



รายงานสหกิจศึกษาฉบับสมบูรณ์

ระบบซอฟต์แวร์อัตโนมัติเพื่อบริหารจัดการอาร์พีเอ

RPA change management tool

นางสาวรนิษฐา อิกหาญสมบัติ

สาขาวิชาวิศวกรรมสารสนเทศ

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2562

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อโครงการสหกิจศึกษา ระบบซอฟต์แวร์อัตโนมัติเพื่อบริหารจัดการอาร์พีเอ

ชื่อ-สกุล นักศึกษา นางสาวรณิษฐา ฮีทหาญสมบัติ

คณะ วิศวกรรมศาสตร์ ภาควิชา วิศวกรรมคอมพิวเตอร์ สาขาวิชา วิศวกรรมสารสนเทศ

ชื่อ-สกุล อาจารย์นิเทศ ผศ. ไพศาล สิทธิโยภาสกุล  
รศ.ดร. ขวลิต เบญจางคประเสริฐ

ชื่อ-สกุล ผู้นิเทศงาน นายเฉลิมชัย ธเนศอนุกุล

ชื่อสถานประกอบการ บริษัทเอ็กซอนโมบิล จำกัด

### บทคัดย่อ

ในปัจจุบัน ระบบอัตโนมัติได้เข้ามามีบทบาทสำคัญในการสร้างกระบวนการทำงานที่มีประสิทธิภาพ โดยมีจุดประสงค์ คือ 1) เพื่อลดกระบวนการที่ซ้ำซ้อน และเพิ่มความสะดวกสบายในการทำงาน 2) เพื่อลดความผิดพลาดที่เกิดจากการทำงานของมนุษย์ 3) เพื่อช่วยให้กระบวนการทำงานมีความรวดเร็วมากยิ่งขึ้น ในโครงการฉบับนี้ผู้พัฒนาได้พัฒนาระบบซอฟต์แวร์อัตโนมัติ เพื่อช่วยให้การบริหารจัดการอาร์พีเอมีประสิทธิภาพมากยิ่งขึ้น และเป็นไปตามความต้องการทางธุรกิจขององค์กร ซึ่งอาร์พีเอเป็นหนึ่งในผลิตภัณฑ์ที่มีความสำคัญต่อบริษัทในการแทนที่กระบวนการทำงานที่ซ้ำซ้อน ให้มีความแม่นยำและรวดเร็วมากยิ่งขึ้น เนื่องด้วยอาร์พีเอได้ถูกนำไปใช้งานอย่างแพร่หลายในแต่ละแผนก และมีแนวโน้มที่สูงขึ้นเรื่อย ๆ เพราะฉะนั้นการบริหารจัดการอาร์พีเอด้วยระบบซอฟต์แวร์อัตโนมัติจึงถูกริเริ่มขึ้น โดยจากผลการพัฒนาพบว่า ระบบซอฟต์แวร์อัตโนมัติเพื่อบริหารจัดการอาร์พีเอมีประสิทธิภาพการทำงานที่ดี และประสิทธิผลของการทำงานตอบสนองต่อความต้องการขององค์กรได้

คำสำคัญ : ระบบอัตโนมัติ การบริหารจัดการอาร์พีเอ การเพิ่มประสิทธิภาพ ความต้องการทางธุรกิจ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**Cooperative Title:** RPA change management tool

**Student Intern Name:** Woranitha Hueghansombat

**Faculty:** Engineering **Department:** Information Engineering

**Advisor Name:** Asst.Prof. Paisan Sithiyopasakul,  
Assoc.Prof.Dr. Chawalit Benjangkaprasert,

**Mentor Name:** Chaloechai Thanetanukun

**Company:** ExxonMobil Limited

## ABSTRACT

These days, automation plays an important role in the making of an efficient working process. By the purposes of automation are 1) Reduce numbers of repetitive job and facilitate user. 2) Reduce the errors that caused by human. 3) Shorten the working process, it helps user to work faster. In this project, Robotic Process Automation (RPA) has change management tool, I developed automation software for enhance the performance of RPA management and fulfil the business needs of the company. RPA is one of the important products for the company in replacing the repetitive process with more accuracy and faster than human, because RPA has been used widely in each department and tend to be used more and more. Therefore, managing RPA with an automation software has been started. The result from development shows that an automation software for management process of RPA has increase the performance improvement and its effectiveness very well response to business needs of the company.

**Keywords:** Automation, RPA change management tool, Performance Improvement, Business Needs

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## กิตติกรรมประกาศ

ข้าพเจ้าได้มีโอกาสเข้าร่วมโครงการสหกิจศึกษาที่ บริษัท เอ็กซอนโมบิล จำกัด ตั้งแต่วันที่ 4 มิถุนายน 2562 ถึงวันที่ 27 ธันวาคม 2562 โดยได้รับฝึกสอนและปฏิบัติหน้าที่ในแผนก Robotics Automation Center Of Expertise (RACE) ในหัวข้อโครงการระบบซอฟต์แวร์อัตโนมัติเพื่อบริหารจัดการอาร์พีเอ การเข้าร่วมฝึกงานในครั้งนี้ข้าพเจ้าได้เรียนรู้การทำงานจากประสบการณ์จริง และลงมือพัฒนาโครงการด้วยตัวเองตั้งแต่ต้นจนจบ ทั้งการออกพื้นที่สังเกตกระบวนการทำงานจริง สัมภาษณ์ผู้ใช้งาน สร้างโปรโตไทป์ และพัฒนาจนสมบูรณ์ บริษัทไม่เพียงแต่ส่งเสริมทักษะความรู้ (Hard skills) แต่ยังให้ความสำคัญกับกิจกรรมเพื่อส่งเสริมทักษะด้านสังคม (Soft skills) โดยข้าพเจ้าได้รับประโยชน์และประสบการณ์มากมาย ข้าพเจ้าขอขอบพระคุณเป็นอย่างสูง

ขอขอบพระคุณ บริษัท เอ็กซอนโมบิล จำกัด และพนักงานที่เกี่ยวข้องที่มอบโอกาสให้ข้าพเจ้าเข้าร่วมทำงาน ตลอดจนการดูแลและให้ความช่วยเหลือเป็นอย่างดีตลอดระยะเวลาโครงการสหกิจศึกษา

ขอขอบพระคุณ คุณเฉลิมชัย ธเนศอนุกุล คุณสุรวดี ตั้งต๋องกิจ และคุณวิริยุทธิ์ เกียรติสูงส่ง ที่เป็นผู้ดูแลข้าพเจ้า คอยให้คำปรึกษา และคำแนะนำไม่ว่าจะเป็นด้านการทำงานหรือการใช้ชีวิตภายในบริษัท ข้าพเจ้านำสิ่งที่ข้าพเจ้าได้รับไปปรับใช้ในการทำงานและการใช้ชีวิตของข้าพเจ้าในอนาคตต่อไป

ขอขอบพระคุณพนักงานแผนก RACE ทุกท่าน ที่คอยช่วยเหลือข้าพเจ้าในเรื่องต่าง ๆ ตลอดจนดูแล ให้ความรู้ และให้คำแนะนำแก่ข้าพเจ้าเสมอมา

ขอขอบพระคุณ ผศ.ไพศาล สิทธิโยภาสกุล และ รศ.ดร.ชวลิต เบญจางคประเสริฐ อาจารย์ที่ปรึกษาที่ให้คำปรึกษาและคำแนะนำแก่ข้าพเจ้า ทำให้การทำงานโครงการสหกิจศึกษา และรายงานฉบับนี้ สำเร็จลุล่วงไปได้ด้วยดี

ขอขอบพระคุณบิดา มารดา และครอบครัวของข้าพเจ้า ที่คอยสนับสนุนในทุก ๆ อย่างก้าวและเป็นกำลังใจสำคัญให้แก่ข้าพเจ้าอยู่ตลอดเวลา

ขอบคุณเพื่อน ๆ นักศึกษาร่วมงานของข้าพเจ้าทุกคน ที่เป็นกำลังใจให้กันและกันเสมอมา และร่วมเติมเต็มให้บรรยากาศการทำงานครั้งนี้มีแต่ความสนุกสนาน

สุดท้ายนี้ ข้าพเจ้าขอขอบพระคุณอาจารย์ทุกท่านของข้าพเจ้าที่ได้ประสิทธิ์ประสาทวิชาความรู้ให้แก่ข้าพเจ้า และขอขอบพระคุณ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง สำหรับโอกาสในการทำโครงการสหกิจศึกษาครั้งนี้

วรนิษฐา ฮึกหาญสมบัติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญ

	หน้า
บทคัดย่อ .....	I
ABSTRACT .....	II
กิตติกรรมประกาศ .....	III
สารบัญ.....	IV
สารบัญตาราง.....	VI
สารบัญรูป .....	VII
<b>บทที่ 1 บทนำ.....</b>	<b>1</b>
1.1 ที่มาและความสำคัญ .....	1
1.2 วัตถุประสงค์ของการพัฒนา.....	2
1.3 ขอบเขตของการพัฒนา .....	2
1.4 วิธีการดำเนินการพัฒนา .....	2
1.5 ประโยชน์ที่คาดว่าจะได้รับ .....	2
1.5.1 ประโยชน์ต่อบริษัท.....	3
1.5.2 ประโยชน์ต่อผู้พัฒนา.....	3
<b>บทที่ 2 แนวคิด ทฤษฎี และงานวิจัยที่เกี่ยวข้อง .....</b>	<b>4</b>
2.1 ระบบอัตโนมัติ.....	4
2.2 UiPath Orchestrator.....	4
2.3 API.....	8
2.4 ระบบ ITSM .....	9
2.5 โปรแกรม Visual Studio Code.....	9
2.6 Angular .....	11
2.7 Azure DevOps Server .....	11
<b>บทที่ 3 ขั้นตอนการดำเนินงานพัฒนา.....</b>	<b>15</b>
3.1 ขั้นตอนการดำเนินงานพัฒนาระบบซอฟต์แวร์อัตโนมัติเพื่อบริหารจัดการอาร์พีเอ.....	15
3.1.1 การประชุมงาน .....	15
3.1.2 การศึกษาการทำงานของระบบ .....	17
3.1.3 ศึกษาและรวบรวมความต้องการปัจจุบัน.....	19

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ IV รมศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.4	วางแผนการพัฒนาระบบ.....	21
3.1.5	การพัฒนาระบบ .....	21
3.1.6	ทดสอบการทำงานของระบบ .....	65
3.1.7	นำระบบไปใช้งานจริง .....	65
บทที่ 4	ผลการพัฒนา.....	66
บทที่ 5	สรุปผลการพัฒนาและข้อเสนอแนะ.....	80
5.1	สรุปผลการพัฒนา .....	80
5.2	ปัญหาที่พบและวิธีการแก้ไข .....	80
เอกสารอ้างอิง.....		82
ประวัติผู้เขียน.....		83



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญตาราง

หน้า

ตารางที่ 3.1 แผนการทำงานพัฒนาระบบซอฟต์แวร์อัตโนมัติเพื่อบริหารจัดการอาร์พีเอ..... 15



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อวีรศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป

	หน้า
รูปที่ 2.1 Palletizer Robot แบบแขนเดี่ยวหลายแกน (Arm Robot).....	4
รูปที่ 2.2 UiPath logo .....	5
รูปที่ 2.3 หน้าต่างหลักของโปรแกรม UiPath Orchestrator .....	5
รูปที่ 2.4 หน้าต่างการใช้งานฟังก์ชัน Robots .....	7
รูปที่ 2.5 หน้าต่างการใช้งานฟังก์ชัน Assets.....	7
รูปที่ 2.6 หน้าต่างการใช้งานฟังก์ชัน Schedules .....	8
รูปที่ 2.7 หน้าต่างการใช้งานฟังก์ชัน Queues .....	8
รูปที่ 2.8 ภาพรวมการใช้งาน API .....	9
รูปที่ 2.9 Visual Studio Code logo.....	10
รูปที่ 2.10 หน้าต่างโปรแกรม Visual Studio Code .....	10
รูปที่ 2.11 ตัวอย่างของเครื่องมือเสริมใน Visual Studio Code .....	10
รูปที่ 2.12 Visual Studio Code logo .....	11
รูปที่ 2.13 หน้าต่างการใช้งาน Azure Repos.....	12
รูปที่ 2.14 หน้าต่างการใช้งาน Azure Repos.....	12
รูปที่ 2.15 หน้าต่างการใช้งาน Azure Test Plans .....	13
รูปที่ 2.16 หน้าต่างการใช้งาน Azure Artifacts .....	13
รูปที่ 2.17 หน้าต่างแสดงการอัปเดตสถานะของงาน .....	14
รูปที่ 2.18 หน้าต่างการจัดกลุ่มและไล่ระดับงาน .....	14
รูปที่ 3.1 ตัวอย่าง Azure board.....	16
รูปที่ 3.2 ตัวอย่าง Physical board.....	17
รูปที่ 3.3 การประชุม PBI planning โดยใช้ Physical board.....	17
รูปที่ 3.4 กระบวนการนำหุ่นยนต์ขึ้นสู่ Production environment แบบเก่า .....	18
รูปที่ 3.5 แบบฟอร์มกรอกข้อมูลในกระบวนการนำขึ้น Production.....	18
รูปที่ 3.6 สถานะของผู้อนุมัติ.....	18
รูปที่ 3.7 แผนการพัฒนาระบบ .....	21
รูปที่ 3.8 หน้าต่าง Team Explorer .....	22
รูปที่ 3.9 หน้าต่าง Add Team Foundation Server .....	23
รูปที่ 3.10 ปุ่ม Source Control Explorer ภายในหน้าต่าง Team Explorer.....	24
รูปที่ 3.11 การวางโครงสร้างหน้า Select assets page.....	25

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ VII ศึกษานี้เท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.12 Assets selection table .....	25
รูปที่ 3.13 Assets selected table .....	26
รูปที่ 3.14 รูปแบบ Response ของ Get license.....	26
รูปที่ 3.15 รูปแบบ Response ของ Get current user .....	27
รูปที่ 3.16 รูปแบบ Response ของ Get Assets .....	28
รูปที่ 3.17 Method getOrgUnit().....	29
รูปที่ 3.18 Method getOrgUnitSelectedProd() .....	29
รูปที่ 3.19 Method getAssetFromAcc() & getAssetFromProd() .....	30
รูปที่ 3.20 Method updateAssets().....	30
รูปที่ 3.21 Method getSelectedAsset().....	31
รูปที่ 3.22 Method selectAll() .....	32
รูปที่ 3.23 Assets searching filter.....	32
รูปที่ 3.24 การวางโครงสร้างหน้า Summarize page.....	33
รูปที่ 3.25 Summarize page.....	34
รูปที่ 3.26 รูปแบบ Response ของ Get user look-up.....	35
รูปที่ 3.27 รูปแบบการตอบกลับของ Get support group .....	36
รูปที่ 3.28 รูปแบบ Response ของ Get members in support group.....	37
รูปที่ 3.29 รูปแบบ Body ของ Get Assets.....	37
รูปที่ 3.30 รูปแบบ Body ของ Post CRQ ticket & Put CRQ ticket.....	38
รูปที่ 3.31 Method addApproveBox() & deleteApproveBox().....	38
รูปที่ 3.32 Method SearchTextChanged() .....	39
รูปที่ 3.33 Method deSearch() .....	39
รูปที่ 3.34 Method checkIsSameAsCurrentUser(), checkIsSelectedUserAlreadyExist(), checkIsApproverListLessThanNumberOfBox(), checkIsApproverSameAsCoordinator...	40
รูปที่ 3.35 Method selectUser() .....	41
รูปที่ 3.36 Method deleteUser() .....	41
รูปที่ 3.37 Method SearchSupportGroupChanged() & doSupportGroupSearched().....	42
รูปที่ 3.38 Method SearchMemberChanged() & doMemberSearch() .....	43
รูปที่ 3.39 Method receiveValid() .....	43
รูปที่ 3.40 Method createCrqTicket().....	44
รูปที่ 3.41 การวางโครงสร้างหน้า View all tickets page.....	45
รูปที่ 3.42 View all tickets (Main component).....	45
รูปที่ 3.43 รูปแบบ Response ของ Cancel ticket .....	46

รูปที่ 3.44	รูปแบบ Response ของ Get CRQ ticket list.....	47
รูปที่ 3.45	Method getCrqTicketList() .....	48
รูปที่ 3.46	Method updateSupportGroup() .....	48
รูปที่ 3.47	Method cancelTicket() .....	49
รูปที่ 3.48	การวางโครงสร้างหน้า Ticket detail page.....	49
รูปที่ 3.49	Track CRQ ticket (Main component).....	49
รูปที่ 3.50	รูปแบบ Response ของ Get current user permission.....	50
รูปที่ 3.51	รูปแบบ Body และ Response ของ Post work detail.....	50
รูปที่ 3.52	รูปแบบของ Body และ Response ของ Post assets.....	52
รูปที่ 3.53	รูปแบบ Body ของ Put assets .....	53
รูปที่ 3.54	Method checkStatus() .....	54
รูปที่ 3.55	Method getTicketApprovers() .....	54
รูปที่ 3.56	Method checkUserIsApprover(), checkUserPermission(), checkIsApproverIsCoordinator().....	55
รูปที่ 3.57	Method checkCrqStatus().....	55
รูปที่ 3.58	Method enableGoLiveButton().....	56
รูปที่ 3.59	Method getAssetFromWorkDetails().....	57
รูปที่ 3.60	Method goLive() .....	58
รูปที่ 3.61	Method checkAsset().....	58
รูปที่ 3.62	Method postAsset().....	60
รูปที่ 3.63	Method putAsset() .....	60
รูปที่ 3.64	การวางโครงสร้าง Value per robot feature.....	61
รูปที่ 3.65	Value per robot feature.....	61
รูปที่ 3.66	รูปแบบ Response ของ Get license.....	62
รูปที่ 3.67	Method getRobots () .....	63
รูปที่ 3.68	Method addRobotValue().....	63
รูปที่ 3.69	Method deleteRobotValue().....	64
รูปที่ 3.70	Method changeRobot().....	64
รูปที่ 3.71	Method checkRobotRedundant() & checkIfRobotHaveValueOrNot().....	64
รูปที่ 3.72	การทดสอบการทำงานของระบบโดยผู้ใช้งานจริงแผนก Fuels & lubes.....	65
รูปที่ 3.73	ตัวอย่างเอกสารข้อมูลเกี่ยวกับการใช้งานระบบ .....	65
รูปที่ 4.1	ตาราง Assets selection.....	66
รูปที่ 4.2	ผู้ใช้งานทำการเลือกชื่อ organization.....	67

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ IX ารศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.3 ช่องเพื่อทำการค้นหา .....	67
รูปที่ 4.4 การแสดงค่าโดยย่อหากมีจำนวนค่าที่ยาวเกินไป และสามารถขยายดูค่าเต็มได้ .....	67
รูปที่ 4.5 ผู้ใช้งานทำการเลือก Asset ที่ต้องการเพิ่มหรือเปลี่ยนแปลง .....	68
รูปที่ 4.6 ผู้ใช้งานทำการเลือก Asset ทั้งหมด โดยการกด select all button.....	68
รูปที่ 4.7 ผู้ใช้งานทำการแก้ไขค่าของ Assets .....	69
รูปที่ 4.8 ผู้ใช้งานทำการเลือกชื่อ organization ที่ต้องการเพิ่ม Assets .....	69
รูปที่ 4.9 กด Next button เพื่อไปยังหน้า Summarize page .....	69
รูปที่ 4.10 หน้า Summary Page โดยตัดช่องกรอกที่ไม่จำเป็นออก เหลือเพียงช่องกรอกที่จำเป็นเท่านั้น	70
รูปที่ 4.11 ผู้ใช้งานทำการกรอก Request summary.....	70
รูปที่ 4.12 ผู้ใช้งานทำการเลือก Approvers .....	71
รูปที่ 4.13 ระบบแสดงคำเตือนหากไม่เจอ Approver ที่ระบุ .....	71
รูปที่ 4.14 ระบบแสดงคำเตือนหากผู้อนุมัติเป็นคนเดียวกับผู้ร้องขอ .....	71
รูปที่ 4.15 ระบบแสดงข้อความ Valid contact หาก Approver ที่ระบุถูกต้อง .....	71
รูปที่ 4.16 ผู้ใช้งานทำการเลือก IT Console group .....	71
รูปที่ 4.17 ผู้ใช้งานทำการเลือก Change coordinator.....	72
รูปที่ 4.18 ระบบแสดงคำเตือนหาก Change coordinator เป็นคนเดียวกับผู้อนุมัติ.....	72
รูปที่ 4.19 หน้า Summary page ที่ผู้ใช้งานทำการกรอกโดยสมบูรณ์แล้ว .....	72
รูปที่ 4.20 หน้าต่างเพื่อให้ผู้ใช้งานทำการยืนยันว่าต้องการสร้าง CRQ Ticket เลย์หรือไม่ .....	72
รูปที่ 4.21 ระบบทำการสร้าง CRQ ticket อัตโนมัติ.....	73
รูปที่ 4.22 ข้อมูลของการเพิ่มหรือเปลี่ยนแปลง Assets ถูกแสดงใน CRQ ticket note .....	73
รูปที่ 4.23 ข้อมูลของการเพิ่มหรือเปลี่ยนแปลง Assets ถูกแสดงใน CRQ ticket work detail.....	74
รูปที่ 4.24 ข้อมูลของ Assets ในส่วนการดำเนินงานของระบบ.....	75
รูปที่ 4.25 ส่วนของผู้อนุมัติใน CRQ ticket .....	75
รูปที่ 4.26 การแสดง Ticket ทั้งหมดของ organization ใน IT Console .....	76
รูปที่ 4.27 หน้า View all tickets Page .....	76
รูปที่ 4.28 หน้าTicket Details page.....	77
รูปที่ 4.29 ผู้ใช้งานทำการเติม credential password.....	77
รูปที่ 4.30 หน้า loading พร้อมแสดงขั้นตอนการทำงานในการเพิ่มหรือเปลี่ยนแปลงAssets .....	78
รูปที่ 4.31 หน้าต่างแสดงข้อความเมื่อระบบอัปเดต Assets เรียบร้อย และปิด CRQ ticket อัตโนมัติ.....	78
รูปที่ 4.32 UiPath Orchestrator ก่อนทำการอัปเดต .....	78
รูปที่ 4.33 UiPath Orchestrator เมื่อทำการอัปเดตสำเร็จแล้ว .....	79
รูปที่ 4.34 UiPath Orchestrator's audit log.....	79

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ X ารศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 1.1 ที่มาและความสำคัญ

บริษัท เอ็กซอนโมบิล จำกัด เป็นบริษัทในเครือเอ็กซอนโมบิล ซึ่งเป็นบริษัทชั้นนำด้านพลังงาน สัญชาติอเมริกัน ประกอบธุรกิจก๊าซธรรมชาติและน้ำมันปิโตรเลียมที่มีขนาดใหญ่ที่สุดในโลก โดยบริษัท เอ็กซอนโมบิล จำกัด ก่อตั้งขึ้นเพื่อดำเนินการศูนย์บริการธุรกิจกรุงเทพ (Global Business Center: GBC) ซึ่งเป็นศูนย์บริการธุรกิจที่ใหญ่ที่สุดในเครือข่ายศูนย์บริการธุรกิจทั่วโลกของเอ็กซอนโมบิล ปัจจุบันมี พนักงานมากกว่า 2,200 คน มีสำนักงานอยู่ 3 แห่งในประเทศไทย ให้บริการแก่บริษัทในเครือเอ็กซอนโมบิลและลูกค้าในกว่า 60 ประเทศทั่วโลก ครอบคลุมทั้งทวีปเอเชีย ทวีปยุโรป และทวีปอเมริกาเหนือ-ใต้ บริการของศูนย์ฯ ครอบคลุมงานด้านบัญชี เทคโนโลยีสารสนเทศ ศูนย์บริการลูกค้าสำหรับน้ำมันเชื้อเพลิง และน้ำมันหล่อลื่น ศูนย์บริการลูกค้าเคมีภัณฑ์ ศูนย์จัดซื้อ ศูนย์ทรัพยากรบุคคล ฝ่ายภาษี และเครดิต ลูกค้า

การเข้าร่วมโครงการสหกิจศึกษาครั้งนี้ ผู้พัฒนาได้เข้าร่วมงานในแผนก Robotics Automation Center Of Expertise (RACE) ที่มีหน้าที่ให้คำปรึกษา ช่วยเหลือ และดูแลการพัฒนาผลิตภัณฑ์อาร์พีเอในแผนกต่าง ๆ รวมถึงพัฒนาซอฟต์แวร์เพื่อสนับสนุนให้การพัฒนาผลิตภัณฑ์อาร์พีเอมีประสิทธิภาพมากขึ้น เนื่องจากอาร์พีเอมีความนิยมมากขึ้นเรื่อย ๆ แต่กระบวนการทำงานบางส่วนยังไม่มีเสถียรภาพเท่าที่ควร อาทิ เช่น กระบวนการนำผลิตภัณฑ์อาร์พีเอขึ้นใช้งานใน Production Environment หลังจากที่นักพัฒนาพัฒนาผลิตภัณฑ์ ผ่านการทดสอบก่อนใช้งานจริง (UAT) และพร้อมที่จะทำนำขึ้น Production แต่เนื่องจากกระบวนการดังกล่าวยังคงมีความซับซ้อน และง่ายต่อการเกิดความผิดพลาด ทำให้เกิดการตามแก้ไขจุดเดิมซ้ำ ๆ เป็นผลให้เกิดความล่าช้าในการนำไปใช้งานจริง จนเกิดความเสียหายทางด้านทรัพย์สิน และความน่าเชื่อถือแก่บริษัท

จากปัญหาข้างต้นผู้พัฒนาจึงได้ริเริ่มความคิดในการพัฒนาระบบซอฟต์แวร์อัตโนมัติเพื่อบริหารจัดการอาร์พีเอ (RPA change management tool) ขึ้น เพื่อช่วยให้นักพัฒนาผลิตภัณฑ์อาร์พีเอดำเนินงานได้สะดวกขึ้น โดยระบบซอฟต์แวร์ดังกล่าวจะทำหน้าที่เป็นศูนย์รวม (All in one) ของกระบวนการนำผลิตภัณฑ์อาร์พีเอขึ้นใช้งานใน Production Environment จากเดิมที่นักพัฒนาจำเป็นต้องกรอกข้อมูลมากมายนับร้อยช่องในโปรแกรม Microsoft Excel และทำการแนบไฟล์ในเว็บไซต์เพื่อสร้างคำร้องขอการอนุมัติ ไปยังผู้อนุมัติ จากนั้นจึงส่งไปยังผู้ที่มีสิทธิ์เข้าถึง Production เพื่อทำการคัดลอกข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นักพัฒนาต้องการเพิ่มและแก้ไข ไปยังเว็บไซต์ที่เรียกว่าศูนย์กลางการควบคุมหุ่นยนต์อย่าง UiPath Orchestrator ซึ่งจะเห็นได้มากกว่าจะได้มาซึ่งผลลัพธ์ก็ใช้เวลาานาน ไม่เพียงเท่านั้นเนื่องจากมีกระบวนการมากมายที่จำเป็นต้องมีความถูกต้องแม่นยำ แต่ยังคงมีวิธีการจัดการที่ไม่เอื้ออำนวย และนำมาซึ่งความผิดพลาดในที่สุด

## 1.2 วัตถุประสงค์ของการพัฒนา

1.2.1 เพื่อลดภาระงานของกระบวนการนำผลิตภัณฑ์อาร์พีเอขึ้นใช้งานใน Production

1.2.2 เพื่อเพิ่มประสิทธิภาพในการทำงานโดยรวม และลดระยะเวลาในการทำงาน

1.2.3 เพื่อเพิ่มระดับความพึงพอใจของผู้ใช้งาน

## 1.3 ขอบเขตของการพัฒนา

1.3.1 ระบบซอฟต์แวร์อัตโนมัติสามารถเชื่อมต่อและทำงานกับ Application Programming Interface (API) ได้

1.3.2 ระบบซอฟต์แวร์อัตโนมัติสามารถระบุตัวตนผู้ใช้งานได้โดยอัตโนมัติ

1.3.3 ระบบซอฟต์แวร์อัตโนมัติสามารถทำงานกับระบบอื่น ๆ ได้

## 1.4 วิธีการดำเนินการพัฒนา

1.4.1 ประชุมวางแผนงาน

1.4.2 ศึกษาการทำงานของระบบ

1.4.3 รวบรวมความต้องการของผู้ใช้

1.4.4 วางแผนและออกแบบการพัฒนาาระบบ

1.4.5 พัฒนาระบบ

1.4.6 ทดสอบการทำงานของระบบ

1.4.7 นำระบบไปใช้งานจริง

## 1.5 ประโยชน์ที่คาดว่าจะได้รับ

ประโยชน์ที่ผู้พัฒนาคาดว่าจะได้รับจากการเข้าร่วมโครงการสหกิจศึกษา กับ บริษัท เอ็กซอนโมบิล จำกัด จำแนกออกเป็น 2 ส่วน คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ 2 ารศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 1.5.1 ประโยชน์ต่อบริษัท

1.5.1.1 ลดความซ้ำซ้อนของกระบวนการทำงาน

1.5.1.2 ลดระยะเวลาการทำงานโดยรวม

1.5.1.3 เพิ่มประสิทธิภาพการทำงานโดยรวม

## 1.5.2 ประโยชน์ต่อผู้พัฒนา

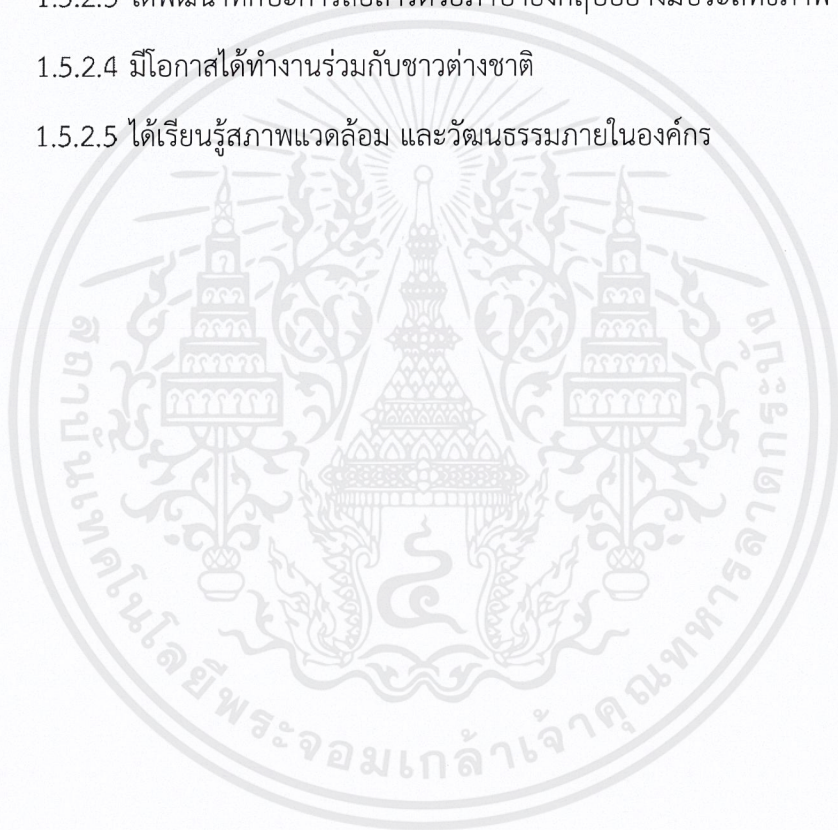
1.5.2.1 สามารถคิด วิเคราะห์ วางแผน และทำงานได้อย่างเป็นระบบ

1.5.2.2 ได้พัฒนาทักษะการสื่อสาร และการทำงานร่วมกับผู้อื่น

1.5.2.3 ได้พัฒนาทักษะการสื่อสารด้วยภาษาอังกฤษอย่างมีประสิทธิภาพ

1.5.2.4 มีโอกาสได้ทำงานร่วมกับชาวต่างชาติ

1.5.2.5 ได้เรียนรู้สภาพแวดล้อม และวัฒนธรรมภายในองค์กร



## แนวคิด ทฤษฎี และงานวิจัยที่เกี่ยวข้อง

### 2.1 ระบบอัตโนมัติ

ระบบอัตโนมัติ คือ ระบบใด ๆ ที่อาจเป็นการใช้กลไกคอมพิวเตอร์ หรืออุปกรณ์อิเล็กทรอนิกส์ในการควบคุมการทำงาน ทำให้ระบบอัตโนมัติสามารถทำงานได้ด้วยตัวเอง โดยจะมีการทำงานตามคำสั่งที่วางไว้ ความถูกต้องในการทำงานจะขึ้นอยู่กับการวางแผน หรือการวางโปรแกรมโดยมนุษย์ทั้งสิ้น ตัวอย่างระบบอัตโนมัติ เช่น ระบบตอบรับอีเมลอัตโนมัติ ระบบตอบรับโทรศัพท์อัตโนมัติ ระบบสายพานลำเลียงอัตโนมัติ เป็นต้น



รูปที่ 2.1 Palletizer Robot แบบแขนเดี่ยวหลายแกน (Arm Robot)

(ที่มา: <https://www.thaiprint.org/2019/12/vol123/knowledge123-02/>)

