริษัท ExxonMobil ซึ่งได้แก่แอปพลิเคชัน Auto User Management และ AE Monitoring Dashboard

1. Auto User Management

เป็นระบบสำหรับให้ผู้ใช้งานขอสิทธิ์เข้าใช้งาน Automation Engine โดยมีการพัฒนาเพื่อเปลี่ยนระบบการขอสิทธิ์เข้าใช้งานรูปแบบเก่าที่ต้องใช้การกรอกข้อมูลบน Excel และมีขั้นตอนการสร้างสิทธิ์เข้าใช้งานด้วยระบบแบบ manual โดยทีม Access admin เป็นผู้จัดการสร้างสิทธิ์เข้าใช้ระบบ ทีละหนึ่งเซิร์ฟเวอร์ ซึ่งใช้เวลาในการดำเนินการประมาณ 2 – 30 วันต่อคำร้อง

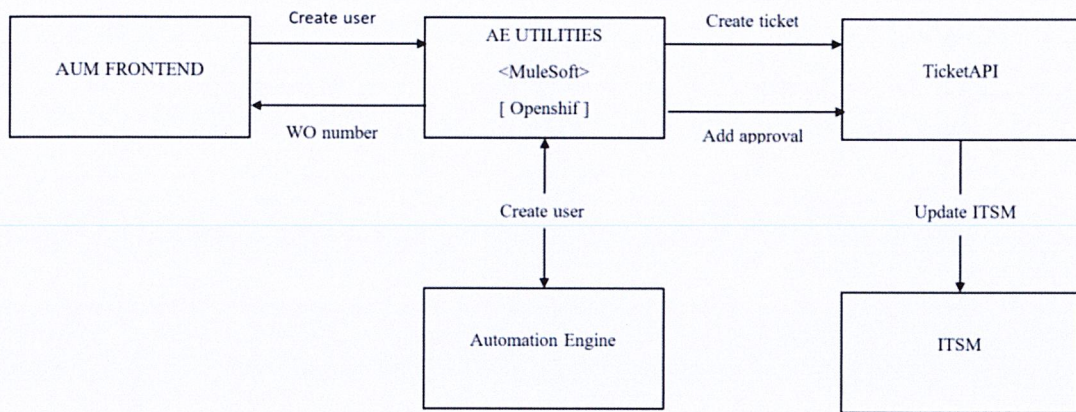
ซึ่งระบบ Auto User Management จะเป็นเว็บไซต์ที่ให้ผู้ใช้งานกรอกข้อมูลแทน Excel เพื่อลดการเกิดความผิดพลาดในการกรอกข้อมูล จากนั้นระบบจะทำการสร้างสิทธิ์การใช้งานให้อัตโนมัติเมื่อคำร้องนั้นถูกรับรอง



ภาพ เว็บไซต์ Auto User Management

โดยมีการพัฒนาเว็บไซต์ด้วย AngularJS และพัฒนาระบบหลังบ้านด้วย MuleSoft ซึ่งระบบมีโครงสร้างดังภาพต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพ Auto User Management architecture

AE Create New User

Step 1: Please select a person
Nuttachai (Oat) Chieocharnnikhit (apinchieo)

Step 2: Please select user type
BU (Business User) [Select user type as BU]

Step 3: Please select Systems
select all | deselect all

production	acceptance	development
104 [UPJ1] - SEA-Stripes (PRD)	304 [UAJ1] - SEA-Stripes (ACC)	504 [UDJ1] - SEA-Stripes (DEV)
108 [UPJ1] - NAPES (PRD)	308 [UAJ1] - NAPES (ACC)	508 [UDJ1] - NAPES (DEV)
109 [UPJ1] - IPES (PRD)	309 [UAJ1] - IPES (ACC)	509 [UDJ1] - IPES (DEV)
110 [UPJ1] - EU-Stripes (PRD)	310 [UAJ1] - EU-Stripes (ACC)	510 [UDJ1] - EU-Stripes (DEV)
202 [UPJ2] - NA-Stripes (PRD)	402 [UAJ2] - NA-Stripes (ACC)	602 [UDJ1] - NA-Stripes (DEV)
203 [UPJ2] - SCL-Stripes (PRD)	403 [UAJ2] - SCL-Stripes (ACC)	603 [UDJ1] - SCL-Stripes (DEV)
205 [UPJ2] - GEMS (PRD)	405 [UAJ2] - GEMS (ACC)	605 [UDJ1] - GEMS (DEV)
206 [UPJ2] - SHARP (PRD)	495 [UAJ2] - GEMS-II (ACC)	606 [UDJ1] - SHARP (DEV)
	406 [UAJ2] - SHARP (ACC)	
	496 [UAJ2] - SHARP-II (ACC)	

