



มหาวิทยาลัยกาฬสินธุ์

## รายงานการปฏิบัติงานสหกิจศึกษา

ระบบโอนถ่ายข้อมูลธุรกรรมการเงินแบบอัตโนมัติ

บนระบบเอสเอพี บิสซิเนส วัน

Automated Financial Transaction Data Transfer System on SAP  
Business One

จัดทำโดย

นายเมฆพัฒน์ การจักร์

## รายงานการปฏิบัติงานสหกิจศึกษา

[ชื่อปริญญา]

คณะบริหารศาสตร์ มหาวิทยาลัยกาฬสินธุ์

ภาคการศึกษาที่ [เลขภาคการศึกษา] [พ.ศ.]

[ชื่อโครงการสหกิจศึกษาภาษาไทย]

[ชื่อเจ้าของโครงการสหกิจศึกษาภาษาไทย]

## รายงานการปฏิบัติงานสหกิจศึกษา

[ชื่อปริญญา]

คณะบริหารศาสตร์ มหาวิทยาลัยกาฬสินธุ์

ภาคการศึกษาที่ [เลขภาคการศึกษา] [พ.ศ.]

ใบรับรองโครงงานสหกิจศึกษา

หัวข้อโครงงาน	ระบบโอนถ่ายข้อมูลธุรกรรมการเงินแบบอัตโนมัติบนระบบเอสเอพี บิซิเนส วัน Automated Financial Transaction Data Transfer System on SAP Business One
ผู้จัดทำ	นายเมฆพัฒน์ การจักร์
สาขาวิชา	[ชื่อสาขาวิชา]
อาจารย์ที่ปรึกษา	[ชื่ออาจารย์ที่ปรึกษา]

คณะบริหารศาสตร์ มหาวิทยาลัยกาฬสินธุ์ อนุมัติให้รายงานการปฏิบัติงานสหกิจฉบับนี้เป็น  
ส่วนหนึ่งของการศึกษาตามหลักสูตร [ชื่อปริญญา] ของคณะบริหารศาสตร์

..... อาจารย์ที่ปรึกษาโครงงานสหกิจศึกษา  
([ชื่ออาจารย์ที่ปรึกษาโครงงานสหกิจศึกษา] )

..... หัวหน้าสาขาวิชา[ชื่อสาขาวิชา]  
([ชื่อหัวหน้าสาขาวิชา] )

..... รองคณบดีงานวิชาการและวิจัย  
([ชื่อรองคณบดีงานวิชาการและวิจัย] )

..... คณบดีคณะบริหารศาสตร์  
([ชื่อคณบดีคณะบริหารศาสตร์] )

วันที่ เดือน [ชื่อเดือน] พ.ศ. [ปี พ.ศ.]

### กิตติกรรมประกาศ

โครงการสหกิจศึกษาเล่มนี้เป็นส่วนหนึ่งของวิชา [ชื่อวิชา] ตามหลักสูตร [ชื่อปริญญา] การปฏิบัติสหกิจศึกษาประจำภาคเรียนที่ [เลขภาคการศึกษา] ปีการศึกษา [พ.ศ.ปีการศึกษา] ระหว่างวันที่ "[ว/ด/ป เริ่มต้น]" ถึง "[ว/ด/ป สิ้นสุด]" ผลจากการปฏิบัติสหกิจศึกษารั้งนี้.....

[ชื่อเจ้าของโครงการสหกิจศึกษา]

ชื่อเรื่อง	[ชื่อโครงการสหกิจศึกษาภาษาไทย]
ชื่อสถานประกอบการ	[ชื่อสถานประกอบการ]
ชื่อนักศึกษา	[ชื่อเจ้าของโครงการสหกิจศึกษา]
รหัสนักศึกษา	[รหัสนักศึกษา]
สาขาวิชา	[ชื่อสาขาวิชา]
อาจารย์ที่ปรึกษา	[ชื่ออาจารย์ที่ปรึกษาโครงการ]
ปีการศึกษา	[พ.ศ.ปีการศึกษา]

### บทคัดย่อ

ในยุคดิจิทัลที่ข้อมูลเปรียบเสมือนสินทรัพย์สำคัญขององค์กร การบริหารจัดการข้อมูลข้ามระบบที่มีประสิทธิภาพถือเป็นปัจจัยชี้วัดความสำเร็จทางธุรกิจ บริษัท ศาลาแดง จำกัด ในฐานะพันธมิตรทางธุรกิจระดับ Gold Partner ของ SAP และผู้เชี่ยวชาญด้านการวางระบบวางแผนทรัพยากรองค์กร (ERP) ได้เล็งเห็นถึงปัญหาเรื่องในการบริหารจัดการข้อมูลของลูกค้าองค์กรขนาดใหญ่ โดยเฉพาะอย่างยิ่ง ปัญหาการแยกส่วนของข้อมูล (DataSilo) ระหว่างระบบบริหารโครงการ (ProjectDevelopmentManagement System: PDMS) และระบบบัญชีการเงิน เอสเอพี บิสซิเนสวัน (SAP Business One) จากการวิเคราะห์กระบวนการทำงาน (Business Process Analysis) พบว่ากระบวนการรับรู้รายได้และการตั้งหนี้ (Account Receivable) ยังคงพึ่งพาการปฏิบัติงานด้วยมือ (Manual Operation) เป็นหลัก พนักงานต้องทำการดึงข้อมูลใบแจ้งหนี้จาก PDMS มาตรวจสอบและบันทึกข้่าลงใน SAP ที่ละรายการ และเมื่อกระบวนการรับชำระเงินเสร็จสิ้น ยังต้องกลับไปอัปเดตสถานะใน PDMS อีกครั้ง ขั้นตอนที่ซ้ำซ้อนและขาดความต่อเนื่องนี้ ส่งผลให้เกิดความล่าช้าในการปิดบัญชี สิ้นเปลืองทรัพยากรบุคคล และมีความเสี่ยงสูงต่อความผิดพลาดของข้อมูล (Data Integrity Issues) ซึ่งกระทบต่อความแม่นยำในการรายงานสถานะทางการเงินต่อผู้บริหาร

เพื่อแก้ไขปัญหาดังกล่าวและยกระดับกระบวนการทำงานให้เป็นระบบอัตโนมัติ (Automation) ตามมาตรฐานสากล ผู้จัดทำโครงการจึงได้พัฒนา "ระบบจัดการข้อมูลและซิงค์สถานะสองทิศทาง (Two-Way Data Management & Status Synchronization System)" ขึ้น โดยออกแบบสถาปัตยกรรมระบบในรูปแบบ Middleware ด้วยภาษา C# บนแพลตฟอร์ม .NET Framework ทำหน้าที่เป็นตัวกลางอัจฉริยะในการเชื่อมต่อ API ระหว่างระบบ PDMS และระบบเอสเอพี บิสซิเนสวัน ผ่านเทคโนโลยี Service Layer บนฐานข้อมูล SAP HANA ระบบที่พัฒนาขึ้นครอบคลุมฟังก์ชันการทำงานสำคัญ 2 ส่วน ได้แก่ 1) Inbound Data Management: ระบบดึงข้อมูล Invoice ที่ได้รับการอนุมัติจาก PDMS ผ่าน RESTful API มาทำการตรวจสอบความถูกต้อง (Validation) เทียบกับฐานข้อมูลหลัก และ

สร้างเอกสารทางบัญชีใน SAP โดยอัตโนมัติ และ 2) Outbound Status Synchronization: ระบบติดตามสถานะการชำระเงินใน SAP และส่งข้อมูลสถานะ "Paid" พร้อมรายละเอียดการชำระเงินกลับไปยัง PDMS ทันทีแบบ Real-time ผ่านกลไกการเรียกกลับ (Callback Mechanism) ช่วยให้ทั้งสองระบบมีข้อมูลที่สอดคล้องตรงกันตลอดเวลาผลการทดสอบประสิทธิภาพการทำงานจริง (User Acceptance Test) พบว่าระบบสามารถลดระยะเวลาการประมวลผลข้อมูลจากเดิมเฉลี่ย 15-20 นาทีต่อเอกสาร เหลือเพียงไม่เกิน 30 วินาที ซึ่งลดลงกว่าร้อยละ 96 ช่วยขจัดปัญหาความผิดพลาดจากการบันทึกข้อมูลด้วยมือ (Human Error) ได้อย่างสมบูรณ์ 100% และทำให้ฝ่ายขายและฝ่ายบัญชีสามารถติดตามสถานะกระแสเงินสดได้อย่างเป็นปัจจุบัน (Real-time Visibility) โครงการนี้ไม่เพียงแต่ช่วยเพิ่มประสิทธิภาพการดำเนินงานให้กับลูกค้าของ บริษัท ศาลาแดง จำกัด แต่ยังเป็นการพิสูจน์ศักยภาพในการประยุกต์ใช้เทคโนโลยี SAP Service Layer เพื่อสร้างนวัตกรรมในการเชื่อมต่อระบบ ERP ระดับองค์กรได้อย่างเป็นรูปธรรม

**คำสำคัญ** : เอสเอพี บิสซิเนส วัน ; เอสเอพี ฮานา ; เซอร์วิส เลเยอร์ ; การจัดการข้อมูล ; ระบบอัตโนมัติ

## สารบัญ

หน้า

กิตติกรรมประกาศ.....	ก
บทคัดย่อ .....	ข
สารบัญ.....	ค
สารบัญตาราง.....	ง
สารบัญภาพ.....	จ
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 รายละเอียดของสถานประกอบการ (ส่วนนี้ให้นักศึกษาเติมข้อมูลจริงของบริษัท).....	2
1.2.1 ตำแหน่งและลักษณะงานที่ได้รับ .....	2
1.2.2 ผลที่ได้รับจากการปฏิบัติงานสหกิจศึกษา .....	2
1.2.3 พนักงานที่ปรึกษา.....	3
1.2.4 ระยะเวลาปฏิบัติงาน .....	3
1.3 วัตถุประสงค์ของโครงการ.....	3
1.4 ขอบเขตของโครงการ.....	3
1.5 นิยามศัพท์เฉพาะ (ถ้ามี) .....	4
1.6 ผลที่คาดว่าจะได้รับ .....	5
บทที่ 2 แนวคิดและทฤษฎีที่เกี่ยวข้อง.....	6
2.1 ระบบ Enterprise Resource Planning (ERP) และ SAP Business One .....	6
2.1.1 แนวคิดของระบบ ERP .....	6
2.1.2 ระบบ SAP Business One บน SAP HANA.....	6
2.2 สถาปัตยกรรม Web API และ RESTful Services.....	7
2.2.1 ความหมายของ API.....	7
2.2.2 สถาปัตยกรรม REST (Representational State Transfer) .....	7
2.3 การเทคโนโลยี SAP Service Layer บน SAP HANA SAP Service Layer .....	8
2.3.1 บทบาทของ Service Layer .....	8
2.3.2 มาตรฐาน OData (Open Data Protocol).....	8
2.4 หลักการออกแบบระบบจัดการข้อมูล (Data Management Design Patterns) .....	8

2.5 การพัฒนาโปรแกรมแบบ Windows Service Windows Service .....	9
2.6 การพัฒนา Add-on บน SAP Business One.....	9
2.7 การรักษาความปลอดภัยของข้อมูลรับรองตัวตน .....	10
บทที่ 3 วิธีดำเนินงานโครงการ.....	13
3.1 การวิเคราะห์ความต้องการและกระบวนการทางธุรกิจ (Requirement & Process Analysis) .....	13
3.2 การออกแบบระบบ (System Design).....	13
3.2.1 แผนภาพบริบท (Context Diagram).....	13
3.2.2 แผนภาพกระแสข้อมูล.....	15
3.2.3 แผนภาพความสัมพันธ์ของข้อมูล ER Diagram .....	16
3.3 การพัฒนาและติดตั้งระบบ Implementation.....	17
3.3.1 ลำดับการทำงานของระบบ Sequence Diagram .....	17
3.3.2 ขั้นตอนการนำเข้าข้อมูล (Import Logic) .....	19
3.3.3 ขั้นตอนการอัปเดตสถานะ (Update Status Logic).....	21
3.4 เครื่องมือและเทคโนโลยีที่ใช้ในการพัฒนา.....	24
3.4.1 ภาษาและแพลตฟอร์มสำหรับการพัฒนา .....	24
3.4.2 สภาพแวดล้อมการพัฒนา .....	24
3.4.3 ไลบรารีและชุดคำสั่งสำหรับการเชื่อมต่อระบบ .....	24
3.4.4 โครงสร้างโปรเจค .....	25
บทที่ 4 ผลการดำเนินงานโครงการ.....	26
4.1 ผลการทดสอบระบบ (System Testing).....	26
4.2 ผลการพัฒนาระบบนำเข้าและส่งออกข้อมูล .....	27
4.3 การเปรียบเทียบประสิทธิภาพก่อนและหลังใช้งาน .....	28
บทที่ 5 สรุปผลโครงการและข้อเสนอแนะ .....	30
5.1 สรุปผลการดำเนินงาน โครงการ .....	30
5.2 ปัญหาและอุปสรรค .....	30
5.3 ข้อเสนอแนะ จากการดำเนินโครงการและทดสอบระบบ .....	30



## สารบัญตาราง

หน้า

ตารางที่ 2.1 ผลการทดลอง .....	11
-------------------------------	----

สารบัญรูป

หน้า

No table of figures entries found.