### 2.2 UiPath Orchestrator

UiPath Orchestrator เป็นเว็บสำหรับช่วยให้แอดมินควบคุมการทำงาน และจัดการ UiPath Robot ให้สามารถทำงานได้โดยการตั้งค่า โดยในเว็บแอปพลิเคชันนี้จะแสดงให้เห็นการทำงานของหุ่นยนต์ รวมถึงสถานะ ณ เวลาปัจจุบัน

UiPath Robot คือ หุ่นยนต์จะถูกออกแบบขึ้นใน UiPath Studio ซึ่งทำงานและถูกควบคุมโดย UiPath Orchestrator โดยการพัฒนาหุ่นยนต์เพื่อใช้งานในองค์กรมีเป้าหมายเพื่อช่วยลดภาระหรือทำหน้าที่แทนบางกระบวนการทางธุรกิจ ทำให้องค์กรสามารถเพิ่มประสิทธิภาพการทำงาน และลดทรัพยากรบุคคลได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

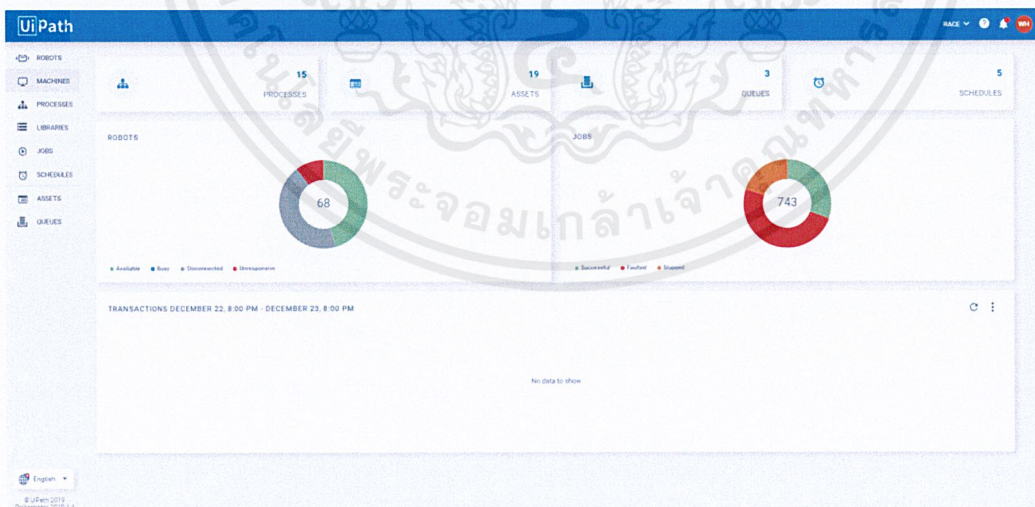
โดยความสามารถหลัก ๆ ของ UiPath Orchestrator มีดังนี้

- 1) สามารถสร้างและดูแลการเชื่อมต่อระหว่างหุ่นยนต์กับเว็บแอปพลิเคชัน
- 2) สามารถรับประกันได้ว่าหุ่นยนต์ที่กำลังเปิดใช้งานมีการตั้งค่าด้วยเวอร์ชันแพ็คเกจที่ถูกต้อง
- 3) สามารถควบคุมและจัดการการตั้งค่า Robot environment และ Process ได้
- 4) สามารถมั่นใจได้ว่าหุ่นยนต์ที่เปิดใช้งานมีการจัดการคิวในการทำงานโดยไม่เกิดการโอเวอร์โหลด
- 5) สามารถติดตามการทำงานของหุ่นยนต์ รวมถึงการจัดการกลุ่มที่มีสิทธิ์เข้าถึงการตั้งค่าได้
- 6) สามารถบันทึกการเปลี่ยนแปลงในอดีตลงในฐานข้อมูลได้



รูปที่ 2.2 UiPath logo

(ที่มา: <https://www.velocity-it.com/business-technology-ui-path/copy-of-untitled>)



รูปที่ 2.3 หน้าต่างหลักของโปรแกรม UiPath Orchestrator

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ 5 ารศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อนักพัฒนาเขียนโปรแกรมสร้างหุ่นยนต์สำเร็จแล้ว ในการเปิดใช้งานและตั้งค่าเพื่อให้หุ่นยนต์ทำงานได้ นักพัฒนาจำเป็นต้องใช้เว็บแอปพลิเคชัน UiPath Orchestrator ในการเก็บข้อมูลแบบตัวแปรที่หุ่นยนต์ตัวนั้นต้องดึงมาใช้งาน การตั้งค่าเพื่อการทำงานของหุ่นยนต์มีหลัก ๆ 4 ส่วน

### 1) Robots

Robots เป็นตัวปฏิบัติการในการทำงานตาม Processes ที่ได้สร้างขึ้นโดย UiPath Studio โดยประเภทของหุ่นยนต์มีทั้งหมด 4 ประเภท Attended, Unattended, NonProduction, Development

### 2) Assets

Assets เป็นตัวแปรชนิดหนึ่ง ซึ่งสามารถแชร์ค่ากันใช้ระหว่างอาร์พีเอได้ โดยการที่มี Assets ทำให้การใช้งานหุ่นยนต์มีความสะดวกมากยิ่งขึ้น เนื่องจาก Assets ทำหน้าที่เก็บค่าของข้อมูลที่เฉพาะเจาะจงเพื่อที่หุ่นยนต์จะได้ดึงไปใช้งานได้

Assets มีทั้งหมด 4 ประเภท

1. Text เก็บเฉพาะค่าเป็น String เท่านั้น
2. Bool เก็บค่า True หรือ False
3. Integer เก็บเฉพาะค่าตัวเลขเท่านั้น
4. Credential เก็บค่าที่เป็น Username และ Password ซึ่งหุ่นยนต์จะนำตัวแปรประเภทนี้ไปใช้ใน Process ที่เฉพาะเช่น การเข้าสู่ระบบ SAP การเข้าสู่ระบบ Salesforce เป็นต้น

การใช้งาน Assets สามารถใช้ได้ 2 รูปแบบ

1. แบบ Global - เป็นการตั้งค่าการใช้งานโดยหุ่นยนต์ทุกตัวสามารถเรียกใช้งานได้
2. แบบ Per Robot - เป็นการตั้งค่าการใช้งานโดยมีเพียงหุ่นยนต์บางตัวเท่านั้นที่สามารถเรียกใช้งานได้

### 3) Schedules

Schedules เป็นส่วนที่นักพัฒนาสามารถจัดการควบคุมตารางเวลาในการเปิดการใช้งานหุ่นยนต์ได้ และนักพัฒนาสามารถตั้งค่าให้หุ่นยนต์เปิดใช้งานตามตารางเวลาที่แตกต่างกันได้ด้วยปัจจัยเหล่านี้

1. All Robots - Schedules ที่ตั้งไว้จะทำงานโดยหุ่นยนต์ทุกตัวใน Environment ที่ถูกตั้งค่าไว้
2. Specific Robots - Schedules ที่ตั้งไว้จะทำงานโดยหุ่นยนต์ที่ผู้ใช้งานเลือก

3. A Number of Robots - Schedules ที่ตั้งไว้จะทำงานตามจำนวนของหุ่นยนต์ การตั้งค่าประเภทนั้นนับว่าเป็นการใช้งานอย่างมีประสิทธิภาพสูงสุด ทั้งนี้ที่มีหุ่นยนต์ตัวใดที่เปลี่ยนสถานะเข้าสู่โหมด Available ระบบจะเริ่มการทำงานตาม Schedule ที่ได้ทำการจัดหามาให้

นอกจากนี้นักพัฒนายังสามารถหยุดการทำงานของหุ่นยนต์ด้วยการตั้งค่า Cancel หรือ Terminate และทำการปรับการตั้งค่าเวลาการทำงานของ Schedules ตาม Time zone ที่ระบุได้

#### 4) Queues

Queues เป็นส่วนที่เปิดให้นักพัฒนาเก็บไฟล์ข้อมูลต่างๆได้ไม่จำกัด โดยไฟล์ที่สามารถจัดเก็บใน Queues มีหลายประเภท เช่น เอกสารใบเสร็จของลูกค้า ซึ่งไฟล์ที่ถูกจัดเก็บสามารถถูกนำไปใช้งานโดยระบบอื่น ๆ เช่น SAP Salesforce เป็นต้น

NAME	MACHINE	TYPE	ENVIRONMENTS	HEARTBEAT	STATUS
DocBot	MBOBOC	NonProduction	doc_env	a few seconds ago	Available
vio	ROPRDWAPP02	Development	doc_env	a few seconds ago	Available
Stan	DESKTOP-PMFQGB	Unattended	doc_env	2 days ago	Disconnected

รูปที่ 2.4 หน้าต่างการใช้งานฟังก์ชัน Robots

(ที่มา: <https://docs.uipath.com/orchestrator/v2018.1/docs/about-robots>)

NAME	TYPE	VALUE
Blatron	Text	CCMC
NANA	Credential	username: LordHeljie
qewqeq	Credential	[Value per robot]

รูปที่ 2.5 หน้าต่างการใช้งานฟังก์ชัน Assets

(ที่มา: <https://docs.uipath.com/orchestrator/v2018.1/docs/about-assets>)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

NAME	PROCESS	ENVIRONMENT	TRIGGER DETAILS	NEXT RUN TIME	STOP AFTER
1VerySimpleInvoke_a_invoke_e...	1VerySimpleInvoke	a_invoke_env	At 22 minutes past the hour, every 02...	in an hour	12 minutes
Add to queue every 10 minutes	AddToNRAQueue	OBE for queue	Every 10 minutes	in 9 minutes	30 minutes
AddTransFAST_andra	AddTransFAST	andra	At 05:05 AM	Disabled	
AddTransFAST_laptop_mrk_env	AddTransFAST	laptop_mrk_env	Every 10 minutes	Disabled	

รูปที่ 2.6 หน้าต่างการใช้งานฟังก์ชัน Schedules

(ที่มา: <https://docs.uipath.com/orchestrator/v2018.1/docs/about-schedules>)

NAME	DESCRIPTION	IN PROGRESS	REMAINING	AVERAGE TIME	SUCCESSFUL	APP EXCEPTIONS	BIZ EXCEPTIONS
DocQueue		0	1369	0 s	0	0	0
Queue1		0	0	0 s	0	0	0
WrtQueue		0	0	0 s	0	0	0

รูปที่ 2.7 หน้าต่างการใช้งานฟังก์ชัน Queues

(ที่มา: <https://docs.uipath.com/orchestrator/v2018.1/docs/about-queues-and-transactions>)

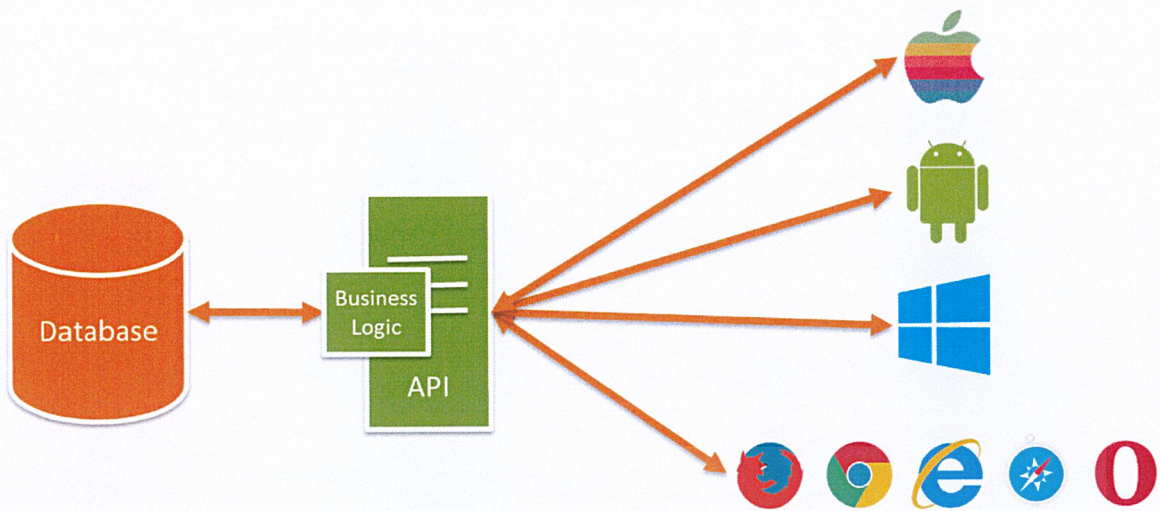
### 2.3 API

API (Application Programming Interface) หมายถึงวิธีการที่ระบบปฏิบัติการ ไลบรารี หรือบริการอื่น ๆ เปิดให้โปรแกรมคอมพิวเตอร์สามารถติดต่อเรียกใช้งานได้

API สามารถแบ่งออกได้เป็น 2 ประเภท คือ

1. API ที่ขึ้นกับภาษา (Language-Dependent API) คือ API ที่สามารถการเรียกใช้จากโปรแกรมที่เขียนขึ้นด้วยภาษาเพียงภาษาใดภาษาหนึ่ง
2. API ที่ไม่ขึ้นกับภาษา (Language-Independent API) คือ API ที่สามารถเรียกได้จากโปรแกรมหลาย ๆ ภาษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ 8 ระดับเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.8 ภาพรวมการใช้งาน API

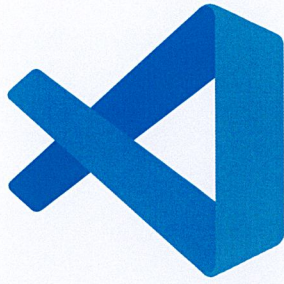
## 2.4 ระบบ ITSM

ITSM เป็นระบบหนึ่งภายในบริษัทที่ใช้สำหรับการสร้างและส่งคำร้อง (Ticket) มายังผู้ดูแลระบบฝั่งไอที (EMIT support group) เมื่อเจอปัญหา หรือต้องการให้ผู้ดูแลระบบทำงานตามคำร้อง (Ticket) โดยเมื่อผู้ใช้มีคำร้องหรือต้องการความช่วยเหลือจากผู้ดูแลระบบ ผู้ใช้จะทำการส่งคำร้องเพื่อขอความช่วยเหลือจากผู้ดูแลระบบ ITSM จะเก็บคำร้องทั้งหมดของผู้ใช้ ทำให้ง่ายต่อการจัดการตามลำดับความสำคัญ โดยจะมีการตั้งสถานะของคำร้องนั้น ๆ ว่าขณะนี้ดำเนินการถึงขั้นตอนไหน โดยบริษัทจะมีคำร้องหลักอยู่ 4 ประเภท คือ Incident (INC), Work Order (WO), Task (TAS) และ Change Request (CRQ) โดยระบบ ITSM มีการพัฒนาให้มี User interface ที่ใช้งานง่ายมากขึ้นเรียกว่า “IT console”

## 2.5 โปรแกรม Visual Studio Code

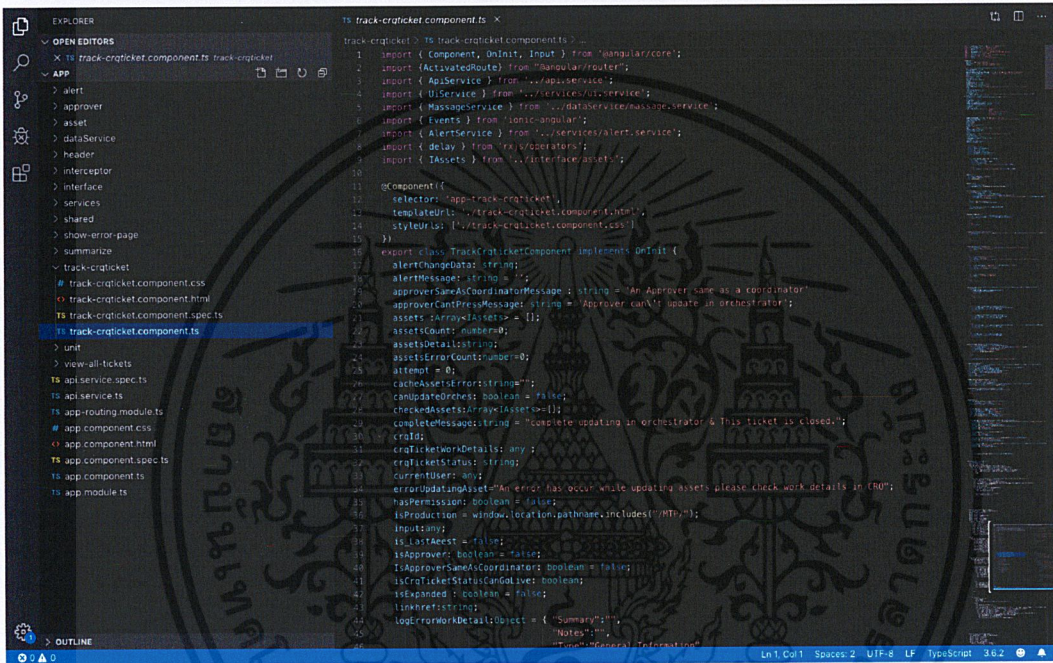
Visual Studio Code เป็นเครื่องมือสำหรับนักพัฒนาโปรแกรมที่ถูกพัฒนาขึ้นโดยบริษัทไมโครซอฟต์ โดยเครื่องมือดังกล่าวมีความสามารถในการเปิดอ่าน แก้ไข และเขียนโค้ดเหมือน editor ตัวอื่น ๆ เช่น Sublime, Atom, Notepad++ แต่มีจุดเด่นคือ ความสามารถในการติดตั้งเครื่องมือเสริม (Extension) ที่ครบครันสำหรับรองรับความต้องการในการพัฒนาโปรแกรม นอกจากนั้น Visual Studio Code ยังสามารถใช้งานได้บนระบบปฏิบัติการวินโดวส์ แมค และลินุกซ์ และทางไมโครซอฟท์เปิดบริการให้ใช้งานโดยไม่ต้องเสียค่าใช้จ่ายใด ๆ รูปแบบของโปรแกรมถูกออกแบบให้สามารถใช้งานได้ง่ายและไม่ซับซ้อน มีความเป็นมืออาชีพ นอกจากนั้นยังสามารถรองรับได้มากกว่า 30 โปรแกรมภาษา เช่น C++, C#, CSS, Dockerfile, HTML, JavaScript, JSON, Less, Markdown, PHP, Python, Sass, TypeScript เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

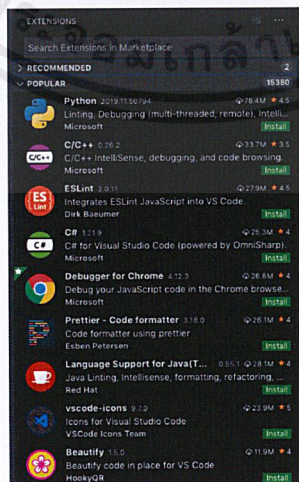


รูปที่ 2.9 Visual Studio Code logo

(ที่มา: <https://worldvectorlogo.com/logo/visual-studio-code-1>)



รูปที่ 2.10 หน้าต่างโปรแกรม Visual Studio Code



รูปที่ 2.11 ตัวอย่างของเครื่องมือเสริมใน Visual Studio Code

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ 10 วิชาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.6 Angular

Angular เป็น Front-end framework สำหรับสร้างแอปพลิเคชันในฝั่งไคลเอนในรูปแบบของ HTML, CSS และ JavaScript/TypeScript ซึ่ง TypeScript จะถูก compile ไปเป็น JavaScript เพื่อนำไปแสดงผล สาเหตุที่ในยุคนี้มีการเขียนโปรแกรม Front-end ด้วย Angular กันอย่างแพร่หลายเนื่องจากในอดีตการใช้งาน JavaScript เพื่อแสดงผลบนเว็บไซต์ต้องเขียนด้วยภาษา JavaScript หรือ jQuery แต่ในปัจจุบันการเขียนโปรแกรมพัฒนาเว็บไซต์มีความซับซ้อนมากยิ่งขึ้นทำให้การเขียนด้วยภาษา JavaScript หรือ jQuery มีความยากลำบากในการดูแลรักษาเพื่อการพัฒนาในอนาคต จึงเป็นที่มาของการพัฒนาเป็น JS Framework เช่น Angular React Vue.js เป็นต้น นอกจากนี้ Angular ยังมีความสามารถอื่น ๆ เช่น มีการ re-usable code ต่าง ๆ เช่น navigation หรือ browser history และ Angular ยังมีการจัดหารูปแบบการ test ที่หลากหลายให้เลือกใช้สำหรับทดสอบแอปพลิเคชัน



รูปที่ 2.12 Visual Studio Code logo

(ที่มา: <https://www.pluralsight.com/paths/angular>)

## 2.7 Azure DevOps Server

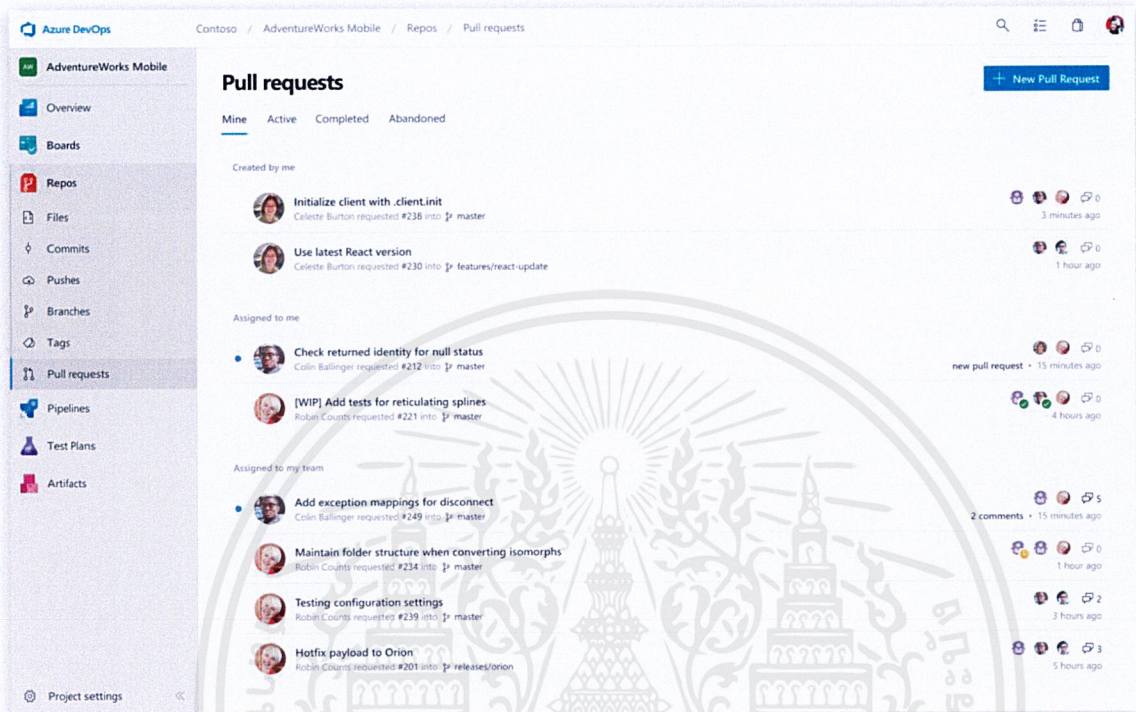
Azure DevOps Server หรือชื่อเดิมคือ Visual Studio Team Foundation Server (TFS) เป็นเครื่องมือสำหรับช่วยกระบวนการของการพัฒนาระบบ ผลิตโดยบริษัทไมโครซอฟท์ ซึ่งประกอบด้วยเครื่องมือต่าง ๆ ที่ช่วยในการพัฒนาระบบในหลาย ๆ ส่วน เช่น Source Code Management, Requirement Management, Project Management, Automation Build, Test and Release เป็นต้น โดย Azure DevOps มีการให้บริการที่ช่วยตอบสนองความต้องการทางธุรกิจทั้งหมด 5 ส่วนหลัก ๆ คือ

### 1. Azure Repos

Azure Repos ทำหน้าที่เป็น Version Control ให้บริการสำหรับเก็บ Source Code Version Control นี้เป็นระบบที่ช่วยบันทึกการเปลี่ยนแปลงของ Source Code ทุกครั้งที่มีการเปลี่ยนแปลง โดยระบบจะทำการเก็บไฟล์ทุกเวอร์ชันที่ถูกเปลี่ยนแปลง ทำให้พัฒนาสามารถเรียกดูหรือย้อนกลับไปยัง Source Code เวอร์ชันก่อนหน้าได้เมื่อครั้งที่ต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ11รศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มากกว่านั้นในการทำงานเป็นทีม หากนำเครื่องมือนี้มาใช้ จะทำให้สามารถจัดการการเปลี่ยนแปลง Source Code ของทุกคนในทีมได้ดียิ่งขึ้น โดย Version Control ของ Azure Repos มี 2 ชนิด Git เป็น Distributed Version Control และ Team Foundation Version Control (TFVC) เป็น Centralized Version Control

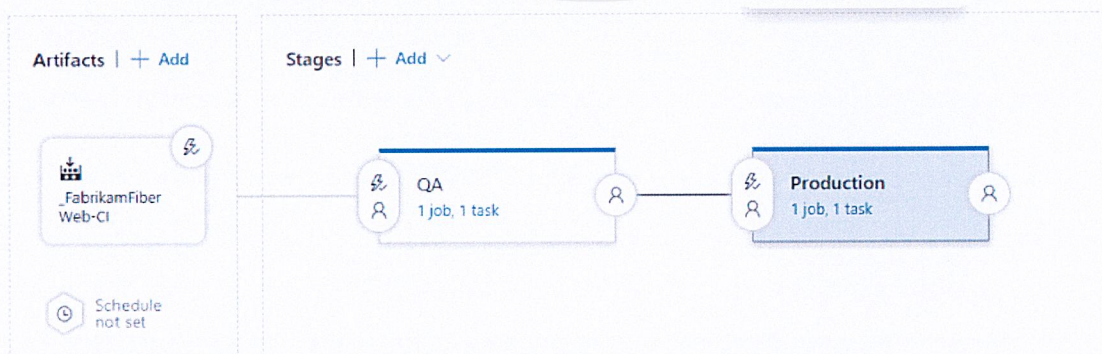


รูปที่ 2.13 หน้าต่างการใช้งาน Azure Repos

(ที่มา: <https://azure.microsoft.com/en-us/services/devops/repos/>)

## 2. Azure Pipelines

Azure Pipelines เป็นเครื่องในการสร้าง Build & Release Pipeline ที่นักพัฒนาสามารถปรับแต่งได้ตามความต้องการ โดย Azure Pipelines มีการรองรับภาษาที่หลากหลาย และช่วยในการตรวจสอบโปรแกรมที่เขียนก่อนนำขึ้น Environment ที่ตั้งค่าไว้



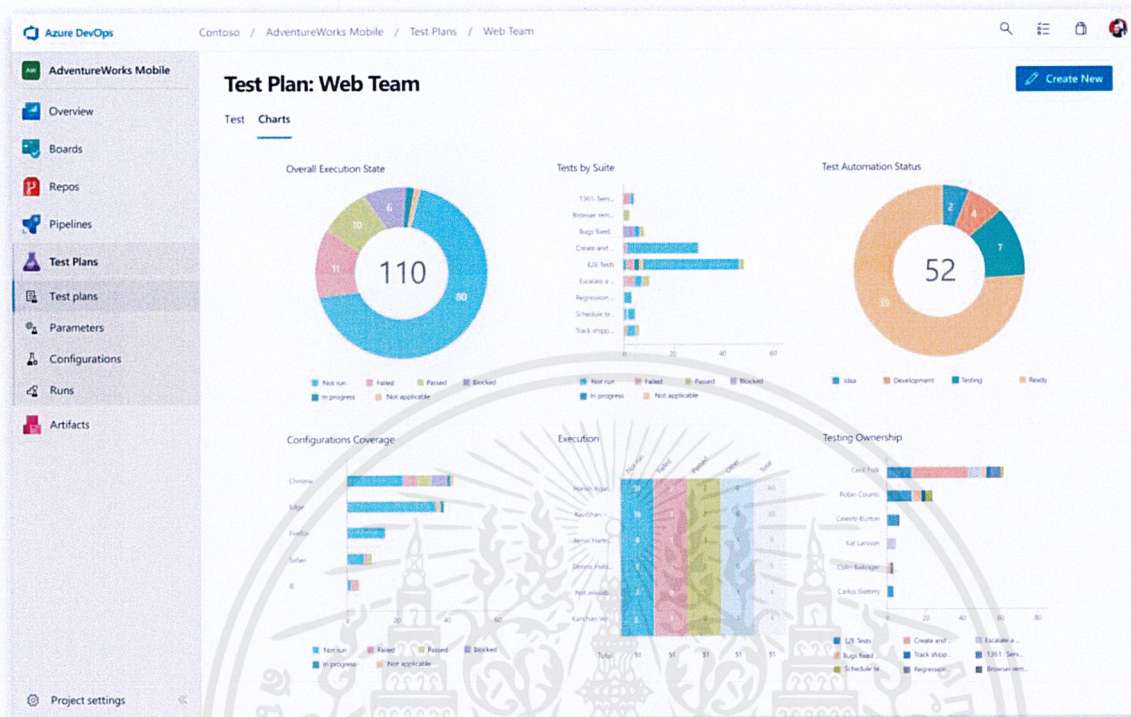
รูปที่ 2.14 หน้าต่างการใช้งาน Azure Repos

(ที่มา: <https://docs.microsoft.com/th-th/azure/devops/pipelines>)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ 12 วิชาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3. Azure Test Plans

Azure Test Plans เป็นเครื่องมือที่รองรับการ Test แบบครบวงจร ตัวอย่างเช่น manual/exploratory testing and continuous testing เป็นต้น

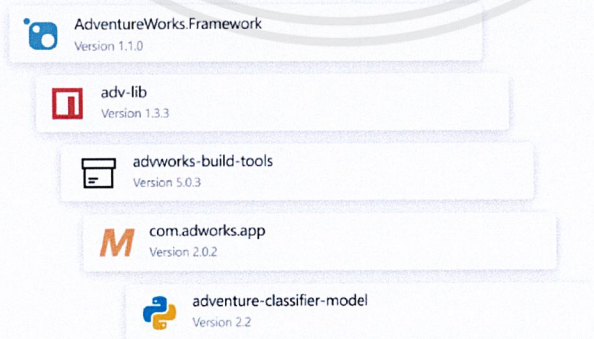


รูปที่ 2.15 หน้าต่างการใช้งาน Azure Test Plans

(ที่มา: <https://azure.microsoft.com/en-us/services/devops/test-plans/>)

### 4. Azure Artifacts

Azure Artifacts เป็นเครื่องมือที่ใช้ในการสร้างแพ็คเกจ เช่น Maven, npm, NuGet และสามารถเลือกได้ระหว่าง Public หรือ Private เพื่อนำมาใช้ร่วมกับ CI/CD pipelines



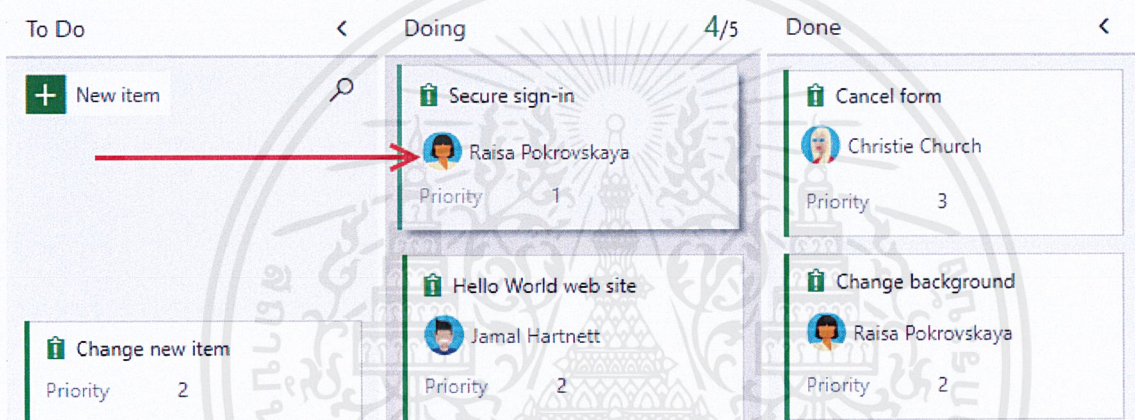
รูปที่ 2.16 หน้าต่างการใช้งาน Azure Artifacts

(ที่มา: <http://microsoftonlineguide.blogspot.com/2018/09/deep-dive-into-azure-artifacts.html>)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ 13 วิชาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

