

บทที่ 3

ขั้นตอนการดำเนินงาน

บทนี้จะกล่าวถึงขั้นตอนในการดำเนินงานของโครงการปริญญาโท ซึ่งจะทำให้ทราบถึงการวิเคราะห์และการออกแบบระบบโดยละเอียดว่ามีแนวทางในการทำงานหรือขั้นตอนในการทำงานของระบบอย่างไร โดยขั้นตอนในการดำเนินงานมีรายละเอียดดังนี้

3.1 กรอบการดำเนินงาน

กรอบการทำงานนี้จะแสดงขั้นตอนการพัฒนา ระบบ แอปพลิเคชันการจองอัตราแลกเปลี่ยนสกุลเงินไทย ซึ่งมีขั้นตอนการดำเนินงานดังนี้



ภาพประกอบที่ 3.1 กรอบการดำเนินงาน

3.1.1 คำอธิบาย

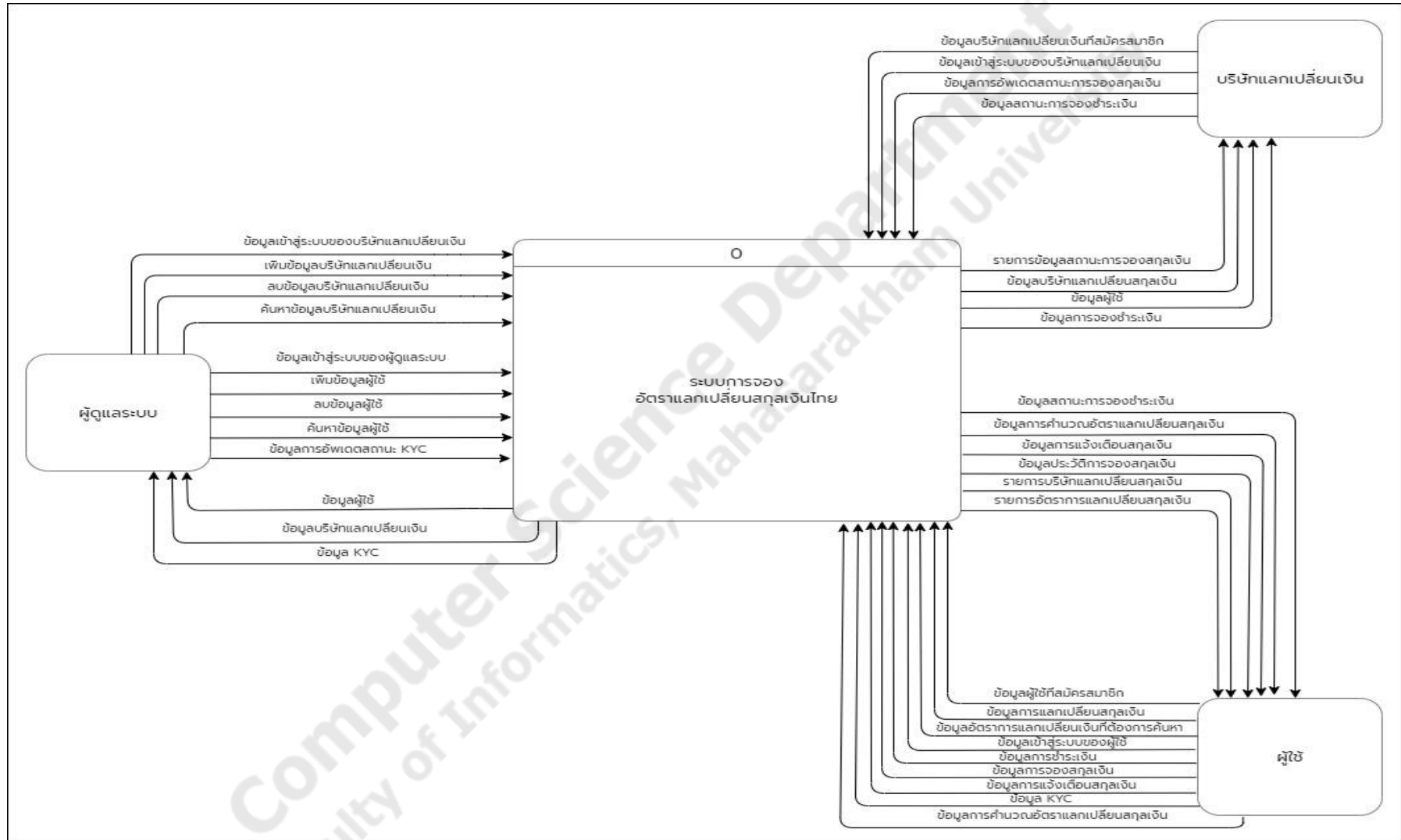
- 1) เก็บข้อมูล ทำการเก็บข้อมูลทั้งหมดของผู้ดูแลระบบ บริษัทแลกเปลี่ยนสกุลเงิน ผู้ใช้ ข้อมูลKYC อัตราแลกเปลี่ยนสกุลเงิน จองสกุลเงินเพื่อทำการบันทึกลงฐานข้อมูล