## บทที่ 1

### บทนำ

#### 1.1 ความเป็นมาและความสำคัญของปัญหา

ในปัจจุบันภูมิทัศน์ของการบริหารจัดการธุรกิจได้เปลี่ยนแปลงไปอย่างมาก เทคโนโลยีสารสนเทศได้เข้ามามีบทบาทสำคัญในการเชื่อมโยงกระบวนการทำงาน (Business Process Integration) ของแต่ละแผนกเข้าด้วยกัน เพื่อให้องค์กรสามารถตอบสนองต่อการแข่งขันได้อย่างรวดเร็ว โดยเฉพาะอย่างยิ่งการบริหารจัดการข้อมูลระหว่างระบบเฉพาะทาง (Specialized Systems) และระบบบริหารทรัพยากรองค์กร (Enterprise Resource Planning: ERP) ที่ต้องทำงานสอดประสานกัน อย่างไรก็ตาม การพึ่งพาการทำงานด้วยเอกสารกระดาษหรือการบันทึกข้อมูลซ้ำซ้อนด้วยมือ (Manual Entry) แบบดั้งเดิม จึงถูกแทนที่ด้วยระบบอัตโนมัติและการเชื่อมต่อข้อมูลผ่าน Application Programming Interface (API) ซึ่งช่วยให้ข้อมูลไหลเวียนระหว่างระบบได้อย่างถูกต้อง รวดเร็ว และเป็นปัจจุบัน (Real-time)

บริษัท ศาลาแดง จำกัด ในฐานะพันธมิตรทางธุรกิจระดับ Gold Partner ของ SAP และผู้เชี่ยวชาญด้านการให้คำปรึกษาและติดตั้งระบบ เอสเอพี บิสซิเนส วัน (SAP Business One) ซึ่งเป็นระบบ ERP ชั้นนำระดับโลก ได้ให้บริการแก่ลูกค้าองค์กรในหลากหลายอุตสาหกรรม โดยครอบคลุมทั้งการจัดการด้านการเงิน การบัญชี การจัดซื้อ และการบริหารสินค้าคงคลัง อย่างไรก็ตาม จากการเข้าไปปฏิบัติงานที่ปรึกษา พบว่าลูกค้าองค์กรหลายรายยังคงประสบปัญหา "คอขวด" ในการปฏิบัติงาน โดยเฉพาะลูกค้าที่มีการใช้งานระบบบริหารโครงการ (PDMS) ควบคู่กับระบบบัญชี ปัญหาหลักที่พบคือ กระบวนการส่งต่อข้อมูลระหว่างสองระบบยังไม่เป็นอัตโนมัติ พนักงานบัญชีต้องใช้เวลาจำนวนมากในการดึงข้อมูลใบแจ้งหนี้จาก PDMS มาตรวจสอบและคีย์ข้อมูลซ้ำลงใน SAP ที่ละรายการ ซึ่งเป็นขั้นตอนที่สิ้นเปลืองเวลา (Time Consuming) และมีต้นทุนแฝงสูง นอกจากนี้ยังมีความเสี่ยงต่อความผิดพลาดจากการกรอกข้อมูล (Human Error) เช่น การระบุตัวเลขผิด หรือการไม่อัปเดตสถานะการชำระเงินกลับไปยังระบบต้นทาง ทำให้ฝ่ายขายและฝ่ายบัญชีมีข้อมูลที่ไม่ตรงกัน (Data Inconsistency) ส่งผลกระทบต่อการตัดสินใจของผู้บริหาร

จากปัญหาดังกล่าว ผู้วิจัยจึงเล็งเห็นความจำเป็นในการพัฒนา "ระบบจัดการข้อมูลธุรกรรมทางการเงินและแจ้งเตือนสถานะอัตโนมัติบนระบบเอสเอพี บิสซิเนส วัน" เพื่อแก้ไขปัญหาที่เกิดขึ้น โดยระบบที่พัฒนาขึ้นจะทำหน้าที่เป็นตัวกลาง (Middleware) ในการดึงข้อมูลการตั้งหนี้ (Invoice) ที่ได้รับการอนุมัติแล้วจากระบบ PDMS มาตรวจสอบและสร้างเอกสารทางบัญชีในระบบ เอสเอพี บิสซิเนส วัน โดยอัตโนมัติ และที่สำคัญระบบยังมีฟังก์ชันการส่งสถานะการชำระเงิน (Payment Status) จาก SAP

ย้อนกลับไปยัง PDMS ทันทีที่มีการรับชำระเงินเสร็จสิ้น การพัฒนาระบบนี้จะช่วยลดระยะเวลาในการทำธุรกรรม ลดความผิดพลาดจากการทำงานของมนุษย์ให้เหลือศูนย์ เพิ่มความถูกต้องแม่นยำของข้อมูล และเพิ่มประสิทธิภาพในการบริหารจัดการทางการเงินให้แก่ลูกค้าของ บริษัท ศาลาแดง จำกัด ได้อย่างเป็นรูปธรรม

## 1.2 รายละเอียดของสถานประกอบการ (ส่วนนี้ให้นักศึกษาเติมข้อมูลจริงของบริษัท)

ชื่อสถานประกอบการ: บริษัทศาลาแดง จำกัด (Saladang Co., Ltd.) ที่ตั้ง: .....

ลักษณะการประกอบธุรกิจ: (Saladang Co., Ltd.) เป็นที่ปรึกษาทางธุรกิจและผู้ให้บริการไอทีครบวงจร เชี่ยวชาญด้านระบบ ERP และได้รับแต่งตั้งเป็น SAP Business One Gold Partner ให้บริการวิเคราะห์ ติดตั้ง และพัฒนาโปรแกรมเชื่อมต่อระบบ (System Integration) แก่องค์กรชั้นนำ

### 1.2.1 ตำแหน่งและลักษณะงานที่ได้รับ

**ตำแหน่ง** นักศึกษาฝึกงาน / สหกิจศึกษา (Technical Services) Software Engineer  
ลักษณะงานที่ได้รับมอบหมาย

1. ศึกษาระบบ ERP SAP Business One และโครงสร้างฐานข้อมูล SAP HANA
2. ศึกษากระบวนการจัดการโครงการ (PDMS) ของลูกค้า และโครงสร้าง API ที่มีการรักษาความปลอดภัยด้วยการยืนยันตัวตนแบบ Token-based Authentication
3. ศึกษาการใช้งาน SAP Service Layer API และเครื่องมือในการพัฒนา
4. พัฒนาโปรแกรมเสริม (Add-on) หรือระบบอัตโนมัติเพื่อช่วยลดภาระงานของแผนกต่างๆ โดยเน้นที่ระบบเชื่อมต่อข้อมูล (Integration)
5. สนับสนุนการแก้ไขปัญหาทางเทคนิค (IT Support) เบื้องต้นให้แก่ผู้ใช้งานภายในและลูกค้าของบริษัท

### 1.2.2 ผลที่ได้รับจากการปฏิบัติงานสหกิจศึกษา

1. ได้เรียนรู้กระบวนการทำงานจริงในระบบธุรกิจและการใช้งานซอฟต์แวร์ระดับองค์กร (Enterprise Resource Planning: ERP)
2. ฝึกทักษะการพัฒนาระบบโปรแกรมเพื่อเชื่อมต่อกับระบบภายนอก (API Integration) และการจัดการฐานข้อมูลขั้นสูง

3. ได้ฝึกทักษะการแก้ปัญหาเฉพาะหน้า (Problem Solving) และการทำงานร่วมกับทีมงานมืออาชีพในองค์กร (Team Collaboration)
4. ได้รับประสบการณ์ในการจัดการโครงการพัฒนาซอฟต์แวร์ (Software Development Life Cycle) ตั้งแต่ขั้นตอนการวิเคราะห์ระบบจนถึงการขึ้นระบบใช้งานจริง

### 1.2.3 พณงานที่ปรึกษา

ชื่อ-นามสกุล: คุณ..... ตำแหน่ง: .....

### 1.2.4 ระยะเวลาปฏิบัติงาน

ตั้งแต่วันที่ ..... ถึงวันที่ .....

## 1.3 วัตถุประสงค์ของโครงการ

1. เพื่อพัฒนาระบบตัวกลาง (Middleware) เชื่อมต่อข้อมูลระหว่าง PDMS และ SAP Business One ผ่าน Service Layer API
2. เพื่อลดขั้นตอนการบันทึกข้อมูลซ้ำซ้อนด้วยมือ (Manual Re-entry) และลดความผิดพลาดในการทำงาน
3. เพื่อสร้างระบบซิงค์สถานะการชำระเงินสองทิศทาง (Two-Way Sync) ให้ข้อมูลถูกต้องและเป็นปัจจุบัน
4. เพื่อศึกษาและประยุกต์ใช้เทคโนโลยี RESTful API ในการพัฒนาระบบระดับองค์กร
5. ด้านการจัดการระบบ ทำงานแบบ Background Service ตามรอบเวลาที่กำหนด และบันทึกประวัติลงใน UDT

## 1.4 ขอบเขตของโครงการ

### 1.4.1 ขอบเขตด้านระบบงาน (Functional Scope)

1. ระบบขาเข้า (Inbound): ดึงข้อมูล Invoice ที่อนุมัติแล้วจาก PDMS มาตรวจสอบและสร้างเอกสารตั้งหนี้ใน SAP โดยอัตโนมัติ
2. ระบบจัดการข้อมูลขาออก (On-Demand Outbound Synchronization): พัฒนาฟังก์ชันส่วนขยาย (Add-on) บนหน้าจอ SAP Business One ให้ผู้ใช้งาน

สามารถกดปุ่มสั่งงานเพื่อส่งสถานะการรับชำระเงิน (Payment Status) ย้อนกลับไปยังระบบ PDMS ได้ทันทีเมื่อทำการเสร็จสิ้น

3. ระบบติดตามและแจ้งเตือน (Monitoring & Alerting Module): พัฒนาระบบบันทึกประวัติการรับ-ส่งข้อมูล (Transaction Logs) ภายในเมนูของ SAP และระบบแจ้งเตือนอัตโนมัติเมื่อเกิดข้อผิดพลาดในการเชื่อมต่อ เพื่อให้ผู้ดูแลระบบสามารถตรวจสอบย้อนหลังได้

### 1.3.2 ขอบเขตด้านเทคนิค (Technical Scope)

1. พัฒนาตัวกลาง (Middleware) ในรูปแบบ Windows Service ด้วยภาษา C# (.NET Framework)
2. เชื่อมต่อระบบ PDMS ผ่าน REST API (JSON)
3. เชื่อมต่อระบบ SAP Business One ผ่าน Service Layer (OData) บนฐานข้อมูล SAP HANA
4. พัฒนาส่วนขยาย (Add-on) ใน SAP Business One เพื่อสร้างปุ่มสั่งงานสำหรับผู้ใช้งาน (User Interface)

## 1.5 นิยามศัพท์เฉพาะ (ถ้ามี)

1. PDMS (Project Development Management System): ระบบบริหารจัดการโครงการที่เป็นแหล่งข้อมูลต้นทางของธุรกรรม ใช้สำหรับบริหารงานขายและวิศวกรรม
2. SAP Service Layer: เทคโนโลยี API ยุคใหม่ของ เอสเอพี บิสซิเนส วัน บน ฮานา (SAP Business One on HANA) ที่ช่วยให้แอปพลิเคชันภายนอกสามารถอ่านและเขียนข้อมูลลงในระบบ SAP ได้ผ่านโปรโตคอล HTTP/HTTPS
3. Two-Way Synchronization: กระบวนการแลกเปลี่ยนข้อมูลสองทิศทาง ในโครงการนี้หมายถึงการดึงข้อมูลตั้งหนี้เข้ามา (Import) และการส่งสถานะการรับชำระเงินออกไป (Update Status)
4. Middleware: ซอฟต์แวร์ตัวกลางที่ผู้วิจัยพัฒนาขึ้น ทำหน้าที่เป็นล่ามแปลภาษาและสะพานเชื่อมระหว่างระบบ PDMS และ SAP ที่มีโครงสร้างข้อมูลแตกต่างกัน ให้สามารถทำงานร่วมกันได้

## 1.6 ผลที่คาดว่าจะได้รับ

1. ประสิทธิภาพการทำงานเพิ่มขึ้น: สามารถลดระยะเวลาในกระบวนการบันทึกข้อมูลตั้งหนี้จากเดิม 15-20 นาทีต่อรายการ เหลือเพียงไม่เกิน 30 วินาทีต่อรายการ คิดเป็นประสิทธิภาพที่เพิ่มขึ้นกว่า 90%
2. ความถูกต้องของข้อมูล (Data Integrity): ขจัดปัญหาความผิดพลาดจากการทำงานของมนุษย์ (Human Error) ได้ 100% ข้อมูลระหว่างระบบ PDMS และ SAP มีความสอดคล้องตรงกัน
3. การบริหารจัดการแบบ Real-time: ฝ่ายบริหารและฝ่ายปฏิบัติการสามารถติดตามสถานะทางการเงินและกระแสเงินสดได้อย่างทันทั่วถึงที่ไม่ต้องรอรายงานสรุปสิ้นวัน
4. องค์ความรู้และนวัตกรรม: บริษัท ศาลาแดง จำกัด ได้รับต้นแบบระบบ (Prototype) และองค์ความรู้ในการเชื่อมต่อระบบผ่าน Service Layer ซึ่งสามารถนำไปต่อยอดเป็นบริการใหม่ให้กับลูกค้ารายอื่นในอนาคต

## บทที่ 2

### แนวคิดและทฤษฎีที่เกี่ยวข้อง

ในการพัฒนาระบบจัดการข้อมูลธุรกรรมทางการเงินและแจ้งเตือนสถานะอัตโนมัติบนระบบเอสเอพี บิสซิเนส วัน คณะผู้จัดทำได้ศึกษาแนวคิด ทฤษฎี และเทคโนโลยีที่เกี่ยวข้อง เพื่อให้ระบบสามารถทำงานได้อย่างมีประสิทธิภาพ มีความปลอดภัย และสอดคล้องกับมาตรฐานทางเทคนิคของระบบบริหารทรัพยากรองค์กรสมัยใหม่ โดยมีหัวข้อสำคัญดังต่อไปนี้