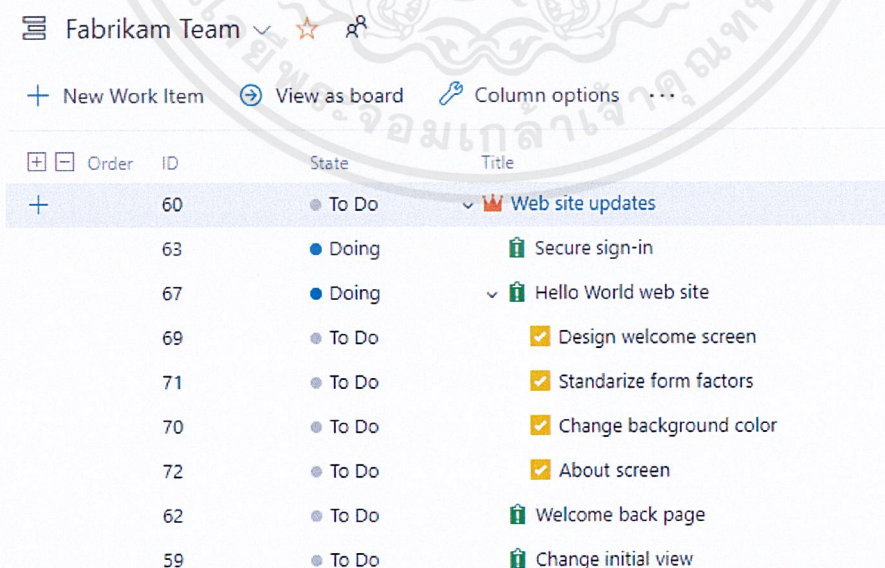
## 5. Azure Boards

Azure Boards เป็นเครื่องมือในการช่วยจัดการงาน ติดตามงาน และกระจายงานให้กับคนในทีม โดยเครื่องมือนี้จะมีหน้าตาการทำงานแบบบอร์ดในแสดงแผนการทำงานต่าง ๆ เช่น การทำ Product Backlog การทำ Planning โดยจะออกมาเป็นลักษณะของ Kanban boards และมีตัวเลือกให้บริการอย่างหลากหลาย เช่น กำหนดระดับงานแบบ Scrum เครื่องมือนี้สามารถกำหนดได้หลายระดับ เช่น Epic > Feature > Story > Task และแต่ละระดับสามารถกำหนดคนรับผิดชอบได้มากกว่า 1 คน นอกจากนี้บอร์ดนี้ยังสามารถปรับแต่งการทำงานได้ตามความต้องการ เช่น การผูก Feature, Story และ Task เข้ากับการ commit งานใน Version Control เพื่อให้ระบบจัดทำ Automation ในการเลื่อนงานที่เสร็จแล้วไปยัง state ต่าง ๆ อีกทั้งยังมีการวิเคราะห์และสร้างกราฟสรุป performance ของทีม หรือรายบุคคล



รูปที่ 2.17 หน้าตาแสดงการอัปเดตสถานะของงาน

(ที่มา: <https://docs.microsoft.com/th-th/azure/devops/boards>)



รูปที่ 2.18 หน้าตาการจัดการจัดกลุ่มและไล่ระดับงาน

(ที่มา: <https://docs.microsoft.com/th-th/azure/devops/boards>)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ14รศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 3

### ขั้นตอนการดำเนินงานพัฒนา

#### 3.1 ขั้นตอนการดำเนินงานพัฒนาระบบซอฟต์แวร์อัตโนมัติเพื่อบริหารจัดการอาร์พีเอ

RPA change management tool เป็นระบบซอฟต์แวร์อัตโนมัติที่ผู้พัฒนาสร้างขึ้น เพื่อช่วยอำนวยความสะดวกแก่นักพัฒนาหุ่นยนต์ และเพิ่มประสิทธิภาพในกระบวนการเพิ่มและเปลี่ยนแปลง การตั้งค่าที่จำเป็นเพื่อให้หุ่นยนต์ดึงมาใช้งาน ซึ่งการทำงานครั้งนี้มีผู้ร่วมพัฒนาคือ นายพิศพลย์ ชันแข็ง โดยผู้พัฒนามีโอกาสได้ทำงานในทีม RACE ซึ่งมีผู้ร่วมงาน 13 คน แบ่งเป็น 2 Regions คือ Bangkok, Thailand 8 คน และ Curitiba, Brazil 5 คน โดยมี Product owner (PO) จาก Houston, USA คอยรับ และติดตามงานของลูกค้า ในการทำงานภายในทีม ผู้พัฒนาได้เรียนรู้รูปแบบการทำงาน ของ Scrum Framework เป็นหลัก เพื่อให้การทำงานร่วมกันสามารถเป็นไปได้อย่างลื่นไหล และสามารถแก้ไขปัญหา รวมถึงปรับเปลี่ยนแผนการทำงานได้อย่างทันท่วงที ทั้งนี้เพื่อให้การทำงานร่วมกันเป็นไปได้อย่างมีประสิทธิภาพ จึงได้มีการจัดประชุมหนึ่งครั้งต่อสัปดาห์ เพื่อติดตามความคืบหน้า วางแผน และร่วมกันแก้ไขปัญหา โดยแผนการทำงานทั้งหมดแสดงได้ ดังตารางที่ 3.1

ตารางที่ 3.1 แผนการทำงานพัฒนาระบบซอฟต์แวร์อัตโนมัติเพื่อบริหารจัดการอาร์พีเอ

งาน \ ระยะเวลา	มิ.ย.	ก.ค.	ส.ค.	ก.ย.	ต.ค.	พ.ย.	ธ.ค.
ประชุมงาน	1 ครั้ง/สัปดาห์						
ศึกษาระบบ	■						
ศึกษาความต้องการ		■					
วางแผนการพัฒนา			■	■	■	■	■
พัฒนาระบบ			■	■	■	■	■
ทดสอบการทำงาน			■	■	■		■
นำไปใช้งานจริง				■	■	■	■

##### 3.1.1 การประชุมงาน

ในการทำงานภายในทีม RACE ได้เลือกรูปแบบการทำงานของ “Scrum Framework” ซึ่ง Scrum เป็นวิธีการทำงานที่ช่วยให้การพัฒนาผลิตภัณฑ์มีขั้นตอนการดำเนินงานและผลลัพธ์ที่ชัดเจน โปร่งใส สามารถตรวจสอบประสิทธิภาพของแต่ละขั้นตอนการดำเนินงาน สามารถปรับปรุงและวัดผลการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ 15 ศึกษานี้เท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปรับปรุงที่เกิดขึ้นได้ “สปรินท์” คือรอบเวลาการดำเนินงานแบบสกรรมในการพัฒนาผลิตภัณฑ์ เมื่อจบสปรินท์จะต้องได้รับชิ้นงานตามที่วางแผนไว้ และสามารถนำไปส่งมอบให้แก่ผู้ใช้ ทำให้ผู้ใช้ไม่จำเป็นต้องรอจนงานเสร็จทั้งหมดก่อน แต่สามารถนำบางส่วนไปใช้งานก่อนเพื่อทดลองใช้เพื่อจะได้ปรับปรุงแก้ไขอย่างทันทีทันใด จากนั้นส่วนอื่น ๆ จะทยอยตามมาเพิ่มเติมในภายหลัง เนื่องจากสมาชิกภายในทีมทำงานต่าง Time zone ทำให้มีการใช้นำเครื่องมือ Azure board เข้ามาช่วยในการอำนวยความสะดวก โดยภายใน RACE ได้ตกลงไว้ว่า 1 สปรินท์เท่ากับ 2 สัปดาห์ เพราะฉะนั้นครบ 2 สัปดาห์จะมีการนำเสนอการดำเนินงานตลอดสปรินท์ และส่งมอบชิ้นงานให้แก่ลูกค้า เรียกว่า “Demo presentation” เมื่อจบการนำเสนอให้กับลูกค้าแล้ว จะมีการประชุมวางแผนงานที่จะเกิดขึ้นในสปรินท์ถัดไปเรียกว่า “PBI Planning” โดยการประชุมนี้จะมีเฉพาะสมาชิกในทีมเท่านั้น เริ่มต้นจาก Product Owner (PO) ทำการนำ Requirement ของลูกค้าใส่เข้าไปใน Azure board จากนั้นสมาชิกในทีมจะเลือกงานที่ตัวเองสนใจ และ PO จะระบุชื่อผู้ที่ได้รับมอบหมายลงในงานนั้น ๆ ดังรูปที่ 3.1 เมื่อถึงวันที่จะต้องดำเนินงานทุกคนในทีมจำเป็นต้องอัปเดตสถานะของงาน โดยสถานะของงานแบ่งออกเป็น 3 สถานะคือ To do คืองานที่จะต้องทำ, Doing คืองานที่กำลังดำเนินอยู่, Done คืองานที่เสร็จแล้ว เพื่อที่เมื่อถึงกลางสปรินท์ PO จะจัดประชุมอีกครั้งเรียกว่า “Grooming” โดยการประชุมครั้งนี้มีจุดประสงค์เพื่อพิจารณาว่ามีงานใดบ้างที่อาจจะเสร็จไม่ทัน เพื่อนำไปต่อรองกับลูกค้าต่อไป และในทุก ๆ วัน จะมีการประชุมในตอนเช้าระยะเวลา 30 นาที เรียกว่า “Daily stand-up” การประชุมนี้จะเข้าร่วมเพียงสมาชิกใน Time zone นั้นและ PO โดยมีจุดประสงค์เพื่อรายงานสิ่งที่ได้ทำไปในวันก่อนหน้า และสิ่งที่จะทำในวันนั้น ๆ รวมถึงการปรึกษาปัญหาที่เกิดขึ้น และขอความช่วยเหลือจากสมาชิกคนอื่น ๆ

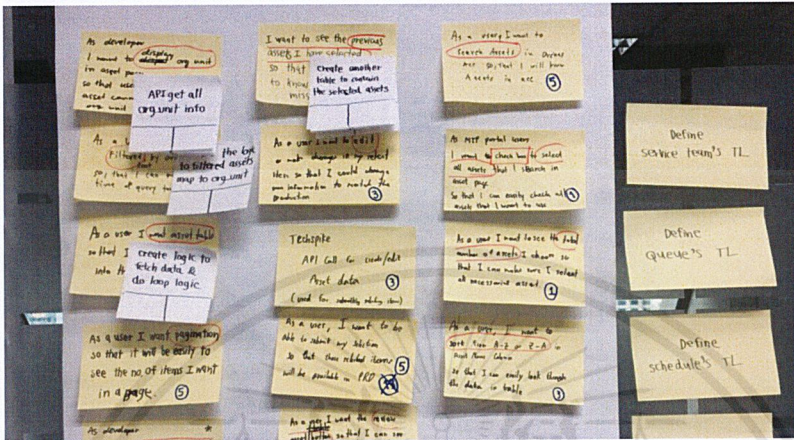
Order	ID	State	Assigned To	Title
1	3763	● Doing	Jamal Hartnett	Secure sign-in
2	3767	● Doing	Raisa Pokrovskaya	Hello World web site
3	3766	● Doing	Christie Church	Resolve service status issues
4	3764	● Doing	Johnnie McLeod	Check performance
5	3773	● To Do	Christie Church	Update welcome page
6	3759	● To Do	... Jamal Hartnett	Change initial view
7	3762	● To Do	Christie Church	Welcome back page

รูปที่ 3.1 ตัวอย่าง Azure board

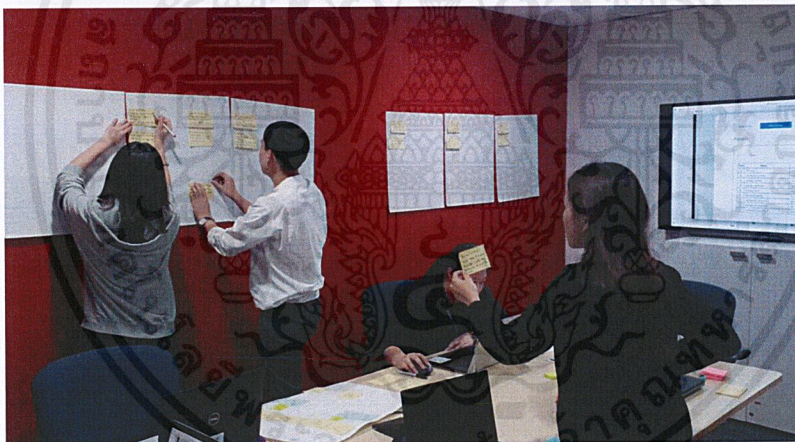
(ที่มา: <https://docs.microsoft.com/th-th/azure/devops/boards>)

ในสองเดือนแรกผู้พัฒนาได้มีการเข้าประชุม และใช้ Azure board ร่วมกันกับทีม แต่เนื่องจากการวัด Performance และการคาดการณ์เวลาในการทำงานในแต่ละ Feature ไม่สามารถวัดโดยอิงจากสมาชิกคนอื่น ๆ ที่ทำงานหลายปี ทำให้ผู้ดูแลได้ทำ Physical board ดังรูปที่ 3.2 และมีการวางแผนการนี้เป็นเอกสารที่ส่งมอบไว้สำหรับการใช้งานเพื่อ 16 วิชาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แผนการประชุมงานใหม่ โดยมีการปรับเปลี่ยนช่วงของสปรินท์เป็น 1 สปรินท์เท่ากับ 1 สัปดาห์ ซึ่ง Grooming และ PBI planning จะจัดในวันแรกของสปรินท์โดยมีตัวอย่างการทำงานดังรูปที่ 3.3 นอกจากนี้ยังมี Extra demo presentation ในทุกสัปดาห์โดยเข้าร่วมเฉพาะผู้พัฒนาและ PO เท่านั้น ส่วนสิ่งที่ยังคงทำร่วมกันกับทีมคือ Demo presentation กับลูกค้ำ และ Daily stand-up



รูปที่ 3.2 ตัวอย่าง Physical board



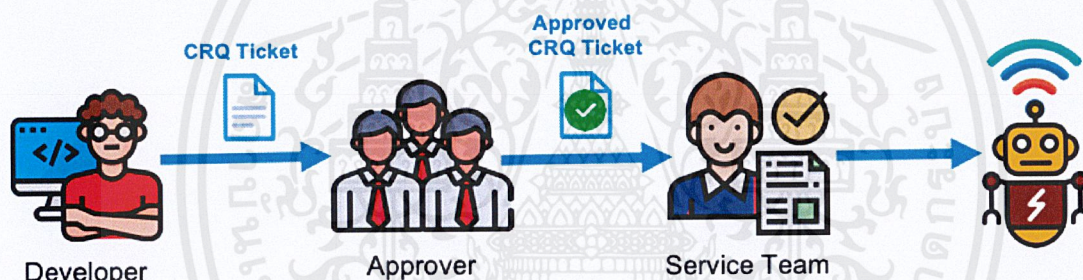
รูปที่ 3.3 การประชุม PBI planning โดยใช้ Physical board

### 3.1.2 การศึกษาการทำงานของระบบ

โครงการที่ผู้พัฒนาได้รับมอบหมายเป็นโครงการเกี่ยวกับการพัฒนาระบบการนำหุ่นยนต์ที่ผ่านการทดสอบแล้วขึ้นสู่ Production Environment ซึ่งเป็นสิ่งแวดล้อมที่เผยแพร่ให้ใช้งานจริงอย่างเป็นทางการ เนื่องด้วยกระบวนการนี้มีความสำคัญและมีส่วนที่ต้องระวังมากมายเพราะหากมีข้อผิดพลาดจะทำให้ส่งผลเสียเป็นอย่างมาก ผู้พัฒนาจึงจำเป็นต้องศึกษาการทำงานและข้อจำกัด รวมถึงนโยบายในกระบวนการนี้อย่างละเอียดเพื่อที่จะได้พัฒนาระบบนี้อย่างสมบูรณ์ที่สุด

ในส่วนของการศึกษาระบบ ผู้พัฒนาได้ศึกษาเกี่ยวกับกระบวนการทำงานของระบบเก่า เพื่อทำความเข้าใจและวิเคราะห์ทุก ๆ จุดอย่างละเอียด และนำมาพัฒนาระบบใหม่ที่จะสร้างขึ้น โดยระบบเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ 17 วิชาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

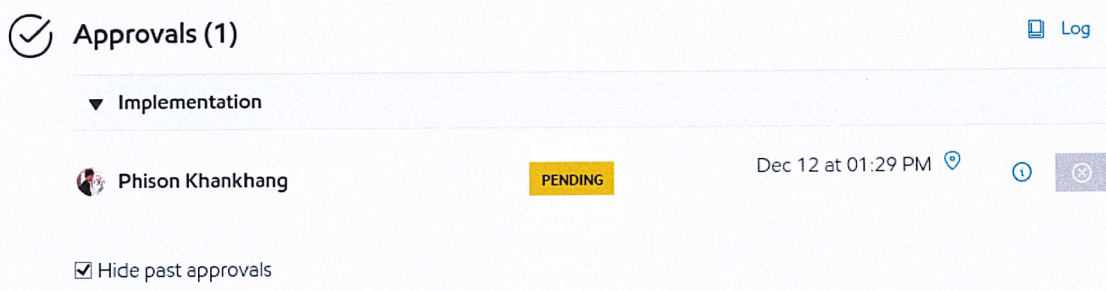
เก่ามีกระบวนการดังรูปที่ 3.4 โดยเริ่มหลังจากที่ผู้พัฒนาหุ่นยนต์ทดสอบการทำงานผ่าน Service Level Agreement (SLA) แล้ว จากนั้นผู้พัฒนาหุ่นยนต์จะทำการกรอกข้อมูลที่ต้องการตั้งค่าใน UiPath Orchestrator เช่น Robot Script คือเวอร์ชันโค้ดของหุ่นยนต์ที่ต้องการ Deploy, Assets คือตัวแปรที่หุ่นยนต์จะดึงค่ามาใช้ในการทำงาน เป็นต้น ซึ่งข้อมูลเหล่านี้ผู้พัฒนาต้องกรอก Excel form ดังรูปที่ 3.5 จากนั้นผู้พัฒนาต้องเข้าสู่การสร้าง Change request ticket (CRQ ticket) ใน IT console และกรอกรายละเอียดเช่น ชื่อ Ticket ระดับความเสี่ยง ระดับความเร่งด่วน ผู้อนุมัติ (Approvers) ผู้ถูกมอบหมาย (Change coordinator) ผู้ดูแลการเปลี่ยนแปลง (Change Manager) จากนั้นผู้พัฒนาจะต้องแนบไฟล์ Excel form หรือในบางแผนกอาจมีการแนบไฟล์อื่น ๆ เพิ่มเติม เช่น QA Report, Business confirmation เป็นต้น จากนั้น CRQ ticket นี้จะถูกส่งไปยังผู้อนุมัติ โดยผู้พัฒนาสามารถติดตามสถานะของการอนุมัติได้ดังรูปที่ 3.6 เมื่อผู้อนุมัติทุกคนอนุมัติเรียบร้อยแล้ว CRQ ticket จะถูกส่งต่อไปยัง Service team ซึ่ง Service team คือบุคคลที่มี IT support role โดยเมื่อ Service team ตรวจสอบเช็คความถูกต้องของ Approvers แล้วก็ทำการคัดลอกรายละเอียดทุกอย่างจาก Excel form ลงในการตั้งค่า UiPath Orchestrator โดยยังคงมีวิธีการทำงานแบบ Manual



รูปที่ 3.4 กระบวนการนำหุ่นยนต์ขึ้นสู่ Production environment แบบเก่า

Create / Deploy process		Create assets							
Process Name	RACS_MTFPv12AgreeWQ	Add Update Comment	Value per Bot?	Process	Type	Bot	Value	User	Password
Package Version	1.0.1117.40817								Control BMT team to update password
Environment ID (linked only in Orchestrator)									Control BMT team to update password
Package description	Update release								Control BMT team to update password
TFIS Package Name	RACS_MTFPv12AgreeWQ								Control BMT team to update password
Release	Release-2								Control BMT team to update password
TFIS Process Line									Control BMT team to update password

รูปที่ 3.5 แบบฟอร์มกรอกข้อมูลในกระบวนการนำขึ้น Production



รูปที่ 3.6 สถานะของผู้อนุมัติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ 18 รัชศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นอกจากการศึกษาระบบการทำงานของระบบเก่าแล้ว ในการพัฒนาระบบใหม่ผู้พัฒนาจำเป็นต้องศึกษาภาษาที่ใช้ในการพัฒนา เนื่องจากเป็นภาษาที่ไม่เคยเขียนมาก่อนซึ่งก็คือ Angular7 และการทำงานของ Orchestrator และ IT console API ซึ่งใช้เวลาเรียนรู้ทั้งหมด 2-3 สัปดาห์ โดยเรียนผ่าน Course ต่าง ๆ ในเว็บไซต์ที่ชื่อว่า “Pluralsight” ทั้งความรู้พื้นฐานและการนำไปต่อยอด

### 3.1.3 ศึกษาและรวบรวมความต้องการปัจจุบัน

หลังจากผู้พัฒนาได้ศึกษาระบบการทำงานของระบบเก่า รวมถึงการศึกษภาษาที่ใช้ในการพัฒนาเรียบร้อยแล้ว ผู้พัฒนาได้ทำการสัมภาษณ์ผู้ที่ใช้งานจริงโดยเกี่ยวข้องกับธุรกิจขององค์กร โดยแผนกที่มีการใช้ระบบการนี้ มีทั้งสิ้น 4 แผนกได้แก่ Procurement HR Supply และ Fuels & Lubes จากการรวบรวมความต้องการต่าง ๆ ที่จำเป็นจากลูกค้าซึ่งเป็นผู้ที่ใช้งานจริงโดยมีความเกี่ยวข้องกับธุรกิจขององค์กร ผู้พัฒนาได้จำแนกประเภทของความต้องการในการพัฒนาระบบออกเป็น 2 ส่วน ได้แก่ ความต้องการของผู้พัฒนาเพื่อพัฒนาระบบ และความต้องการทางธุรกิจขององค์กร

#### 3.1.3.1 ความต้องการของผู้พัฒนาเพื่อพัฒนาระบบ

เพื่อให้การพัฒนาระบบเป็นไปตามข้อกำหนดของบริษัท และตรงต่อความต้องการทางธุรกิจขององค์กร ผู้พัฒนาจึงได้จัดเตรียมสิ่งที่จำเป็นต่อการพัฒนาระบบไว้ ดังนี้

1. ขอและติดตั้งโปรแกรม Visual Studio และ Visual studio code
2. ขอและติดตั้งโปรแกรม Adobe XD และ Photoshop
3. ขอสิทธิ์เข้าถึง UiPath Orchestrator ของทีม และทีมที่เกี่ยวข้อง
4. ขอสิทธิ์เข้าถึง ITSM system ของทีมเพื่อการจัดการ Ticket ที่เกี่ยวข้อง
5. ขอสิทธิ์เข้าถึง Azure DevOps Server และ E-mail ของทีม

#### 3.1.3.2 ความต้องการทางธุรกิจขององค์กร

สำหรับความต้องการทางธุรกิจขององค์กร ผู้ใช้งานต้องการทั้งหมด 4 ฟีเจอร์คือ ฟีเจอร์ Robot script, ฟีเจอร์ Assets, ฟีเจอร์ Schedules และ ฟีเจอร์ Queues โดยผู้ใช้จากแผนกต่าง ๆ ได้เลือกฟีเจอร์ที่ต้องการมากที่สุด และต้องการให้พัฒนาเป็นอันดับแรกคือ ฟีเจอร์ Assets สามารถจำแนกออกได้เป็น 4 ขอบเขตหลัก โดยมีรายละเอียดดังนี้

1. สามารถระบุตัวตน
  - ข้อมูลพื้นฐานของผู้ใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ 19 ศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. สามารถลดความผิดพลาด ความซ้ำซ้อน และเวลาในกระบวนการทำงานแบบเก่า

- การเพิ่มหรือแก้ไข Assets ใน UiPath Orchestrator อัตโนมัติ
- การลดจำนวนช่องในการกรอกข้อมูลใน CRQ ticket เหลือเพียงช่องที่จำเป็นเท่านั้น
- การเพิ่มหรือแก้ไข Assets ใน UiPath Orchestrator อัตโนมัติ
- การสร้างระบบเพื่อนเป็น All-in-one Portal โดยระบบเดียวสามารถทำกระบวนการตั้งแต่ต้นจนจบ
- การสร้าง User interface แบบใหม่ที่ง่ายต่อการใช้งาน โดยใช้ UiPath Orchestrator API
- การพัฒนา Feature ใหม่โดยแก้ไขสิ่งที่ UiPath Orchestrator ไม่สามารถได้

3. สามารถทำงานร่วมกับระบบอื่น ๆ ได้

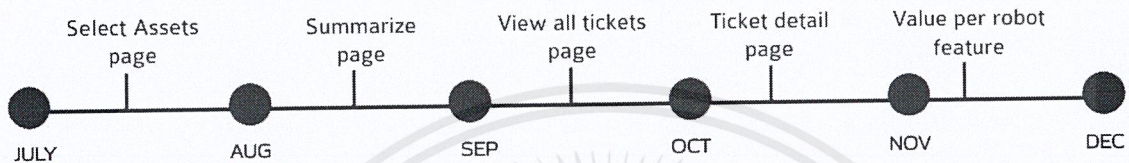
- ITSM API
- UiPath Orchestrator API

4. สามารถควบคุมการทำงานให้ตรงตามนโยบายของบริษัท

- การคัดกรองตัวเลือกบุคคลในการกรอก Approvers, Change coordinator, Change Manager โดยมีเฉพาะบุคคลที่มี Role ตามที่ระบุ
- การตรวจสอบความสมบูรณ์ครบถ้วนของ Approvals
- การตรวจสอบความเหมาะสมของการกรอก เช่น Change coordinator ไม่สามารถเป็นบุคคลเดียวกันกับ Approvers ได้
- การตรวจสอบการมีสิทธิ์เข้าถึง ITSM
- การตรวจสอบการมีสิทธิ์เข้าถึง UiPath orchestrator

### 3.1.4 วางแผนการพัฒนาระบบ

จากการศึกษาและรวบรวมข้อมูลความต้องการทางธุรกิจตามหัวข้อที่ 3.1.3.2 นั้น ผู้พัฒนาได้ทำการวางแผนและออกแบบการพัฒนา โดยระบบที่พัฒนามีทั้งหมด 4 หน้าและ 1 ฟีเจอร์เสริม โดยมีรายละเอียด ดังรูปที่ 3.7



รูปที่ 3.7 แผนการพัฒนาระบบ

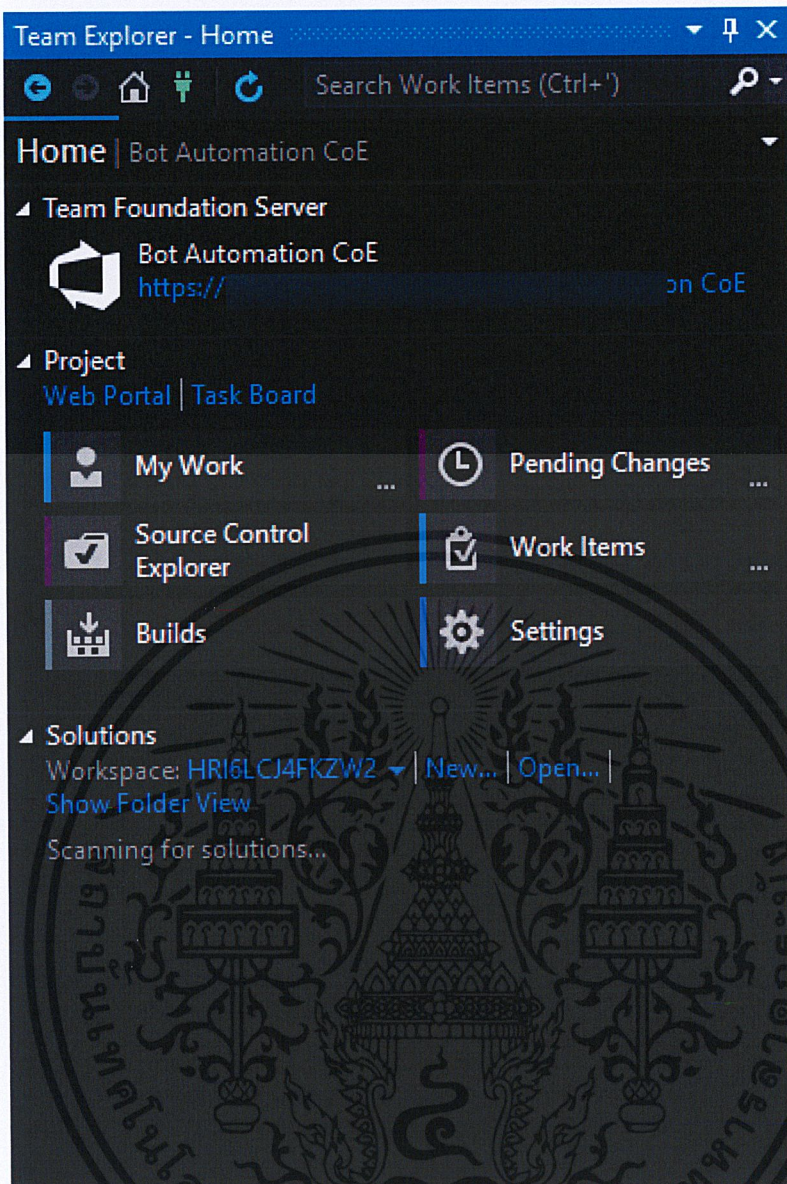
### 3.1.5 การพัฒนาระบบ

จากการวางแผนงานในหัวข้อที่ 3.1.4 ผู้พัฒนาสามารถแจกแจงรายละเอียดการพัฒนาได้ ดังนี้

#### 3.1.5.1 การเชื่อมต่อเข้า Team Foundation Server (TFS)

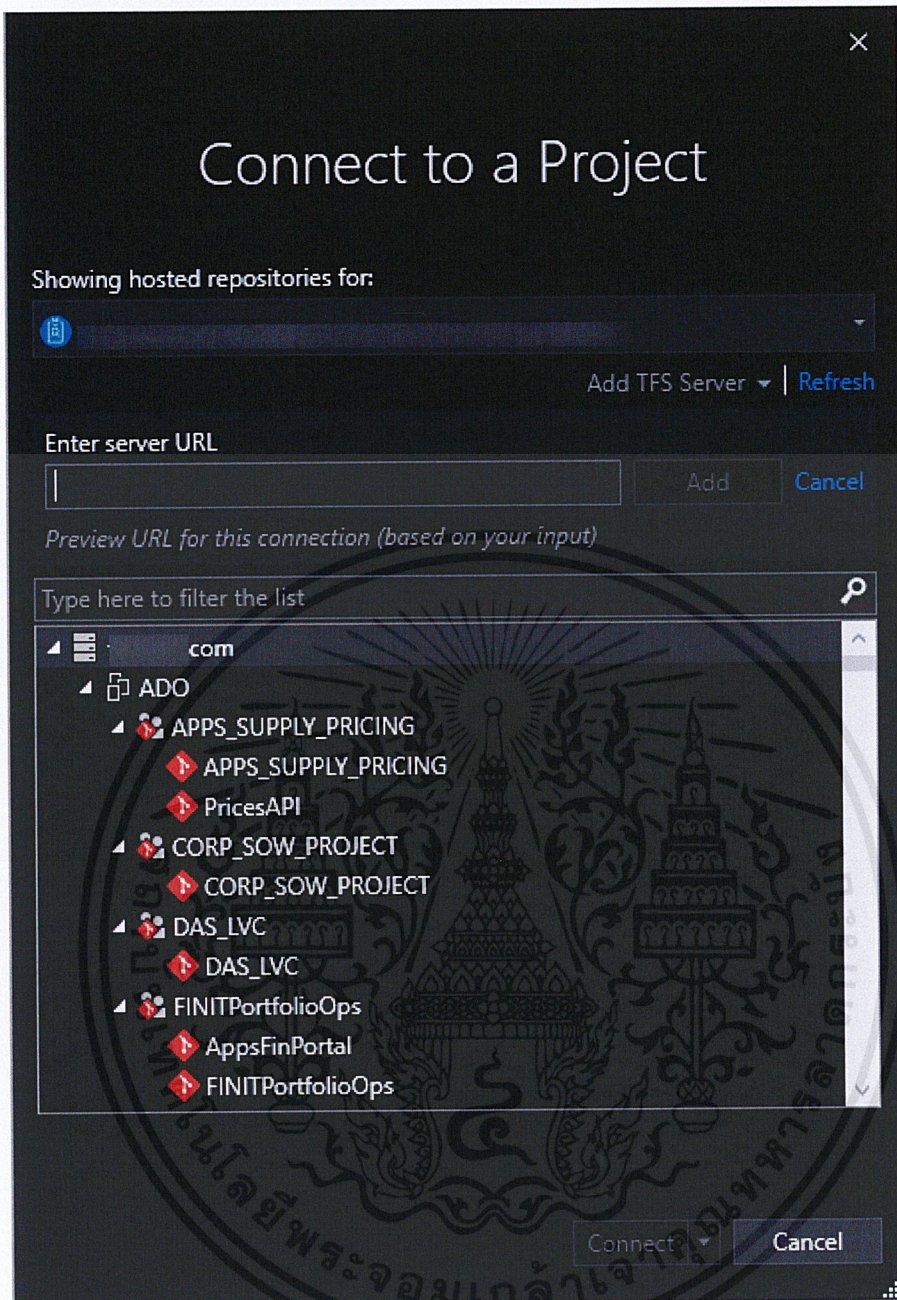
ในการพัฒนาครั้งนี้ ผู้พัฒนาได้ใช้บริการของ Team Foundation Server เพื่อทำ Version Control โดยได้ทำการสร้าง branch production เพื่อเก็บ code ส่วนที่พร้อมจะนำไปใช้งาน และสร้าง เพื่อเก็บ code หลักโดยหากต้องการพัฒนาฟังก์ชันใหม่ จะนำไปใส่ใน branch นี้ สาเหตุที่ผู้พัฒนาต้องสร้าง branch แยกกันเพื่อให้สามารถทำงานที่แตกต่างกันไปได้พร้อม ๆ กันโดยที่ code ไม่ทับซ้อนกัน เมื่อเสร็จสิ้นการพัฒนาฟังก์ชันการทำงานนั้น ๆ แล้ว จึงจะทำการรวม branch development สู่ branch production เมื่อพร้อมที่จะนำไปใช้จริงต่อไป สำหรับขั้นตอนการเชื่อมต่อ Visual Studio กับ Team Foundation Server ทำได้ดังนี้