- 2) ฐานข้อมูล นำข้อมูลจากการเก็บข้อมูลมาบันทึกลงในฐานข้อมูลโดยจะมีฐานข้อมูล 2 ส่วนคือ
 - 2.1 Firebase เก็บข้อมูลทั้งหมดของผู้ดูแลระบบ บริษัทแลกเปลี่ยนสกุลเงิน ผู้ใช้ ข้อมูลKYC อัตราแลกเปลี่ยนสกุลเงินลงในฐานข้อมูล Firebase
 - 2.2 Blockchain เก็บข้อมูลการจองสกุลเงินของผู้ใช้ลงในฐานข้อมูล Blockchain
- 3) แอปพลิเคชัน พัฒนาโดย Flutter Framework ซึ่งเป็นภาษา Dart ในการเขียนโปรแกรมและพัฒนาแอปพลิเคชัน โดยจะใช้การเชื่อมต่อกับ Infura sepolia API ผ่านเครื่องมือ Firebase, Blockchain, Sepolia Test Network
- 4) ทดสอบระบบ ทำการทดสอบระบบผ่าน Sepolia Test Network เพื่อทดสอบและพัฒนาโดยการใช้ Test Network ไม่ทำให้เกิดการเสียค่าใช้จ่ายจึงเหมาะแก่การทดสอบดูสถานะและข้อมูลการจองสกุลเงินของลูกค้า

3.2 การไหลของข้อมูล(Data Flow Diagram)

3.2.1 แผนภาพคอนเท็กซ์(Context Diagram)

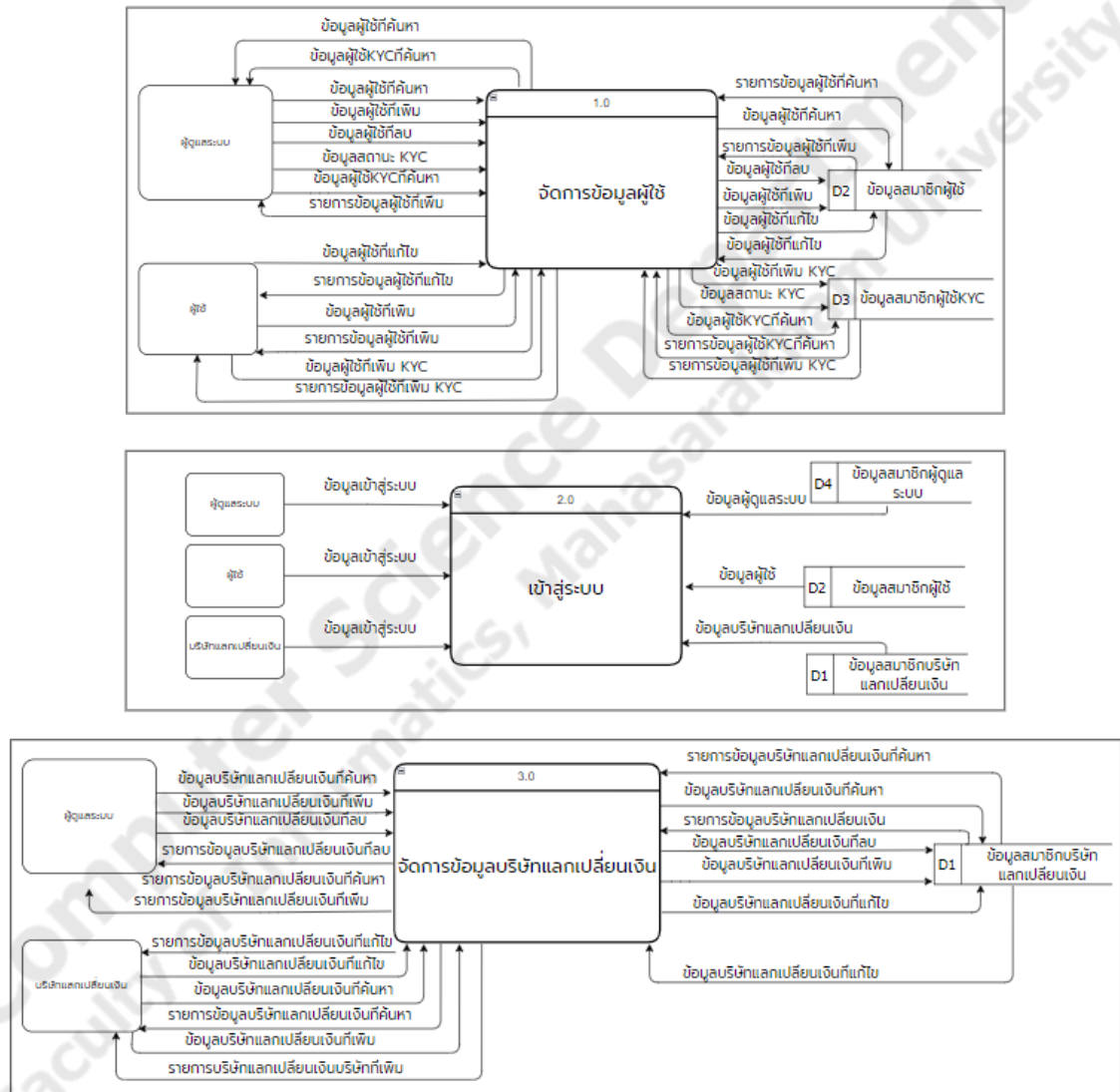
แผนภาพกระแสข้อมูลที่แสดงภาพรวมการทำงานของระบบที่มีความสัมพันธ์กับผู้ดูแลระบบ บริษัทแลกเปลี่ยน และผู้ใช้ เพื่อแสดงถึงขอบเขตของระบบการจองอัตราแลกเปลี่ยนสกุลเงินไทย