#### 2.1 ระบบ Enterprise Resource Planning (ERP) และ SAP Business One

ระบบ Enterprise Resource Planning (ERP) เป็นหัวใจสำคัญในการบริหารจัดการธุรกิจยุคใหม่ โดยทำหน้าที่รวบรวมข้อมูลจากทุกแผนกไว้ในฐานข้อมูลเดียว สำหรับโครงการนี้ได้เลือกใช้ซอฟต์แวร์ SAP Business One ซึ่งเป็นระบบ ERP ขั้นนำที่รองรับการทำงานบนฐานข้อมูลประสิทธิภาพสูงอย่าง SAP HANA เพื่อให้การประมวลผลและเชื่อมต่อข้อมูลเป็นไปอย่างรวดเร็ว

##### 2.1.1 แนวคิดของระบบ ERP

EnterpriseResourcePlanning(ERP)คือระบบสารสนเทศที่ทำหน้าที่บูรณาการกระบวนการทำงานหลักภายในองค์กรเข้าด้วยกัน (Integrated Business Processes) ไม่ว่าจะเป็นการเงิน การบัญชี การจัดซื้อ การขาย สินค้าคงคลัง และการผลิต เพื่อให้ข้อมูลสามารถไหลเวียนระหว่างแผนกได้อย่างต่อเนื่องและเป็นปัจจุบัน (Real-time) ช่วยให้ผู้บริหารสามารถตัดสินใจบนพื้นฐานของข้อมูลที่ต้องการและครบถ้วน ลดความซ้ำซ้อนของการบันทึกข้อมูล และเพิ่มประสิทธิภาพในการดำเนินงานขององค์กร

##### 2.1.2 ระบบ SAP Business One บน SAP HANA

SAP Business One เป็นซอฟต์แวร์ ERP ที่พัฒนาโดยบริษัท SAP SE ออกแบบมาเพื่อรองรับธุรกิจขนาดกลางและขนาดย่อม (SME) โดยเฉพาะ มีความยืดหยุ่นสูงและครอบคลุมฟังก์ชันงานธุรกิจที่จำเป็นสำหรับเวอร์ชันที่ทำงานบนฐานข้อมูล SAP HANA (High-Performance Analytic Appliance) นั้นใช้เทคโนโลยี In-Memory Computing ซึ่งเก็บและประมวลผลข้อมูลในหน่วยความจำหลัก (RAM) แทนการอ่านเขียนลงฮาร์ดดิสก์แบบดั้งเดิม ทำให้สามารถประมวลผลข้อมูลปริมาณมหาศาลและออกรายงาน



วิเคราะห์(Analytics)ได้ด้วยความเร็วสูงซึ่งเป็นรากฐานสำคัญที่ช่วยให้ระบบเชื่อมต่อข้อมูล (Integration) สามารถทำงานได้อย่างรวดเร็ว

## 2.2 สถาปัตยกรรม Web API และ RESTful Services

การเชื่อมต่อระบบสารสนเทศในปัจจุบันนิยมใช้สถาปัตยกรรม Web API และ RESTful Services เป็นมาตรฐานกลางในการสื่อสารข้อมูล เนื่องจากมีความเรียบง่าย ลดความซับซ้อนในการเชื่อมต่อ และสามารถทำงานร่วมกันได้บนโครงสร้างพื้นฐานของระบบเครือข่ายอินเทอร์เน็ตที่มีอยู่เดิม โดยมีรายละเอียดดังนี้

### 2.2.1 ความหมายของ API

Application Programming Interface (API) คือชุดของคำสั่ง โปรโตคอล และเครื่องมือที่ใช้ในการสร้างซอฟต์แวร์ ซึ่งทำหน้าที่เป็น "สะพานเชื่อม" (Interface) ให้โปรแกรมประยุกต์ต่างระบบสามารถสื่อสารและแลกเปลี่ยนข้อมูลกันได้ โดยที่ผู้พัฒนาไม่จำเป็นต้องทราบรายละเอียดการทำงานภายในของอีกระบบหนึ่ง เพียงแค่รู้คำสั่งในการเรียกใช้ (Request) และรูปแบบข้อมูลที่ได้รับกลับมา (Response)

### 2.2.2 สถาปัตยกรรม REST (Representational State Transfer)

REST คือรูปแบบสถาปัตยกรรมซอฟต์แวร์สำหรับการพัฒนาระบบแบบกระจายศูนย์ (Distributed Systems) บนเครือข่ายอินเทอร์เน็ต โดยใช้โปรโตคอล HTTP มาตรฐานในการสื่อสาร หลักการสำคัญของ RESTful API ประกอบด้วย:

1. Stateless การสื่อสารแต่ละครั้ง (Request) จะต้องมียุทธศาสตร์ครบถ้วนสำหรับการประมวลผล โดยที่เซิร์ฟเวอร์จะไม่เก็บสถานะของผู้ใช้ไว้
2. Standard Methods ใช้ HTTP Methods ในการกระทำกับทรัพยากร (Resources) ได้แก่
  - GET: สำหรับดึงข้อมูล (Retrieve)
  - POST สำหรับสร้างข้อมูลใหม่ (Create)
  - PUT/PATCH สำหรับแก้ไขข้อมูล (Update)
  - DELETE สำหรับลบข้อมูล (Delete)
3. Data Format นิยมใช้รูปแบบ JSON (JavaScript Object Notation) ในการรับส่งข้อมูล เนื่องจากมีโครงสร้างที่เข้าใจง่ายและมีขนาดเล็กกว่า XML

## 2.3 การเทคโนโลยี SAP Service Layer บน SAP HANA SAP Service Layer

คือเทคโนโลยี API ยุคใหม่ที่ถูกรออกแบบมาเพื่อการเข้าถึงและจัดการข้อมูลใน SAP Business One version for SAP HANA โดยเฉพาะ ซึ่งทำงานบนมาตรฐานเว็บ (Web Standards) ช่วยให้นักพัฒนาสามารถสร้างแอปพลิเคชันส่วนต่อขยาย (Extensions) ที่มีความทันสมัยและตอบสนองต่อความต้องการทางธุรกิจได้อย่างรวดเร็ว

### 2.3.1 บทบาทของ Service Layer

SAP Service Layer เป็นเทคโนโลยี API รุ่นใหม่ (Next-generation API) ที่พัฒนาขึ้นสำหรับ SAP Business One on HANA โดยเฉพาะ ออกแบบมาเพื่อทดแทนเทคโนโลยีเก่าอย่าง DI API (Data Interface API) ซึ่งมีข้อจำกัดเรื่องประสิทธิภาพเมื่อใช้งานบน Cloud หรือ Mobile

Service Layer ทำงานบนพื้นฐานของมาตรฐาน OData (Open Data Protocol) และสถาปัตยกรรมแบบ RESTful ซึ่งอนุญาตให้แอปพลิเคชันภายนอกสามารถเข้าถึงและจัดการ Business Objects (เช่น ใบแจ้งหนี้, ข้อมูลลูกค้า) ใน SAP Business One ได้ผ่าน HTTP/HTTPS Request ทำให้การพัฒนาระบบเชื่อมต่อ (Integration) มีความยืดหยุ่นสูง ปลอดภัย และสามารถรองรับการทำรายการจำนวนมาก (High Volume Transaction) ได้อย่างมีประสิทธิภาพ

### 2.3.2 มาตรฐาน OData (Open Data Protocol)

OData เป็นมาตรฐานสากลสำหรับการสร้างและใช้งาน RESTful APIs ที่ช่วยให้การสืบค้นข้อมูล (Querying) มีความยืดหยุ่นคล้ายกับภาษา SQL เช่น การกรองข้อมูล (\$filter), การเลือกฟิลด์ที่ต้องการ (\$select), หรือการเรียงลำดับ (\$orderby) ผ่าน URL โดยตรง ช่วยให้นักพัฒนาสามารถดึงข้อมูลที่ต้องการได้อย่างเจาะจง ลดภาระการส่งข้อมูลที่ไม่จำเป็น

## 2.4 หลักการออกแบบระบบจัดการข้อมูล (Data Management Design Patterns)

ในการพัฒนาระบบเชื่อมต่อข้อมูล (Data Integration) ระหว่างระบบ PDMS และ SAP Business One คณะผู้จัดทำได้ประยุกต์ใช้หลักการสำคัญดังนี้:

1. Middleware Architecture: การใช้ซอฟต์แวร์ตัวกลาง (Middleware) ในการเชื่อมต่อ ช่วยลดความยึดโยง (Decoupling) ระหว่างสองระบบ หากมีการเปลี่ยนแปลงในระบบใดระบบหนึ่ง จะกระทบเพียงแค่การตั้งค่าที่ตัวกลาง โดยไม่ต้องแก้โค้ดของระบบหลักทั้งสอง
2. Token-based Authentication: เพื่อความปลอดภัยในการเข้าถึงข้อมูลจากระบบ PDMS และ SAP Service Layer ระบบจะใช้วิธีการยืนยันตัวตนผ่าน Token (เช่น Session ID หรือ Bearer Token) ซึ่งมีความปลอดภัยสูงกว่าการส่ง Username/Password ไปในทุกครั้งที่มีการร้องขอข้อมูล
3. ETL Concept (Extract, Transform, Load): กระบวนการทำงานของระบบประยุกต์ใช้แนวคิด ETL ในแบบ Real-time: - Extract: ดึงข้อมูลดิบ (JSON) จาก API ของ PDMS - Transform: แปลงโครงสร้างข้อมูล ตรวจสอบความถูกต้อง และจับคู่ฟิลด์ (Mapping) ให้ตรงกับมาตรฐานของ SAP - Load: บันทึกข้อมูลลงสู่ฐานข้อมูล SAP HANA ผ่าน Service Layer

## 2.5 การพัฒนาโปรแกรมแบบ Windows Service Windows Service

คือรูปแบบหนึ่งของแอปพลิเคชันบนระบบปฏิบัติการ Microsoft Windows ที่ถูกออกแบบมาให้ทำงานอยู่เบื้องหลัง (Background Process) โดยมีลักษณะเด่นที่เหมาะสมกับการพัฒนาระบบอัตโนมัติ (Automation) ดังนี้ Automatic Start: สามารถตั้งค่าให้โปรแกรมทำงานทันทีที่เครื่องเซิร์ฟเวอร์เริ่มทำงาน (Boot) โดยไม่ต้องรอให้ผู้ใช้งานล็อกอิน (User Login) Long-running เหมาะสำหรับงานที่ต้องทำอย่างต่อเนื่องตลอดเวลา 24 ชั่วโมง หรือการเฝ้าระวัง (Monitoring) ไฟล์และข้อมูล No User Interface: ไม่มีหน้าจอติดต่อกับผู้ใช้โดยตรง แต่จะสื่อสารสถานะการทำงานผ่าน Log File หรือ Windows Event Viewer แทน

## 2.6 การพัฒนา Add-on บน SAP Business One

นอกจากการพัฒนาระบบ Windows Service ที่ทำหน้าที่ดึงข้อมูลและประมวลผลในเบื้องหลังแล้ว โครงการนี้ยังได้พัฒนาส่วนขยาย (Add-on) เพิ่มเติมบนหน้าจอการทำงานของ SAP Business One ด้วย เพื่อให้ผู้ใช้งานสามารถโต้ตอบกับระบบได้โดยตรงผ่านหน้าจอที่คุ้นเคยส่วนขยายที่พัฒนาขึ้นใช้เทคโนโลยี SAPbouiCOM ซึ่งย่อมาจาก SAP Business One User Interface Component Object Model เป็นชุดคำสั่งที่ SAP จัดเตรียมไว้ให้สำหรับนักพัฒนาที่ต้องการสร้างส่วนเสริมต่างๆ บนตัว SAP Client โดย SAPbouiCOM จะให้ความสามารถในการเข้าถึงและควบคุมองค์ประกอบต่างๆ บนหน้าจอ SAP ได้ เช่น การเพิ่มปุ่ม กล่องข้อความ หรือแม้กระทั่งหน้าจอใหม่ทั้งหมดหน้าที่หลักของส่วนขยายที่

พัฒนาขึ้นมาคือการเพิ่มปุ่ม อัปเดต PDMS ลงไปในหน้าจอ Outgoing Payment หรือหน้าจอการจ่ายเงินของระบบ SAP Business One โดยปุ่มนี้จะปรากฏขึ้นมาโดยอัตโนมัติเมื่อผู้ใช้เปิดหน้าจอดังกล่าว ตำแหน่งของปุ่มได้รับการออกแบบให้อยู่ในบริเวณที่เข้าถึงได้ง่าย ไม่รบกวนการทำงานปกติของผู้ใช้เมื่อผู้ใช้งานกดปุ่ม "อัปเดต PDMS" ระบบจะเริ่มทำงานโดยอ่านข้อมูลการจ่ายเงินจากเอกสารที่กำลังเปิดอยู่ในขณะนั้น ข้อมูลที่จะถูกอ่านออกมาประกอบด้วย เลขที่เอกสารการจ่ายเงิน วันที่ทำรายการ จำนวนเงินที่จ่าย รหัสผู้จำหน่าย และข้อมูลการอ้างอิงต่างๆ ที่เกี่ยวข้อง หลังจากนั้นระบบจะจัดรูปแบบข้อมูลเหล่านี้ให้อยู่ในรูปแบบ JSON แล้วส่งสถานะ "Paid" พร้อมกับรายละเอียดทั้งหมดกลับไปยังระบบ PDMS ผ่านทาง RESTful API โดยอัตโนมัติข้อดีของการมีส่วนขยายนี้คือผู้ใช้งานสามารถอัปเดตสถานะการชำระเงินไปยังระบบ PDMS ได้ทันทีหลังจากที่บันทึกข้อมูลการจ่ายเงินเสร็จสิ้น โดยไม่ต้องรอให้ Windows Service ทำงานตามรอบที่กำหนดไว้ ซึ่งโดยปกติจะเป็นทุก 15 นาที ทำให้การอัปเดตข้อมูลระหว่างสองระบบมีความรวดเร็วและทันสมัยมากขึ้นนอกจากนี้ยังช่วยลดความผิดพลาดที่อาจเกิดขึ้นจากการรอคอย เพราะในบางกรณีผู้ใช้อาจต้องการให้ข้อมูลถูกส่งไปทันทีเพื่อให้ฝ่ายอื่นๆ สามารถเห็นสถานะที่อัปเดตแล้วได้โดยเร็วที่สุด โดยเฉพาะในกรณีที่มีความเร่งด่วนหรือต้องการความแม่นยำของข้อมูลในเวลาจริงการพัฒนาส่วนขยายด้วย SAPbouiCOM นั้นต้องมีความเข้าใจในโครงสร้างของ SAP Business One UI API เป็นอย่างดี เพราะต้องทำงานร่วมกับระบบที่มีอยู่แล้วโดยไม่สร้างความเสียหายหรือขัดแย้งกับฟังก์ชันเดิมการเขียนโค้ดต้องระมัดระวังในการจัดการหน่วยความจำและทรัพยากรต่างๆ เพื่อไม่ให้กระทบต่อประสิทธิภาพของ SAP Client