1. เปิดโปรแกรม Microsoft Visual Studio
2. ไปที่ View > Team Explorer จะปรากฏหน้าต่าง ดังรูปที่ 3.8



รูปที่ 3.8 หน้าต่าง Team Explorer

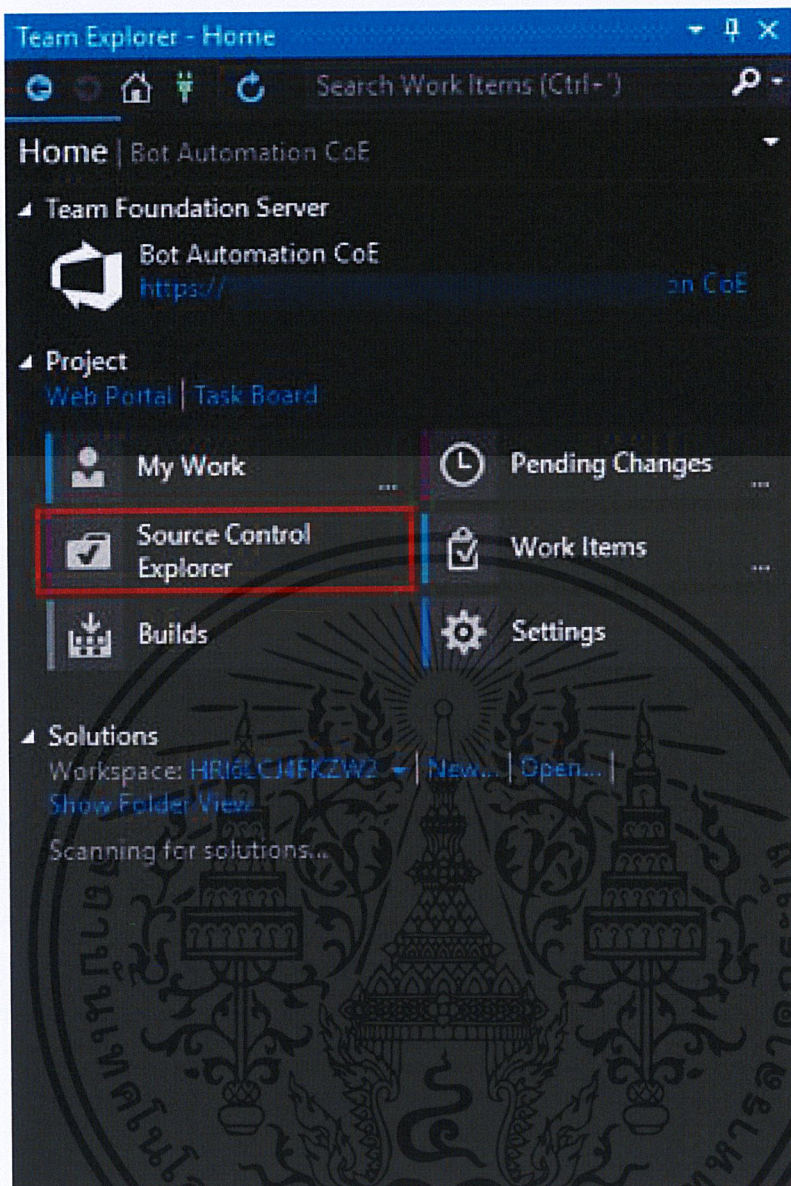
3. หน้าต่าง Team Explorer กด Manage Connection > Connect to Team Project แล้วกดที่ Server..
4. จะปรากฏหน้าต่าง ดังรูปที่ 3.9 กรอก URL ของ Team Foundation Server ที่ต้องการ และกด OK



รูปที่ 3.9 หน้าต่าง Add Team Foundation Server

5. เลือก Project ที่ต้องการคือ APPS1 > Bot\_Automation\_CoE และกด Connect
6. หลังจากเสร็จสิ้นการสร้างการเชื่อมต่อ ไปที่ Source Control Explorer ดังรูปที่ 3.10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ 23 วิชาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.10 ปุ่ม Source Control Explorer ภายในหน้าต่าง Team Explorer

7. ไปที่ Bot\_Automation\_CoE คลิกขวาที่ mtpportal-dev > Get Latest Version
8. จะได้ไฟล์สำหรับพัฒนาทั้งหมดของโปรแกรม

### 3.1.5.2 การพัฒนา Select assets page

ในการพัฒนา Select assets page ผู้พัฒนาได้ทำการการวางโครงสร้างของการเขียนโปรแกรม ดังรูปที่ 3.11

```

  ✓ asset
    > add-value-per-robot
  ✓ asset-list.component
    # asset-list.component.css
    <> asset-list.component.html
    TS asset-list.component.spec.ts
    TS asset-list.component.ts
  ✓ asset-selected-table
    # asset-selected-table.component.css
    <> asset-selected-table.component.html
    TS asset-selected-table.component.spec.ts
    TS asset-selected-table.component.ts
  ✓ asset-table.component
    # asset-table.component.css
    <> asset-table.component.html
    TS asset-table.component.spec.ts
    TS asset-table.component.ts
    TS assets-filter.pipe.ts
    TS assetsProd-filter.pipe.ts

```

รูปที่ 3.11 การวางโครงสร้างหน้า Select assets page

หลังจากวางโครงสร้างเป็นที่เรียบร้อย ผู้พัฒนาได้สร้าง User interface โดยในหน้า Select assets page มีส่วนประกอบหลัก ๆ 3 ส่วนคือ Assets list (Main component), Assets selection table และ Assets selected table ดังรูปที่ 3.12 และ 3.13

## Assets

SELECT ALL	ASSET_NAME	VALUE ACC	VALUE PROD	TYPE
<input checked="" type="checkbox"/>	RACE_MTPPortalExportOil_AccountantName	Grace W.	Grace W.	Text
<input checked="" type="checkbox"/>	RACE_MTPPortalExportOil_ConfigAccount	[Value per robot]		Text
	Patom_Bot : Patom S.			
	Varis_Test : Varis K.			
<input checked="" type="checkbox"/>	RACE_MTPPortalExportOil_InboxAccount	username: woranitha@exxonmo...	username: woranitha@exxonmo...	Credential

รูปที่ 3.12 Assets selection table

## Selected Assets

ASSET NAME	VALUE	TYPE	ENVIRONMENT
RACE_MTPPortalExportOil_Acc...	Grace W.	Text	Production
RACE_MTPPortalExportOil_Inbo...	Username: woranitha@exxonmobil.com	Credential	Production

รูปที่ 3.13 Assets selected table

API ที่ใช้สำหรับหน้า Assets page มีทั้งหมด 3 API ได้แก่

- Get license (UiPath Orchestrator API)  
เรียกใช้งานผ่าน URL {UiPathURL}/odata/Settings/UiPath.Server.

Configuration.Odata.GetLicense() โดยใช้ HTTP Method GET โดย API นี้เป็นการทำ Authentication เพื่อใช้งาน UiPath orchestrator โดย API นี้มีรูปแบบของ Response ดังรูปที่ 3.14

```
GET /odata/Settings/UiPath.Server.Configuration.OData.GetLicense() Retrieves the current license information.

Implementation Notes
Requires authentication.

Response Class (Status 200)
Model Example Value

LicenseDto {
  HostLicenseId (integer, optional),
  Id (integer, optional): License Id,
  ExpireDate (integer, optional): License expiration date in Epoch format,
  Allowed (LicenseFields, optional): Contains the number of allowed licenses for each type,
  Used (LicenseFields, optional): Contains the number of used licenses for each type,
  AttendedConcurrent (boolean, optional): States whether the license is Attended Concurrent,
  DevelopmentConcurrent (boolean, optional): States whether the license is Development Concurrent,
  IsRegistered (boolean, optional): True if the current tenant is registered with a license. False otherwise.,
  IsExpired (boolean, optional): States whether the license is still valid or not.,
  CreationTime (string, optional): The date when the license was uploaded.,
  Code (string, optional): The license code.
}

LicenseFields {
  Unattended (integer, optional),
  Attended (integer, optional),
  NonProduction (integer, optional),
  Development (integer, optional)
}
```

รูปที่ 3.14 รูปแบบ Response ของ Get license

- Get user's organization unit (UiPath Orchestrator API)  
เรียกใช้งานผ่าน URL {UiPathURL}/odata/Users/UiPath.Server.

Configuration.Odata.GetCurrentUser?\$expand=OrganizationUnits() โดยใช้ HTTP Method GET โดยที่ API จะส่งข้อมูล array object ซึ่งเป็นข้อมูลของผู้ใช้งาน จากนั้นผู้พัฒนาจะนำข้อมูลของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ 26 รัชกาลที่ 10 ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Organization unit ทั้งหมดที่ผู้ใช้งานนั้น ๆ สามารถเข้าถึงได้ไปใช้งานต่อ โดย API นี้มีรูปแบบของ Response ดังรูปที่ 3.15

```

GET /odata/Users/UiPath.Server.Configuration.OData.GetCurrentUser() Returns details about the user currently logged into Orchestrator.

Response Class (Status 200)
Model Example Value

UserDto {
  Name (string, optional): The name of the person for which the user is created.stringMax. Length:32.
  Surname (string, optional): The surname of the person for which the user is created.stringMax. Length:32.
  UserName (string, optional): The name used to login to Orchestrator..
  Domain (string, optional): The domain from which the user is imported,
  FullName (string, optional): The full name of the person constructed with the format Name Surname..
  EmailAddress (string, optional): The e-mail address associated with the user.stringMax. Length:256.
  IsEmailConfirmed (boolean, optional): States if the email address is valid or not..
  LastLoginTime (string, optional): The date and time when the user last logged in, or null if the user never logged in..
  IsActive (boolean, optional): States if the user is active or not. An inactive user cannot login to Orchestrator..
  CreationTime (string, optional): The date and time when the user was created..
  AuthenticationSource (string, optional): The source which authenticated this user..
  Password (string, optional): The password used during application login..
  UserRoles (Array[UserRoleDto], optional): The collection of roles associated with the user..
  RolesList (Array[string], optional): The collection of role names associated with the user..
  LoginProviders (Array[string], optional): The collection of entities that can authenticate the user..
  OrganizationUnits (Array[OrganizationUnitDto], optional): The collection of organization units associated with the user..
  TenantId (integer, optional): The id of the tenant owning the user..
  TenancyName (string, optional): The name of the tenant owning the user..
  Type (string, optional): The user type. = [User, Robot]stringEnum."User", "Robot".
  NotificationSubscription (UserNotificationSubscription, optional): User can choose which notifications does he want to receive.
  Id (integer, optional)
}UserRoleDto {
  UserId (integer, optional): The Id of the associated user..
  RoleId (integer, optional): The Id of the associated role..
  UserName (string, optional): The name of the associated user.
  RoleName (string, optional): The name of the associated role.
  Id (integer, optional)
}OrganizationUnitDto {
  DisplayName (string): The name of the organization unit..
  Id (integer, optional)
}UserNotificationSubscription {
  Queues (boolean, optional),
  Robots (boolean, optional),
  Jobs (boolean, optional),
  Schedules (boolean, optional)
}
  
```

รูปที่ 3.15 รูปแบบ Response ของ Get current user

- Get Assets (UiPath Orchestrator API)

เรียกใช้งานผ่าน URL {UiPathURL}/odata/Assets โดยใช้ HTTP Method GET โดยที่ผู้เรียกใช้จะต้องระบุ Organization ID และ API จะส่งข้อมูล array object ซึ่งเป็น Assets List ที่มีใน organization unit ที่ผู้ใช้งานได้เลือกไว้ก่อนหน้ากลับมา โดย API นี้มีรูปแบบของ Response ดังรูปที่ 3.16

## Implementation Notes

Required permissions: Assets.View.

## Response Class (Status 200)

Model Example Value

```
ODataResponse[List[AssetDto]] {
  @odata.context (string, optional)
  value (Array[AssetDto], optional)
}AssetDto {
  Name (string): A custom name for the asset.
  CanBeDeleted (boolean, optional): States if an assets can be deleted. The default value of this property is true.
  ValueScope (string): Defines the scope of the asset. = ['Global', 'PerRobot']stringEnum:"Global", "PerRobot",
  ValueType (string, optional): Defines the type of value stored by the asset. = ['DBConnectionString', 'HttpConnectionString', 'Text', 'Bool', 'Integer', 'Credential',
  'WindowsCredential', 'KeyValueList']stringEnum:"DBConnectionString", "HttpConnectionString", "Text", "Bool", "Integer", "Credential", "WindowsCredential",
  "KeyValueList",
  Value (string, optional): The textual representation of the asset value, irrespective of value type.
  StringValue (string, optional): The value of the asset when the value type is Text. Empty when the value type is not Text.
  BoolValue (boolean, optional): The value of the asset when the value type is Bool. False when the value type is not Bool.
  IntValue (integer, optional): The value of the asset when the value type is Integer. 0 when the value type is not Integer.
  CredentialUsername (string, optional): The user name when the value type is Credential. Empty when the value type is not Credential.
  CredentialPassword (string, optional): The password when the value type is Credential. Empty when the value type is not Credential.
  KeyValueList (Array[CustomKeyValuePair], optional): A collection of key value pairs when the type is KeyValueList. Empty when the value type is not
  KeyValueList.
  RobotValues (Array[AssetRobotValueDto], optional): The collection of asset values per robot. Empty if the asset type is Global.
  Id (integer, optional)
}CustomKeyValuePair {
  Key (string, optional): A piece of text representing the key.
  Value (string, optional): A piece of text representing the value.
}AssetRobotValueDto {
  RobotId (integer, optional): The Id of the robot with which the asset is associated.
  RobotName (string, optional): The name of the robot with which the asset is associated.
  KeyTrail (string, optional): Masked value of the robot key.
  ValueType (string, optional): Defines the type of value stored by the asset. = ['DBConnectionString', 'HttpConnectionString', 'Text', 'Bool', 'Integer', 'Credential',
  'WindowsCredential', 'KeyValueList']stringEnum:"DBConnectionString", "HttpConnectionString", "Text", "Bool", "Integer", "Credential", "WindowsCredential",
  "KeyValueList",
  StringValue (string, optional): The value of the asset when the value type is Text. Empty when the value type is not Text.
  BoolValue (boolean, optional): The value of the asset when the value type is Bool. False when the value type is not Bool.
  IntValue (integer, optional): The value of the asset when the value type is Integer. 0 when the value type is not Integer.
  Value (string, optional): The textual representation of the asset value, irrespective of value type.
  CredentialUsername (string, optional): The user name when the value type is Credential. Empty when the value type is not Credential.
  CredentialPassword (string, optional): The password when the value type is Credential. Empty when the value type is not Credential.
  KeyValueList (Array[CustomKeyValuePair], optional): A collection of key value pairs when the type is KeyValueList. Empty when the value type is not
  KeyValueList.
  Id (integer, optional)
}
```

## รูปที่ 3.16 รูปแบบ Response ของ Get Assets

เมื่อจัดเตรียม API ที่จำเป็นต่อการใช้งานในหน้า Assets page เสร็จสมบูรณ์แล้ว ผู้พัฒนาได้เริ่มพัฒนาโดยสร้างเป็น method หลัก ๆ ดังนี้

- Method getOrgUnit() ดังรูปที่ 3.17 เริ่มต้นโดยการเรียกใช้ Get license API ทั้งใน acceptance (ACC) และ production (PROD) จากนั้นระบบทำการเรียกใช้ Get current user's organization unit API ทั้งใน ACC และ PROD ซึ่งได้รับการตอบกลับเป็น array object ของ organization unit ที่ผู้ใช้งานมีสิทธิ์เข้าถึงได้ เมื่อผู้ใช้งานทำการเลือก ACC organization unit ที่ต้องการเรียบร้อยแล้ว ระบบจะทำการ mapping Organization unit ดังกล่าวกับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ 28 ศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PROD organization unit และเก็บ organization unit ID ของทั้งสอง environment

```
async getOrgUnit(){
  await this.apiService.getCurrentUserUI().toPromise();
  let orgUnits = await this.apiService.getOrgUnitById().toPromise();
  await this.apiService.getCurrentUserUIProd().toPromise();
  let orgUnitsProd = await this.apiService.getOrgUnitProdById().toPromise();
  let orgMapListAccId:Array<object> = [];
  let orgMapListProdId:Array<object> = [];
  orgUnits['OrganizationUnits'].forEach(orgUnit => {
    orgUnitsProd['OrganizationUnits'].forEach(orgUnitProd => {
      if(orgUnitProd['DisplayName']===orgUnit['DisplayName']){
        orgMapListAccId.push(orgUnit)
        orgMapListProdId.push(orgUnitProd)
      }
    });
  });
  return {Acc:orgMapListAccId,Prod:orgMapListProdId}
}
```

รูปที่ 3.17 Method getOrgUnit()

- Method getOrgUnitSelectedProd() ดังรูปที่ 3.18 ทำการวน for loop เพื่อเปรียบเทียบ organization unit ที่ผู้ใช้งานเลือกกับ PROD organization unit list หากมี DisplayName ตรงกัน ระบบจะทำการเก็บค่า organization unit ID และ array object ของ organization unit เพื่อเป็น Endpoint organization unit

```
async getOrgUnitSelectedProd(){
  for(let orgUnitProd of this.orgUnits.Prod){
    if(this.orgUnitSelected['DisplayName']===orgUnitProd['Displayname']){
      this.OrgIdProd = orgUnitProd['Id'];
      this.orgUnitSelectedProd = orgUnitProd;
      return this.orgUnitSelectedProd
    }
  }
}
```

รูปที่ 3.18 Method getOrgUnitSelectedProd()

- Method getAssetFromAcc() & getAssetFromProd() ดังรูปที่ 3.19 มีการเรียกใช้งาน Get Assets API ทั้งของ ACC และ PROD และได้รับการตอบกลับเป็น array object ของ Assets ทั้งหมดที่มีใน ACC และ PROD จากนั้นทำการวน Loop เพื่อเพิ่ม Flag ในการระบุ environment ให้แก่ Assets แต่ละตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ 29 รัชศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

async getAssetFromAcc(orgId){
    let responseAllAccAssets = await this.apiService.getAssets(orgId).toPromise();
    let allAccAssets:Array<object> = responseAllAccAssets['value'];
    for(var i = 0 ; i < allAccAssets.length ; i++){
        allAccAssets[i]['environment'] = 'Acceptance';
    }
    return allAccAssets as Array<IAsets>
}

async getAssetFromProd(OrgIdProd){
    let responseAllProdAssets = await this.apiService.getAssetsProd(OrgIdProd).toPromise();
    let allProdAssets:Array<object> = responseAllProdAssets['value'];
    for(var i = 0 ; i < allProdAssets.length ; i++){
        allProdAssets[i]['environment'] = 'Production';
    }
    return allProdAssets as Array<IAsets>
}

```

รูปที่ 3.19 Method getAssetFromAcc() & getAssetFromProd()

- Method updateAssets() ดังรูปที่ 3.20 มีการทำงานคือหากผู้ใช้ทำการเปลี่ยน Organization unit ที่แตกต่างจากเดิม ระบบจะทำการเก็บค่า Organization unit ใหม่แทน และเรียกใช้ method ที่ได้กล่าวถึงก่อนหน้านี้คือ getOrgUnitSelectedProd(), getAssetFromAcc() และ getAssetFromProd() ในการนำ Assets ทั้งหมดที่อยู่ใน Organization unit ใหม่มาแสดงใน Assets selection table

```

public async updateAssets(orgId){
    this.uiService.setLoading(true);
    this.assetSelected = [];
    this.isCanNext = false;
    this.OrgId = orgId;
    await this.getOrgUnitSelectedProd();
    this.assetFromAcc = await this.getAssetFromAcc(orgId);
    this.assetFromProd = await this.getAssetFromProd(this.OrgIdProd);
    this.messageService.changeAssetFromProd(this.assetFromProd);
    this.assets2 = this.assetFromAcc.concat(this.assetFromProd);
    this.assetGroupByName = this.assets2.reduce((assets, asset) =>{
        assets[asset.Name] = assets[asset.Name] + 1 || 1;
    }, {});
    return assets;
}, {});
this.assets2 = this.assignIsExpanded(this.assets2);
this.orgUnitSelectedOld = this.orgUnitSelected;
this.newList();
this.uiService.setLoading(false);
this.isAssetsLoaded = true;
}

```

รูปที่ 3.20 Method updateAssets()

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ 30 รัชการศึกษานี้ ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Method `getSelectedAsset()` ดังรูปที่ 3.21 ทำการรับค่า `isChecked` และ array object ของ Assets โดย `isChecked` เป็นตัวแปรในการตรวจสอบว่า Asset ตัวนั้นถูกเลือกแล้วหรือไม่ ก่อนที่จะเก็บ Asset ตัวดังกล่าวลงในตัวแปร `assetSelected` หากยังไม่ได้เลือก ระบบจะทำการตรวจสอบทั้งหมด 2 กรณี คือถ้าหาก Asset มีค่าทั้งใน ACC และ PROD ค่าของ Asset ที่ทำการเก็บในตัวแปรจะมีค่าเหมือน PROD asset's value แต่ถ้าหาก Asset มีค่าแค่เพียงใน ACC ค่าของ Asset ที่ทำการเก็บในตัวแปรจะมีค่าเหมือน ACC asset's value

```

getSelectedAsset(isChecked: boolean ,assetList:Object){
  if(isChecked){
    if(!this.assetSelected.some(item => item['Id'] == assetList['Id'])){
      let tempAssetSelected = JSON.parse(JSON.stringify(assetList));
      if(this.assetGroupByName[tempAssetSelected['Name']]==2){
        for(let prodValue of this.pipe.transform(this.assetFromProd,tempAssetSelected['Name'],this.orgUnitSelected)){
          if(prodValue['Name'] == tempAssetSelected['Name']){
            tempAssetSelected['environment'] = JSON.parse(JSON.stringify(prodValue['environment']));
            tempAssetSelected['CredentialUsername'] = JSON.parse(JSON.stringify(prodValue['CredentialUsername']));
            if(tempAssetSelected['ValueType']!=prodValue['ValueType']){
              tempAssetSelected['ValueType']=JSON.parse(JSON.stringify(prodValue['ValueType']));
            }
            if(tempAssetSelected['ValueScope']=="PerRobot"||prodValue['ValueScope']=="PerRobot"){
              tempAssetSelected['ValueScope']="PerRobot";
              tempAssetSelected['Value'] = "[Value per robot]"
              tempAssetSelected["RobotValues"]=JSON.parse(JSON.stringify(prodValue["RobotValues"]));
            }
            else{
              tempAssetSelected['Value'] = JSON.parse(JSON.stringify(prodValue['Value']));
            }
          }
        }
      }
      this.assetSelected.push(tempAssetSelected);
    }
  }
  else{
    this.assetSelected = this.assetSelected.filter(item => item['Id'] != assetList['Id']);
  }
  if(this.assetSelected.length){
    for(let eachAssetSelected of this.assetSelected){
      if(eachAssetSelected.ValueScope=="PerRobot"){
        this.isCanNext = false;
        break
      }
      else{
        this.isCanNext = true;
      }
    }
  }
  else{
    this.isCanNext = false;
  }
  JSON.stringify(this.assetSelected)
}

```

รูปที่ 3.21 Method `getSelectedAsset()`

- Method `selectAll()` ดังรูปที่ 3.22 ทำการรับค่า `isSelectAll` และรับค่า Assets ทุกตัว ถ้าหากเลือก Select all แล้ว ระบบจะทำการตรวจสอบหน้าที่ผู้ใช้งานเลือกใน ณ ขณะนั้น และรวมถึงจำนวน items ต่อหน้าที่ผู้ใช้งานเลือก แล้วจึงเรียกใช้งาน method `getSelectedAsset` เพื่อเก็บ Assets ทุกตัวในหน้านั้นเก็บในตัวแปร `assetSelected`

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ 31 ศึกษเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

selectAll(isSelectAll:boolean,asset) {
  let items = this.paginationService.getInstance('non-select');
  if(isSelectAll){
    for (var i = (this.currentPage-1)*this.itemsPerPage; i < this.absoluteIndex(this.itemsPerPage,this.currentPage); i++) {
      if(asset[i].isChecked = false){
        this.getSelectedAsset(asset[i].isChecked = !asset[i].isChecked, asset[i])
      } else{
        this.getSelectedAsset(asset[i].isChecked = !asset[i].isChecked, asset[i])
      }
    }
  }
  else{
    for (var i = (this.currentPage-1)*this.itemsPerPage; i < this.absoluteIndex(this.itemsPerPage,this.currentPage); i++) {
      if(asset[i].isChecked = true){
        this.getSelectedAsset(asset[i].isChecked = !asset[i].isChecked, asset[i])
      }
    }
  }
}
}

```

รูปที่ 3.22 Method selectAll()

- Method transform() ใน Assets searching filter ดังรูปที่ 3.22 จะทำการรับค่า Assets ทุกตัว, คำที่ผู้ใช้งานค้นหาใน search box และ Organization unit จากนั้นระบบจะทำการตรวจสอบเป็นกรณีที่แตกต่างกัน ตัวอย่างเช่น หากระบบมีการรับค่าของคำค้นหาเป็น string เปล่าและ organization unit เป็น “undefined” method จะ return ค่า Assets ทุกตัวกลับคืน หรือหากระบบมีการรับค่าของคำที่ค้นหา และ organization unit ที่มีค่า method นี้จะทำการแปลงคำค้นหาให้เป็นอักษรพิมพ์เล็ก และกรองเพียง Assets ที่มี Asset name, Asset value หรือ Asset type ที่บรรจุคำดังกล่าว จากนั้นระบบจึง return ค่าเป็น Assets ที่กรองเรียบร้อยแล้วแสดงใน Selection table เป็นต้น

```

transform(items: Array<Assets>, searchText: string, orgUnitSelected: any): any {
  items = items.reduce((acc, val) => {
    if (!acc.find(e => e.Name === val.Name)) {
      acc.push(val);
    }
    return acc;
  }, []);
  let newItem = JSON.parse(JSON.stringify(items));
  if(!items) {
    return [];
  }
  else if (searchText === undefined && orgUnitSelected === undefined) {
    return items;
  }
  else if (searchText && orgUnitSelected === undefined) {
    searchText = searchText.toLowerCase();
    return items.filter(it => {
      return it['Name'].toLowerCase().includes(searchText) || it['Value'].toLowerCase().includes(searchText) || it['ValueType'].toLowerCase().includes(searchText);
    });
  }
  else if (searchText === undefined && orgUnitSelected) {
    this.orgId = orgUnitSelected['id'];
    return items;
  }
  else if (searchText === "" && orgUnitSelected === undefined) {
    return items;
  }
  else if (searchText && orgUnitSelected) {
    searchText = searchText.toLowerCase();
    orgUnitSelected = orgUnitSelected['DisplayName'].toLowerCase();
    return items.filter(it => {
      return it['Name'].toLowerCase().includes(searchText) || it['Value'].toLowerCase().includes(searchText) || it['ValueType'].toLowerCase().includes(searchText);
    });
  }
  else if (searchText === "" && orgUnitSelected) {
    searchText = searchText.toLowerCase();
    orgUnitSelected = orgUnitSelected['DisplayName'].toLowerCase();
    return items.filter(it => {
      return it['Name'].toLowerCase().includes(searchText) || it['Value'].toLowerCase().includes(searchText) || it['ValueType'].toLowerCase().includes(searchText);
    });
  }
}

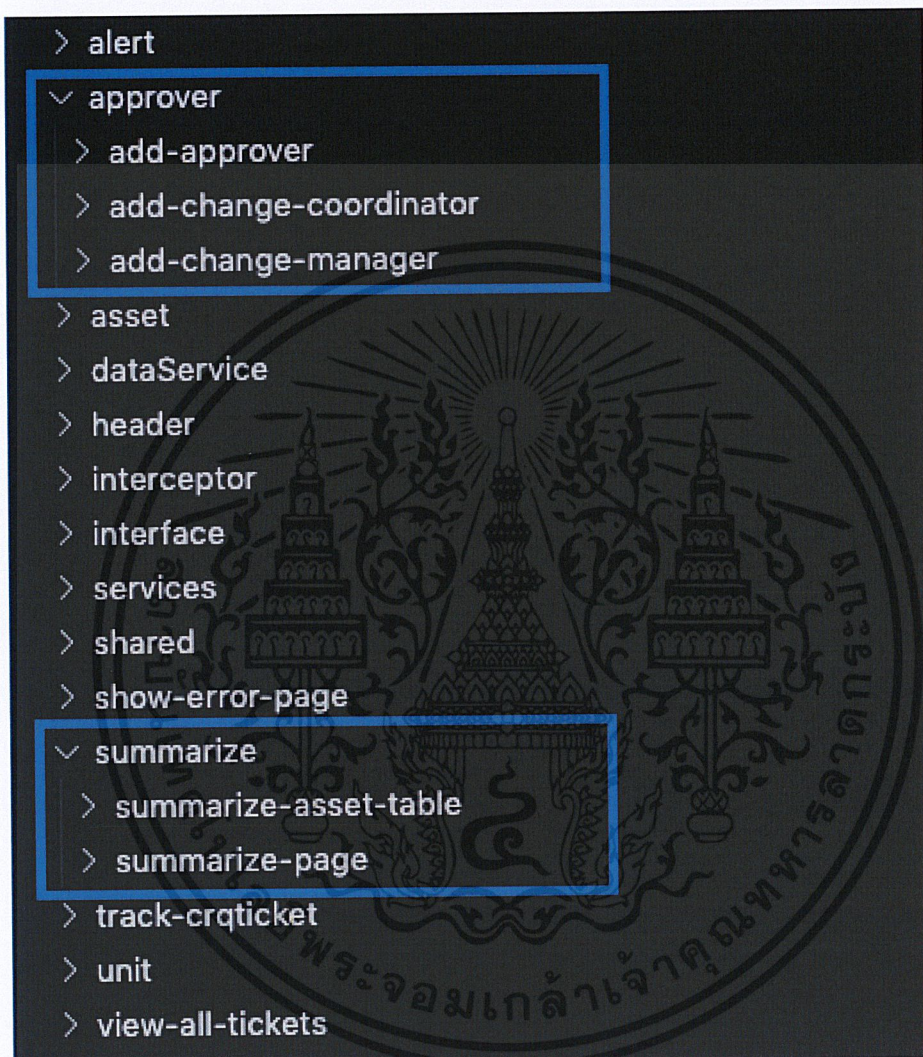
```

รูปที่ 3.23 Assets searching filter

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ 32 รัชศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.1.5.3 การพัฒนา Summarize page

ในการพัฒนา Summarize page ผู้พัฒนาได้ทำการวางโครงสร้างของการเขียนโปรแกรม ดังรูปที่ 3.23



รูปที่ 3.24 การวางโครงสร้างหน้า Summarize page

หลังจากวางโครงสร้างเป็นที่เรียบร้อยแล้ว ผู้พัฒนาได้สร้าง User interface โดยในหน้า Summarize page มีส่วนประกอบหลัก ๆ 5 ส่วนคือ Summarize page (Main component), Summarize table, Add approver, Add change coordinator และ Add change manager ดังรูปที่ 3.24

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ 33 ศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# MTP Portal

New request View requests

## Summary

### Assets

ASSET NAME	VALUE	TYPE	ENVIRONMENT
RACE_MTPPortalExportOil_Acc...	Phison K.	Text	Production
RACE_MTPPortalExportOil_Conf...	[Value per robot]	Text	Acceptance
<a href="#">Show ▶</a>			
RACE_MTPPortalExportOil_Inbo...	username: phison@exxonmobil.c...	Credential	Production
RACE_MTPPortalExportOil_infol...	False	Bool	Production
RACE_MTPPortalExportOil_Name	John R.	Text	Production

Items per page:

5 ▼

Previous 1 2 Next ▶

### Request Summary

RACE- AssetChange

### Approver



Approver 1 :

Select a person

### Change coordinator

Select a group

Select a person

### Change manager

Select a group

Select a person

Submit

Back

รูปที่ 3.25 Summarize page

API ที่มีเพิ่มเติมในหน้า Summarize page มีทั้งหมด 9 API ได้แก่

- Get user look-up (ITSM API)

เรียกใช้งานผ่าน URL {ITSM URL}/user/all?query=\${search Approver} โดยใช้ HTTP Method GET โดยที่ผู้เรียกใช้จะต้องระบุค่าที่จะค้นหา และ API จะส่งข้อมูลที่เป็น array object ของชื่อพนักงานในองค์กรที่บรรจุค่าค้นหา นั้น ๆ กลับมา โดย API นี้มีรูปแบบของ Response ดังรูปที่ 3.25

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ 34 ศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