Step 4: submit

ภาพ ขั้นตอนการใช้งาน Auto User Management

การใช้งานเว็บไซต์สามารถแบ่งได้เป็น 4 ขั้นตอนได้แก่

1. การกรอกชื่อของบุคคลที่จะขอสิทธิ์เข้าใช้งาน
2. เลือกประเภทของยูสเซอร์ที่จะใช้ในระบบ
3. เลือกเซิร์ฟเวอร์ที่จะขอสิทธิ์เข้าใช้งาน
4. กดปุ่ม submit เพื่อส่งข้อมูล

และการกำหนดประเภทของยูสเซอร์จะส่งผลกระทบต่อรายชื่อเซิร์ฟเวอร์ที่สามารถขอสิทธิ์เข้าใช้งานได้รวมถึงสิทธิ์ในการกระทำการใดๆก็ตามในเซิร์ฟเวอร์ โดยการกำหนดสิทธิ์ต่างๆมีข้อมูลดังตารางต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง สิทธิ์ของแต่ละประเภทยูสเซอร์

Client	Team	Prefix	Suffix	AE user group		
				DEV	ACC	PRD
Business client	AE	BA	/BATCH	BATCH.DEVELOP.DEV	BATCH.DEVELOP.ACC	BATCH.DEVELOP.PRD
	MSP	-	/BATCH	BATCH.MONITOR	BATCH.MONITOR	BATCH.MONITOR
	AE Admin	SA	/BASIS	BASIS	BASIS	BASIS
	BASIS	SA	/VIEW	VIEW	VIEW	VIEW
	COE	BU	/VIEW	VIEW	VIEW	VIEW
Technical client	AE	BA	/BATCH	-	-	BATCH
	MSP	-	/BATCH	BASIS.MAINT	BASIS.MAINT	BASIS.MAINT
	AE Admin	SA	/BASIS	BASIS.DEVELOPMENT	BASIS.MAINT	BASIS.MAINT
	BASIS	SA	/BASIS	BASIS.MAINT	BASIS.MAINT	BASIS.MAINT
					BASIS.TRANSPORT	BASIS.TRANSPORT
Configuration client	AE Admin	SA	/BASIS	BASIS.MAINT	BASIS.MAINT	BASIS.MAINT

นอกจากนี้ประเภทของยูสเซอร์ และแผนกของผู้ขอ ยังส่งผลต่อการขอการรับรองของคำร้องด้วย ซึ่งมีความสัมพันธ์กันดังตารางต่อไปนี้

ตาราง การกำหนดผู้รับรอง

Client	User groups	Team	Prefix	Approvers											
Business client	BU	COE	BU	No approver											
	BA	AE	BA	AE's POs											
	BM	MSP	-	AE's POs											
	SA	AE Admin	SA	AE's POs											
	SA	BASIS	SA	No approver											
Technical client	BA	AE	BA	AE's POs											
	BM	MSP	-	AE's POs											
	SA	AE Admin	SA	AE's POs											
	SA	BASIS	SA												
				<table border="1"> <thead> <tr> <th>BASIS groups</th> <th>BASIS approvers</th> </tr> </thead> <tbody> <tr> <td>Platform Admin-Chemicals</td> <td></td> </tr> <tr> <td>Platform Admin-F&L</td> <td></td> </tr> <tr> <td>Platform Admin-Upstream/Corporate</td> <td></td> </tr> <tr> <td>Global Basis and Application Services-LEVEL 1</td> <td></td> </tr> <tr> <td>Other</td> <td></td> </tr> </tbody> </table>	BASIS groups	BASIS approvers	Platform Admin-Chemicals		Platform Admin-F&L		Platform Admin-Upstream/Corporate		Global Basis and Application Services-LEVEL 1		Other
BASIS groups	BASIS approvers														
Platform Admin-Chemicals															
Platform Admin-F&L															
Platform Admin-Upstream/Corporate															
Global Basis and Application Services-LEVEL 1															
Other															
Configuration client	SA	AE Admin	SA	AE's POs											