## 2.7 การรักษาความปลอดภัยของข้อมูลรับรองตัวตน

ในการพัฒนาระบบที่ต้องเชื่อมต่อกับ API ของระบบภายนอก ไม่ว่าจะเป็น SAP Service Layer หรือ PDMS API นั้น จำเป็นต้องมีการจัดเก็บข้อมูลรับรองตัวตน (Credentials) เช่น ชื่อผู้ใช้ รหัสผ่าน กุญแจ API และข้อมูลสำคัญอื่นๆ ไว้ในระบบ เพื่อให้สามารถทำการยืนยันตัวตนและเชื่อมต่อกับระบบเหล่านั้นได้ปัญหาที่ตามมาคือถ้าเราเก็บข้อมูลเหล่านี้ไว้แบบข้อความธรรมดาในไฟล์ตั้งค่า จะเกิดความเสี่ยงสูงมากที่ข้อมูลจะรั่วไหลออกไป โดยเฉพาะในกรณีที่มีคนสามารถเข้าถึงไฟล์หรือเครื่องเซิร์ฟเวอร์ได้ หรือในกรณีที่ไฟล์ตั้งค่าถูกสำรองข้อมูลไปยังที่อื่น คนที่เข้าถึงไฟล์เหล่านั้นก็จะสามารถเห็นข้อมูลรับรองตัวตนทั้งหมดได้ทันทีเพื่อแก้ปัญหานี้ โครงการนี้จึงเลือกใช้เทคโนโลยี DPAPI ซึ่งย่อมาจาก Data Protection API เป็นเครื่องมือที่ Microsoft พัฒนาขึ้นมาสำหรับระบบปฏิบัติการ Windows โดยเฉพาะ เพื่อให้บริการด้านการเข้ารหัสและถอดรหัสข้อมูลอย่างปลอดภัย DPAPI ทำงานโดยใช้หลักการเข้ารหัสแบบสมมาตร (Symmetric Encryption) แต่สิ่งที่ทำให้มันพิเศษกว่าการเข้ารหัสทั่วไปคือมันจะผูกกับเครื่องคอมพิวเตอร์หรือบัญชีผู้ใช้ที่ทำการเข้ารหัส ซึ่งเรียกว่า Machine-level Encryption หรือ User-level Encryption ก็ได้ ในโครงการนี้เลือกใช้แบบ Machine-level เพราะเป็น

Windows Service ที่ทำงานในระดับเครื่องกลไกการทำงานของ DPAPI นั้นจะใช้กุญแจเข้ารหัสที่ถูกสร้างขึ้นจากข้อมูลเฉพาะของเครื่องคอมพิวเตอร์นั้นๆ เช่น รหัสเครื่อง ข้อมูลฮาร์ดแวร์ และข้อมูลอื่นๆ ที่ไม่ซ้ำกันสำหรับเครื่องนั้น เมื่อทำการเข้ารหัสข้อมูลด้วย DPAPI แล้ว ข้อมูลที่ได้จะสามารถถอดรหัสได้เฉพาะบนเครื่องที่ทำการเข้ารหัสเท่านั้นผลที่ตามมาคือแม้ว่าจะมีคนเอาไฟล์ตั้งค่าที่มีข้อมูลเข้ารหัสไปเปิดที่เครื่องอื่น ก็จะไม่สามารถถอดรหัสออกมาเป็นข้อความที่อ่านได้ เพราะกุญแจเข้ารหัสของทั้งสองเครื่องไม่เหมือนกัน วิธีการนี้จึงช่วยป้องกันการรั่วไหลของข้อมูลรับรองตัวตนได้อย่างมีประสิทธิภาพการใช้งาน DPAPI ใน C# ไม่ยากเกินไป Microsoft ได้จัดเตรียมคลาสที่ชื่อว่า ProtectedData ไว้ใน namespace System.Security.Cryptography โดยมีเมธอดหลักๆ อยู่ 2 ตัวคือ Protect สำหรับเข้ารหัส และ Unprotect สำหรับถอดรหัส เราสามารถเลือกได้ว่าจะให้ผูกกับ CurrentUser หรือ LocalMachine ในโครงการนี้เลือกใช้ DataProtectionScopeLocalMachine เพราะเป็น Windows Service ที่ต้องทำงานโดยไม่ต้องมีผู้ใช้เข้าสู่ระบบดังนั้นการผูกกับเครื่องจะเหมาะสมกว่าผูกกับบัญชีผู้ใช้ใดๆ โดยเฉพาะตัวอย่างการใช้งานเบื้องต้นคือ เวลาที่เราจะเก็บรหัสผ่านลงไฟล์ตั้งค่า เราจะเข้ารหัสผ่านที่เป็นข้อความธรรมดาตามาแปลงเป็น byte array ก่อน แล้วเรียกใช้ ProtectedData.Protect เพื่อเข้ารหัส ได้ข้อมูลที่เข้ารหัสแล้วออกมา จากนั้นก็แปลงเป็น Base64 string เพื่อความสะดวกในการเก็บ เวลาจะใช้ก็ทำกลับกัน อ่าน Base64 string มาแปลงเป็น bytes แล้วเรียก ProtectedData.Unprotect ก็จะได้ข้อความธรรมดากลับมาอย่างไรก็ตาม DPAPI ไม่ได้ปลอดภัย 100% เพราะถ้ามีคนสามารถเข้าถึงเครื่องเซิร์ฟเวอร์โดยตรงในฐานะผู้ดูแลระบบ ก็ยังสามารถถอดรหัสได้อยู่ดี แต่มันก็ช่วยป้องกันได้หลายกรณี เช่นกรณีไฟล์ถูกคัดลอกออกไป กรณีสำรองข้อมูลถูกเก็บไว้ที่อื่น หรือกรณีที่มีคนพยายามอ่านไฟล์โดยตรงโดยไม่ผ่านแอปพลิเคชันที่ถูกต้องนอกจาก DPAPI แล้ว ระบบยังมีการใช้ TLS 1.2 ในการสื่อสารกับ API ต่างๆ ด้วย เพื่อให้แน่ใจว่าข้อมูลที่ส่งผ่านเครือข่ายจะถูกเข้ารหัสตลอดเส้นทาง ไม่สามารถดักจับข้อมูลระหว่างทางได้ การใช้ TLS 1.2 ร่วมกับ DPAPI จึงทำให้ระบบมีความปลอดภัยทั้งในส่วนของการเก็บข้อมูล (Data at Rest) และการส่งข้อมูล (Data in Transit)

คุณสมบัติ (Criteria)	การบันทึกด้วยมือ (Manual Input)	Data Transfer Workbench (DTW)	Windows Service (โครงการนี้)
ความเร็ว (Speed)	ช้ามาก (Slow)	ปานกลาง - เร็ว (Batch)	เร็วมาก (Real-time)
ความอัตโนมัติ (Automation)	ต้องใช้คน (Manual)	ต้องกดสั่งงาน (Manual Trigger)	อัตโนมัติสมบูรณ์ (Fully Auto)
ความเสี่ยงต่อ ข้อผิดพลาด(Error Risk)	สูง (Human Error)	ปานกลาง (Format Error)	ต่ำ (Programmatic Validation)
การทำงานตลอด24 ชม. การแจ้งเตือน (Notification)	ไม่ได้	ไม่ได้	ได้ (Always On)
การแจ้งเตือน (Notification)	ไม่มี	ดูจาก Log ท้ายรายการ	แจ้งเตือนทันทีผ่านระบบ

ตารางที่ 1 เปรียบเทียบคุณสมบัติของวิธีการนำเข้าข้อมูลระหว่างการบันทึกด้วยมือเครื่องมือ DTW และ Windows Service

## บทที่ 3

### วิธีดำเนินงานโครงการ

ในการดำเนินโครงการระบบจัดการข้อมูลธุรกรรมทางการเงินและแจ้งเตือนสถานะอัตโนมัติบนระบบ เอสเอพี บิสซิเนส วัน คณะผู้จัดทำได้กำหนดขั้นตอนการดำเนินงานอย่างเป็นระบบ เพื่อให้การพัฒนา ระบบเป็นไปตามวัตถุประสงค์และขอบเขตที่กำหนดไว้ โดยมีลำดับขั้นตอนการดำเนินงานตั้งแต่การ วิเคราะห์ปัญหา การออกแบบระบบ การพัฒนาโปรแกรม การทดสอบ และการนำไปใช้งานจริง ซึ่งมี รายละเอียดในแต่ละขั้นตอนดังนี้

#### 3.1 การวิเคราะห์ความต้องการและกระบวนการทางธุรกิจ (Requirement & Process Analysis)

จากการศึกษากระบวนการทำงานเดิม (As-Is Process) ร่วมกับทีมที่ปรึกษาของบริษัท พบจุดอ่อนใน ขั้นตอนการส่งต่อข้อมูล ผู้วิจัยจึงได้ออกแบบกระบวนการทำงานใหม่ (To-Be Process) ดังนี้

1. Data Acquisition การดึงข้อมูล Middleware ทำการเรียก (Request) ไปยัง API ของ PDMS ตามรอบเวลา หรือเมื่อได้รับคำสั่ง เพื่อดึงรายการ Invoice ที่รอการตั้งหนี้
2. Data Validation & Transformation (การตรวจสอบและแปลงข้อมูล) ระบบทำการตรวจสอบความสมบูรณ์ของข้อมูล เช่น ตรวจสอบว่ารหัสลูกค้า CardCode ใน PDMS ตรงกับใน SAP หรือไม่ หากไม่ตรงระบบจะแจ้งเตือน นอกจากนี้ยังทำการแปลงโครงสร้างข้อมูลจาก JSON ของ PDMS ให้เป็น Format ที่ SAP Service Layer ต้องการ

#### 3.2 การออกแบบระบบ (System Design)

เพื่อให้เห็นภาพรวมและรายละเอียดของระบบ ผู้จัดทำได้ออกแบบแผนภาพต่างๆ ดังนี้:

##### 3.2.1 แผนภาพบริบท (Context Diagram)

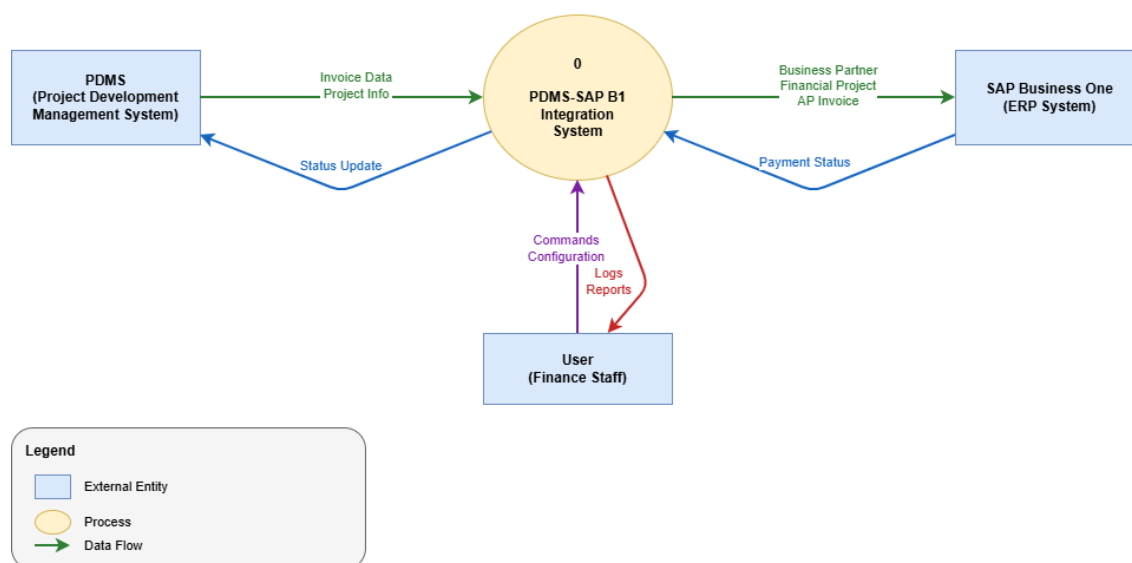
แผนภาพบริบท (Context Diagram) แสดงขอบเขตและภาพรวมของการทำงานของระบบเชื่อมต่อข้อมูลอัตโนมัติ โดยมุ่งเน้นการแสดงความสัมพันธ์ระหว่างระบบที่

พัฒนาขึ้นกับปัจจัยภายนอก (External Entities) ที่เกี่ยวข้อง ประกอบด้วย 3 ส่วนหลัก ได้แก่