[
  {
    "ID": "string",
    "Person_ID": "string",
    "Corporate_ID": "string",
    "Name": "string",
    "First_Name": "string",
    "Last_Name": "string",
    "Middle_Initial": "string",
    "Email": "string",
    "Phone": "string",
    "Region": "string",
    "Country": "string",
    "Site": "string",
    "Desk": "string",
    "City": "string",
    "Company": "string",
    "Organization": "string",
    "Department": "string",
    "Job_Title": "string",
    "Sensitivity": "string",
    "Litigation_Hold": true,
    "VIP": true
  }
]

```

รูปที่ 3.26 รูปแบบ Response ของ Get user look-up

- Get support group (ITSM API)

เรียกใช้งานผ่าน URL {ITSM URL}/group/all/{searchSupport Group} โดยใช้ HTTP Method GET โดยที่ผู้เรียกใช้จะต้องระบุค่าที่จะค้นหา และ API จะส่งข้อมูลที่เป็น array object ของ Support group ทั้งหมดในองค์กรที่บรรจุค่าคั่นหา นั้น ๆ กลับมา โดย API นี้มีรูปแบบของ Response ดังรูปที่ 3.26

- Get members in support group (ITSM API)

เรียกใช้งานผ่าน URL {ITSM URL}/group/{searchMemberIn SupportGroup}/members โดยใช้ HTTP Method GET โดยที่ผู้เรียกใช้จะต้องระบุ Support group และ API จะส่งข้อมูลที่เป็น array object ของสมาชิกทั้งหมดใน Support group นั้น ๆ กลับมา โดย API นี้มีรูปแบบของ Response ดังรูปที่ 3.27

- Add approvers (ITSM API)

เรียกใช้งานผ่าน URL {ITSM URL}/change/{crqlid}/approver โดยใช้ HTTP Method POST ในการเพิ่มผู้อนุมัติลงใน CRQ ticket โดยที่ผู้เรียกใช้จะต้องระบุ CRQ ticket's ID ของ ticket ที่ต้องการทำการ Add approvers และมีรูปแบบ Body ดังรูปที่ 3.28

```
[
  "Change_ID": "string",
  "Impact": "string",
  "Urgency": "string",
  "Priority": "string",
  "Has_Attachments": true,
  "Status": "string",
  "Change_Reason": "string",
  "SLM": "string",
  "SLM_Hold": true,
  "Risk_Level": "string",
  "Stage": "string",
  "Class": "string",
  "Summary": "string",
  "Completed_Date": "2019-12-30T09:59:24.931Z",
  "Closed_Date": "2019-12-30T09:59:24.931Z",
  "Scheduled_Start_Date": "2019-12-30T09:59:24.931Z",
  "Scheduled_End_Date": "2019-12-30T09:59:24.931Z",
  "Change_Coordinator": {
    "ID": "string",
    "Name": "string",
    "Group_ID": "string",
    "Group": "string",
    "Group_Organization": "string",
    "Group_Company": "string",
    "Intent": {
      "ID": "string",
      "Person_ID": "string",
      "Corporate_ID": "string",
      "Name": "string",
      "First_Name": "string",
      "Last_Name": "string",
      "Middle_Initial": "string",
      "Email": "string",
      "Phone": "string",
      "Region": "string",
      "Country": "string",
      "Site": "string",
      "Desk": "string",
      "City": "string",
      "Company": "string",
      "Organization": "string",
      "Department": "string",
      "Job_title": "string",
      "Sensitivity": "string",
      "Litigation_Hold": true,
      "VIP": true
    }
  },
  "Change_Manager": {
    "ID": "string",
    "Name": "string",
    "Group_ID": "string",
    "Group": "string",
    "Group_Organization": "string",
    "Group_Company": "string"
  },
  "Product": {
    "ID": "string",
    "Name": "string",
    "Model_Version": "string",
    "Manufacturer": "string",
    "Tier_1": "string",
    "Tier_2": "string",
    "Tier_3": "string"
  },
  "Service_Level": {
    "ID": "string",
    "Title": "string",
    "Category": "string",
    "Progress": "string",
    "Goal": "string",
    "Due": "2019-12-30T09:59:24.931Z",
    "Due_In": "string"
  },
  "Submit_Date": "2019-12-30T09:59:24.931Z",
  "Submitted_By": {
    "ID": "string",
    "Person_ID": "string",
    "Corporate_ID": "string",
    "Name": "string",
    "First_Name": "string",
    "Last_Name": "string",
    "Middle_Initial": "string",
    "Email": "string",
    "Phone": "string",
    "Region": "string",
    "Country": "string",
    "Site": "string",
    "Desk": "string",
    "City": "string",
    "Company": "string",
    "Organization": "string",
    "Department": "string",
    "Job_title": "string",
    "Sensitivity": "string",
    "Litigation_Hold": true,
    "VIP": true
  },
  "Last_Modified_Date": "2019-12-30T09:59:24.931Z",
  "Last_Modified_By": {
    "ID": "string",
    "Person_ID": "string",
    "Corporate_ID": "string",
    "Name": "string",
    "First_Name": "string",
    "Last_Name": "string",
    "Middle_Initial": "string",
    "Email": "string",
    "Phone": "string",
    "Region": "string",
    "Country": "string",
    "Site": "string",
    "Desk": "string",
    "City": "string",
    "Company": "string",
    "Organization": "string",
    "Department": "string",
    "Job_title": "string",
    "Sensitivity": "string",
    "Litigation_Hold": true,
    "VIP": true
  }
}
]
```

รูปที่ 3.27 รูปแบบการตอบกลับของ Get support group

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ 36 ศึกษเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

{
  {
    "ID": "string",
    "Name": "string",
    "Email": "string",
    "Phone": "string",
    "Region": "string",
    "Country": "string",
    "Site": "string",
    "Available": true,
    "Role_Available": {
      "Incident": {
        "Assignee": true
      },
      "Problem": {
        "Assignee": true,
        "Coordinator": true
      },
      "Change": {
        "Approver": true,
        "Coordinator": true,
        "Manager": true
      },
      "Work_Order": {
        "Assignee": true,
        "Manager": true
      },
      "Task": {
        "Assignee": true
      },
      "Asset": {
        "Asset_Admin": true,
        "Asset_User": true
      },
      "Support_Group": {
        "Global_Admin": true,
        "Support_Admin": true
      }
    },
    "Roles": {
      "Work_Order": {
        "Assignee": true,
        "Manager": true
      },
      "Change": {
        "Approver": true,
        "Coordinator": true,
        "Manager": true
      },
      "Problem": {
        "Coordinator": true
      },
      "Support_Group": {
        "Support_Admin": true
      }
    }
  }
}

```

รูปที่ 3.28 รูปแบบ Response ของ Get members in support group

```

{
  "Approver": "string"
}

```

รูปที่ 3.29 รูปแบบ Body ของ Get Assets

- Post CRQ ticket (ITSM API)

เรียกใช้งานผ่าน URL {ITSM URL}/change โดยใช้ HTTP Method POST ในการสร้าง CRQ ticket โดยมีรูปแบบ Body ดังรูปที่ 3.29

- Put CRQ ticket (ITSM API)

เรียกใช้งานผ่าน URL {ITSM URL}/change/{crqlid}/next โดยใช้ HTTP Method PUT ในการอัปเดต CRQ ticket โดยที่ผู้เรียกใช้จะต้องระบุ CRQ ticket's ID ของ ticket ที่ต้องการอัปเดต โดยมีรูปแบบ Body ดังรูปที่ 3.29

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ 37 วิชาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
  "Change_Coordinator_Group": "string",
  "Change_Coordinator": "string",
  "Change_Manager_Group": "string",
  "Change_Manager": "string",
  "Summary": "string",
  "Notes": "string",
  "Service": "string",
  "Change_Type": "string",
  "Change_Reason": "string",
  "Class": "string",
  "Timing_Reason": "string",
  "Impact": "string",
  "Urgency": "string",
  "Risk_Level": "string",
  "Status": "string",
  "Status_Reason": "string",
  "Scheduled_Start_Date": "2019-12-30T03:53:24.397Z",
  "Scheduled_End_Date": "2019-12-30T03:53:24.397Z",
  "Actual_Start_Date": "2019-12-30T03:53:24.397Z",
  "Actual_End_Date": "2019-12-30T03:53:24.397Z",
  "Product_Name": "string",
  "Product_Manufacturer": "string",
  "Product_Model_Version": "string",
  "Operational_Categorization_Tier_1": "string",
  "Operational_Categorization_Tier_2": "string",
  "Operational_Categorization_Tier_3": "string",
  "Vendor_Group": "string",
  "Vendor_Ticket_ID": "string",
  "Template": "string",
  "Work_Info": "string"
}
```

รูปที่ 3.30 รูปแบบ Body ของ Post CRQ ticket & Put CRQ ticket

เมื่อจัดเตรียม API ที่จำเป็นต่อการใช้งานในหน้า Summarize page เสร็จสมบูรณ์แล้ว ผู้พัฒนาได้เริ่มพัฒนาโดยสร้างเป็น method หลัก ๆ ดังนี้

- Method addApproveBox() และ deleteApproveBox() ดังรูปที่ 3.30 ทำหน้าที่ในการเพิ่มจำนวน Approvers box เมื่อผู้ใช้งานกด Add button และทำการลบจำนวน Approvers box เมื่อผู้ใช้งานกด Remove button โดยระบบจะทำการลบ Approvers field โดยระบุ ID ของ field ที่ผู้ใช้งานต้องการ

```
addApproveBox() {
  if(this.approverBoxes.length < 4){
    this.approverBoxes.push({id:this.approverBoxes.length, searchApprover: '', validStatus:null, showSearch: false, showLoaders:false,
    userLookup:[], selectedUser:this.selectedUser, approverIsCurrentUser:false, approverIsExisted:false, isApproverSameAsCoordinator: false});
  }
}

deleteApproveBox(){
  if(this.approverBoxes.length-1){
    this.deleteUser(this.approverBoxes.length-1);
    this.approverBoxes = this.approverBoxes.
    slice(0,this.approverBoxes.length-1);
  }
}
```

รูปที่ 3.31 Method addApproveBox() & deleteApproveBox()

- Method SearchTextChanged() ดังรูปที่ 3.31 ทำการรับค่าจาก field ที่ผู้ใช้งานทำการกรอก เมื่อเวลาผ่านไปทุก ๆ 500 millisecond ระบบจะทำการส่งค่าที่ผู้ใช้งานได้ทำการกรอกไปค้นหาโดยการเรียกใช้งาน method deSearch()

```

searchTextChanged(approverBox: approverBox) {
    approverBox.approverIsCurrentUser = false;
    if (this.textChangeTimeout) {
        clearTimeout(this.textChangeTimeout);
    }
    approverBox.validStatus = null;
    this.textChangeTimeout = setTimeout(() => {
        this.sendApproverValid()
        this.doSearch(approverBox);
    }, 500);
}

```

รูปที่ 3.32 Method SearchTextChanged()

- Method doSearch() ดังรูปที่ 3.32 โดยภายใน method หากผู้ใช้งานทำการกรอกหารายชื่อใน field ระบบจะทำการเรียกใช้ Get user look-up API และแสดงรายชื่อบรรจุมูลค่าที่ค้นหาเพื่อให้ผู้ใช้งานได้ทำการเลือก ถ้าหากผู้ใช้งานทำการลบชื่อออกจากช่อง ระบบจะทำการล้างค่าในตัวแปรที่เคยเก็บไว้ก่อนหน้า

```

doSearch(approverBox:approverBox){
    if (approverBox.searchApprover == "") {
        this.deleteUser(approverBox.id);
        this.showSearch = false;
    }
    else {
        approverBox.showLoader = true;
        this.apiService.getUserLookUp(approverBox.searchApprover).subscribe((userLookUp:Array<Object>)->{
            approverBox.userLookUp = userLookUp;
            approverBox.showLoader = false;
            approverBox.showSearch = true;
            approverBox.validStatus = null;
            approverBox.approverIsCurrentUser = false;
            approverBox.approverIsExisted = false;

            if (approverBox.userLookUp.length == 0 || approverBox.userLookUp == null) {
                approverBox.validStatus = false;
                this.deleteUser(approverBox.id);
            }
        })
    }
}

```

รูปที่ 3.33 Method deSearch()

- ในการตรวจสอบความถูกต้องของวิธีการและเพื่อให้สอดคล้องกับนโยบายขององค์กร ผู้พัฒนาจึงทำการสร้าง method ในการตรวจสอบ 4 กรณี ดังรูปที่ 3.33 โดย Method checksSameAsCurrentUser() ทำการตรวจสอบว่าผู้ใช้งานไม่ใช่คนเดียวกันกับผู้อนุมัติ, Method checksSelectedUserAlready Exist() ทำการตรวจสอบว่าผู้อนุมัติทุกคนที่ผู้ใช้งานทำการกรอกต้องไม่ซ้ำกัน, Method checksApproverListLessThan NumberOfBox() ทำการตรวจสอบว่า Approvers field ถูกกรอกครบทุกกล่อง, Method checksApproverSameAsCoordinator () ทำการตรวจสอบว่า Change coordinator หรือผู้ที่ถูกมอบหมาย CRQ ticket นี้ไม่ใช่คนเดียวกันกับผู้อนุมัติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ 39 ศึกษเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

checkIsSameAsCurrentUser(user:IPeople){
  if(this.userProfileLANid == user.ID){
    return true
  }
  else{
    return false
  }
}

checkIsSelectedUserAlreadyExist(user:IPeople, approverBox: approverBox){
  let isUserAlreadyExist:boolean = false
  for(let approverBoxx of this.approverBoxes){
    if(approverBoxx.id!==approverBox.id && approverBoxx.selectedUser.ID === user.ID){
      isUserAlreadyExist = true
      return isUserAlreadyExist
    }
  }

  return isUserAlreadyExist
}

checkIsApproverListLessThanNumberOfBox(approverBox: approverBox){
  var tempLength = this.approvers.length;
  if(tempLength<=approverBox['id']){
    return true
  }else{
    return false
  }
}

checkIsApproverSameAsCoordinator(selectedApprover:IPeople){
  let isApproverSameAsCoordinator : boolean = false
  this.messageService.currentCoordinator.subscribe((coordinator: IMember) => {
    if(selectedApprover.ID === coordinator.ID){
      isApproverSameAsCoordinator = true;
    }
  })
  return isApproverSameAsCoordinator
}

```

รูปที่ 3.34 Method checkIsSameAsCurrentUser(), checkIsSelectedUserAlreadyExist(), checkIsApproverListLessThanNumberOfBox(), checkIsApproverSameAsCoordinator

- Method selectUser() ดังรูปที่ 3.34 เป็น method ในการเลือกผู้อนุมัติ โดยมีการตรวจสอบกรณีต่าง ๆ ดังที่ได้กล่าวไว้ก่อนหน้านี้ แต่ถ้าหากผู้ใช้งานทำการเลือกผู้อนุมัติโดยไม่มีข้องเกี่ยวกับ 4 กรณีดังกล่าว ระบบจะทำการเก็บ array object ของผู้อนุมัติลงในตัวแปร approverBox.selectedUser ซึ่งมี type เป็น array object เพื่อเก็บผู้อนุมัติใน approvers field ต่าง ๆ โดยสามารถรับข้อมูลผู้อนุมัติสูงสุด 4 คนเท่านั้น
- Method deleteUser() ดังรูปที่ 3.35 เป็น method ในการลบผู้อนุมัติออก ระบบจะทำการตรวจสอบโดยถ้าหาก ID ของ approver field ที่ผู้ใช้งานต้องการลบ ตรงกับ ID ของ approver field ที่มีการเก็บไว้ ระบบจะทำการลบบุคคลดังกล่าวออก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ 40 วิชาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

selectUser(user:IPeople, approverBox: approverBox) {
    if(this.checkIsSameAsCurrentUser(user)){
        this.onClickOutSide(approverBox)
        approverBox.approverIsCurrentUser = true;
    }

    else if(this.checkIsApproverSameAsCoordinator(user)){
        approverBox.isApproverSameAsCoordinator = true;
        this.onClickOutSide(approverBox)
    }

    else{
        approverBox.searchApprover = user.Name;
        approverBox.isApproverSameAsCoordinator = false;
        user['boxId'] = approverBox.id;
        if(this.approvers.length !== 0){
            if(this.checkIsSelectedUserAlreadyExist(user,approverBox)){
                approverBox.validStatus = false
                this.onClickOutSide(approverBox);
                approverBox.approverIsExisted = true;
                return
            }
            else{
                this.deleteUser(approverBox.id)
                this.approvers.push(user);
            }
        }
        else{
            this.deleteUser(approverBox.id)
            this.approvers.push(user);
        }
    }

    approverBox.selectedUser = user;
    approverBox.userLookUp = [];
    approverBox.validStatus = true;
    approverBox.approverIsCurrentUser = false;
    approverBox.approverIsExisted = false;
    this.messageService.pushToAllApprovers(this.approverBoxes);
    this.sendApproverValid()
}
}

```

รูปที่ 3.35 Method selectUser()

```

deleteUser(id: number) {
    this.sendApproverValid()
    this.approverBoxes[id].selectedUser={} as IPeople;
    for(let approver of this.approvers){
        if(approver['boxId']==id){
            this.approvers.splice(this.approvers.indexOf(approver),1);
        }
    }
}
}

```

รูปที่ 3.36 Method deleteUser()

- Method SearchSupportGroupChanged() ดังรูปที่ 3.36 ทำการรับค่าจาก field ที่ผู้ใช้งานทำการกรอก เมื่อเวลาผ่านไปทุก ๆ 500 millisecond ระบบจะทำการส่งค่าที่ผู้ใช้งานได้ทำการกรอกไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ 41 ศึกษเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค้นหาโดยการเรียกใช้งาน method doSupportGroup Searched() ซึ่งหากผู้ใช้งานทำการกรอกหารายชื่อภายใน field ระบบจะทำการเรียกใช้ Get support group API และแสดงรายชื่อ support group ที่บรรจุค่าค้นหานั้น ๆ เพื่อให้ผู้ใช้งานได้ทำการเลือก ถ้าหากผู้ใช้งานทำการลบชื่อออกจากช่อง ระบบจะทำการล้างค่าในตัวแปรที่เคยเก็บไว้ก่อนหน้านี้

```
searchSupportGroupChanged(searchSupportGroup:string) {
  if (this.supportGroupChangeTimeout) {
    clearTimeout(this.supportGroupChangeTimeout);
  }
  if(this.memberValidStatus == true){
    this.deleteMember()
    this.memberValidStatus = null;
    this.searchMember = "";
  }
  this.supportGroupValidStatus = null;
  this.supportGroupChangeTimeout = setTimeout(() => {
    this.sendCoordinatorValid()
    this.doSupportGroupSearch(searchSupportGroup);
  }, 500);
}

doSupportGroupSearch(searchSupportGroup:string){
  this.coordinatorMember = [];
  this.coordinatorForSelecting = [];
  this.showMemberSearch = false;
  if (this.searchSupportGroup == "") {
    this.showSupportGroupSearch = false;
  }
  else {
    this.showSupportGroupLoader = true;
    this.apiService.getSupportGroup(searchSupportGroup).subscribe({supportGroupLookup}>=>{
      this.showSupportGroupLoader = false;
      this.showSupportGroupSearch = true;
      this.supportGroupLookup = supportGroupLookup;
      this.supportGroupValidStatus = null;
      if (this.supportGroupLookup.length == 0) {
        this.supportGroupValidStatus = false;
        this.deleteSupportGroup();
      }
    })
  }
}
```

รูปที่ 3.37 Method SearchSupportGroupChanged() & doSupportGroupSearched()

- Method SelectSupportGroup() ดังรูปที่ 3.37 เป็น method ในการเลือก support group และระบบจะทำการเก็บ array object ของ support group ลงในตัวแปร selectedSupport Group จากนั้นระบบทำการเรียกใช้ method showAllMembers InSupportGroup() โดย method นี้ทำการเรียกใช้ Get members in support group API และได้รับการตอบกลับเป็น array object ของสมาชิกใน support group นั้น ๆ จากนั้นระบบ จะทำการวน loop เพื่อคัดกรองเพียงสมาชิกที่มี Role ตามที่ระบุ เช่น เมื่อผู้ใช้งานกรอกใน change coordinator field ระบบจะทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ 42 วิชาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การกรองสมาชิกที่มีเพียง Role change coordinator มาแสดงใน dropdown list แต่ถ้าหากผู้ใช้งานกรอกใน change manager field ระบบจะทำการกรองสมาชิกที่มีเพียง Role change manager มาแสดงใน dropdown list เพื่อทำการเลือก

```
selectSupportGroup(supportGroup) {
    this.searchSupportGroup = supportGroup.Name;
    this.selectedSupportGroup = supportGroup;
    this.supportGroupLookUp = [];
    this.supportGroupValidStatus = true;
    this.showAllMembersInSupportGroup();
}

showAllMembersInSupportGroup(){
    this.showMemberLoader = true;
    this.apiService.getMemberInSupportGroup(this.selectedSupportGroup.Name).subscribe((memberLookUp: Array<IMember>)=>{
        this.memberLookUp = memberLookUp;
        this.memberValidStatus = null;
        this.sendCoordinatorValid()

        for (var i = 0 ; i < this.memberLookUp.length ; i++){
            if(this.memberLookUp[i].Role_Available.Change.Coordinator){
                this.coordinatorMember.push(this.memberLookUp[i])
            }
        }

        this.coordinatorForSelecting = this.coordinatorMember;
        this.showMemberLoader = false;
        this.showMemberSearch = true;

        if (this.memberLookUp.length == 0 ) {
            this.memberValidStatus = null;
            this.deleteMember();
        }
    })
}
```

รูปที่ 3.38 Method SearchMemberChanged() & doMemberSearch()

- Method receiveValid() ดังรูปที่ 3.38 เป็นการตรวจสอบว่าช่องกรอกทุกช่อง เช่น approver, change coordinator, change manager ถูกกรอกครบถ้วนด้วยข้อมูลที่ถูกต้องหรือไม่ หากยังคงมีบางช่องที่ยังไม่ถูกกรอก หรือมีความผิดพลาด ปุ่ม Next button ของหน้า summarize page จะไม่สามารถกดได้

```
receiveValid($event){
    if($event == 'truemanager'){
        this.managerMemberValidStatus = true;
    }
    else if($event == 'truecoordinator'){
        this.coordinatorMemberValidStatus = true;
    }
    else if($event == 'trueapprover'){
        this.allApproversValidStatus = true;
    }
    else{
        this.showSubmitButton = false;
        return
    }
    if(this.managerMemberValidStatus==true && this.coordinatorMemberValidStatus && this.allApproversValidStatus){
        this.showSubmitButton = true;
    }
}
```

รูปที่ 3.39 Method receiveValid()

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ 43 ศึกษานี้เท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Method createCrqTicket() ดังรูปที่ 3.39 เป็น method ที่เขียนขึ้นเพื่อสร้าง CRQ ticket โดยผู้พัฒนาได้ทำการ mapping ตัวแปรต่าง ๆ ที่เก็บข้อมูลส่วนต่าง ๆ ของผู้ใช้งาน ไว้กับ attribute ที่ต้องใช้ในส่ง Post CRQ ticket API เช่น map ตัวแปร request Summary, change coordinator, change manager ในโปรแกรมกับ attribute ที่จำเป็นในการส่ง API และสำหรับบาง attribute ที่ automate ได้ซึ่งสามารถลดภาระของผู้ใช้งานเป็นอย่างมาก ผู้พัฒนาทำการใส่ hard code ในการ mapping เช่น Schedule ผู้พัฒนาได้ทำการใส่ start date เป็นเวลา ณ ขณะที่ผู้ใช้งานทำการสร้าง CRQ ticket และ end date เป็นเวลา 3 วันหลังจาก CRQ ticket ถูกสร้างขึ้น หลังจากที่ CRQ ticket ถูกสร้างขึ้นเป็นที่เรียบร้อย UiPath orchestrator มีกระบวนการก่อนการ add approvers คือผู้พัฒนาจำเป็นต้องทำการเรียกใช้ Post CRQ ticket API ซึ่งเปรียบเสมือนการกด Next button ใน UiPath orchestrator's user interface เมื่อไปยัง stage ถัดไปแล้วผู้พัฒนาจึงทำการ mapping ตัวแปร Approver ID และ CRQ ticket ID เพื่อใช้ในการส่ง Add approvers API ในการเพิ่ม approvers ลงใน CRQ ticket

```

createCrqTicket(){
    this.uiService.setLoading(true);
    this.approverBoxes = this.addApproverComponent.approverBoxes;
    for(let approver of this.approverBoxes){
        this.approversId.push(approver.selectedUser.ID)
    };
    let today = new Date();
    let strToDate = formatDate(today,"y-MM-ddTHH:mm:ss.SSS'Z'",'en-US','+0');
    let endDate = today.setDate(today.getDate()+2);
    let strEndDate = formatDate(endDate,"y-MM-ddTHH:mm:ss.SSS'Z'",'en-US','+0');
    this.crq2["Summary"] = "<MTP_Portal>" + this.requestSummary;
    this.crq2["Change_Coordinator_Group"] = this.addChangeCoordinatorComponent.selectedSupportGroup['ID'];
    this.crq2["Change_Coordinator"] = this.addChangeCoordinatorComponent.selectedMember['ID'];
    this.crq2["Change_Manager_Group"] = this.addChangeManagerComponent.selectedSupportGroup['ID'];
    this.crq2["Change_Manager"] = this.addChangeManagerComponent.selectedMember['ID'];
    this.crq2["Notes"] = this.summarizeAssetTableComponent.finalAssetsString;
    this.crq2["Work_Info"] = this.summarizeAssetTableComponent.detailForNotes;
    this.crq2["Scheduled_Start_Date"] = strToDate;
    this.crq2["Scheduled_End_Date"] = strEndDate;
    this.orgUnitSelected = this.summarizeAssetTableComponent.orgUnitSelected;
    this.api.getCurrentUserUI().subscribe((res)=>{
    this.api.postCrq(this.crq2).subscribe((res)=>{
        this.crqId = res;
        this.api.putCrq(this.crqId).subscribe((res)=>{
            this.api.putCrq(this.crqId).subscribe((res)=>{
                for(let Id of this.approversId){
                    this.api.addApprover(Id,this.crqId).subscribe((res)=>{
                        this.crqTicketNumber = this.crqId;
                        if(this.isProduction){
                            this.ticketLink = " " + this.crqTicketNumber;
                        }else{
                            this.ticketLink = " " + this.crqTicketNumber;
                        }
                    });
                }
            });
        });
        this.approversId = [];
        this.afterAddApproval();
    });
    });
}

```

รูปที่ 3.40 Method createCrqTicket()

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ 44 ศึกษเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

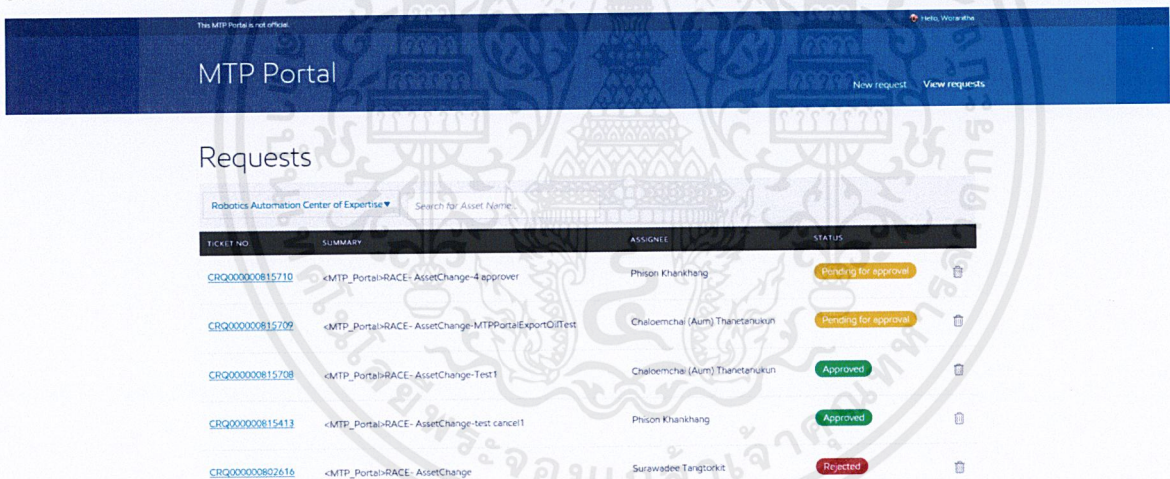
### 3.1.5.4 การพัฒนา View all tickets page

ในการพัฒนา View all tickets page ผู้พัฒนาได้ทำการการวางโครงสร้างของการเขียนโปรแกรม ดังรูปที่ 3.40

```
view-all-tickets
├── summary-filter.pipe.spec.ts
├── summary-filter.pipe.ts
├── view-all-tickets.component.css
├── view-all-tickets.component.html
├── view-all-tickets.component.spec.ts
└── view-all-tickets.component.ts
```

รูปที่ 3.41 การวางโครงสร้างหน้า View all tickets page

หลังจากวางโครงสร้างเป็นที่เรียบร้อย ผู้พัฒนาได้สร้าง User interface โดยในหน้า View all tickets page มีส่วนประกอบหลัก คือ View all tickets (Main component) ดังรูปที่ 3.41



รูปที่ 3.42 View all tickets (Main component)

API ที่มีเพิ่มเติมในหน้า View all tickets มีทั้งหมด 2 API ได้แก่

- Cancel ticket (ITSM API)

เรียกใช้งานผ่าน URL {ITSM URL}/change/{crqId}/cancel โดยใช้ HTTP Method GET โดยที่ API จะส่งข้อมูล array object ซึ่งเป็นข้อมูลของผู้ใช้งาน จากนั้นผู้พัฒนาจะนำข้อมูลของ Organization unit ทั้งหมดที่ผู้ใช้งานนั้น ๆ สามารถเข้าถึงได้ไปใช้งานต่อ โดย API นี้มีรูปแบบของ Response ดังรูปที่ 3.42

- Get CRQ ticket list (ITSM API)

เรียกใช้งานผ่าน URL {ITSM URL/change/group/{userSupport Group}?DataSource=Submitted โดยใช้ HTTP Method GET โดยที่ผู้เรียกใช้จะต้องระบุ support group และ API จะส่งข้อมูล array object ซึ่งเป็น CRQ ticket ทั้งหมดของ support group นั้น ๆ กลับมา โดย API นี้มีรูปแบบการตอบดังรูปที่ 3.43

```
{
  "Template_Name": "string",
  "Change_ID": "string",
  "Request_Number": "string",
  "Change_Type": "string",
  "Impact": "string",
  "Urgency": "string",
  "Priority": "string",
  "Has_Attachments": true,
  "Status": "string",
  "Status_Reason": "string",
  "Change_Reason": "string",
  "SLM": "string",
  "SLM_Hold": true,
  "Risk_Level": "string",
  "Stage": "string",
  "Class": "string",
  "Site": "string",
  "Service": "string",
  "Summary": "string",
  "Notes": "string",
  "Timing_Reason": "string",
  "Completed_Date": "2019-12-30T03:53:24.898Z",
  "Closed_Date": "2019-12-30T03:53:24.898Z",
  "Scheduled_Start_Date": "2019-12-30T03:53:24.898Z",
  "Scheduled_End_Date": "2019-12-30T03:53:24.898Z",
  "Actual_Start_Date": "2019-12-30T03:53:24.898Z",
  "Actual_End_Date": "2019-12-30T03:53:24.898Z",
  "Ticket_Locked": true,
  "Tasks_Count": 0,
  "Next_Approval_Phase": "string",
```

รูปที่ 3.43 รูปแบบ Response ของ Cancel ticket

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ 46 รัชศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
  "Change_ID": "string",
  "Effect": "string",
  "Urgency": "string",
  "Priority": "string",
  "Has Attachments": true,
  "Status": "string",
  "Change_Reason": "string",
  "SLM_Hold": true,
  "SLM_Hold": true,
  "Risk Level": "string",
  "Stage": "string",
  "Class": "string",
  "Summary": "string",
  "Completed Date": "2019-12-30T03:53:24.932Z",
  "Closed Date": "2019-12-30T03:53:24.932Z",
  "Scheduled_Start_Date": "2019-12-30T03:53:24.932Z",
  "Scheduled_End_Date": "2019-12-30T03:53:24.932Z",
  "Change_Coordinator": {
    "ID": "string",
    "Name": "string",
    "Group_ID": "string",
    "Group": "string",
    "Group_Organization": "string",
    "Group_Company": "string",
    "Extent": {
      "ID": "string",
      "Person_ID": "string",
      "Corporate_ID": "string",
      "Name": "string",
      "First Name": "string",
      "Last Name": "string",
      "Middle Initial": "string",
      "Email": "string",
      "Phone": "string",
      "Region": "string",
      "Country": "string",
      "Site": "string",
      "Desk": "string",
      "City": "string",
      "Company": "string",
      "Organization": "string",
      "Department": "string",
      "Job Title": "string",
      "Sensitivity": "string",
      "Litigation Hold": true,
      "VIP": true
    }
  },
  "Change_Manager": {
    "ID": "string",
    "Name": "string",
    "Group_ID": "string",
    "Group": "string",
    "Group_Organization": "string",
    "Group_Company": "string"
  },
  "Product": {
    "ID": "string",
    "Name": "string",
    "Model Version": "string",
    "Manufacturer": "string",
    "Tier 1": "string",
    "Tier 2": "string",
    "Tier 3": "string"
  },
  "Service_Level": {
    "ID": "string",
    "Title": "string",
    "Category": "string",
    "Progress": "string",
    "Email": "string",
    "Due": "2019-12-30T03:53:24.932Z",
    "Due In": "string"
  },
  "Submit Date": "2019-12-30T03:53:24.932Z",
  "Submitted_By": {
    "ID": "string",
    "Person_ID": "string",
    "Corporate_ID": "string",
    "Name": "string",
    "First Name": "string",
    "Last Name": "string",
    "Middle Initial": "string",
    "Email": "string",
    "Phone": "string",
    "Region": "string",
    "Country": "string",
    "Site": "string",
    "Desk": "string",
    "City": "string",
    "Company": "string",
    "Organization": "string",
    "Department": "string",
    "Job Title": "string",
    "Sensitivity": "string",
    "Litigation Hold": true,
    "VIP": true
  },
  "Last Modified Date": "2019-12-30T03:53:24.932Z",
  "Last Modified By": {
    "ID": "string",
    "Person_ID": "string",
    "Corporate_ID": "string",
    "Name": "string",
    "First Name": "string",
    "Last Name": "string",
    "Middle Initial": "string",
    "Email": "string",
    "Phone": "string",
    "Region": "string",
    "Country": "string",
    "Site": "string",
    "Desk": "string",
    "City": "string",
    "Company": "string",
    "Organization": "string",
    "Department": "string",
    "Job Title": "string",
    "Sensitivity": "string",
    "Litigation Hold": true,
    "VIP": true
  }
}
```