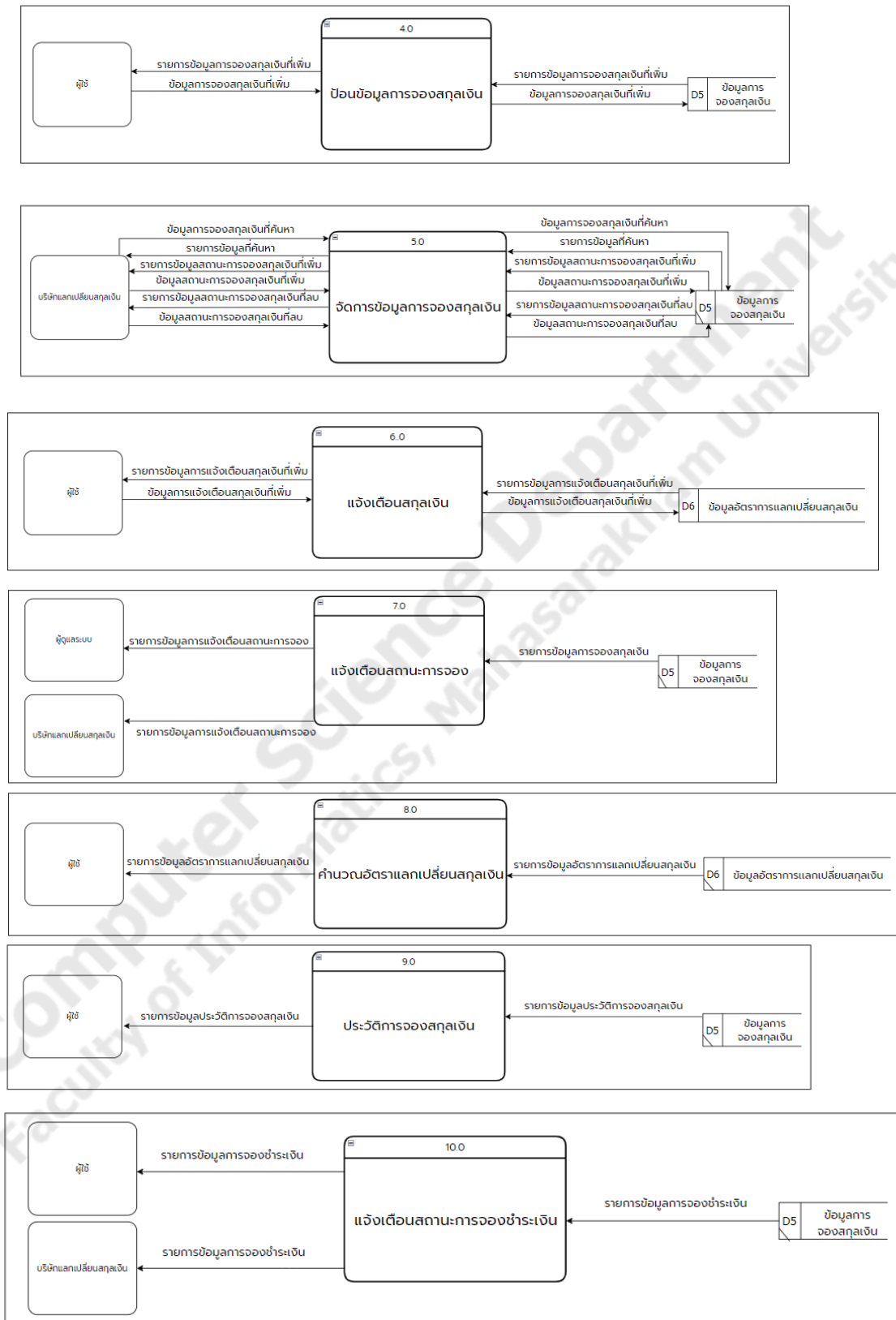
ภาพประกอบที่ 3.2 แผนภาพคอนแท็กซ์ (Context Diagram)