- 1) ระบบบริหารโครงการ (PDMS) ทำหน้าที่เป็นแหล่งข้อมูลต้นทาง (Data Source) โดยส่งข้อมูลโครงการ (Project Info) และรายละเอียดงาน (Invoice Data) ที่ได้รับการอนุมัติแล้วเข้าสู่ระบบ
- 2) ระบบ SAP Business One ทำหน้าที่เป็นระบบปลายทาง (Destination) สำหรับรับข้อมูลเพื่อสร้างเอกสารทางบัญชี และเป็นแหล่งข้อมูลสำหรับการตรวจสอบสถานะการชำระเงิน (Payment Status) เพื่อส่งกลับไปยังระบบต้นทาง
- 3) ผู้ใช้งาน (User) เจ้าหน้าที่ฝ่ายบัญชีหรือผู้ดูแลระบบที่ทำหน้าที่ตรวจสอบรายงานความผิดพลาด (Error Report) และดูแลการทำงานของระบบ

โดยกระบวนการไหลของข้อมูล (Data Flow) จะเริ่มจากระบบ PDMS ส่งข้อมูลเข้ามายังระบบกลาง (System) เพื่อทำการประมวลผลตรวจสอบความถูกต้อง (Validation) ก่อนจะส่งต่อไปบันทึกยังฐานข้อมูล SAP Business One และเมื่อกระบวนการชำระเงินเสร็จสิ้น ระบบจะดึงสถานะจาก SAP เพื่อส่งกลับไปยัง PDMS โดยอัตโนมัติ

#### DFD Context Level - PDMS SAP B1 Integration System



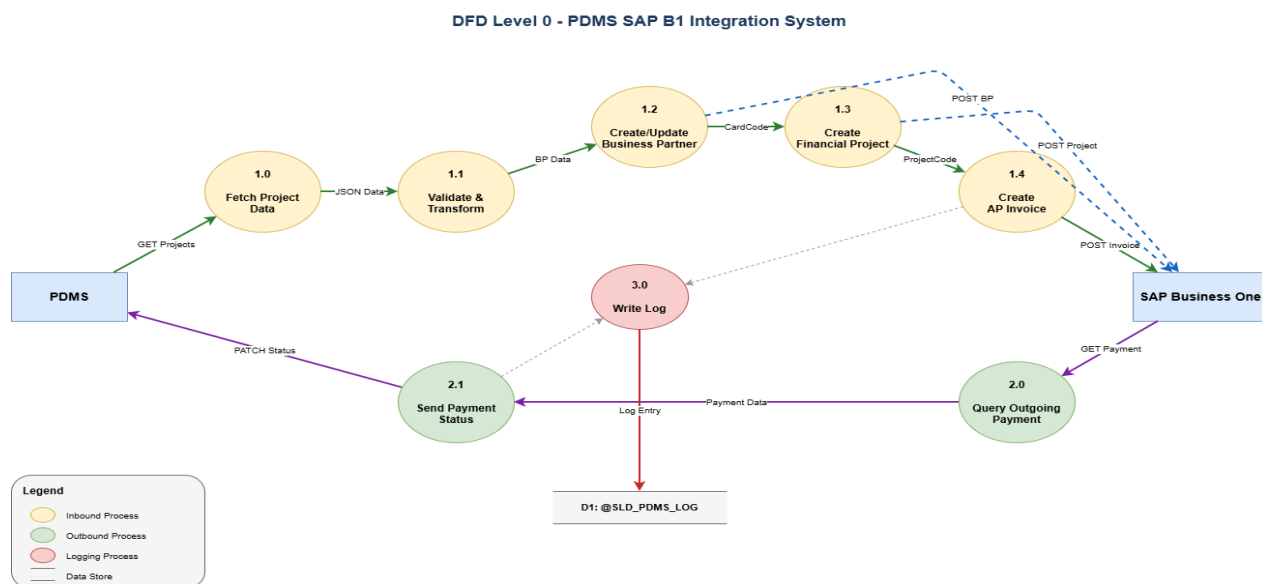
ภาพที่ 1 แผนภาพบริบทแสดงการเชื่อมต่อระหว่าง External Entities



### 3.2.2 แผนภาพกระแสข้อมูล

แผนภาพกระแสข้อมูลระดับ 0 (Data Flow Diagram Level 0) แสดงรายละเอียดของกระบวนการหลัก (Processes) ภายในระบบ ซึ่งแตกย่อยจากแผนภาพบริบทเพื่อแสดงขั้นตอนการทำงานภายใน โดยแบ่งกลุ่มการทำงานออกเป็น 3 ส่วนสำคัญ (ตามรหัสสีในแผนภาพ) ดังนี้:

1. กระบวนการนำเข้าข้อมูล (Inbound Process - สีเหลือง) เริ่มต้นจากการดึงข้อมูลโครงการจาก PDMS (Fetch Project Data) ผ่านเข้าสู่ขั้นตอนการตรวจสอบความถูกต้องและแปลงรูปแบบข้อมูล (Validate and Transform) เพื่อให้สอดคล้องกับโครงสร้างของ SAP จากนั้นระบบจะทำการตรวจสอบและสร้างข้อมูลหลัก ได้แก่ ข้อมูลคู่ค้า (Create/Update BP) ข้อมูลโครงการ (Create Project) และสุดท้ายคือการสร้างใบสำคัญจ่าย (Create A/P Invoice)
2. กระบวนการส่งออกข้อมูล (Outbound Process - สีเขียว) เป็นกระบวนการที่ทำงานคู่ขนานเพื่อตรวจสอบสถานะการชำระเงิน โดยเริ่มจากการสอบถามข้อมูลการจ่ายเงิน (Query Outgoing Payment) จากฐานข้อมูล SAP หากพบรายการที่ชำระเงินแล้ว ระบบจะทำการส่งสถานะ (Send Payment Status) กลับไปยัง PDMS เพื่ออัปเดตข้อมูลให้ตรงกัน
3. กระบวนการบันทึกประวัติ (Logging Process - สีแดง) ทุกขั้นตอนการทำงาน ไม่ว่าจะเป็นความสำเร็จหรือข้อผิดพลาด จะถูกส่งมายังกระบวนการบันทึกประวัติ (Write Log) เพื่อจัดเก็บลงในถังเก็บข้อมูล (Data Store D1: @SLD\_PDMS\_LOG) สำหรับใช้ในการตรวจสอบย้อนหลัง (Audit Trail)



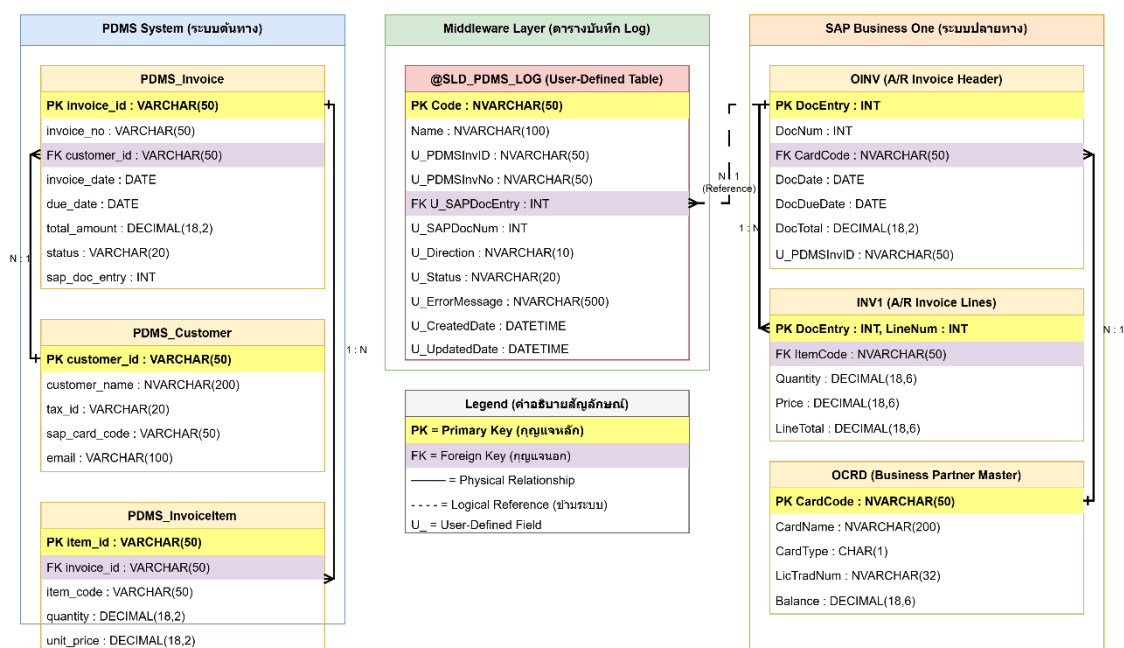
ภาพที่ 2 แผนภาพกระแสข้อมูลระดับ 0 แสดงกระบวนการนำเข้า ส่งออก และบันทึกผล

### 3.2.3 แผนภาพความสัมพันธ์ของข้อมูล ER Diagram

แผนภาพความสัมพันธ์ของข้อมูล (Entity Relationship Diagram) แสดงโครงสร้างการจัดเก็บข้อมูลภายในฐานข้อมูล SAP HANA และความสัมพันธ์เชิงตรรกะระหว่างตารางข้อมูลต่างๆ ซึ่งถูกออกแบบตามมาตรฐานของระบบ SAP Business One โดยเน้นการรักษาความถูกต้องของข้อมูล (Data Integrity) ในรูปแบบ One-to-Many (1:N) ดังนี้:

- ลำดับชั้นข้อมูลหลัก (Master Data Hierarchy): เริ่มต้นจากตาราง OCRD (Business Partner) ซึ่งจัดเก็บข้อมูลคู่ค้าทางธุรกิจ จะมีความสัมพันธ์แบบ 1:N กับตาราง OPRJ (Project) เนื่องจากคู่ค้าหนึ่งรายสามารถรับผิดชอบโครงการก่อสร้างได้หลายโครงการพร้อมกัน
- ลำดับชั้นข้อมูลรายการค้า (Transaction Data Hierarchy) ข้อมูลโครงการจากตาราง OPRJ จะทำหน้าที่เป็นศูนย์กลางในการเชื่อมโยงไปยังตาราง OPCH (A/P Invoice) หรือใบสำคัญจ่าย ซึ่งหมายความว่าภายใต้หนึ่งโครงการจะประกอบไปด้วยงวดงานและการตั้งหนี้หลายรายการ และเมื่อกระบวนการชำระเงินเสร็จสิ้น ตาราง OPCH จะถูกเชื่อมโยงต่อไปยังตาราง OVPM (Outgoing Payment) เพื่อบันทึกการจ่ายเงิน
- ส่วนขยายเพื่อการตรวจสอบ (Auditing Extension) เพื่อรองรับการทำงานของระบบอัตโนมัติ ได้มีการสร้างตาราง @SLD\_PDMS\_LOG (User Defined Table) ขึ้นมาเพื่อทำหน้าที่บันทึกประวัติการนำเข้าและส่งออกข้อมูล (Transaction Log) โดยตารางนี้จะเก็บรหัสโครงการ (U\_ProjectCode) และเลขที่เอกสาร (U\_DocNum) เป็นฟิลด์อ้างอิง เพื่อให้สามารถตรวจสอบย้อนกลับไปยังข้อมูลต้นทางในระบบ SAP ได้อย่างถูกต้อง

**Entity Relationship Diagram**  
**ระบบจัดการข้อมูลธุรกรรมทางการเงินและแจ้งเดือนสถานะอัตโนมัติ**  
**บนระบบ SAP Business One**



ภาพที่ 3 แผนภาพความสัมพันธ์ของข้อมูล (ER Diagram)

### 3.3 การพัฒนาและติดตั้งระบบ Implementation

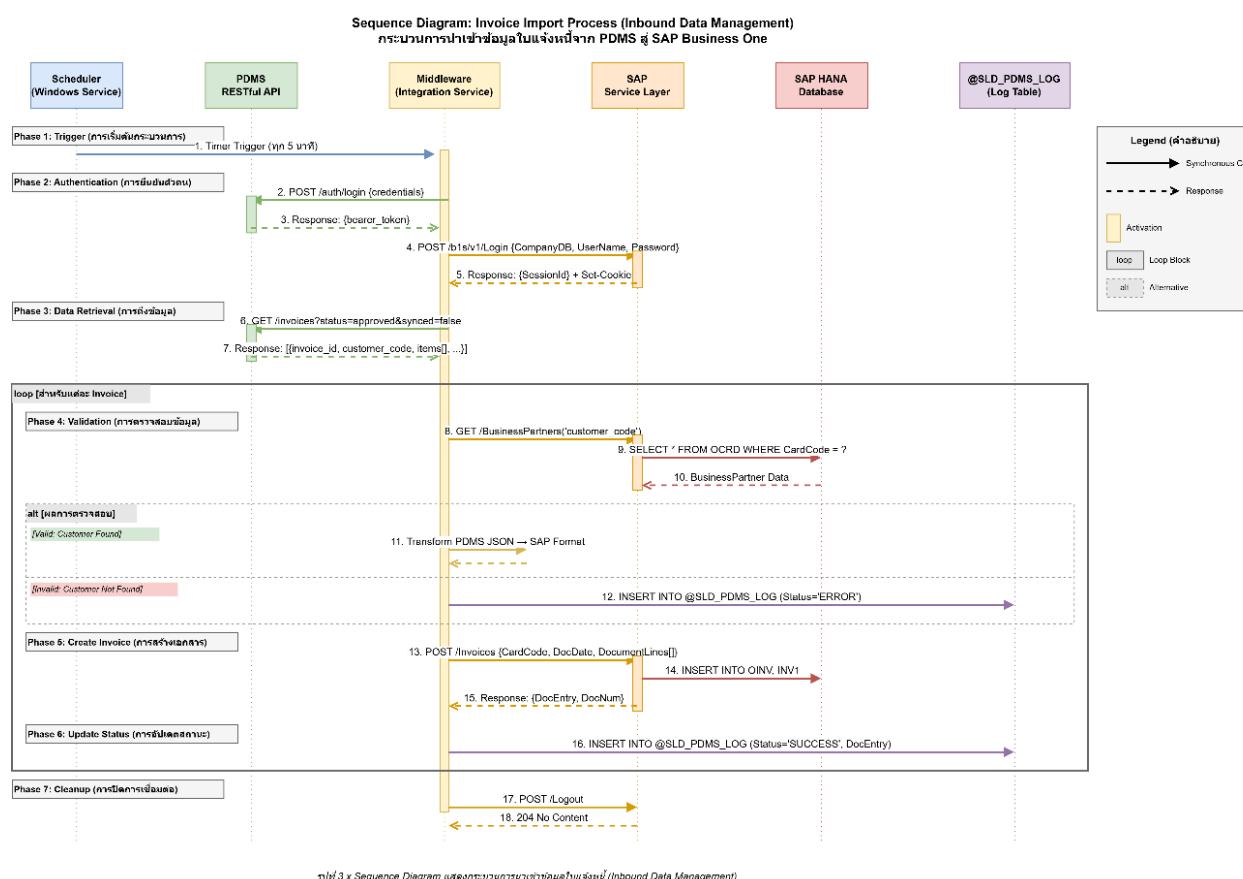
สำหรับการเริ่มขั้นตอนการพัฒนาและติดตั้งระบบ ผู้จัดทำได้นำแบบจำลองระบบที่ออกแบบไว้ในขั้นตอนก่อนหน้า มาใช้เป็นต้นแบบในการเขียนโปรแกรมจริง โดยเครื่องมือหลักที่เลือกใช้คือ โปรแกรม Visual Studio 2015 ร่วมกับภาษา C# บนแพลตฟอร์ม .NET Framework เพื่อพัฒนาแอปพลิเคชันในรูปแบบ Windows Service ซึ่งมีความเสถียรและเหมาะสมกับงานที่ต้องประมวลผลเบื้องหลังตลอดเวลา ทั้งนี้ ได้แบ่งส่วนการพัฒนาโปรแกรมออกเป็น 3 ส่วนงานหลักตามหน้าที่การทำงาน ได้แก่ ส่วนจัดการลำดับการทำงาน (Sequence Logic), ส่วนนำเข้าข้อมูล (Import Logic) และส่วนการอัปเดตสถานะ (Update Status Logic) โดยมีรายละเอียดดังนี้