รูปที่ 3.44 รูปแบบ Response ของ Get CRQ ticket list

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ 47 รัชศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อจัดเตรียม API ที่จำเป็นต่อการใช้งานในหน้า View all tickets page เสร็จสมบูรณ์แล้ว ผู้พัฒนาได้เริ่มพัฒนาโดยสร้างเป็น method หลัก ๆ ดังนี้

- Method getCrqTicketList() ดังรูปที่ 3.44 ทำการเรียกใช้ Get CRQ ticket list API ซึ่งมีการตอบกลับเป็น array object ของ CRQ ticket ทั้งหมดที่อยู่ใน support group ที่ได้ทำการส่งไป จากนั้นระบบจะทำการกรอง CRQ ticket ให้เหลือเพียง CRQ ticket ที่ถูกสร้างขึ้นจากระบบที่กำลังพัฒนาเท่านั้น โดยกรองจาก tag <MTP\_Portal> ที่ระบบได้ทำการ automate added เข้าไปใน CRQ ticket summary ขณะที่ทำการสร้าง CRQ ticket ในหน้า summarize page เมื่อระบบทำการกรองเรียบร้อยแล้ว CRQ ticket ที่ถูกกรองจะถูกแสดงใน CRQ ticket table

```
async getCrqTicketList(){
  this.apiService.getCrqTicketList(this.firstSupportGroup).subscribe(({ticketLists})=>{
    this.ticketLists = ticketLists;
    this.finalTicketLists = this.ticketLists.filter(item => item['Summary'].includes("<MTP_Portal>"));
  })
}
```

รูปที่ 3.45 Method getCrqTicketList()

- Method updateSupportGroup() ดังรูปที่ 3.45 จะทำงานเมื่อผู้ใช้งานทำการเลือก support group ใหม่ จากนั้นระบบจะทำการเรียกใช้ Get CRQ ticket list API เพื่อนำ CRQ ticket ทั้งหมดที่อยู่ใน support group ใหม่มาแสดงใน CRQ ticket table

```
updateSupportGroup(supportGroupSelected:string){
  this.uiService.setLoading(true);
  if(this.supportGroupSelected == ""){
    this.ticketLists = [];
    this.uiService.setLoading(false);
  }
  else{
    this.apiService.getCrqTicketList(supportGroupSelected).subscribe(({ticketLists})=>{
      this.ticketLists = ticketLists;
      this.uiService.setLoading(false);
    })
  }
}
```

รูปที่ 3.46 Method updateSupportGroup()

- Method cancelTicket() ดังรูปที่ 3.46 เป็น method ที่ใช้สำหรับการยกเลิก CRQ ticket โดยผู้ใช้งานสามารถทำการยกเลิกได้ หาก CRQ ticket มีความผิดพลาด หรือไม่มีความจำเป็นต้องใช้งาน CRQ ticket นั้น ๆ อีกต่อไป

```

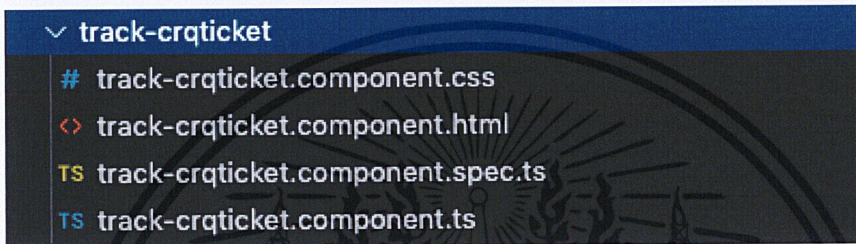
async cancelTicket(crqId){
  let cancelReturn = this.apiService.cancelTicket(crqId).toPromise();
  this.uiService.setLoading(true);
  await cancelReturn.then((data)=>{
  }, (error)=>{
    this.alertService.error(`${error} You do not have permission to move to the status of \"Cancelled\".`);
    console.log("Promise rejected with " + JSON.stringify(error));
  })
  await this.getCrqTicketList();
  this.uiService.setLoading(false);
}

```

รูปที่ 3.47 Method cancelTicket()

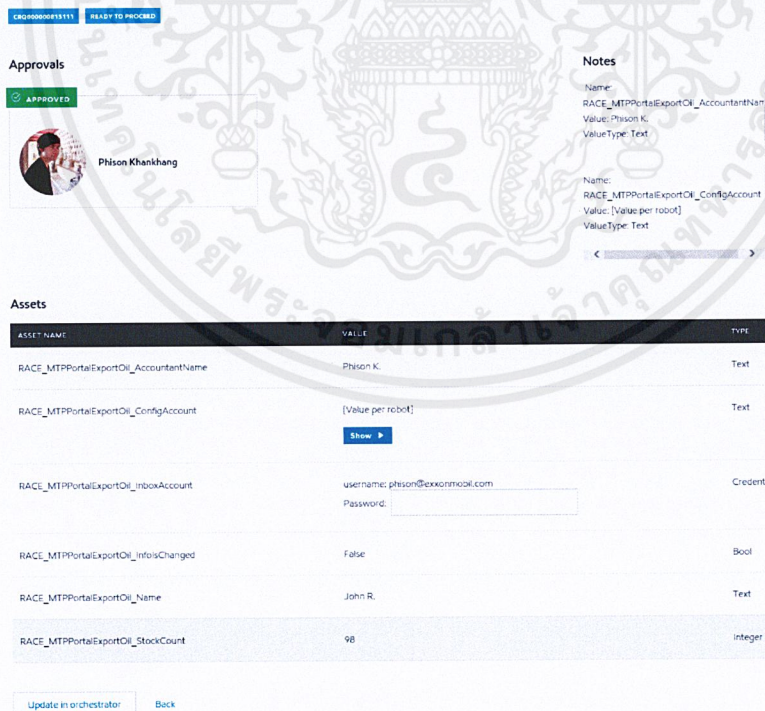
### 3.1.5.5 การพัฒนา Ticket detail page

ในการพัฒนา Ticket detail page ผู้พัฒนาได้ทำการการวางโครงสร้างของการเขียนโปรแกรม ดังรูปที่ 3.47



รูปที่ 3.48 การวางโครงสร้างหน้า Ticket detail page

หลังจากวางโครงสร้างเป็นที่เรียบร้อย ผู้พัฒนาได้สร้าง User interface โดยในหน้า Ticket detail page มีส่วนประกอบหลักคือ Track CRQ ticket (Main component) ดังรูปที่ 3.48



รูปที่ 3.49 Track CRQ ticket (Main component)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ 49 ศึกษเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

API ที่มีเพิ่มเติมในสำหรับหน้า Ticket detail page มีทั้งหมด 5 API ได้แก่

- Get current user permission (UiPath Orchestrator API)

เรียกใช้งานผ่าน URL {UiPathURL}/odata/Users/UiPath.Server.

Configuration.OData.GetCurrentPermissions() โดยใช้ HTTP Method GET จากนั้น API จะทำการส่งข้อมูลการมีสิทธิ์เข้าถึงส่วนต่าง ๆ ใน UiPath orchestrator ของผู้ใช้งานกลับมา โดย API นี้มีรูปแบบของ Body และ Response ดังรูปที่ 3.49

**GET** /odata/Users/UiPath.Server.Configuration.OData.GetCurrentPermissions()  
Returns a user permission collection containing data about the current user and all the permissions it has.

**Implementation Notes**  
Requires authentication.

**Response Class (Status 200)**  
Model Example Value

**UserPermissionsCollection {**  
  **UserId** (integer, optional): The Id of the user associated with the permissions.  
  **Permissions** (Array[string], optional): The collection of names of the permissions the user is associated with.  
}

รูปที่ 3.50 รูปแบบ Response ของ Get current user permission

- Post work detail (ITSM Orchestrator API)

เรียกใช้งานผ่าน URL {ITSM URL}/ change/{crqid}/workdetail()

โดยใช้ HTTP Method Post โดยที่ผู้เรียกใช้จะต้องทำการส่งข้อมูลของ Work detail ที่ต้องการเพิ่มพร้อมทั้งระบุ CRQ ticket ID ที่ต้องการทำการเพิ่ม จากนั้น API จะส่งข้อมูลที่ได้ทำการเพิ่มสำเร็จแล้วกลับมาโดย API นี้มีรูปแบบของ Body และ Response ดังรูปที่ 3.50

**BODY SAMPLE**

```
{
  "Summary": "string",
  "Status": "string",
  "Type": "string",
  "is_internal": true,
  "is_locked": true
}
```

**BODY SCHEMA**

**RESPONSE SAMPLE**

```
{
  "Instance_ID": "string",
  "Work_Log_ID": "string",
  "Summary": "string",
  "Notes": "string",
  "Submitter": "string",
  "Full Name": "string",
  "Submit Date": "2020-12-09T05:51:24.999Z",
  "Last Modified Date": "2020-12-09T05:51:24.999Z",
  "Type": "string",
  "Entry_Locked": true,
  "is_internal": true,
  "From_Request": true,
  "Attachment": [
    {
      "Name": "string",
      "Size": "string",
      "URL": "string",
      "Number": 0
    }
  ]
}
```

รูปที่ 3.51 รูปแบบ Body และ Response ของ Post work detail

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ 50 รัชศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Get CRQ ticket detail (ITSM API)

เรียกใช้งานผ่าน URL {ITSM URL}/change/{crqlid} โดยใช้ HTTP Method GET โดยที่ผู้เรียกใช้จะต้องระบุ CRQ ticket's ID และ API จะส่งข้อมูล array object ซึ่งเป็นรายละเอียดของ CRQ ticket นั้น ๆ กลับมา

- Post assets (UiPath Orchestrator API)

เรียกใช้งานผ่าน URL {UiPathURL}/odata/Assets() โดยใช้ HTTP Method Post โดยที่ผู้เรียกใช้จะต้องทำการส่งข้อมูลของ Assets ที่ต้องการเพิ่มใน UiPath orchestrator จากนั้น API จะส่งข้อมูล Assets ที่ได้ทำการเพิ่มแล้วกลับมา โดย API นี้มีรูปแบบของ Body และ Response ดังรูปที่ 3.51

- Put assets (UiPath Orchestrator API)

เรียกใช้งานผ่าน URL {UiPathURL}/odata/Assets ({Id})() โดยใช้ HTTP Method Put โดยที่ผู้เรียกใช้จะต้องทำการส่งข้อมูลของ Assets พร้อมทั้งระบุ Asset ID ที่ต้องการทำการอัปเดตค่า จากนั้น API จะส่ง Assets ที่ได้ทำการอัปเดตค่าแล้วกลับมา โดย API นี้มีรูปแบบของ Response ดังรูปที่ 3.52

POST /odata/Assets Creates an asset

**Implementation Notes**  
 Required permissions: Assets.Create

**Response Class (Status 201)**  
 Successfully created the asset

Model: Example Value

```

AssetDto {
  Name (string): A custom name for the asset.
  CanBeDeleted (boolean, optional): States if an assets can be deleted. The default value of this property is true.
  ValueScope (string): Defines the scope of the asset = ["Global", "PerRobot"];stringEnum: "Global", "PerRobot"
  ValueType (string, optional): Defines the type of value stored by the asset = [DBConnectionString, "HttpConnectionString", "Text", "Bool", "Integer", "Credential", "WindowsCredential", "KeyValueList"];stringEnum: "DBConnectionString", "HttpConnectionString", "Text", "Bool", "Integer", "Credential", "WindowsCredential", "KeyValueList"
  Value (string, optional): The textual representation of the asset value, irrespective of value type.
  StringValue (string, optional): The value of the asset when the value type is Text. Empty when the value type is not Text.
  BoolValue (boolean, optional): The value of the asset when the value type is Bool. False when the value type is not Bool.
  IntValue (integer, optional): The value of the asset when the value type is Integer. 0 when the value type is not Integer.
  CredentialUsername (string, optional): The user name when the value type is Credential. Empty when the value type is not Credential.
  CredentialPassword (string, optional): The password when the value type is Credential. Empty when the value type is not Credential.
  KeyValueList (Array<CustomKeyValuePair>): optional: A collection of key value pairs when the type is KeyValueList. Empty when the value type is not KeyValueList.
  RobotValues (Array<AssetRobotValueDto>): optional: The collection of asset values per robot. Empty if the asset type is Global.
  Id (integer, optional)
}
CustomKeyValuePair {
  Key (string, optional): A piece of text representing the key.
  Value (string, optional): A piece of text representing the value
}
AssetRobotValueDto {
  RobotId (integer, optional): The Id of the robot with which the asset is associated.
  RobotName (string, optional): The name of the robot with which the asset is associated.
  KeyTrail (string, optional): Masked value of the robot key.
  ValueType (string, optional): Defines the type of value stored by the asset = [DBConnectionString, "HttpConnectionString", "Text", "Bool", "Integer", "Credential", "WindowsCredential", "KeyValueList"];stringEnum: "DBConnectionString", "HttpConnectionString", "Text", "Bool", "Integer", "Credential", "WindowsCredential", "KeyValueList"
  StringValue (string, optional): The value of the asset when the value type is Text. Empty when the value type is not Text.
  BoolValue (boolean, optional): The value of the asset when the value type is Bool. False when the value type is not Bool.
  IntValue (integer, optional): The value of the asset when the value type is Integer. 0 when the value type is not Integer.
  Value (string, optional): The textual representation of the asset value, irrespective of value type.
  CredentialUsername (string, optional): The user name when the value type is Credential. Empty when the value type is not Credential.
  CredentialPassword (string, optional): The password when the value type is Credential. Empty when the value type is not Credential.
  KeyValueList (Array<CustomKeyValuePair>): optional: A collection of key value pairs when the type is KeyValueList. Empty when the value type is not KeyValueList.
  Id (integer, optional)
}
  
```

Response Content Type: application/json

Parameter	Value	Description	Parameter Type	Data Type
assetDto	(required)		body	Model: Example Value

Parameter content type: application/json

```

AssetDto {
  Name (string): A custom name for the asset.
  CanBeDeleted (boolean, optional): States if an assets can be deleted. The default value of this property is true.
  ValueScope (string): Defines the scope of the asset = ["Global", "PerRobot"];stringEnum: "Global", "PerRobot"
  ValueType (string, optional): Defines the type of value stored by the asset = [DBConnectionString, "HttpConnectionString", "Text", "Bool", "Integer", "Credential", "WindowsCredential", "KeyValueList"];stringEnum: "DBConnectionString", "HttpConnectionString", "Text", "Bool", "Integer", "Credential", "WindowsCredential", "KeyValueList"
  Value (string, optional): The textual representation of the asset value, irrespective of value type.
  StringValue (string, optional): The value of the asset when the value type is Text. Empty when the value type is not Text.
  BoolValue (boolean, optional): The value of the asset when the value type is Bool. False when the value type is not Bool.
  IntValue (integer, optional): The value of the asset when the value type is Integer. 0 when the value type is not Integer.
  CredentialUsername (string, optional): The user name when the value type is Credential. Empty when the value type is not Credential.
  CredentialPassword (string, optional): The password when the value type is Credential. Empty when the value type is not Credential.
  KeyValueList (Array<CustomKeyValuePair>): optional: A collection of key value pairs when the type is KeyValueList. Empty when the value type is not KeyValueList.
  RobotValues (Array<AssetRobotValueDto>): optional: The collection of asset values per robot. Empty if the asset type is Global.
  Id (integer, optional)
}
CustomKeyValuePair {
  Key (string, optional): A piece of text representing the key.
  Value (string, optional): A piece of text representing the value
}
AssetRobotValueDto {
  RobotId (integer, optional): The Id of the robot with which the asset is associated.
  RobotName (string, optional): The name of the robot with which the asset is associated.
  KeyTrail (string, optional): Masked value of the robot key.
  ValueType (string, optional): Defines the type of value stored by the asset = [DBConnectionString, "HttpConnectionString", "Text", "Bool", "Integer", "Credential", "WindowsCredential", "KeyValueList"];stringEnum: "DBConnectionString", "HttpConnectionString", "Text", "Bool", "Integer", "Credential", "WindowsCredential", "KeyValueList"
  StringValue (string, optional): The value of the asset when the value type is Text. Empty when the value type is not Text.
  BoolValue (boolean, optional): The value of the asset when the value type is Bool. False when the value type is not Bool.
  IntValue (integer, optional): The value of the asset when the value type is Integer. 0 when the value type is not Integer.
  Value (string, optional): The textual representation of the asset value, irrespective of value type.
  CredentialUsername (string, optional): The user name when the value type is Credential. Empty when the value type is not Credential.
  CredentialPassword (string, optional): The password when the value type is Credential. Empty when the value type is not Credential.
  KeyValueList (Array<CustomKeyValuePair>): optional: A collection of key value pairs when the type is KeyValueList. Empty when the value type is not KeyValueList.
  Id (integer, optional)
}
  
```

รูปที่ 3.52 รูปแบบของ Body และ Response ของ Post assets

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ 52 รัชศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Parameter	Value	Description	Parameter Type	Data Type
Id	[required]	key: Id	path	long
assetId	[required]		body	Model

Parameter content type: application/json	Example Value
	<pre> <b>AssetDo</b> {   <b>Name</b> (string): A custom name for the asset.   <b>CanBeDeleted</b> (boolean, optional): Status if an asset can be deleted. The default value of this property is true.   <b>ValueScope</b> (string): Defines the scope of the asset.   * [Global]   * <b>PerRobot</b> (stringEnum: "Global", "PerRobot")   <b>ValueType</b> (string, optional): Defines the type of value stored by the asset.   * [DBConnectionString, HttpConnectionString, Text, Bool, Integer, Credential, WindowsCredential]   * <b>KeyValueList</b> (stringEnum: "DBConnectionString", "HttpConnectionString", "Text", "Bool", "Integer", "Credential", "WindowsCredential")   * <b>KeyValueList</b>   <b>Value</b> (string, optional): The textual representation of the asset value, irrespective of value type.   <b>StringValue</b> (string, optional): The value of the asset when the value type is Text. Empty when the value type is not Text.   <b>BoolValue</b> (boolean, optional): The value of the asset when the value type is Bool. False when the value type is not Bool.   <b>IntValue</b> (integer, optional): The value of the asset when the value type is Integer. 0 when the value type is not Integer.   <b>CredentialUsername</b> (string, optional): The user name when the value type is Credential. Empty when the value type is not Credential.   <b>CredentialPassword</b> (string, optional): The password when the value type is Credential. Empty when the value type is not Credential.   <b>KeyValueList</b> (Array&lt;CustomKeyValuePair&gt;, optional): A collection of key value pairs when the type is KeyValueList. Empty when the value type is not KeyValueList.   <b>RobotValues</b> (Array&lt;AssetInfoValueDo&gt;, optional): The collection of asset values per robot. Empty if the asset type is Global.   <b>Id</b> (string, optional)   <b>CustomKeyValuePair</b> {     <b>Key</b> (string, optional): A piece of text representing the key.     <b>Value</b> (string, optional): A piece of text representing the value.   }   <b>AssetInfoValueDo</b> {     <b>RobotId</b> (string, optional): The id of the robot with which the asset is associated.     <b>RobotName</b> (string, optional): The name of the robot with which the asset is associated.     <b>Key</b> (string, optional): Masked value of the root key.     <b>ValueType</b> (string, optional): Defines the type of value stored by the asset.   * [DBConnectionString, HttpConnectionString, Text, Bool, Integer, Credential, WindowsCredential]   * <b>KeyValueList</b> (stringEnum: "DBConnectionString", "HttpConnectionString", "Text", "Bool", "Integer", "Credential", "WindowsCredential")   * <b>KeyValueList</b>   <b>StringValue</b> (string, optional): The value of the asset when the value type is Text. Empty when the value type is not Text.   <b>BoolValue</b> (boolean, optional): The value of the asset when the value type is Bool. False when the value type is not Bool.   <b>IntValue</b> (integer, optional): The value of the asset when the value type is Integer. 0 when the value type is not Integer.   <b>Value</b> (string, optional): The textual representation of the asset value, irrespective of value type.   <b>CredentialUsername</b> (string, optional): The user name when the value type is Credential. Empty when the value type is not Credential.   <b>CredentialPassword</b> (string, optional): The password when the value type is Credential. Empty when the value type is not Credential.   <b>KeyValueList</b> (Array&lt;CustomKeyValuePair&gt;, optional): A collection of key value pairs when the type is KeyValueList. Empty when the value type is not KeyValueList.   <b>Id</b> (string, optional) </pre>

รูปที่ 3.53 รูปแบบ Body ของ Put assets

เมื่อจัดเตรียม API ที่จำเป็นต่อการใช้งานในหน้า Ticket detail page เสร็จสมบูรณ์แล้ว ผู้พัฒนาได้เริ่มพัฒนาโดยสร้างเป็น method หลัก ๆ ดังนี้

- Method checkStatus() ดังรูปที่ 3.53 เริ่มต้นโดยระบบจะทำการเรียกใช้งาน Get CRQ ticket detail API โดยมีการระบุ CRQ ticket ID ที่ผู้ใช้งานได้ทำการเลือก จากนั้นเมื่อได้ Response กลับมาแล้ว ระบบจะทำการ mapping ตัวแปรในโปรแกรมกับบาง attribute ของ response เช่น Notes, Approvals, Status และ Work detail เป็นต้น เมื่อตัวแปรที่ได้สร้างขึ้นทำการจัดเก็บ attribute ที่จำเป็นต้องใช้งานต่อแล้ว ระบบจะทำการเรียกใช้ Get current user permission ในการรับข้อมูลการมีสิทธิ์เข้าถึงระบบ UiPath orchestrator ของผู้ใช้งาน และส่งค่าที่ได้รับไปตรวจสอบโดยเรียกใช้งาน method checkUserPermission() และมีการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ 53 ปรึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตรวจสอบข้อมูลอื่นโดยเรียกใช้งาน method checkUsersIs Approver() และ checkIsApproverIsCoordinator() เพื่อความปลอดภัยในการทำรายการต่าง ๆ จากนั้นระบบได้ทำการตรวจสอบกรณีเพื่อความถูกต้องสมควรตามนโยบายองค์กร

```
checkStatus(){
  this.api.getCurrentUser().subscribe((currentUser)=>{
    this.currentUser = currentUser;
    this.userProfileAWid = currentUser['Login_ID'];
  })

  this.api.getCrq(this.crqId).subscribe((crqDetail)=>{
    this.ticketNotes = crqDetail["Notes"];
    this.ticketApprovals = crqDetail["Approvals"];
    this.crqTicketStatus = crqDetail["Status"];
    this.crqTicketWorkDetails = crqDetail["Work_Details"];
    this.getTicketApprovers(this.ticketApprovals);

    this.api.getCurrentUserPermission().subscribe((resPermission)=>{
      this.isApproverSameAsCoordinator = this.checkIsApproverIsCoordinator(crqDetail["Change_Coordinator"]["ID"]);
      this.hasPermission = this.checkUserPermission(resPermission);
      this.isApprover = this.checkUserIsApprover();
      this.isCrqTicketStatusCanGoLive = this.checkCrqStatus(this.crqTicketStatus);
      this.showGoLiveButton = this.enableGoLiveButton(this.hasPermission, this.isApprover, this.isCrqTicketStatusCanGoLive, this.isApproverSameAsCoordinator);
    })

    return this.getAssetFromWorkDetails(this.crqTicketWorkDetails);
  })
}
```

รูปที่ 3.54 Method checkStatus()

- Method getTicketApprovers () ดังรูปที่ 3.54 เป็น method ที่นำตัวแปร ticketApprovals มาทำการวน loop และสร้างตัวแปรใหม่เพื่อเก็บ ID ของผู้อนุมัติทุกคน จากนั้นจึงทำการดึง profile picture ของผู้อนุมัติแต่ละคน โดยนำ ID เติมเข้ากับ URL รูปภาพ และแสดงในหน้า Ticket detail page

```
getTicketApprovers(ticketApprovals){
  if(ticketApprovals.length == 0){
    for(let ticketApproval of ticketApprovals){
      this.ticketApproverId.push(ticketApproval.Approver_ID)
      ticketApproval['profilepicture'] = this.userService.getUserProfilePicture(ticketApproval.Approver_ID ? ticketApproval.Approver_ID.replace('\', '') : null);
      this.uiService.setLoading(false);
    }
  } else{
    this.alertService.error(this.noApproveMessage);
    window.scrollTo(0,0);
  }
}
```

รูปที่ 3.55 Method getTicketApprovers()

- ในการตรวจสอบความถูกต้องของวิธีการและเพื่อให้สอดคล้องกับนโยบายขององค์กร ผู้พัฒนาจึงมีการตรวจสอบทั้งในหน้า Summarize page คือกระบวนการตรวจสอบก่อนการสร้าง CRQ ticket และมีการตรวจสอบอีกหนึ่งครั้งในหน้านี้ ซึ่งเป็นกระบวนการก่อนดำเนินงานจริงใน UiPath orchestrator โดยผู้พัฒนาทำการสร้าง method ในการตรวจสอบ 3 กรณี ดังรูปที่ 3.55 โดย Method checkUsersIsApprover() ทำการตรวจสอบว่า

ผู้ใช้งานไม่ใช่คนเดียวกับผู้อนุมัติ, Method checkUserPermission() ทำการตรวจสอบว่าผู้ใช้งานมีสิทธิ์เข้าถึงส่วนของการดำเนินงานที่กำลังดำเนิน ณ ขณะนี้, Method checkIsApproverIsCoordinator() ทำการตรวจสอบว่า Change coordinator หรือ ผู้ที่ถูกมอบหมาย CRQ ticket นี้ไม่ใช่คนเดียวกับผู้อนุมัติ

```
checkUserIsApprover(){
  for( var i = 0 ; i < this.ticketApproversLanId.length ; i++ ){
    if(this.userProfileLanId == this.ticketApproverLanId(i)){
      return true;
    }
  }
  else{
    return false;
  }
}

checkUserPermission(resPermission){
  if(resPermission['Permissions'].indexOf("Assets.Create") != -1 && resPermission['Permissions'].indexOf("Assets.Edit") != -1){
    return true;
  }
  else{
    return false;
  }
}

checkIsApproverIsCoordinator(coordinator:string){
  if(this.ticketApproversLanId.indexOf(coordinator) != -1){
    return true;
  }else{
    return false;
  }
}
```

รูปที่ 3.56 Method checkUserIsApprover(), checkUserPermission(), checkIsApproverIsCoordinator()

- Method checkCrqStatus() ดังรูปที่ 3.56 ทำการตรวจสอบสถานะของ CRQ ticket ที่ผู้ใช้ได้ทำการเลือก โดยระบบจะทำการตรวจสอบเป็นกรณี ๆ โดยแต่ละกรณีมีผลต่อการทำงานของ Update in orchestrator button

```
checkCrqStatus(crqTicketStatus){
  if(crqTicketStatus=="Scheduled" || crqTicketStatus=="Implementation In Progress"){
    this.status = "Ready to proceed";
    return true;
  }
  else if(crqTicketStatus=="Pending"){
    this.status = "Rejected";
    this.alertService.error(this.status+" : Ticket is rejected. Please cancel CRQ from ITConsole");
    return false;
  }
  else if(crqTicketStatus=="Completed" || crqTicketStatus=="Closed"){
    this.status = "completed moving assets";
    this.alertService.success(this.status);
    return false;
  }
  else if(crqTicketStatus!="Scheduled"){
    this.status = "Pending for approval";
    return false;
  }
  else{
    return false;
  }
}
```

รูปที่ 3.57 Method checkCrqStatus()

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ 55 ารศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Method enableGoLiveButton() ดังรูปที่ 3.57 ระบบได้ทำการตรวจสอบกรณีเพื่อความถูกต้องสมควรตามนโยบายองค์กร เช่น ผู้อนุมัติไม่สามารถเป็นคนเดียวกันกับ change coordinator, ผู้ใช้งานไม่สามารถเป็นหนึ่งในผู้อนุมัติใน CRQ ticket ที่จะทำการนำขึ้น production, สถานะของ CRQ ticket ต้องอยู่ในสถานะที่พร้อมนำขึ้น production และสิทธิ์การเข้าถึงของผู้ใช้งานจะต้องสอดคล้องกับการทำรายการ ระบบจึงจะดำเนินการต่อไปได้ หากระบบทำการตรวจสอบว่าข้อมูลเป็นไปตามกรณีทั้งหมดที่ได้กล่าวมา Update in orchestrator button จะสามารถทำงานได้ แต่ถ้าหากข้อมูลไม่เป็นไปตามกรณีใดกรณีหนึ่ง Update in orchestrator button จะไม่สามารถทำงานได้

```

enableGoLiveButton(hasPermission, isApprover, isCrqTicketStatusCanGoLive, isApproverSameAsCoordinator){
    if(hasPermission){
        if(!isApprover&&!isApproverSameAsCoordinator){
            if(isCrqTicketStatusCanGoLive){
                return true;
            }
            else{
                return false;
            }
        }
        else if(!isApprover&&isApproverSameAsCoordinator){
            this.alertMessage = this.approverSameAsCoordinatorMessage ;
            this.alertService.error(this.alertMessage);
            return false;
        }
        else if(isApprover&&!isApproverSameAsCoordinator){
            this.alertMessage = this.approverCantPressMessage ;
            this.alertService.error(this.alertMessage);
            return false;
        }
        else{
            this.alertMessage = this.approverSameAsCoordinatorMessage + ' & ' +this.approverCantPressMessage ;
            this.alertService.error(this.alertMessage);
            return false;
        }
    }
    else{
        this.alertMessage = this.noPermissionMessage ;
        this.alertService.error(this.alertMessage);
        return false;
    }
}

```

รูปที่ 3.58 Method enableGoLiveButton()

- Method getAssetFromWorkDetails() ดังรูปที่ 3.58 ระบบรับข้อมูล CRQ ticket's work detail และทำการตรวจสอบ work detail ส่วนที่เป็นข้อมูล Assets ที่ต้องการทำรายการใน UiPath orchestrator และทำการเก็บข้อมูลส่วนนี้ในตัวแปรที่ถูกสร้างขึ้นเพื่อนำไปแสดงใน Assets confirmation table แต่ถ้าหากระบบทำการตรวจสอบ และไม่พบ work detail ดังกล่าว Alert service

จะถูกทำงานโดยมีการแจ้งเตือนแก่ผู้ใช้งานในการดำเนินการติดต่อ  
ทีมที่ดูแลในการติดตามช่วยเหลือ

```
getAssetFromWorkDetails(crqTicketWorkDetails){
  for(let crqTicketWorkDetail of crqTicketWorkDetails){
    if(!crqTicketWorkDetail["Is_Internal"] && crqTicketWorkDetail["Notes"].indexOf(",") !== -1){
      this.assetsDetail = crqTicketWorkDetail["Notes"]
      this.alertChangeData = "Founded Change";
      return this.prepareAssetsDetail();
    }
  }
  this.alertService.error("Not found change pls contact race team");
}
```

รูปที่ 3.59 Method getAssetFromWorkDetails()