โดยหลังจากที่ผู้ใช้งานกดปุ่ม Submit แล้วระบบจะทำการสร้าง ticket ใน ITSM เพื่อเก็บข้อมูล และใช้สำหรับการให้ผู้รับรองยืนยันการขอคำร้องนั้นๆ โดยรูปแบบของ ticket เป็นดังภาพต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

[[AUTO]] [[AUM]] Auto User Management - Create User ap\amongk1

On Monday, Dec 16 2019 at 05:26 PM

LOW PRIORITY TAG

Resolution

Notes

Work Details

Approvals

Tasks

Relationships

Last Modified: Dec 16 2019 at 05:36 PM

Last Modified By: AE Automation Production (mlvongpap)

Resolution

See below for additional features available through IT Services.

Notes

*** AE Access Request Form **

*** Request Details

- Type of Change (New / Change / Delete USER): New

- Change Requestor Name: Anusara Mongkon

- user Type: BU

- user ID: [REDACTED]

- Systems: [504,508,509,510,602,603,605,304,308,309,399,310,402,403,405,495,104,108,109,110,202,203,205]

- Approvers: -

[[ROBOT section do not disturb!]]

"RequestDetail": ####

[[{"First_Name": [REDACTED], "Last_Name": [REDACTED], "Email_Address": [REDACTED], "approvers": "-", "approversDisplayName": "-", "id": "ap\amongk1", "type": "BU", "userID": "AMONGK1", "MGroups": ["Procurement-Accounts Payable", "Payable-Powers"], "systems": ["504", "508", "509", "510", "602", "603", "605", "304", "308", "309", "399", "310", "402", "403", "405", "495", "104", "108", "109", "110", "202", "203", "205"], "basisSelectedTechnicalUserGroups": []}]] ####

[Read Less](#)

ภาพ ticket ของ Auto User Management

ซึ่งระบบจะนำข้อมูลในส่วนของ [[ROBOT section do not disturb!]] จาก ticket ไปสร้างสิทธิ์เข้าใช้งานระบบ Automation Engine โดยอัตโนมัติทุกๆ 20 นาที

[[ROBOT section do not disturb!]]

"RequestDetail": ####


[[{"First_Name": [REDACTED], "Last_Name": [REDACTED], "Email_Address": [REDACTED], "approvers": "-", "approversDisplayName": "-", "id": "ap\amongk1", "type": "BU", "userID": "AMONGK1", "MGroups": ["Procurement-Accounts Payable", "Payable-Powers"], "systems": ["504", "508", "509", "510", "602", "603", "605", "304", "308", "309", "399", "310", "402", "403", "405", "495", "104", "108", "109", "110", "202", "203", "205"], "basisSelectedTechnicalUserGroups": []}]] ####

[Read Less](#)

ภาพ robot section ของ ticket

โดยระบบหลังบ้านจะทำการแปลงข้อมูลข้อความเป็น JSON แล้วนำไปสร้างสิทธิ์เข้าใช้งานระบบโดยอัตโนมัติ ซึ่งหลังจากการทำงานของระบบ จะมีการเพิ่มข้อความแสดงผลการทำงานลงใน work details ของ ticket เพื่อเป็นการแจ้งสถานะการทำงานกับผู้ใช้งานและใช้สำหรับ AE analyst กรณีระบบเกิดปัญหาขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



 Work Details (1)

New Work Detail


[WO0000015744275](#)


User ID'AMONGK1/VIEW' is created successfully in business client [504, 508, 509, 510, 602, 603, 605, 304, 308, 309, 399, 310, 402, 403, 405, 495, 104, 108, 109, 110, 202, 203, 205]

This ticket will be closed.

 AE-Automation Production  

Dec 16 at 05:38 PM

ภาพ success log

 Work Details (1)

New Work Detail



failed in [504]. Please check!

Error Log:



504: An object named 'BUNSUWANW/VIEW' already exists.

Please choose a different name. in 504

Please contact AE team AEMonitoring/SM

This ticket will be closed.