#### 3.3.1 ลำดับการทำงานของระบบ Sequence Diagram

แผนภาพลำดับการทำงาน (Sequence Diagram) แสดงปฏิสัมพันธ์ (Interaction) ระหว่างองค์ประกอบต่างๆ ของระบบตามลำดับเวลา (Time-ordered):

- 1) การเริ่มทำงาน (Trigger): ระบบทำงานแบบ Background Service โดยมี Scheduler เป็นตัวกระตุ้น (Trigger) การทำงานตามรอบเวลาที่กำหนด (เช่น ทุก 5 นาที)

- 2) การนำเข้าข้อมูล (Inbound): เมื่อถึงเวลา ระบบ (Integration System) จะส่งคำสั่ง GET ไปยัง PDMS API เพื่อดึงข้อมูลโครงการ จากนั้นจะทำการยืนยันตัวตน (Login) และส่งคำสั่ง POST ไปยัง SAP Service Layer เพื่อสร้างข้อมูล BP, Project และ Invoice ตามลำดับ
- 3) การส่งออกข้อมูล (Outbound): ในรอบการทำงานถัดไป ระบบจะทำการค้นหา (Query) ข้อมูลจาก SAP เพื่อตรวจสอบสถานะการจ่ายเงิน หากพบการเปลี่ยนแปลง ระบบจะส่งคำสั่ง PATCH กลับไปยัง PDMS API เพื่ออัปเดตสถานะ
- 4) การบันทึกผล: ในทุกขั้นตอน ระบบจะเขียนข้อมูลลงในฐานข้อมูล Log DB เพื่อบันทึกผลการทำงาน



ภาพที่ 4 แผนภาพลำดับการทำงานแสดงการโต้ตอบระหว่างระบบ

### 3.3.2 ขั้นตอนการนำเข้าข้อมูล (Import Logic)

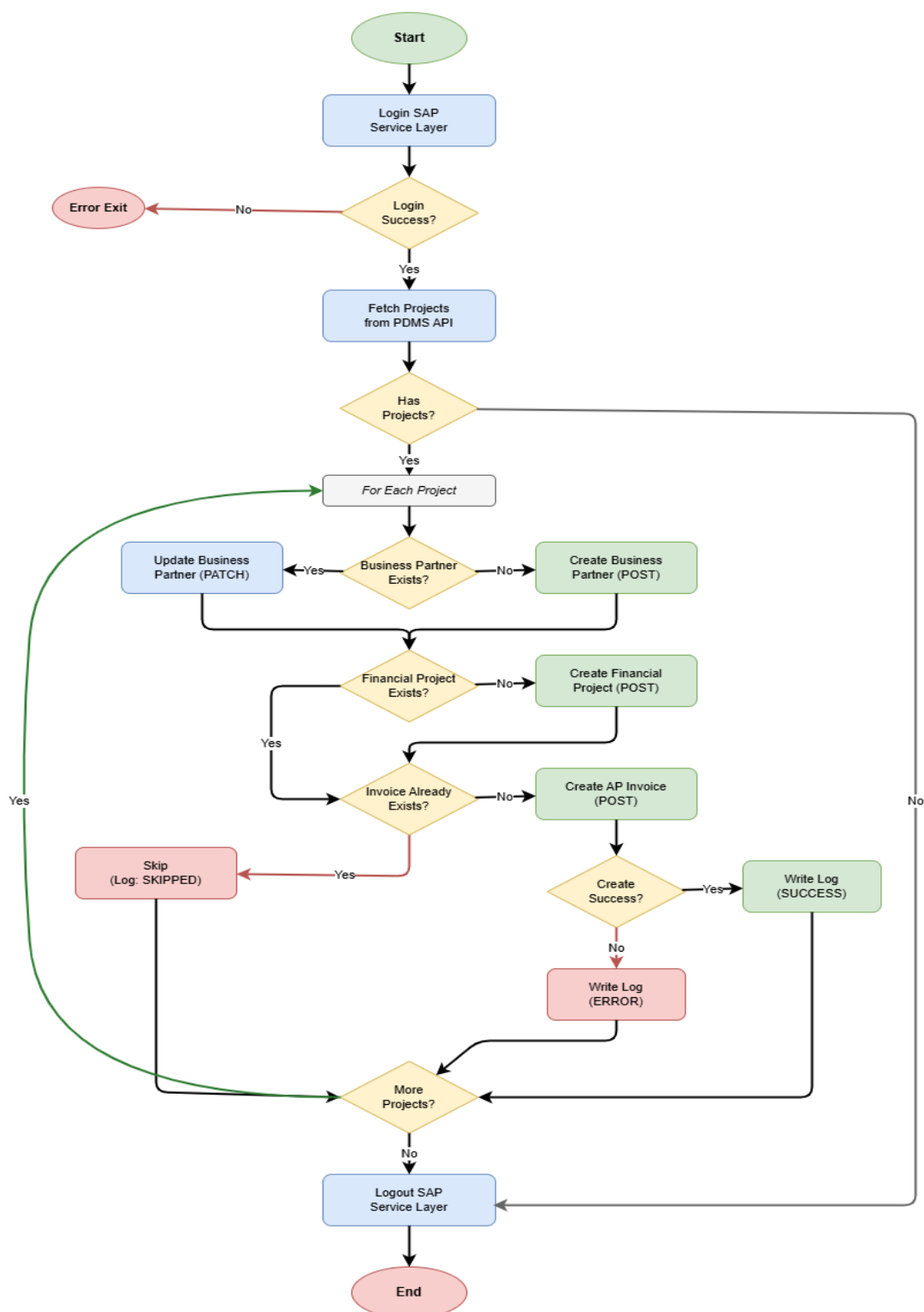
ผังงานกระบวนการนำเข้าข้อมูล (Import Flowchart) แสดงตรรกะการตัดสินใจ (Decision Logic) ของโปรแกรมอย่างละเอียด:

เริ่มต้นจากการเข้าสู่ระบบ SAP (Login) หากไม่สำเร็จจะจบการทำงานทันที หากสำเร็จจะดึงข้อมูลจาก PDMS และเข้าสู่กระบวนการวนซ้ำ (Loop) ที่ละโครงการ ระบบจะทำการตรวจสอบเงื่อนไข "มีอยู่หรือไม่ (Exists Check)" ใน 3 ระดับ เพื่อป้องกันความผิดพลาดของข้อมูล:

1. ระดับที่ 1 การตรวจสอบคู่ค้า (Business Partner Check) ระบบจะค้นหาว่ามีรหัสคู่ค้านี้อยู่ในระบบแล้วหรือไม่ หากตรวจพบว่ามีอยู่แล้ว ระบบจะข้ามขั้นตอนการสร้างไป แต่หากยังไม่มีระบบจะดำเนินการสร้างข้อมูลคู่ค้าใหม่โดยอัตโนมัติ
2. ระดับที่ 2 การตรวจสอบโครงการ (Project Check) ระบบจะดำเนินการตรวจสอบรหัสโครงการในทำนองเดียวกัน หากพบว่ามีข้อมูลโครงการนี้อยู่แล้วก็จะข้ามไป แต่หากไม่พบก็จะสร้างข้อมูลโครงการใหม่ทันที
3. ระดับที่ 3 การตรวจสอบเอกสาร (Invoice Check) เป็นขั้นตอนสำคัญที่สุดเพื่อป้องกันข้อมูลซ้ำซ้อน (Duplicate Data) โดยระบบจะตรวจสอบจากเลขที่เอกสารอ้างอิง หากพบว่าเป็นเอกสารนี้เคยถูกนำเข้าแล้ว ระบบจะข้ามการทำงานในส่วนนี้ไป แต่หากเป็นเอกสารใหม่ ระบบจะดำเนินการสร้างใบสำคัญจ่าย (Invoice) เข้าสู่ระบบ

เมื่อเสร็จสิ้นกระบวนการทั้งหมดในแต่ละรายการ ระบบจะทำการบันทึกผลการทำงาน (Log Result) ว่าสำเร็จหรือล้มเหลว ก่อนจะวนไปทำรายการถัดไปจนครบทุกรายการ

Flowchart: Import Process (PDMS → SAP B1)



ภาพที่ 5 ผังงานกระบวนการนำเข้าข้อมูลและการตัดสินใจ

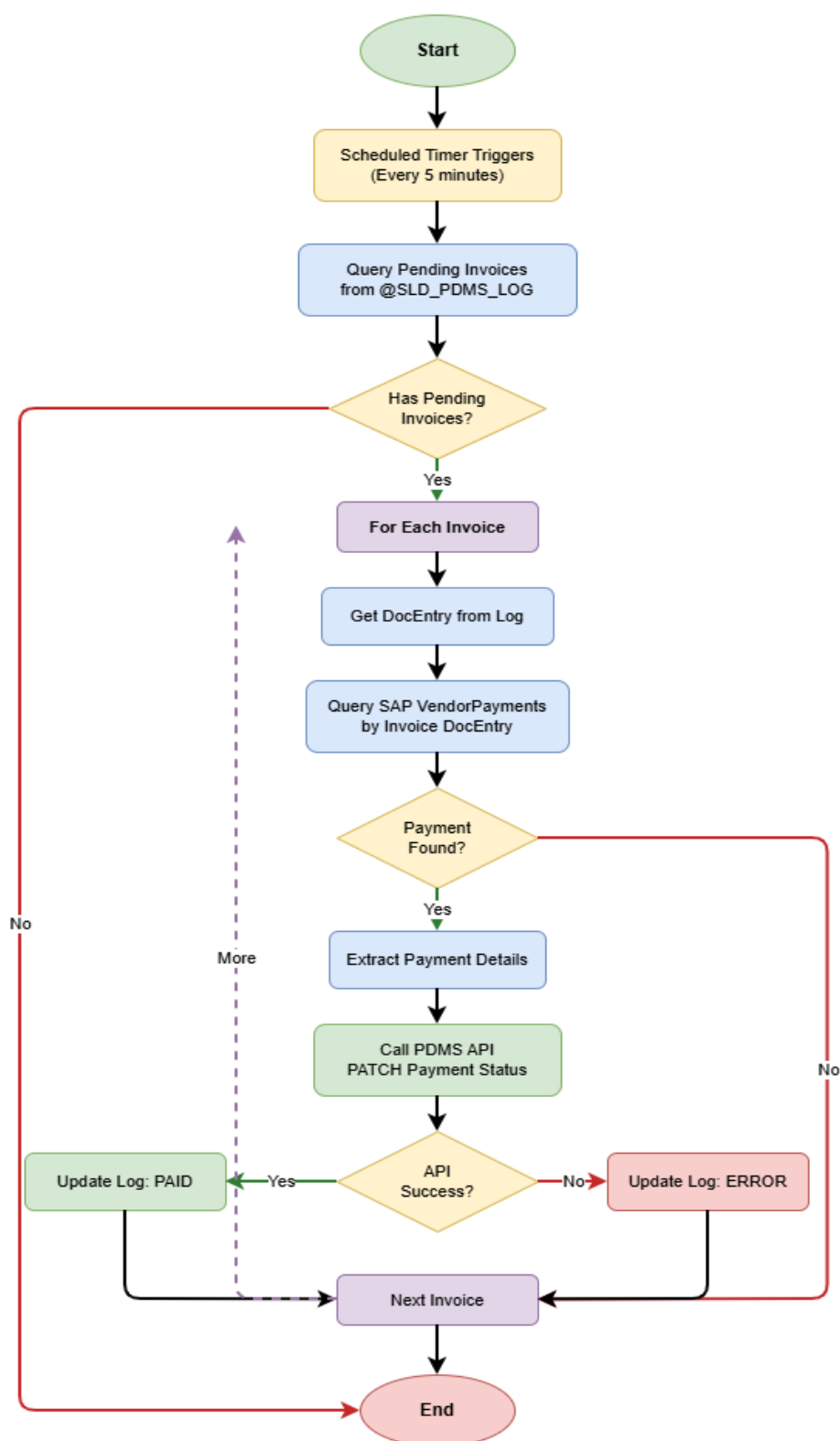
### 3.3.3 ขั้นตอนการอัปเดตสถานะ (Update Status Logic)