- Method goLive() ดังรูปที่ 3.59 เป็น Method ที่สำคัญมากในกระบวนการนำขึ้น production โดย method นี้เริ่มจากระบบทำการเรียกใช้งาน Get license API ในการทำ Authentication จากนั้นระบบจัดการเรียกใช้งาน Get user's organization unit API ใน production และเก็บข้อมูลดังกล่าวไว้ในตัวแปรที่สร้างขึ้น จากนั้นทำการวน loop เพื่อเปรียบเทียบ organization unit's display name ที่ผู้ใช้งานทำการเลือก กับ organization unit's display name ใน production เพื่อระบบจะนำเอา organization unit's ID ใน production เก็บในตัวแปร เพื่อการดำเนินการต่อ จากนั้นระบบทำการเรียกใช้ Get assets ใน production และได้รับข้อมูล assets ทั้งหมดใน production กลับมา จากนั้นระบบทำการนำข้อมูลที่ได้รับส่งต่อเพื่อทำการเรียกใช้ method checkAsset() ดังรูปที่ 3.60 โดย method นี้จะทำการวน loop เพื่อเปรียบเทียบระหว่าง asset ที่อยู่ใน production กับ asset ที่ผู้ใช้งานต้องการทำการเปลี่ยนแปลงหรือเพิ่มเติม ถ้าหาก asset ตัวใดที่ผู้ใช้งานต้องการทำการเปลี่ยนแปลงหรือเพิ่มเติมเป็น asset ที่มีอยู่แล้วใน production ระบบนำ asset นี้เข้าไปทำ method postAsset() แต่ถ้าหาก asset ตัวใดที่ผู้ใช้งานต้องการทำการเปลี่ยนแปลงหรือเพิ่มเติมเป็น asset ยังไม่มีใน production ระบบนำ asset นี้เข้าไปทำ method postAsset()

```

goLive(assets: Array<Object>){
  this.assetsCount = 0;
  this.assetsErrorCount = 0;
  this.api.getCurrentUserUIProd().subscribe((res)=>{
    this.api.getOrgUnitProdById().subscribe((orgUnitsProd)=>{
      for(let orgUnit of orgUnitsProd['OrganizationUnits']){
        if(this.orgUnitSelected['DisplayName']==orgUnit['DisplayName']){
          this.orgUnitSelected = JSON.parse(JSON.stringify(orgUnit))
          break
        }
      }
    })
  })
  this.api.getAssetsProd(this.orgUnitSelected['Id']).subscribe((prodAssets)=>{
    this.prodAssets = prodAssets['value'];
    this.checkedAssets = this.checkAsset(assets);
    this.messageService.changeUpdateOrchestratorMessage(this.updateOrchestratorStatus);
    this.uiService.setLoading(true);
    this.assetsCount=0;
    for(let asset of this.checkedAssets){
      if(asset['isPost']==true){
        this.postAsset(asset,0);
      }
      else if(asset['isPost']==false){
        this.putAsset(asset,0);
      }
    }
  })
}

```

รูปที่ 3.60 Method goLive()

```

checkAsset(selectedAssets){
  for( var i = 0 ; i < selectedAssets.length ; i++ ){
    if(this.prodAssets.length!= 0){
      for(var j = 0 ; j < this.prodAssets.length ; j++ ){
        if(selectedAssets[i]['Name'] === this.prodAssets[j]['Name']){
          selectedAssets[i]['Id'] = this.prodAssets[j]['Id'];
          selectedAssets[i]['isPost'] = false;
          break
        }
        else{
          selectedAssets[i]['isPost'] = true;
        }
      }
    }
    else{
      selectedAssets[i]['isPost'] = true;
    }
  }
  return selectedAssets
}

```

รูปที่ 3.61 Method checkAsset()

- Method postAsset() ดังรูปที่ 3.61 ทำการรับข้อมูล asset ที่ผู้ใช้งานต้องการเพิ่ม และตัวแปร postfailed ซึ่งเป็นตัวแปรที่ทำการนับจำนวนครั้งของ asset ที่ทำดำเนินการเพิ่มลงใน UiPath orchestrator ไม่สำเร็จ การทำงานเริ่มต้นโดยการนำข้อมูล asset มาทำการลบ attribute ที่ไม่สอดคล้องกับ Body ในการส่ง Post

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ 58 ศึกษเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

assets API จากนั้นระบบทำการแยก asset ที่เป็น single value type กับ value per robot type และทำการ mapping ค่าที่ผู้ใช้งานต้องการทำรายการ กับ attribute ของ Body เข้าด้วยกัน เมื่อทำการ mapping เรียบร้อยแล้ว ระบบจะทำการเรียกใช้งาน Get license API ในการทำ Authentication และต่อด้วย Post assets API ในการ post asset แต่ละตัว โดยระบบได้ทำการสร้างตัวแปรในการรับจำนวนของ asset ที่ระบบทำรายการสำเร็จ และ ไม่สำเร็จ ทำให้ตรวจสอบได้ว่า asset ถูกทำรายการได้ครบถ้วนหรือไม่ โดยหากมี asset ตัวใดที่ระบบทำรายการไม่สำเร็จ ระบบจำทำรายการนั้นใหม่อีกครั้ง ในกรณีที่มี error เกิดขึ้นขณะที่ทำรายการ ระบบจะทำการเรียกใช้งาน Post work detail API และใส่รายละเอียด error ใน CRQ ticket นั้น ๆ เพื่อให้ผู้ใช้งานสามารถตรวจสอบความผิดพลาดและแก้ไขได้

- Method putAsset() ดังรูปที่ 3.62 ทำการรับข้อมูล asset ที่ผู้ใช้งานต้องการเปลี่ยนแปลงค่า และตัวแปร putfailed ซึ่งเป็นตัวแปรที่ทำการนับจำนวนครั้งของ asset ที่ทำดำเนินการเปลี่ยนแปลงลงใน UiPath orchestrator ไม่สำเร็จ การทำงานเริ่มต้นโดยการนำข้อมูล asset มาทำการลบ attribute ที่ไม่สอดคล้องกับ Body ในการส่ง Put assets API จากนั้นระบบทำการแยก asset ที่เป็น single value type กับ value per robot type และทำการ mapping ค่าที่ผู้ใช้งานต้องการทำรายการ กับ attribute ของ Body เข้าด้วยกัน เมื่อทำการ mapping เรียบร้อยแล้ว ระบบจะทำการเรียกใช้งาน Get license API ในการทำ Authentication และต่อด้วย Put assets API ในการ put asset แต่ละตัว โดยระบบได้ทำการสร้างตัวแปรในการรับจำนวนของ asset ที่ระบบทำรายการสำเร็จ และ ไม่สำเร็จ ทำให้ตรวจสอบได้ว่า asset ถูกทำรายการได้ครบถ้วนหรือไม่ โดยหากมี asset ตัวใดที่ระบบทำรายการไม่สำเร็จ ระบบจำทำรายการนั้นใหม่อีกครั้ง ในกรณีที่มี error เกิดขึ้นขณะที่ทำรายการ ระบบจะทำการเรียกใช้งาน Post work detail API และใส่รายละเอียด error ใน CRQ ticket นั้น ๆ เพื่อให้ผู้ใช้งานสามารถตรวจสอบความผิดพลาดและแก้ไขได้



### 3.1.5.6 การพัฒนา Value per robot feature (Enhancement)

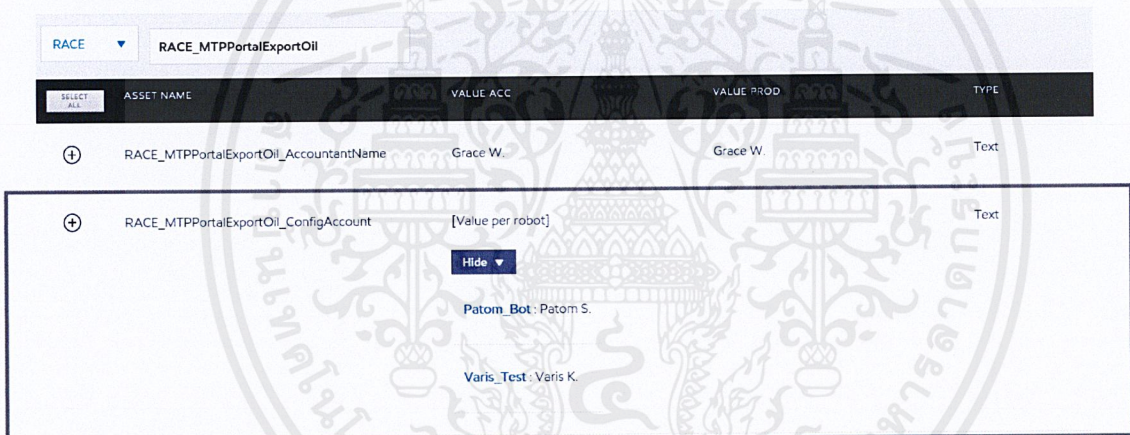
ในการพัฒนา Value per robot feature ผู้พัฒนาได้ทำการการวางโครงสร้างของการเขียนโปรแกรม ดังรูปที่ 3.63

```
▼ add-value-per-robot
# add-value-per-robot.component.css
<> add-value-per-robot.component.html
TS add-value-per-robot.component.spec.ts
TS add-value-per-robot.component.ts
```

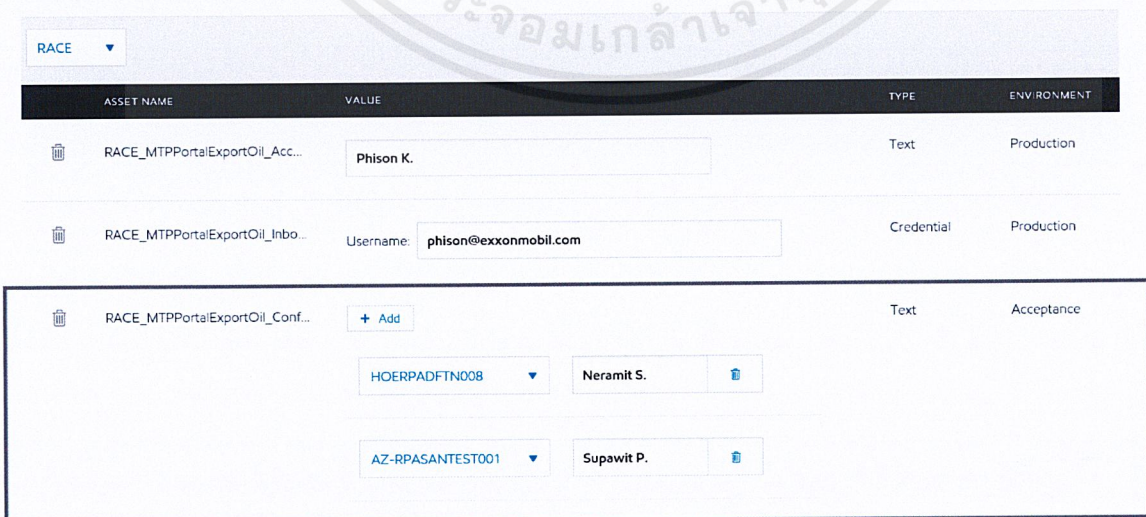
รูปที่ 3.64 การวางโครงสร้าง Value per robot feature

หลังจากวางโครงสร้างเป็นที่เรียบร้อย ผู้พัฒนาได้สร้าง User interface เพิ่มเติมจากหน้า Select assets page โดย Value per robot feature มีส่วนประกอบหลักคือ Value per robot component ดังรูปที่ 3.64

#### Assets



#### Selected Assets



รูปที่ 3.65 Value per robot feature

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ 61 ารศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

API ที่ใช้เพิ่มเติมสำหรับ Value per robot feature มี 1 API ได้แก่

- Get robots (UiPath Orchestrator API)

เรียกใช้งานผ่าน URL {UiPathURL}/odata/Robots โดยใช้ HTTP Method GET ซึ่ง API นี้จะส่งกลับมาเป็น array object ของ robots ทั้งหมดที่มีอยู่ใน organization unit ที่ทำการเลือกไว้ โดย API นี้มีรูปแบบของ Response ดังรูปที่ 3.65

**GET /odata/Robots** Gets robots.

**Implementation Notes**  
Required permissions: Robots.View.

**Response Class (Status 200)**  
Model Example Value

```
ODataResponse[List[RobotDto]] {
  @odata.context (string, optional),
  value (Array[RobotDto], optional)
}
RobotDto {
  LicenseKey (string, optional): The key is automatically generated from the server for the Robot machine. For the robot to work, the same key must exist on both the robot and Orchestrator. All robots on a machine must have the same license key in order to register correctly.stringMax. Length:255.
  MachineName (string, optional): The name of the machine a Robot is hosted on.stringMax. Length:50.
  MachineId (integer, optional): The Id of the machine a Robot is hosted on.
  Name (string): A custom name for the robot.stringMax. Length:19.
  Username (string): The machine username. If the user is under a domain, you are required to also specify it in a DOMAIN\username format. Note: You must use short domain names, such as desktop\administrator and NOT desktop.local/administrator.stringMax. Length:100.
  Description (string, optional): Used to add additional information about a robot in order to better identify it.stringMax. Length:500.
  Version (string, optional): The Robot's Version.
  Type (string): The Robot type. = ['NonProduction', 'Attended', 'Unattended', 'Development']stringEnum:"NonProduction", "Attended", "Unattended", "Development".
  HostingType (string): The Robot hosting type (Standard / Floating). = ['Standard', 'Floating']stringEnum:"Standard", "Floating".
  Password (string, optional): The Windows password associated with the machine username.stringMax. Length:100.
  CredentialType (string, optional): The robot credentials type (Default/ SmartCard) = ['Default', 'SmartCard']stringEnum:"Default", "SmartCard".
  Environments (Array[EnvironmentDto], optional): The collection of environments the robot is part of.
  RobotEnvironments (string, optional): The comma separated textual representation of environment names the robot is part of.
  ExecutionSettings (inline_model_5, optional): A collection of key value pairs containing execution settings for this robot.
  Id (integer, optional)
}
EnvironmentDto {
  Name (string): A custom name for the environment.stringMax. Length:100.
  Description (string, optional): Used to add additional information about an environment in order to better identify it.stringMax. Length:500.
  Robots (Array[SimpleRobotDto], optional): The collection of robots associated with the current environment.
  Type (string, optional): DEPRECATED. The environment type specifies how it should be used. This property is deprecated and should no longer be used. = ['Dev', 'Test', 'Prod']stringEnum:"Dev", "Test", "Prod".
  Id (integer, optional)
}
inline_model_5 {} SimpleRobotDto {
  LicenseKey (string, optional): The key is automatically generated from the server for the Robot machine. For the robot to work, the same key must exist on both the robot and Orchestrator. All robots on a machine must have the same license key in order to register correctly.stringMax. Length:255.
  MachineName (string, optional): The name of the machine a Robot is hosted on.stringMax. Length:50.
  MachineId (integer, optional): The Id of the machine a Robot is hosted on.
  Name (string): A custom name for the robot.stringMax. Length:19.
  Username (string): The machine username. If the user is under a domain, you are required to also specify it in a DOMAIN\username format. Note: You must use short domain names, such as desktop\administrator and NOT desktop.local/administrator.stringMax. Length:100.
  Description (string, optional): Used to add additional information about a robot in order to better identify it.stringMax. Length:500.
  Version (string, optional): The Robot's Version.
  Type (string): The Robot type. = ['NonProduction', 'Attended', 'Unattended', 'Development']stringEnum:"NonProduction", "Attended", "Unattended", "Development".
  HostingType (string): The Robot hosting type (Standard / Floating). = ['Standard', 'Floating']stringEnum:"Standard", "Floating".
  Password (string, optional): The Windows password associated with the machine username.stringMax. Length:100.
  CredentialType (string, optional): The robot credentials type (Default/ SmartCard) = ['Default', 'SmartCard']stringEnum:"Default", "SmartCard".
  Environments (Array[EnvironmentDto], optional): The collection of environments the robot is part of.
  RobotEnvironments (string, optional): The comma separated textual representation of environment names the robot is part of.
  ExecutionSettings (inline_model_4, optional): A collection of key value pairs containing execution settings for this robot.
  Id (integer, optional)
}
inline_model_4 {}
```

รูปที่ 3.66 รูปแบบ Response ของ Get license

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ 62 ปรึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อจัดเตรียม API ที่จำเป็นต่อการใช้งานใน Value per robot feature เสร็จสมบูรณ์แล้ว ผู้พัฒนาได้เริ่มพัฒนาโดยสร้างเป็น method หลัก ๆ ดังนี้

- Method getRobots() ดังรูปที่ 3.66 ทำการเรียกใช้งาน Get license API ในการทำ Authentication จากนั้นจึงทำการเรียกใช้ Get robots API โดยระบุ organization unit's ID ของ endpoint ที่ผู้ใช้งานทำการเลือก และเมื่อได้รับข้อมูลหุ่นยนต์ใน organization unit นั้น ๆ แล้ว จึงสร้างตัวแปร robots ในเก็บค่า robot's value เพื่อดำเนินการต่อไป

```
async getRobots(){
  await this.apiService.getCurrentUserUIProd().toPromise();
  let resrobots = await this.apiService.getRobots(this.orgUnitEndPointSelected["Id"]).toPromise();
  this.robots = this.robots.concat(resrobots["value"] as Array<IRobot>);
}
```

รูปที่ 3.67 Method getRobots ()

- Method addRobotValue() ดังรูปที่ 3.67 เป็น method ที่ทำหน้าที่เมื่อผู้ใช้งานทำการเพิ่มหุ่นยนต์ใน selected table โดยระบบจะทำการตรวจสอบว่าจำนวนช่องที่เพิ่มต้องไม่เกินจำนวนหุ่นยนต์ทั้งหมดที่มีใน organization unit นั้นจึงจะทำการเพิ่มหุ่นยนต์ได้

```
addRobotValue(){
  if(this.defaultPerRobot.length<this.robots.length){
    let robot :IRobotValue = {} as IRobotValue;
    robot.Id = this.generateId;
    robot.ValueType = this.asset.ValueType;
    robot.RobotId = this.nonSelectRobot.Id;
    robot.RobotName = this.nonSelectRobot.Name;
    this.defaultPerRobot.push(robot)
    this.robotSelected[robot.Id] = {} as IRobot;
    this.robotSelected[robot.Id].Id = robot.RobotId;
    this.robotSelected[robot.Id].Name = robot.RobotName;
    this.robotSelected[robot.Id]["IsRobotAlreadyExist"] = false;
    this.generateId = this.generateId-1;
    this.updateIsCanNextStage("False",this.asset.Name);
  }
}
```

รูปที่ 3.68 Method addRobotValue()

- Method deleteRobotValue() ดังรูปที่ 3.68 เป็น method ที่ทำหน้าที่เมื่อผู้ใช้งานทำการลบหุ่นยนต์ใน selected table โดยระบบจะทำการตรวจสอบว่าจำนวนช่องของหุ่นยนต์ที่มีอยู่ต้องมีมากกว่า 1 ช่องจึงจะทำการลบหุ่นยนต์ได้

```

deleteRobotValue(valuePerRobotToDelete: IRobotValue){
  if(this.defaultPerRobot.length>1){
    for(let valuePerRobot of this.defaultPerRobot){
      if(valuePerRobot.Id == valuePerRobotToDelete.Id){
        this.defaultPerRobot.splice(this.defaultPerRobot.indexOf(valuePerRobotToDelete),1);
        delete this.robotSelected [valuePerRobotToDelete.Id];
      }
    }
  }
  this.checkRobotRedundant();
  this.checkIfRobotHaveValueOrNot();
}

```

รูปที่ 3.69 Method deleteRobotValue()

- Method changeRobot() ดังรูปที่ 3.69 เป็น method ที่ทำหน้าที่เมื่อผู้ใช้งานทำการเปลี่ยนชื่อหุ่นยนต์ จากนั้นระบบจะทำการตรวจสอบโดยการเรียกใช้งาน method checkRobotRedundant() และ checkIfRobotHaveValueOrNot()

```

changeRobot (valuePerRobot:IRobotValue, robotSelected:IRobot){
  this.updateIsCanNextStage("True",this.asset.Name);
  valuePerRobot.RobotId = JSON.parse(JSON.stringify(robotSelected.Id));
  valuePerRobot.RobotName = JSON.parse(JSON.stringify(robotSelected.Name));
  this.checkRobotRedundant();
  this.checkIfRobotHaveValueOrNot();
}

```

รูปที่ 3.70 Method changeRobot()

- ในการตรวจสอบความถูกต้องของวิธีการและเพื่อให้สอดคล้องกับนโยบายขององค์กร ผู้พัฒนาจึงมีการตรวจสอบความถูกต้องของวิธีการดำเนินการเพิ่มหุ่นยนต์ 2 กรณี ดังรูปที่ 3.70 โดยmethod checkRobotRedundant() จะทำการตรวจสอบว่าหุ่นยนต์ที่ผู้ใช้ทำการเลือกต้องไม่ซ้ำกับหุ่นยนต์ทำการเลือกก่อนหน้า และ method checkIfRobotHaveValueOrNot() จะทำการตรวจสอบว่าหุ่นยนต์ที่ผู้ใช้ทำการเลือกต้องทำการใส่ค่าของ Asset ให้ครบถ้วนทุกช่อง

```

checkRobotRedundant(){
  let valuePerRobotGroupByName = this.defaultPerRobot.reduce((valuesPerRobot,valuePerRobot)={
    valuesPerRobot[valuePerRobot.RobotId] = valuesPerRobot[valuePerRobot.RobotId] + 1 || 1;
    return valuesPerRobot
  },{});
  this.updateIsCanNextStage("True",this.asset.Name);
  for(let valuePerRobot of this.defaultPerRobot){
    if(valuePerRobotGroupByName[valuePerRobot.RobotId]>1){
      if(valuePerRobot.RobotId!==-99){
        this.robotSelected [valuePerRobot.Id] ["IsRobotAlreadyExist"] = true;
        this.updateIsCanNextStage("False",this.asset.Name);
      }
    }
    else{
      this.robotSelected [valuePerRobot.Id] ["IsRobotAlreadyExist"] = false;
    }
  }
}

checkIfRobotHaveValueOrNot(){
  for(let valuePerRobot of this.defaultPerRobot){
    if(valuePerRobot.RobotId === -99){
      this.updateIsCanNextStage("False",this.asset.Name);
      break
    }
  }
}

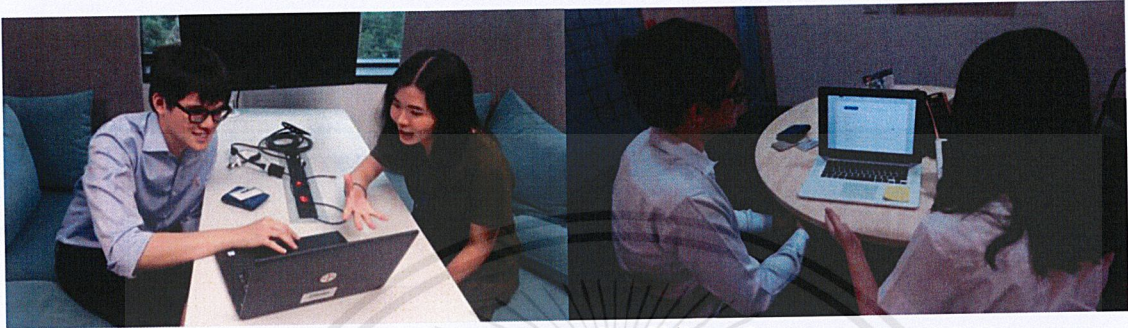
```

รูปที่ 3.71 Method checkRobotRedundant() & checkIfRobotHaveValueOrNot()

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ 64 รัชศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.1.6 ทดสอบการทำงานของระบบ

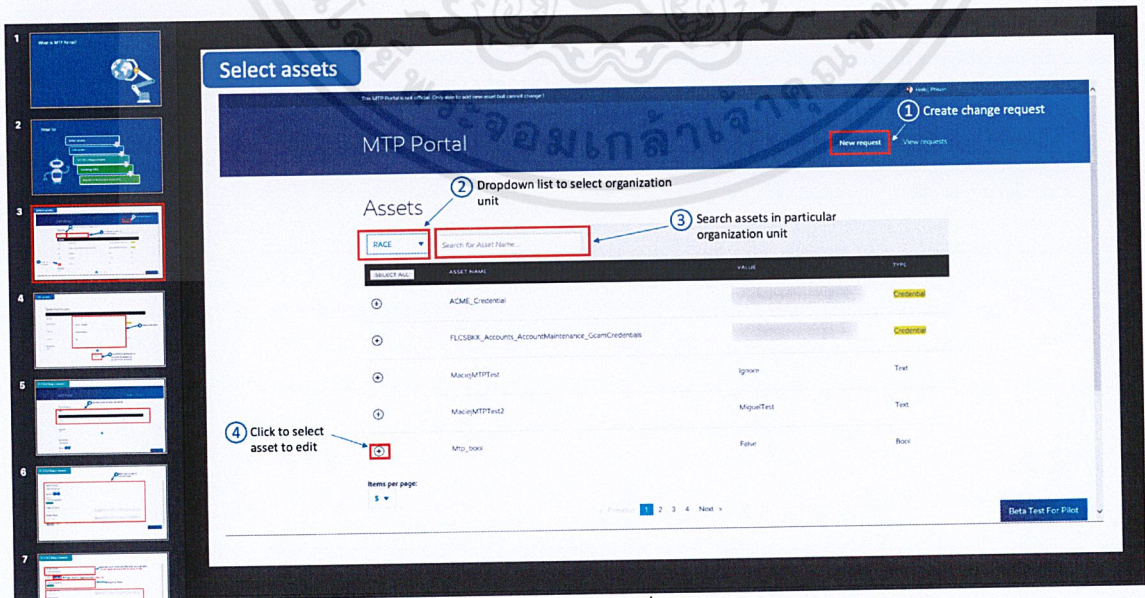
ผู้พัฒนาได้ทำการทดสอบการใช้งานของระบบตลอดช่วงระยะเวลาของการพัฒนา โดยการทดสอบด้วยตนเอง ทดสอบพร้อมกับเพื่อนร่วมงาน และให้พนักงานภายในแผนกรวมถึงผู้ใช้งานจริงแผนกต่าง ๆ ทำการทดสอบ ดังรูปที่ 3.71 โดยผลจากการทดสอบในแต่ละครั้ง ผู้พัฒนาจะนำผลการทดสอบมาวิเคราะห์เพื่อพัฒนา และปรับปรุงการทำงานให้มีประสิทธิภาพมากยิ่งขึ้นต่อไป



รูปที่ 3.72 การทดสอบการทำงานของระบบโดยผู้ใช้งานจริงแผนก Fuels & lubes

### 3.1.7 นำระบบไปใช้งานจริง

หลังจากการทดสอบการทำงานเรียบร้อยแล้ว และได้รับผลลัพธ์ที่น่าพึงพอใจ ผู้พัฒนาและผู้ร่วมพัฒนาจึงได้ทำการปรึกษากับพนักงานที่ปรึกษา และ Product owner (PO) จากนั้นจึงทำการ deploy โดย TFS เพื่อให้ระบบอยู่บนสถานะพร้อมใช้งานตลอดเวลา นอกจากนี้ผู้พัฒนายังได้เขียนเอกสารเกี่ยวกับการใช้งานระบบที่พัฒนาขึ้นโดยมีลำดับขั้นตอนอย่างละเอียด และเตรียมเอกสารข้อมูลเกี่ยวกับการพัฒนาเพื่อส่งต่อให้ผู้พัฒนาคนต่อไป ดังรูปที่ 3.72 และทำ Knowledge Transfer เพื่อส่งต่อความรู้ให้พนักงานในทีมที่รับผิดชอบดูแลงานนี้ต่อไป



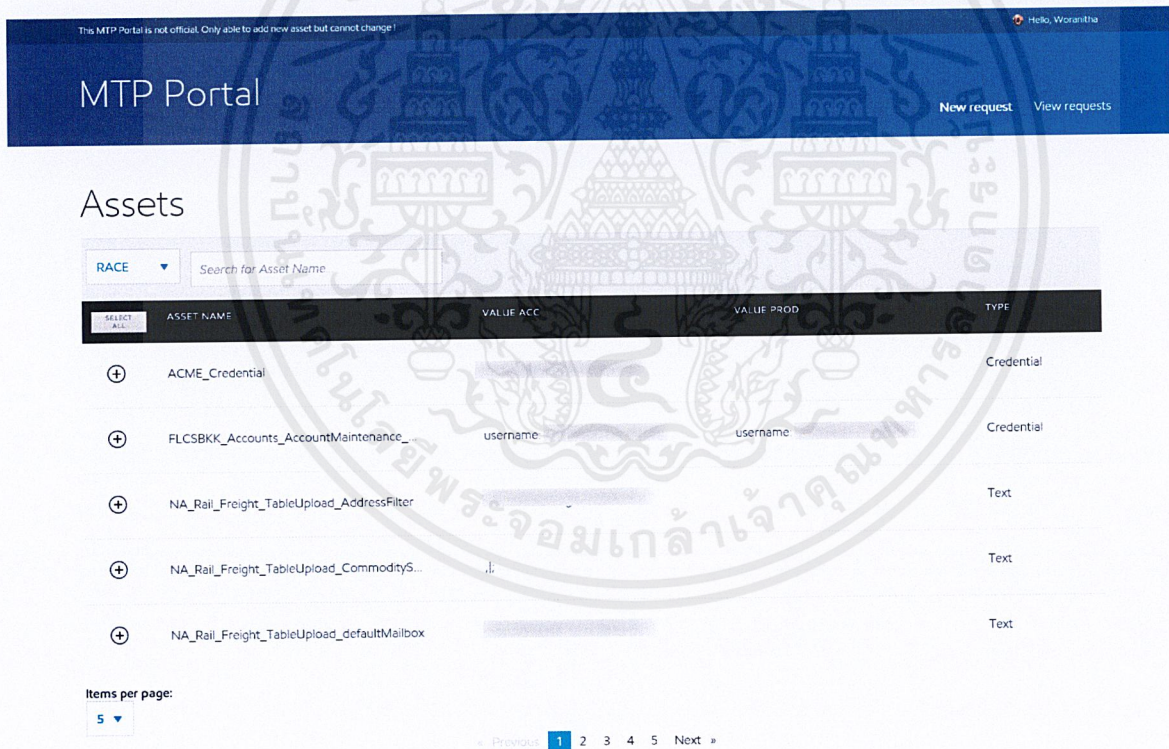
รูปที่ 3.73 ตัวอย่างเอกสารข้อมูลเกี่ยวกับการใช้งานระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ 65 ศึกษานี้เท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 4

### ผลการพัฒนา

ในส่วนของการพัฒนาระบบ ผู้พัฒนาได้วัดประสิทธิผลของการพัฒนาระบบซอฟต์แวร์อัตโนมัติเพื่อบริหารจัดการอาร์พีเอ โดยจากผลการทดสอบการใช้งานระบบพบว่า ระบบซอฟต์แวร์อัตโนมัติเพื่อบริหารจัดการอาร์พีเอสามารถดำเนินงานโดยลดขั้นตอน และทำการ automate ทำให้สามารถลดเวลาของกระบวนการโดยรวมถึง 30 นาทีต่อการนำขึ้น production 1 ครั้ง นอกจากนี้ระบบยังสามารถเชื่อมต่อกับระบบอื่น ๆ ได้แก่ UiPath orchestrator และ ITSM API เพื่อทำงานได้เป็นอย่างดี โดยขั้นตอนการทำงานในการนำขึ้น production ของระบบซอฟต์แวร์อัตโนมัติเพื่อบริหารจัดการอาร์พีเอได้แสดง ดังรูปที่ 4.1 ถึงรูปที่ 4.34



รูปที่ 4.1 ตาราง Assets selection

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ 66 รัชศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SELECT ALL	ASSET NAME	VALUE ACC	VALUE PROD	TYPE
+	RACE_MTPPortalExportOil_AccountantName	Grace W.	Grace W.	Text

รูปที่ 4.2 ผู้ใช้งานทำการเลือกชื่อ organization

### Assets

SELECT ALL	ASSET NAME	VALUE ACC	VALUE PROD	TYPE
+	RACE_MTPPortalExportOil_AccountantName	Grace W.	Grace W.	Text
+	RACE_MTPPortalExportOil_ConfigAccount	[Value per robot]		Text
+	RACE_MTPPortalExportOil_InboxAccount	username	username	Credential
+	RACE_MTPPortalExportOil_InfolsChanged	True	True	Bool
+	RACE_MTPPortalExportOil_Name	Alexandra H.	Alexandra H.	Text
+	RACE_MTPPortalExportOil_StockCount	64	64	Integer

รูปที่ 4.3 ช่องเพื่อทำการค้นหา

### Assets

SELECT ALL	ASSET NAME	VALUE ACC	VALUE PROD	TYPE
+	RACE_MTPPortalExportOil_AccountantName	Grace W.	Grace W.	Text
+	RACE_MTPPortalExportOil_ConfigAccount	[Value per robot]		Text
+	RACE_MTPPortalExportOil_InboxAccount	username	username	Credential
+	RACE_MTPPortalExportOil_InfolsChanged	True	True	Bool
+	RACE_MTPPortalExportOil_Name	Alexandra H.	Alexandra H.	Text
+	RACE_MTPPortalExportOil_StockCount	64	64	Integer

รูปที่ 4.4 การแสดงค่าโดยย่อหากมีจำนวนคำที่ยาวเกินไป และสามารถขยายดูคำเต็มได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## Assets

RACE		RACE_MTPPortalExportOil		
SELECT ALL	ASSET NAME	VALUE ACC	VALUE PROD	TYPE
<input checked="" type="checkbox"/>	RACE_MTPPortalExportOil_AccountantName	Grace W.	Grace W.	Text
<input type="checkbox"/>	RACE_MTPPortalExportOil_ConfigAccount	[Value per robot]		Text
		<input type="button" value="Hide"/>		
		Patom_Bot : Patom S.		
		Varis_Test : Varis K.		
<input checked="" type="checkbox"/>	RACE_MTPPortalExportOil_InboxAccount	username: <input type="text"/>	username: <input type="text"/>	Credential
<input type="checkbox"/>	RACE_MTPPortalExportOil_InfolsChanged	True	True	Bool
<input type="checkbox"/>	RACE_MTPPortalExportOil_Name	Alexandra H.	Alexandra H.	Text
<input type="checkbox"/>	RACE_MTPPortalExportOil_StockCount	64	64	Integer

รูปที่ 4.5 ผู้ใช้งานทำการเลือก Asset ที่ต้องการเพิ่มหรือเปลี่ยนแปลง

## Assets

RACE		RACE_MTPPortalExportOil		
CLEAR SELECTORS	ASSET NAME	VALUE ACC	VALUE PROD	TYPE
<input checked="" type="checkbox"/>	RACE_MTPPortalExportOil_AccountantName	Grace W.	Grace W.	Text
<input checked="" type="checkbox"/>	RACE_MTPPortalExportOil_ConfigAccount	[Value per robot]		Text
		<input type="button" value="Show"/>		
<input checked="" type="checkbox"/>	RACE_MTPPortalExportOil_InboxAccount	username: <input type="text"/>	username: <input type="text"/>	Credential
<input checked="" type="checkbox"/>	RACE_MTPPortalExportOil_InfolsChanged	True	True	Bool
<input checked="" type="checkbox"/>	RACE_MTPPortalExportOil_Name	Alexandra H.	Alexandra H.	Text
<input checked="" type="checkbox"/>	RACE_MTPPortalExportOil_StockCount	64	64	Integer

รูปที่ 4.6 ผู้ใช้งานทำการเลือก Asset ทั้งหมด โดยการกด select all button

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ 68 ศึกษเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## Selected Assets

ASSET NAME	VALUE	TYPE	ENVIRONMENT
RACE_MTPPortalExportOil_Acc...	<input type="text" value="Phison K."/>	Text	Production
RACE_MTPPortalExportOil_Inbo...	Username: <input type="text" value=""/>	Credential	Production
RACE_MTPPortalExportOil_Conf...	<input type="button" value="+ Add"/> <input type="text" value="HOERPADFTN008"/> <input type="text" value="Neramit S."/> <input type="text" value="AZ-RPASANTEST001"/> <input type="text" value="Supawit P."/>	Text	Acceptance
RACE_MTPPortalExportOil_Infol...	<input type="radio"/> True <input checked="" type="radio"/> False	Bool	Production
RACE_MTPPortalExportOil_Name	<input type="text" value="John R."/>	Text	Production
RACE_MTPPortalExportOil_Stoc...	<input type="text" value="98"/>	Integer	Production

รูปที่ 4.7 ผู้ใช้งานทำการแก้ไขค่าของ Assets

- BSA
- CHCSCUR
- CHLDGBL
- CHSCMON
- CSMS
- FLCSBKK
- RACE

## Selected Assets

ASSET NAME	VALUE	TYPE	ENVIRONMENT
RACE_MTPPortalExportOil_Acc...	<input type="text" value="Grace W."/>	Text	Production
RACE_MTPPortalExportOil_Conf...	<input type="button" value="+ Add"/> <input type="text" value="HOERPADFTN013_sa"/> <input type="text" value="Petom S."/> <input type="text" value="HOERPADFTN013_sa"/> <input type="text" value="Varis K."/>	Text	Acceptance

รูปที่ 4.8 ผู้ใช้งานทำการเลือกชื่อ organization ที่ต้องการเพิ่ม Assets

RACE_MTPPortalExportOil_Name	<input type="text" value="John R."/>	Text	Production
------------------------------	--------------------------------------	------	------------

Items per page:

« Previous **1** 2 Next »

รูปที่ 4.9 กด Next button เพื่อไปยังหน้า Summarize page

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## Summary

### Assets

ASSET NAME	VALUE	TYPE	ENVIRONMENT
RACE_MTPPortalExportOil_Acc...	Phison K.	Text	Production
RACE_MTPPortalExportOil_Conf...	[Value per robot]	Text	Acceptance
<a href="#">Show ▶</a>			
RACE_MTPPortalExportOil_Inbo...	username: [redacted]	Credential	Production
RACE_MTPPortalExportOil_Infol...	False	Bool	Production
RACE_MTPPortalExportOil_Name	John R.	Text	Production

Items per page:

5 ▼

Previous 1 2 Next ▶

### Request Summary

RACE- AssetChange

Approver



Approver 1:

Select a person

Change coordinator

Select a group

Select a person

Change manager

Select a group

Select a person

[Submit](#)

[Back](#)

รูปที่ 4.10 หน้า Summary Page โดยตัดช่องกรอกที่ไม่จำเป็นออก เหลือเพียงช่องกรอกที่จำเป็นเท่านั้น

### Request Summary

RACE- AssetChange-MTPPortalExportOilTest

รูปที่ 4.11 ผู้ใช้งานทำการกรอก Request summary

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ 70 รัชศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Approver

Approver 1 :

suru
Aranzazu Guisuraga
Chad Sura
Claudia Y Katsurayama
Dario G Sura
George P Emsurak

รูปที่ 4.12 ผู้ใช้งานทำการเลือก Approvers

Approver

Approver 1 :

Phisonk
---------

Invalid contact!