[Read Less](#)

 AE-Automation Production  

Jul 18 at 05:17 PM

ภาพ fail log

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. AE Monitoring Dashboard

เป็นระบบสำหรับเฝ้าดูสถานะผิดพลาดของ Automation Engine โดยการนำข้อมูลเกี่ยวกับความผิดพลาดบางส่วนที่สำคัญมาแสดงลงบนเว็บไซต์ เพื่อช่วยลดภาระการทำงานของทีม AE Monitor เพราะว่าปกติการเฝ้าดูสถานะผิดพลาดทีมจะต้องเข้าระบบ Automation Engine ทุกเซิร์ฟเวอร์เพื่อดูสถานะของแต่ละเซิร์ฟเวอร์เอง ซึ่งปัจจุบันมีมากกว่า 10 เซิร์ฟเวอร์ ทำให้ยากต่อการตรวจพบ

ผู้จัดทำจึงพัฒนาเว็บไซต์ซึ่งรวมข้อมูลที่จำเป็นสำหรับการเฝ้าดูสถานะของ Automation Engine จากทุกๆเซิร์ฟเวอร์เข้ามารวมอยู่ในเว็บไซต์เดียวเพื่อให้ง่ายต่อการทำงานของทีม AE Monitor

The screenshot displays the AE Monitor Dashboard with the following sections:

- System Status:** Shows the status of AE SERVER, AE DB, and JAS DB. AE SERVER includes UAJ1, UAJ2, UDJ1, UPJ1, and UPJ2. AE DB includes UPJ2, UPJ1, UAJ1, UAJ2, and UDJ1. JAS DB includes IORDERPRD.
- Agents Disconnected:** A table listing disconnected agents with columns for Client, NAME, TYPE, and IP ADDRESS.

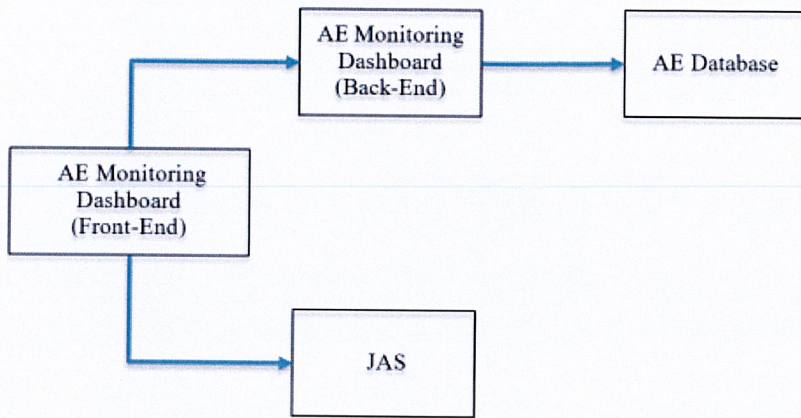
Client	NAME	TYPE	IP ADDRESS
202	DALDSP02	WINDOWS	10.216.177.23
202	DALDSP03	WINDOWS	10.216.177.23
205	SNGZUB44	WINDOWS	172.30.181.114
- Job Failure:** A table listing failed jobs with columns for Client, Job, and Status.

Client	Job	Status
110	LS.9998.ES.0229204_0229205.C.TMS6BNLPL0305.COMMON	ENDED NOT OK - Aborted
110	LS.9998.ES.0229205.LTMS6BNLPL0305.MAIN	ENDED NOT OK - Aborted
110	LS.9998.ES.TMS6BNLPLC.OVERALL.COMMON.ECC_MFS	Workflow is blocked
109	R3.0000.AE.0270724.C.MY_ONIQUA_IN_MGRP.MAIN	ENDED NOT OK - Aborted
109	R3.0000.AE.0301857.C.EMURALD_T4X.MAIN	ENDED NOT OK - Aborted
109	R3.0000.AE.0302252.R.MZ_MM_AUTO_PO_SERVICES.REP	ENDED NOT OK - Aborted

ภาพ เว็บไซต์ AE monitor Dashboard

โดยระบบมีการเรียกข้อมูลจาก Automation Engine ผ่านระบบหลังบ้านของ AE Monitor dashboard และเรียกใช้ API ของระบบ Job Automation Services ด้วย ซึ่งมีโครงสร้างระบบดังภาพต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพ AE Monitor Dashboard architecture



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้