ในส่วนของกระบวนการอัปเดตสถานะการชำระเงิน ผู้จัดทำได้ออกแบบตรรกะการทำงาน (Update Status Logic) เพื่อให้ระบบมีความสามารถในการตรวจสอบสถานะการชำระเงินภายในระบบ SAP Business One และดำเนินการส่งข้อมูลยืนยันกลับไปยังระบบ PDMS ได้อย่างถูกต้อง แม่นยำ และเป็นปัจจุบัน โดยกระบวนการดังกล่าวสามารถอธิบายลำดับขั้นตอนการทำงานผ่านแผนภาพได้ดังนี้

ผังงานกระบวนการส่งสถานะกลับ (Flowchart) กระบวนการทำงานเริ่มต้นด้วยการที่ระบบดำเนินการสืบค้นข้อมูลรายการใบสำคัญจ่ายที่ยังคงค้างอยู่ในสถานะรอการชำระเงิน (Pending Status) จากฐานข้อมูลบันทึกประวัติ (System Log) จากนั้นระบบจะนำเลขที่เอกสารอ้างอิง (Document Entry) ไปตรวจสอบสถานะปัจจุบันในตารางข้อมูลการชำระเงิน (OVPM Table) ของระบบ SAP

ในขั้นตอนนี้ หากระบบตรวจสอบพบว่าเอกสารดังกล่าวยังมีสถานะเป็นรอดำเนินการ (Open) ระบบจะข้ามรายการนั้นไปและดำเนินการตรวจสอบรายการถัดไป แต่หากตรวจสอบพบว่าเอกสารมีสถานะเป็นชำระเงินแล้ว (Paid) หรือปิดรายการแล้ว (Closed) ระบบจะดำเนินการดึงข้อมูลสำคัญ ได้แก่ วันที่ชำระเงินและเลขที่เอกสารการโอนเงิน เพื่อส่งข้อมูลยืนยันสถานะ "ชำระแล้ว (Paid)" กลับไปยังระบบ PDMS ผ่านช่องทาง API ทันที และเมื่อกระบวนการสื่อสารเสร็จสิ้นสมบูรณ์ ระบบจะทำการบันทึกสถานะการทำงานลงในฐานข้อมูลประวัติว่าเป็น "ดำเนินการเสร็จสิ้น (Completed)" เพื่อป้องกันการนำข้อมูลเดิมกลับมาประมวลผลซ้ำซ้อนในอนาคต

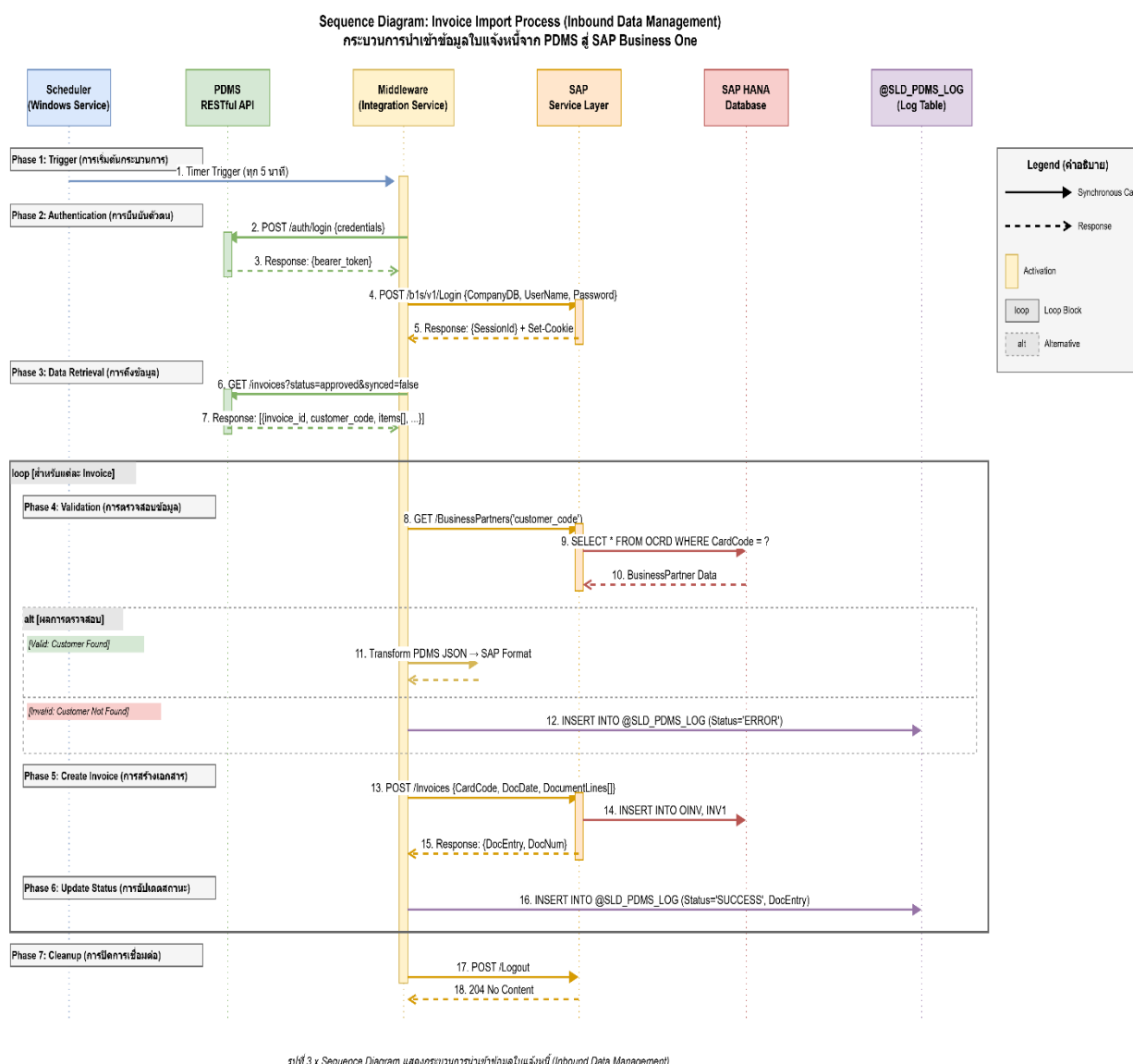
### Flowchart: Update Payment Status (Outbound Process)



ภาพที่ 6 ฟังก์ชันกระบวนการตรวจสอบและส่งสถานะกลับ



แผนภาพลำดับการทำงานกระบวนการอัปเดตสถานะ (Sequence Diagram) เพื่อให้เห็นภาพขั้นตอนการสื่อสารระหว่างระบบที่ชัดเจนยิ่งขึ้นแผนภาพนี้แสดงให้เห็นถึงการทำงานร่วมกันระหว่าง Scheduler, Integration Service, SAP Database และ PDMS API ในการตรวจสอบสถานะการชำระเงิน (Check Payment Status) เมื่อพบรายการที่ชำระแล้ว ระบบจะทำการส่งข้อมูลยืนยัน (Ack) กลับไปยัง PDMS ทันที



ภาพที่ 7 แผนภาพลำดับการทำงานกระบวนการอัปเดตสถานะการชำระเงิน

### 3.4 เครื่องมือและเทคโนโลยีที่ใช้ในการพัฒนา

ในการพัฒนาโครงการนี้ได้มีการเลือกใช้เครื่องมือและเทคโนโลยีหลายอย่างมาประกอบกัน โดยพิจารณาจากความเหมาะสมกับงานที่ต้องทำ ความเสถียรของเทคโนโลยี และความสามารถในการทำงานร่วมกับระบบ SAP Business One ที่มีอยู่แล้ว โดยมีรายละเอียดดังนี้

#### 3.4.1 ภาษาและแพลตฟอร์มสำหรับการพัฒนา

เลือกใช้ภาษา C# บนแพลตฟอร์ม .NET Framework version 4.5.2 เป็นหลักในการพัฒนา เหตุผลที่เลือกใช้ภาษา C# เนื่องจากเป็นภาษาที่ SAP รองรับอย่างเป็นทางการสำหรับการพัฒนาส่วนขยายและการเชื่อมต่อต่างๆ นอกจากนี้ภาษา C# ยังเป็นภาษาที่มีเฟรมเวิร์คและไลบรารีครบครัน เหมาะสมกับการพัฒนาแอปพลิเคชันสำหรับองค์กร สำหรับ .NET Framework version 4.5.2 นั้นถึงแม้จะไม่ใชเวอร์ชันล่าสุด แต่เป็นเวอร์ชันที่มีความเสถียรสูง มีการสนับสนุนระยะยาวจากผู้พัฒนา และที่สำคัญคือรองรับการทำงานบน Windows Server ที่ใช้ในองค์กรได้ดี รวมถึง SAP Business One SDK ที่ใช้ก็รองรับ .NET Framework เวอร์ชันนี้อย่างสมบูรณ์

#### 3.4.2 สภาพแวดล้อมการพัฒนา

ใช้ Microsoft Visual Studio 2015 และ 2017 เป็นเครื่องมือหลักในการเขียนโค้ด เหตุผลที่เลือก Visual Studio เนื่องจากเป็นเครื่องมือที่มีฟีเจอร์ครบครัน มีระบบช่วยเขียนโค้ดอัตโนมัติ (IntelliSense) มีเครื่องมือตรวจหาข้อผิดพลาดที่ทรงพลัง และรองรับการพัฒนาทั้ง Windows Service และ Windows Application ได้อย่างสมบูรณ์ Visual Studio ยังมีการทำงานร่วมกับ SAP Business One SDK อีกด้วย ทำให้สามารถติดตั้งการอ้างอิงไปยัง SAPbouiCOM และไลบรารีอื่นๆ ของ SAP ได้ง่าย รวมถึงมีเครื่องมือตรวจสอบข้อผิดพลาดที่ช่วยในการตรวจสอบส่วนขยายที่ทำงานร่วมกับ SAP Client ได้

#### 3.4.3 ไลบรารีและชุดคำสั่งสำหรับการเชื่อมต่อระบบ

ในการพัฒนาระบบเชื่อมต่อข้อมูลได้มีการใช้ไลบรารีและชุดคำสั่งที่สำคัญดังนี้ HttpClient เป็นคลาสที่มาพร้อมกับ .NET Framework สำหรับการส่ง HTTP requests และรับ responses โดยถูกออกแบบมาให้ทำงานแบบไม่ประสานเวลา (Asynchronous) ได้ดี มี

ประสิทธิภาพสูง และรองรับฟีเจอร์ HTTP สมัยใหม่ต่างๆ ในโครงการนี้ใช้ HttpClient เป็นหลักในการเรียก RESTful APIs ทั้งของ SAP Service Layer และ PDMS โดยตั้งค่าให้ใช้ TLS 1.2 เพื่อความปลอดภัย Newtonsoft.Json หรือที่รู้จักกันในชื่อ Json.NET เป็นไลบรารีที่ได้รับ ความนิยมสูงสุดสำหรับการทำงานกับ JSON ใน .NET โดยมีความสามารถในการแปลงวัตถุ (Objects) เป็น JSON string และแปลง JSON string กลับมาเป็นวัตถุได้อย่างรวดเร็วและมี ประสิทธิภาพ ในโครงการนี้ใช้ Json.NET เป็นหลักในการแปลงข้อมูลระหว่าง C# objects กับ รูปแบบ JSON ที่ใช้สื่อสารกับ APIs ต่างๆ

### 3.4.4 โครงสร้างโปรเจค

การออกแบบโครงสร้างโปรเจคเป็นสิ่งสำคัญที่จะช่วยให้ระบบมีความเป็นระเบียบง่ายต่อการบำรุงรักษา และสามารถพัฒนาต่อยอดได้ในอนาคต ระบบที่พัฒนาในโครงการนี้จึงได้รับการออกแบบให้แบ่งแยกความรับผิดชอบออกเป็น 3 โปรเจคย่อย โดยแต่ละโปรเจคมีหน้าที่ และวัตถุประสงค์ที่ชัดเจน ดังนี้

- 1) SldImportPdms.Core เป็นโปรเจคไลบรารีหลักที่รวบรวมตรรกะทางธุรกิจ และฟังก์ชันหลักทั้งหมดของระบบ โดยออกแบบให้เป็นไลบรารีที่ใช้ร่วมกันซึ่ง สามารถถูกเรียกใช้ได้จากทั้งโปรเจค Service และโปรเจคส่วนขยาย วิธีนี้ช่วย ป้องกันการเขียนโค้ดซ้ำซ้อนและทำให้การบำรุงรักษาทำได้ง่ายขึ้น เมื่อมีการ แก้ไขหรือเพิ่มฟีเจอร์ก็แค่แก้ที่ Core project เดียวแล้วโปรเจคอื่นๆ ก็จะได้รับ การอัปเดตไปด้วย
- 2) SldImportPdms.Service เป็นโปรเจค Windows Service ที่ทำหน้าที่รัน อยู่เบื้องหลังและทำงานตามรอบเวลาที่กำหนด เช่น ทุก 5 นาที โดยจะ เรียกใช้ฟังก์ชันจาก Core เพื่อดึงข้อมูลจาก PDMS และสร้างเอกสารใน SAP
- 3) SldImportPdms.Addon เป็นโปรเจคส่วนขยายที่ทำงานบนหน้าจอ SAP Client โดยเพิ่มปุ่มอัปเดต PDMS บนหน้าจอ Outgoing Payment เพื่อให้ ผู้ใช้สามารถส่งสถานะการชำระเงินกลับไป PDMS ได้ทันทีโดยไม่ต้องรอรอบ การทำงานของ Service