รูปที่ 4.13 ระบบแสดงคำเตือนหากไม่เจอ Approver ที่ระบุ

Approver

Approver 1 :

Select a person

\* Requester can't be an approver

รูปที่ 4.14 ระบบแสดงคำเตือนหากผู้อนุมัติเป็นคนเดียวกับผู้ร้องขอ

Approver

Approver 1 :

Phison Khankhang
------------------

Valid contact

รูปที่ 4.15 ระบบแสดงข้อความ Valid contact หาก Approver ที่ระบุถูกต้อง

Change coordinator

Robotics
Financial Robotics Process Automation (Applications-Financial)
Robotics Automation Center of Expertise (Applications-Technical Services)
Robotics Automation CoE-Celonis (Applications-Technical Services)

Select a person

Change manager

Select a group

Select a person

Submit

Back

รูปที่ 4.16 ผู้ใช้งานทำการเลือก IT Console group

**Change coordinator**

Robotics Automation Center of Expertise  
Valid group

**Change manager**

Robotics Automation Center of Expertise  
Valid group

Submit Back

- Alex Weber
- Amanda Mendes Simoni
- Bruno R Sella
- Chaloemchai (Aum) Thanetanukun
- Felipe M Santos

รูปที่ 4.17 ผู้ใช้งานทำการเลือก Change coordinator

**Approver**

Approver 1 :  
Phison Khankhang  
Valid contact

**Change coordinator**

Robotics Automation Center of Expertise  
Valid group

Select a person  
\* Change coordinator can't be an approver

รูปที่ 4.18 ระบบแสดงคำเตือนหาก Change coordinator เป็นคนเดียวกับผู้อนุมัติ

**Request Summary**

RACE- AssetChange-MTPPortalExportOilTest

**Approver**

Approver 1 :  
Phison Khankhang  
Valid contact

**Change coordinator**

Robotics Automation Center of Expertise  
Valid group

Chaloemchai (Aum) Thanetanukun  
Valid user

**Change manager**

Robotics Automation Center of Expertise  
Valid group

Chaloemchai (Aum) Thanetanukun  
Valid user

Submit Back

รูปที่ 4.19 หน้า Summary page ที่ผู้ใช้งานทำการกรอกโดยสมบูรณ์แล้ว

**Request Summary**

RACE- AssetChange-MTPPortalExportOilTest

**Approver**

Approver 1 :  
Phison Khankhang  
Valid contact

**Change coordinator**

Create CRQ ticket?  
Are you sure to proceed?

Create CRQ ticket Stay on this page

รูปที่ 4.20 หน้าต่างเพื่อให้ผู้ใช้งานทำการยืนยันว่าต้องการสร้าง CRQ Ticket เลยหรือไม่

# <MTP\_Portal>RACE- AssetChange- MTPPortalExportOilTest

Scheduled Dec 12 2019 at 01:29 PM Class Expedited  
End Dec 14 2019 at 01:29 PM Reason Upgrade

LOW RISK TAG

Scheduled For Approval

Scheduled For Approval  
1 Pending Approvals on Review

- Notes
- Work Details
- Approvals
- Tasks
- Relationships

**Notes**  
Name: RACE\_MTPPortalExportOil\_AccountantName  
Value: Phison K  
ValueType: Text

Name: RACE\_MTPPortalExportOil\_ConfigAccount  
Value: [Value per robot]...  
[Read More](#)

Last Modified Dec 12 2019 at 01:29 PM

Last Modified By Woranitha Hueghansombat (ap/whuegha)

## Work Details (2)

New Work Detail

Status Transition: Draft > Request For Authorization > Request For Change > Planning In Progress > Scheduled For Review > Scheduled For Approval > Scheduled > Implementation In Progress > Completed > Closed

Woranitha Hueghansombat

Dec 12 at 01:29 PM

### รูปที่ 4.21 ระบบทำการสร้าง CRQ ticket อัตโนมัติ

- Notes
- Work Details
- Approvals
- Tasks
- Relationships

Last Modified Dec 12 2019 at 01:29 PM

Last Modified By Woranitha Hueghansombat (ap/whuegha)

**Notes**  
Name: RACE\_MTPPortalExportOil\_AccountantName  
Value: Phison K  
ValueType: Text

Name: RACE\_MTPPortalExportOil\_ConfigAccount  
Value: [Value per robot]  
ValueType: Text

Name: RACE\_MTPPortalExportOil\_InboxAccount  
Value: username  
ValueType: Credential

Name: RACE\_MTPPortalExportOil\_InfosChanged  
Value: False  
ValueType: Bool

Name: RACE\_MTPPortalExportOil\_Name  
Value: John R  
ValueType: Text

Name: RACE\_MTPPortalExportOil\_StockCount  
Value: 98  
ValueType: Integer  
[Read Less](#)

**Coordinator**  
 Chaloechai (Aum) Thanetanukun  
Robotics Automation Center of Expertise

**Change Manager**  
 Chaloechai (Aum) Thanetanukun

- Documents**
- Backout Plan (0) [View All](#)  
How to restore a system's previous state
  - Install Plan (0) [View All](#)  
How to install/perform this change
  - Test Plan (0) [View All](#)  
How to assure the change was successful

**Coordinator**  
 Chaloechai (Aum) Thanetanukun  
Robotics Automation Center of Expertise

**Change Manager**  
 Chaloechai (Aum) Thanetanukun

- Documents**
- Backout Plan (0) [View All](#)  
How to restore a system's previous state
  - Install Plan (0) [View All](#)  
How to install/perform this change
  - Test Plan (0) [View All](#)  
How to assure the change was successful

► Categorization

► Additional Details

### รูปที่ 4.22 ข้อมูลของการเพิ่มหรือเปลี่ยนแปลง Assets ถูกแสดงใน CRQ ticket note

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ 73 ศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

----- Asset -----

Name: RACE\_MTPPortalExportOil\_AccountantName

Value: Phison K.

ValueType: Text

Name: RACE\_MTPPortalExportOil\_ConfigAccount

Value: [Value per robot]

ValueType: Text

Name: RACE\_MTPPortalExportOil\_InboxAccount

Value: username: [REDACTED]

ValueType: Credential

Name: RACE\_MTPPortalExportOil\_InfolsChanged

Value: False

ValueType: Bool

Name: RACE\_MTPPortalExportOil\_Name

Value: John R.

ValueType: Text

Name: RACE\_MTPPortalExportOil\_StockCount

Value: 98

ValueType: Integer

รูปที่ 4.23 ข้อมูลของการเพิ่มหรือเปลี่ยนแปลง Assets ถูกแสดงใน CRQ ticket work detail

----- Details for MTP Portal -----

```
[{"Name": "RACE_MTPPortalExportOil_AccountantName", "CanBeDeleted": true, "ValueScope": "Global", "ValueType": "Text", "Value": "Phison K.", "StringValue": "Grace W.", "BoolValue": false, "IntValue": 0, "CredentialUsername": "", "CredentialPassword": "", "Id": 6003, "KeyValueList": [], "RobotValues": [], "environment": "Production", "isExpanded": false, "isChecked": true}, {"Name": "RACE_MTPPortalExportOil_ConfigAccount", "CanBeDeleted": true, "ValueScope": "PerRobot", "ValueType": "Text", "Value": "[Value per robot]", "StringValue": null, "BoolValue": false, "IntValue": 0, "CredentialUsername": null, "CredentialPassword": null, "Id": 6037, "KeyValueList": [], "RobotValues": [{"RobotId": 27, "RobotName": "HOERPADFT-N008", "KeyTrail": "****0e2c1fc", "ValueType": "Text", "StringValue": "Patom S.", "BoolValue": false, "IntValue": 0, "Value": "Neramit S.", "CredentialUsername": "", "CredentialPassword": "", "Id": 6337, "KeyValueList": []}, {"RobotId": 29, "RobotName": "AZ-RPASANTEST001", "KeyTrail": "****766c027", "ValueType": "Text", "StringValue": "Varis K.", "BoolValue": false, "IntValue": 0, "Value": "Supawit P.", "CredentialUsername": "", "CredentialPassword": "", "Id": 6338, "KeyValueList": []}], "environment": "Acceptance", "isExpanded": true, "isChecked": true}, {"Name": "RACE_MTPPortalExportOil_InboxAccount", "CanBeDeleted": true, "ValueScope": "Global", "ValueType": "Credential", "Value": "userna", "StringValue": "", "BoolValue": false, "IntValue": 0, "CredentialUsername": "", "CredentialPassword": "", "Id": 6004, "KeyValueList": [], "RobotValues": [], "environment": "Production", "isExpanded": false, "isChecked": true}, {"Name": "RACE_MTPPortalExportOil_IsChanged", "CanBeDeleted": true, "ValueScope": "Global", "ValueType": "Bool", "Value": "False", "StringValue": "", "BoolValue": true, "IntValue": 0, "CredentialUsername": "", "CredentialPassword": "", "Id": 6005, "KeyValueList": [], "RobotValues": [], "environment": "Production", "isExpanded": false, "isChecked": true}, {"Name": "RACE_MTPPortalExportOil_Name", "CanBeDeleted": true, "ValueScope": "Global", "ValueType": "Text", "Value": "John R.", "StringValue": "Alexandra H.", "BoolValue": false, "IntValue": 0, "CredentialUsername": "", "CredentialPassword": "", "Id": 6006, "KeyValueList": [], "RobotValues": [], "environment": "Production", "isExpanded": false, "isChecked": true}, {"Name": "RACE_MTPPortalExportOil_StockCount", "CanBeDeleted": true, "ValueScope": "Global", "ValueType": "Integer", "Value": 98, "StringValue": "", "BoolValue": false, "IntValue": 64, "CredentialUsername": "", "CredentialPassword": "", "Id": 6007, "KeyValueList": [], "RobotValues": [], "environment": "Production", "isExpanded": false, "isChecked": true}], {"DisplayName": "RACE", "Id": 6}]
```

[Read Less](#)

Woranitha Hueghansombat

Dec 12 at 01:29 PM

รูปที่ 4.24 ข้อมูลของ Assets ในส่วนการดำเนินงานของระบบ

## ✓ Approvals (1)

Log

▼ Implementation

Phison Khankhang

PENDING

Dec 12 at 01:29 PM

Hide past approvals

รูปที่ 4.25 ส่วนของผู้อนุมัติใน CRQ ticket

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ 75 ศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## Ticket List

5  
Critical/High  
My Group's

14  
Unassigned  
My Group's

17  
Incidents  
My Group's

60  
Work Orders  
My Group's

12  
Change Requests  
My Group's

My Queue
Group Queue
Closed Tickets

Type ▾
Priority ▾
Assignee ▾
Tags ▾
Product ▾
Filters

Search Summary 🔍

TYPE	TICKET	SUMMARY	STATUS	ASSIGNEE	SUBMITTED	
CRQ	CRQ000000815111	<MTP_Portal>RACE- AssetChange-MTPPortalExportOilTest + TAG	Scheduled...	Chaloemchai (...)	Dec 12, 01:29 PM	🔗 ⋮
WO	WO0000015654387	[BULK] Please add these groups to RPA.ORCHESTRATOR ALL UG + TAG	Assigned	Unassigned	Dec 06, 12:38 AM	🔗 ⋮
WO	WO0000015654478	test + TAG	Assigned	Jonasta J Dos...	Dec 05, 10:49 PM	🔗 ⋮
WO	WO0000015650047	Server SW: Back Office Robot 2018 4 4 Modules Selected + TAG	Assigned	Unassigned	Nov 29, 06:52 PM	🔗 ⋮
WO	WO0000015650483	Please connect RPAFLCSIP042_TEST3 HOERPDPFO165 to 2017 + TAG	Assigned	Bruno R Sella	Nov 29, 06:32 PM	🔗 ⋮
WO	WO0000015650176	FLCSIND_Account_Revactivation_Fuels_NASA + TAG	Assigned	Surawadee Ta...	Nov 29, 06:16 PM	🔗 ⋮

รูปที่ 4.26 การแสดง Ticket ทั้งหมดของ organization ใน IT Console

This MTP Portal is not official.
👤 Hello, Worathna

# MTP Portal

New request   View requests

## Requests

Robotics Automation Center of Expertise ▾

Search for Asset Name

TICKET NO.	SUMMARY	ASSIGNEE	STATUS	
<a href="#">CRQ000000815710</a>	<MTP_Portal>RACE- AssetChange-4 approver	Phison Khankhang	Pending for approval	🗑️
<a href="#">CRQ000000815709</a>	<MTP_Portal>RACE- AssetChange-MTPPortalExportOilTest	Chaloemchai (Aum) Thanetanakun	Pending for approval	🗑️
<a href="#">CRQ000000815708</a>	<MTP_Portal>RACE- AssetChange-Test1	Chaloemchai (Aum) Thanetanakun	Approved	🗑️
<a href="#">CRQ000000815413</a>	<MTP_Portal>RACE- AssetChange-test cancel	Phison Khankhang	Approved	🗑️
<a href="#">CRQ000000802616</a>	<MTP_Portal>RACE- AssetChange	Surawadee Tengtorikit	Rejected	🗑️

รูปที่ 4.27 หน้า View all tickets Page

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### Approvals

**APPROVED**



Phison Khankhang

### Notes

Name: RACE\_MTPPortalExportOil\_AccountantName  
 Value: Phison K.  
 ValueType: Text

Name: RACE\_MTPPortalExportOil\_ConfigAccount  
 Value: [Value per robot]  
 ValueType: Text

### Assets

ASSET NAME	VALUE	TYPE
RACE_MTPPortalExportOil_AccountantName	Phison K.	Text
RACE_MTPPortalExportOil_ConfigAccount	[Value per robot] <a href="#">Show ▶</a>	Text
RACE_MTPPortalExportOil_InboxAccount	username: Password:	Credential
RACE_MTPPortalExportOil_InfosChanged	False	Bool
RACE_MTPPortalExportOil_Name	John R.	Text
RACE_MTPPortalExportOil_StockCount	98	Integer

[Update in orchestrator](#) [Back](#)

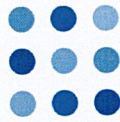
รูปที่ 4.28 หน้า Ticket Details page

### Assets

ASSET NAME	VALUE	TYPE
RACE_MTPPortalExportOil_AccountantName	Phison K.	Text
RACE_MTPPortalExportOil_ConfigAccount	[Value per robot] <a href="#">Show ▶</a>	Text
RACE_MTPPortalExportOil_InboxAccount	username: <input type="text"/> Password: <input type="password"/>	Credential
RACE_MTPPortalExportOil_InfosChanged	False	Bool
RACE_MTPPortalExportOil_Name	John R.	Text
RACE_MTPPortalExportOil_StockCount	98	Integer

[Update in orchestrator](#) [Back](#)

รูปที่ 4.29 ผู้ใช้งานทำการเติม credential password



Orchestrator is updated successfully. Closing CRQ ticket ...

รูปที่ 4.30 หน้า loading พร้อมแสดงขั้นตอนการทำงานในการเพิ่มหรือเปลี่ยนแปลงAssets

This MTP Portal is not official. Only able to add new asset but cannot change!

Hello, Woranit.na

# MTP Portal

New request View requests

CRQ000000815111 COMPLETED MOVING complete updating in orchestrator & This ticket is closed. Close

### Approvals

APPROVED

Phison Khankhang

### Notes

Name: RACE\_MTPPortalExportOil\_AccountantName  
Value: Phison K.  
ValueType: Text

Name: RACE\_MTPPortalExportOil\_ConfigAccount  
Value: [Value per robot]  
ValueType: Text

### Assets

ASSET NAME	VALUE	TYPE
RACE_MTPPortalExportOil_AccountantName	Phison K	Text
RACE_MTPPortalExportOil_ConfigAccount	[Value per robot]	Text

Show ▶

รูปที่ 4.31 หน้าต่างแสดงข้อความเมื่อระบบอัปเดต Assets เรียบร้อย และปิด CRQ ticket อัตโนมัติ

## UiPath Assets

NAME	TYPE	VALUE
RACE_MTPPortalExportOil_InboxAccount	Credential	Username
RACE_MTPPortalExportOil_InfoIsChanged	Bool	True
RACE_MTPPortalExportOil_Name	Text	Alexandra H.
RACE_MTPPortalExportOil_StockCount	Integer	64
RACETestingBot_Email	Text	
RACETestingBot_Sap_List	Text	ISSO: AMB [NA Stripes Accept FU TAX];ISSO: APB [ASIA PACIFIC STRIPES Accept FU TAX]
renew	Integer	[Value per robot]
RP4DEV14	Credential	Username: RP4DEV14
VM_Initialization_Sap_List	Text	ISSO: AMB [NA Stripes Accept FU TAX];ISSO: APB [ASIA PACIFIC STRIPES Accept FU TAX]

Items: 10 Page: 2/2 19 Items

รูปที่ 4.32 UiPath Orchestrator ก่อนทำการอัปเดต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

NAME	TYPE	VALUE
BSA_PricingLetterGeneration_ColumnsMPCFile	Text	Ship-to Number;Customer Name;Customer Location;Error;Mobil Source;MOT;Product Name;Price Change;N
FLCSBKK_Accounts_AccountMaintenance_GcamCredentials	Credential	Username: [REDACTED]
MTP_Robot_credential	Credential	[Value per robot]
MTP_Robot_int	Integer	[Value per robot]
MTP_SingleValue_Test_OnlyInProd	Credential	Username: [REDACTED]
RACE_MTFPortalExportOil_AccountantName	Text	Phason K.
RACE_MTFPortalExportOil_ConfigAccount	Text	[Value per robot]
RACE_MTFPortalExportOil_InboxAccount	Credential	Username: [REDACTED]
RACE_MTFPortalExportOil_InfocChanged	Bool	False
RACE_MTFPortalExportOil_Name	Text	John R.
RACE_MTFPortalExportOil_StockCount	Integer	58
RACETestingBot_Email	Text	[REDACTED]
RACETestingBot_Sap_List	Text	ISSO AMB [NA-Stripes Accept FI/TAX];ISSO APB [ASIA PACIFIC STRIPES Accept FI/TAX]

รูปที่ 4.33 UiPath Orchestrator เมื่อทำการอัปเดตสำเร็จแล้ว

COMPONENT	USER	ACTION	OPERATION	TIME
Assets	whuegha	update	User whuegha updated asset RACE_MTFPortalExportOil_InfocChanged	4 minutes ago
Assets	whuegha	update	User whuegha updated asset RACE_MTFPortalExportOil_StockCount	4 minutes ago
Assets	whuegha	Create	User whuegha created asset RACE_MTFPortalExportOil_ConfigAccount	4 minutes ago
Assets	whuegha	update	User whuegha updated asset RACE_MTFPortalExportOil_InboxAccount	4 minutes ago
Assets	whuegha	update	User whuegha updated asset RACE_MTFPortalExportOil_Name	4 minutes ago
Assets	whuegha	update	User whuegha updated asset RACE_MTFPortalExportOil_AccountantName	4 minutes ago
Queues	twitlay	Delete	User twitlay deleted queue blank queue	12 hours ago
Queues	twitlay	Delete	User twitlay deleted queue second queue create	12 hours ago
Queues	twitlay	Delete	User twitlay deleted queue create first queue	12 hours ago
Queues	twitlay	Create	User twitlay created queue third queue	12 hours ago

รูปที่ 4.34 UiPath Orchestrator's audit log

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สรุปผลการพัฒนาและข้อเสนอแนะ

### 5.1 สรุปผลการพัฒนา

ผู้พัฒนาได้มีโอกาสเข้าร่วมโครงการสหกิจศึกษาที่ บริษัท เอ็กซอนโมบิล จำกัด ในแผนก Robotics automation center of expertise (RACE) โดยในขณะที่เข้าร่วมโครงการสหกิจศึกษาผู้พัฒนาได้รับมอบหมายให้พัฒนาระบบซอฟต์แวร์อัตโนมัติเพื่อบริหารจัดการอาร์พีเอ โดยได้รับความต้องการจากเจ้าของผลิตภัณฑ์และผู้ที่เกี่ยวข้องกับอาร์พีเอในแผนกต่าง ๆ ผู้พัฒนาได้ศึกษากระบวนการและระบบแบบก้าวรบรวมความต้องการ ร่วมวางแผนการพัฒนาโปรแกรมกับทีม พัฒนาระบบที่สร้างขึ้นใหม่ ทดสอบการทำงานของระบบ และนำระบบซอฟต์แวร์อัตโนมัติไปใช้งานจริง

เมื่อการดำเนินงานพัฒนาระบบซอฟต์แวร์อัตโนมัติเสร็จสิ้น พบว่าประสิทธิผลของการทำงานของระบบนั้นน่าพึงพอใจ ลดขั้นตอนและเวลาการทำงานได้ และระบบสามารถทำงานได้ตามความต้องการทางธุรกิจขององค์กร ตามที่ผู้พัฒนาได้รับมอบหมายความต้องการจากเจ้าของผลิตภัณฑ์และผู้ใช้งานจริง

### 5.2 ปัญหาที่พบและวิธีการแก้ไข

5.2.1 เนื่องจากแผนกที่ผู้พัฒนาอยู่มีการนำเสนอเป็นภาษาอังกฤษบ่อยครั้ง โดยมีการจัด Demo presentation ทาง Skype for business ทุก ๆ สองสัปดาห์ ซึ่งเป็นการประชุมเพื่อนำเสนองานของสมาชิกแต่ละคนที่ได้ทำในสปรินท์ที่ผ่านมา โดยมีลูกค้าจากทั่วโลกเข้าร่วมเฉลี่ย 40 คนในการรับชม โดยผู้นำเสนอแต่ละคนมีเวลานำเสนอเพียง 5 นาที นอกจากนี้ผู้พัฒนาต้องจัดการนำเสนอ Extra demo presentation ทุก ๆ วันศุกร์ให้กับ Product owner แต่เนื่องจากผู้พัฒนามีประสบการณ์การนำเสนองานไม่มากในตอนเริ่มต้น ทำให้เป็นการยากในการจัดการนำเสนอแต่ละครั้ง

วิธีแก้ไข : เนื่องจากใน Demo presentation ผู้พัฒนาต้องเล่าถึงสิ่งที่ได้พัฒนาไปในสองสัปดาห์ที่ผ่านมา, สิ่งที่จะพัฒนาสองสัปดาห์ข้างหน้า, ปัญหาที่เกิดขึ้นและการแก้ไข ซึ่งข้อมูลทั้งหมดมีเวลาพูดเพียง 5 นาที ทำให้ผู้พัฒนาต้องบริหารจัดการเวลาให้มีประสิทธิภาพ โดยแบ่งเวลาพัฒนาระบบ, ประชุมกับทีม และเวลาที่ใช้เตรียมตัวเพื่อการนำเสนองาน โดยผู้พัฒนาต้องเตรียมอุปกรณ์และสไลด์ในการนำเสนอ รวมถึงฝึกการนำเสนอให้ครบถ้วน กระชับที่สุด และให้เกิดความเคยชินกับภาษาอังกฤษ

5.2.2 เนื่องจาก บริษัท เอ็กซ์คอนโมบิล จำกัด เป็นบริษัทที่อยู่ในเครือเอ็กซ์คอนโมบิล ซึ่งมีขนาดใหญ่ ทำให้การทำงานต่าง ๆ ต้องคำนึงถึงความปลอดภัยของข้อมูลและระบบอยู่ตลอดเวลา ทำให้การทำงาน เช่น การขอซอฟต์แวร์ที่จำเป็นในการพัฒนาโปรแกรม ส่งผลกระทบต่อความล่าช้าในการทำงาน

วิธีการแก้ไข : ในช่วงก่อนการเริ่มทำงาน ผู้ดูแลได้ทำการวางแผนล่วงหน้าเกี่ยวกับโปรแกรมพื้นฐานเพื่อใช้ในการพัฒนา ทำให้เมื่อเข้าทำงานจริงผู้พัฒนาสามารถเริ่มเรียนรู้งานได้ทันทีโดยไม่ต้องทำเรื่องขอโปรแกรม และรอการอนุมัติเพื่อติดตั้งซึ่งใช้เวลาประมาณ 3-4 วัน แต่ระหว่างการดำเนินการงานผู้พัฒนาและผู้ร่วมพัฒนามีการวางแผนในการออกแบบแบบจำลองของระบบซอฟต์แวร์ที่กำลังพัฒนาขึ้น โดยจำเป็นต้องใช้ Macbook และโปรแกรมออกแบบ เช่น Adobe XD ผู้พัฒนาจึงทำเรื่องขอโปรแกรมก่อนการทำงานจริง 1 สัปดาห์ ทำให้อุปกรณ์และโปรแกรมพร้อมใช้งานทันทีเมื่อถึงเวลาใช้งาน ทำให้ความล่าช้าของการร้องขอไม่กระทบกับแผนงานที่ได้วางไว้

5.2.3 ภายในทีม RACE ซึ่งเป็นทีมที่ผู้พัฒนาทำงานประกอบด้วย ทีม Bangkok, Thailand และทีม Curitiba, Brazil ซึ่งทั้งสองประเทศมีเขตเวลาที่ต่างกันถึง 14 ชั่วโมง โดยในทุก ๆ สองสัปดาห์จะมีการประชุม 2 ครั้ง เริ่มในเวลา 19.00 น. - 21.00 น. ทำให้เกิดความเหนื่อยล้าในการทำงาน

วิธีแก้ไข : เนื่องจากบริษัทไม่มีการกำหนดเวลาในการเข้า-ออก และบริษัทตั้งอยู่ในเมืองซึ่งมีการจราจรหนาแน่นในชั่วโมงเร่งด่วน ทำให้วันที่มีประชุมผู้พัฒนาและพนักงานในทีมสามารถออกจากบริษัทก่อนเวลาได้ แต่ในทุก ๆ เข้าทีม RACE ยังคงมีการประชุม Daily stand-up กับ Product owner จากต่างประเทศในเวลา 10.00 น. ซึ่งไม่สามารถเลื่อนได้เนื่องจากเป็นเวลาที่เหมาะสมสำหรับ 2 เขตเวลา ทำให้ทุก ๆ คนในทีมต้องบริหารจัดการเวลาโดยไม่เหนื่อยล้าจนเกินไป

## เอกสารอ้างอิง

1. UiPath. 2018. **UiPath orchestrator's introduction**. สืบค้นเมื่อ 20 ธันวาคม 2562, จาก <https://docs.uipath.com/orchestrator/v2018.1/docs/introduction>
2. Permpol Thanapunnamas. 2019. **Visual Studio Code คืออะไร มารู้จักกับ VS Code ดีงามอย่างไร ทำอะไรได้บ้าง**. สืบค้นเมื่อ 20 ธันวาคม 2562, จาก <https://eleceasy.com/t/visual-studio-code-vs-code/2042api/overview/getting-started-with-aspnet-web-api/tutorial-your-first-web-api>
3. มานพ กองอุ้น. 2017. **Angular คืออะไร**. สืบค้นเมื่อ 20 ธันวาคม 2562, จาก <https://www.programmerthailand.com/tutorial/post/view/208/angular-คืออะไร>
4. Microsoft. 2019. **Azure DevOps documentation**. สืบค้นเมื่อ 21 ธันวาคม 2562, จาก <https://docs.microsoft.com/en-us/azure/devops/?view=azure-devops>
5. Microsoft. 2019. **Team Foundation Server is now called Azure DevOps Server**. สืบค้นเมื่อ 21 ธันวาคม 2562, จาก <https://docs.microsoft.com/en-us/azure/devops/server/tfs-is-now-azure-devops-server?view=azure-devops>
6. Salaspuk 2019. **Azure DevOps**. สืบค้นเมื่อ 21 ธันวาคม 2562, จาก <https://saladpuk.gitbook.io/learn/cloud/azure-devops>

## ประวัติผู้เขียน



ชื่อ - ชื่อสกุล

นางสาวรณิษฐา ฮึกหาญสมบัติ

รหัสนักศึกษา

59011166

คณะ

วิศวกรรมศาสตร์

ภาควิชา

วิศวกรรมคอมพิวเตอร์

สาขาวิชา

วิศวกรรมสารสนเทศ

วัน เดือน ปีเกิด

12 พฤศจิกายน 2540

สถานที่เกิด

จังหวัดกรุงเทพมหานคร

สถานที่อยู่ปัจจุบัน

119/46 หมู่ที่ 5 ตำบลสระกรวด อำเภอศรีเทพ จังหวัดเพชรบูรณ์  
รหัสไปรษณีย์ 67170

### ประวัติการศึกษา

พ.ศ. 2559 - พ.ศ. 2562

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ. 2553 - พ.ศ. 2558

โรงเรียนวัฒนาวิทยาลัย (ระดับมัธยมศึกษา)

พ.ศ. 2547 - พ.ศ. 2552

โรงเรียนอัสสัมชัญคอนแวนต์ลำานารายณ์ (ระดับประถมศึกษา)