บทที่ 4

ผลการดำเนินงานโครงการ

จากการพัฒนาระบบจัดการข้อมูลธุรกรรมทางการเงินและแจ้งเตือนสถานะอัตโนมัติ บนระบบเอสเอพีบีเอสซีเนสวัน คณะผู้จัดทำได้ดำเนินการพัฒนาจนเสร็จสมบูรณ์ตามขอบเขตงาน และได้ดำเนินการทดสอบประสิทธิภาพการทำงานของระบบ ซึ่งมีรายละเอียดผลการดำเนินงานในด้านต่างๆ ดังนี้

4.1 ผลการทดสอบระบบ (System Testing)

จากการพัฒนาระบบเชื่อมต่อข้อมูล PDMS กับ SAP Business One ผู้จัดทำได้ดำเนินการทดสอบระบบตามแผนการทดสอบ (Test Plan) ที่กำหนดไว้ โดยแบ่งการทดสอบออกเป็นกรณีทดสอบ (Test Cases) ต่างๆ เพื่อตรวจสอบความถูกต้องของฟังก์ชันการทำงานหลัก (Functional Testing) และความสามารถในการจัดการข้อผิดพลาด(Exception Handling) ผลการทดสอบสามารถสรุปรายละเอียดได้ดังตาราง

กรณีทดสอบ(Test Case)	รายละเอียดการทดสอบ ( Test Description)	ผลลัพธ์ที่คาดหวัง (Expected Result)	ผลการทดสอบ (Actual Result)	สถานะ (Status)
TC-01: Data Import	นำเข้าข้อมูลโครงการจาก PDMS ที่มีข้อมูลครบถ้วนถูกต้อง	ทำรายการชำระเงิน ( Outgoing Payment) ใน SAP และรอรอบการ Sync	ระบบสร้างเอกสารได้ครบถ้วน ข้อมูลถูกต้อง 100%	
TC-02: Duplicate Data Check	ทำการนำเข้าข้อมูลชุดเดิมซ้ำเป็นครั้งที่ 2	ระบบต้องตรวจสอบพบว่าเป็นข้อมูลซ้ำและข้ามการสร้างเอกสาร โดยไม่มี Error	ระบบตรวจสอบพบข้อมูลซ้ำและข้ามการทำงานได้อย่างถูกต้อง	
TC-03: Missing Field Validation	จำลองการส่งข้อมูลที่ขาดฟิลด์จำเป็น (เช่น ไม่มี Tax ID)	ระบบต้องไม่สร้างเอกสารที่สมบูรณ์ และบันทึกข้อผิดพลาดลงใน Log	ระบบปฏิเสธการสร้างเอกสารและบันทึก Error Log ระบุสาเหตุชัดเจน	
TC-04: Network Interruption	จำลองสถานการณ์อินเทอร์เน็ตหลุดขณะเชื่อมต่อ API	ระบบต้องหยุดการทำงานโดยไม่เกิดการ Crash และ	ระบบแจ้งเตือน Connection Timeout และหยุด	

		สามารถเริ่มทำงานใหม่ได้ในรอบถัดไป	การทำงานอย่างปลอดภัย	
<b>TC-04: Network Interruption</b>	ทำรายการชำระเงิน (Outgoing Payment) ใน SAP และรอรอบการ Sync	สถานะในระบบ PDMS ต้องเปลี่ยนเป็น "Paid" และระบุวันที่ชำระเงินถูกต้อง	สถานะใน PDMS เปลี่ยนแปลงถูกต้องภายในเวลาไม่เกิน 5 นาที	

ตารางที่ 2 ผลการทดสอบการทำงานของระบบตามกรณีทดสอบ

## 4.2 ผลการพัฒนาระบบนำเข้าและส่งออกข้อมูล

ผลจากการพัฒนาโปรแกรมในรูปแบบ Windows Service ทำให้ได้ระบบที่สามารถทำงานอยู่เบื้องหลังได้อย่างมีประสิทธิภาพ โดยมีส่วนติดต่อผู้ใช้และผลลัพธ์ที่สำคัญดังนี้

- 1) ส่วนแสดงสถานะการทำงาน (Console/Log Monitor) เนื่องจากระบบทำงานแบบ Background Service การตรวจสอบสถานะจึงทำผ่าน Log File หรือ Console Monitor (ในโหมด Debug) ดังภาพ ซึ่งแสดงให้เห็นถึงขั้นตอนการทำงานแบบ Real-time ตั้งแต่การเริ่มต้นเชื่อมต่อ การดึงข้อมูล การตรวจสอบข้อมูล จนถึงผลลัพธ์การสร้างเอกสารที่สำเร็จ (Success) ซึ่งช่วยให้ผู้ดูแลระบบสามารถติดตามสถานะและวิเคราะห์ปัญหาได้อย่างรวดเร็ว

หน้าจอแสดงสถานะการทำงานของโปรแกรม Background Service แสดง Log การนำเข้าข้อมูลสำเร็จ

- 2) ผลลัพธ์เอกสารในระบบ SAP Business One เมื่อระบบทำการนำเข้าข้อมูลสำเร็จ ข้อมูลเหล่านั้นจะถูกแปลงเป็นเอกสาร A/P Invoice ในระบบ SAP Business One โดยอัตโนมัติ ดังภาพที่ 4.2 ซึ่งแสดงให้เห็นว่าข้อมูลสำคัญทางบัญชี ได้แก่ รหัสผู้ขาย (Vendor Code), วันที่เอกสาร

(Posting Date), รายละเอียดสินค้า/บริการ (Item/Service Description), การเชื่อมโยงรหัสโครงการ (Project Code), และยอดเงินรวม (Total Amount) ถูกบันทึกอย่างถูกต้องแม่นยำและครบถ้วนตามหลักการบัญชี พร้อมสำหรับการนำไปประมวลผลต่อในขั้นตอนการจ่ายเงิน

#### 4.3 การเปรียบเทียบประสิทธิภาพก่อนและหลังใช้งาน

จากการดำเนินการเก็บรวบรวมสถิติการปฏิบัติงานจริงในระยะเวลา 1 เดือน เพื่อนำมาวิเคราะห์เปรียบเทียบประสิทธิผล (Effectiveness) และประสิทธิภาพ (Efficiency) ระหว่างกระบวนการทำงานแบบดั้งเดิม (Manual Process) ที่พึ่งพาการปฏิบัติงานของเจ้าหน้าที่ กับกระบวนการทำงานรูปแบบใหม่ที่ใช้ระบบอัตโนมัติ (Automated System) ที่ผู้จัดทำได้พัฒนาขึ้น สามารถจำแนกผลการวิเคราะห์ในมิติต่างๆ ได้อย่างชัดเจน ทั้งในด้านระยะเวลา ความถูกต้อง และการบริหารทรัพยากร ดังแสดงรายละเอียดในตาราง

หัวข้อเปรียบเทียบ	ระบบเดิม (Manual)	ระบบใหม่ (Automated)	การวิเคราะห์ผลลัพธ์ (Analysis)
ระยะเวลาการทำงาน	15-20 นาที / เอกสาร	25-30 วินาที / เอกสาร	ลดลง 96% ระบบสามารถประมวลผลได้รวดเร็วกว่ามนุษย์มากช่วยลดปัญหาคอขวดของงานเอกสาร
ระยะเวลาการทำงาน	5-10% (Human Error)	0% (Systematic)	ความถูกต้อง 100% ระบบบริหารจัดการความผิดพลาดจากการพิมพ์ผิดหรือดูข้อมูลผิดพลาดได้แม่นยำขึ้น
ความทันสมัยของข้อมูล	ล่าช้า 1-2 วัน (Batch)	ทันที (Near Real-time)	เป็นปัจจุบัน ผู้บริหารสามารถดูรายงานสถานะการเงินที่สะท้อนความเป็นจริงได้ตลอดเวลา
ทรัพยากรบุคคล	ใช้พนักงาน 2 คน (ทำ/ตรวจ)	ไม่ต้องใช้ (System)	ลดต้นทุนแรงงานสามารถนำพนักงานไป

			ปฏิบัติงานเชิงวิเคราะห์ที่มีมูลค่าสูงกว่าได้
<b>ก า ร ต ร ว จ ส อ บ ย้อนหลัง</b>	ค้นหาจากแฟ้มเอกสาร	ค้นหาจาก System Log	<b>ตรวจสอบได้ง่าย</b> มีหลักฐานการทำงาน (Audit Trail) ที่ชัดเจนในทุกขั้นตอน

จากตารางเปรียบเทียบ แสดงให้เห็นอย่างชัดเจนว่าการนำระบบอัตโนมัติมาใช้งาน สามารถเพิ่มประสิทธิภาพในการทำงานได้อย่างก้าวกระโดด ทั้งในแง่ของความเร็ว ความถูกต้อง และความคุ้มค่า ซึ่งตอบโจทย์วัตถุประสงค์หลักของโครงการทุกประการ

## บทที่ 5

### สรุปผลโครงการและข้อเสนอแนะ

การดำเนินโครงการสหกิจศึกษาเรื่อง ระบบเชื่อมต่อข้อมูล PDMS กับ SAP Business One มีวัตถุประสงค์เพื่อพัฒนาระบบจัดการข้อมูลธุรกรรมทางการเงินแบบอัตโนมัติ เพื่อลดความซ้ำซ้อนและเพิ่มประสิทธิภาพในการปฏิบัติงาน จากผลการดำเนินงานและการทดสอบระบบ สามารถสรุปผลอภิปรายผล ปัญหาและอุปสรรค รวมถึงข้อเสนอแนะได้ดังนี้

#### 5.1 สรุปผลการดำเนินงาน โครงการ

ระบบจัดการข้อมูลธุรกรรมทางการเงินและแจ้งเตือนสถานะอัตโนมัติ ประสบความสำเร็จตามวัตถุประสงค์ที่ตั้งไว้ โดยสามารถพัฒนาระบบตัวกลาง (Middleware) ที่เชื่อมโยงข้อมูลระหว่างสองระบบได้อย่างมีประสิทธิภาพ ช่วยลดเวลาในการทำงานของพนักงานบัญชีได้อย่างมีนัยสำคัญ และขจัดปัญหาความผิดพลาดจากการคีย์ข้อมูลด้วยมือ ทำให้ข้อมูลมีความถูกต้องน่าเชื่อถือและเป็นปัจจุบัน

#### 5.2 ปัญหาและอุปสรรค

- 1) ความซับซ้อนของโครงสร้างข้อมูล ข้อมูลในระบบ PDMS และ SAP มีโครงสร้างที่แตกต่างกันมาก ทำให้ต้องใช้เวลาในการทำ Data Mapping และปรับจูน Logic การแปลงข้อมูล
- 2) การจัดการ Session: ในช่วงแรกพบปัญหา Service Layer Session หหมดอายุเมื่อไม่ได้ใช้งานนานๆ จึงต้องพัฒนาระบบ Auto Re-login เพิ่มเติม
- 3) ข้อจำกัดด้าน Network: การเชื่อมต่อระหว่าง Server ของลูกค้ากับภายนอกมีความเข้มงวดด้าน Firewall ทำให้ต้องประสานงานฝ่าย IT เพื่อเปิด Port ที่จำเป็น

#### 5.3 ข้อเสนอแนะ จากการดำเนินโครงการและทดสอบระบบ

มีข้อเสนอแนะเพื่อการพัฒนาต่อยอดดังนี้

- 1) ควรเพิ่มระบบ Dashboard แสดงสถานะการทำงานของ Service ในรูปแบบกราฟิก เพื่อให้ผู้ดูแลระบบตรวจสอบได้ง่ายขึ้น



- 2) ควรพิจารณาเพิ่มการแจ้งเตือนผ่านช่องทางอื่นๆ เช่น Email หรือ LINE Notify ในกรณีที่เกิดข้อผิดพลาดร้ายแรง (Critical Error)
- 3) ควรมีการสำรองข้อมูล Log File อย่างสม่ำเสมอ เพื่อใช้ในการตรวจสอบย้อนหลังในระยะยาว

บรรณานุกรม

## บรรณานุกรม

"[คลิกที่นี่เพื่อเริ่มพิมพ์รายละเอียด เขียนอ้างอิงแบบ APA]"

ภาคผนวก

ภาคผนวก ก

[คลิกที่นี่เพื่อเริ่มพิมพ์รายละเอียด]





ภาคผนวก ข

[คลิกที่นี่เพื่อเริ่มพิมพ์รายละเอียด]





ประวัติย่อผู้ทำโครงการ

## ประวัติผู้ทำโครงการ

ชื่อ นามสกุล [ชื่อผู้วิจัย]  
 วัน เดือน ปีเกิด วันที่ [วันที่ เดือน ปี พ.ศ.ผู้วิจัย]  
 จังหวัด และประเทศที่เกิด อำเภอ[ชื่ออำเภอ] จังหวัด[ชื่อจังหวัด]  
 ประวัติการศึกษา พ.ศ. [ปีพ.ศ.] [ระดับการศึกษาปัจจุบันที่สำเร็จ]  
 [อักษรย่อวุฒิการศึกษา] [สาขาวิชา] [สถาบัน]  
 (ประวัติการศึกษา ควรระบุตั้งแต่ มัธยมศึกษาตอนต้น เป็นต้นไป)  
 ตำแหน่ง สถานที่ทำงาน [ตำแหน่งงานของผู้วิจัย] [สถานที่ทำงานผู้วิจัย]  
 ที่อยู่ที่สามารถติดต่อได้ "[ระบุที่อยู่ปัจจุบันของผู้วิจัย ที่สามารถติดต่อได้]"  
 รางวัลเรียนดี พุทธวิทย์ และทุนการศึกษา  
 "[ระบุชื่อรางวัลเรียนดี/ พุทธวิทย์ /ทุนการศึกษาที่ผู้วิจัยได้รับ (ถ้ามี)]"

## ผลงานวิจัย

[ระบุผลงานวิจัยตามรูปแบบหนังสืออ้างอิง]