3.2.2 Data Flow Diagram Level 1

แผนภาพกระแสข้อมูลในระดับที่แสดงขั้นตอนการทำงานหลักทั้งหมด (Process หลัก) ของ ระบบ แสดงทิศทางไหลของ Data Flow และแสดงรายละเอียดของแหล่งจัดเก็บข้อมูล (Data Store)



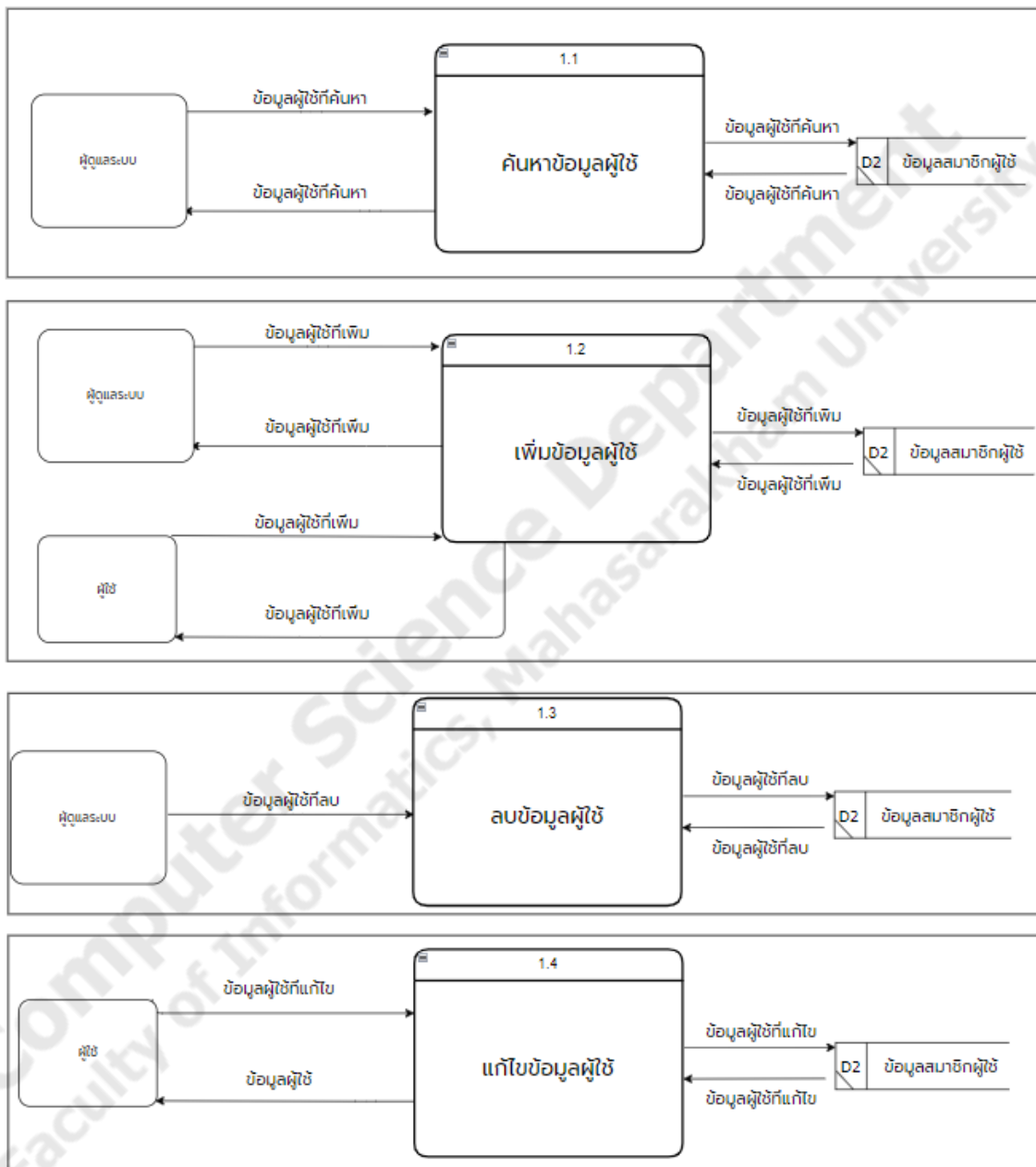
ภาพประกอบที่ 3.3 แผนภาพกระแสข้อมูลเลเวล-1



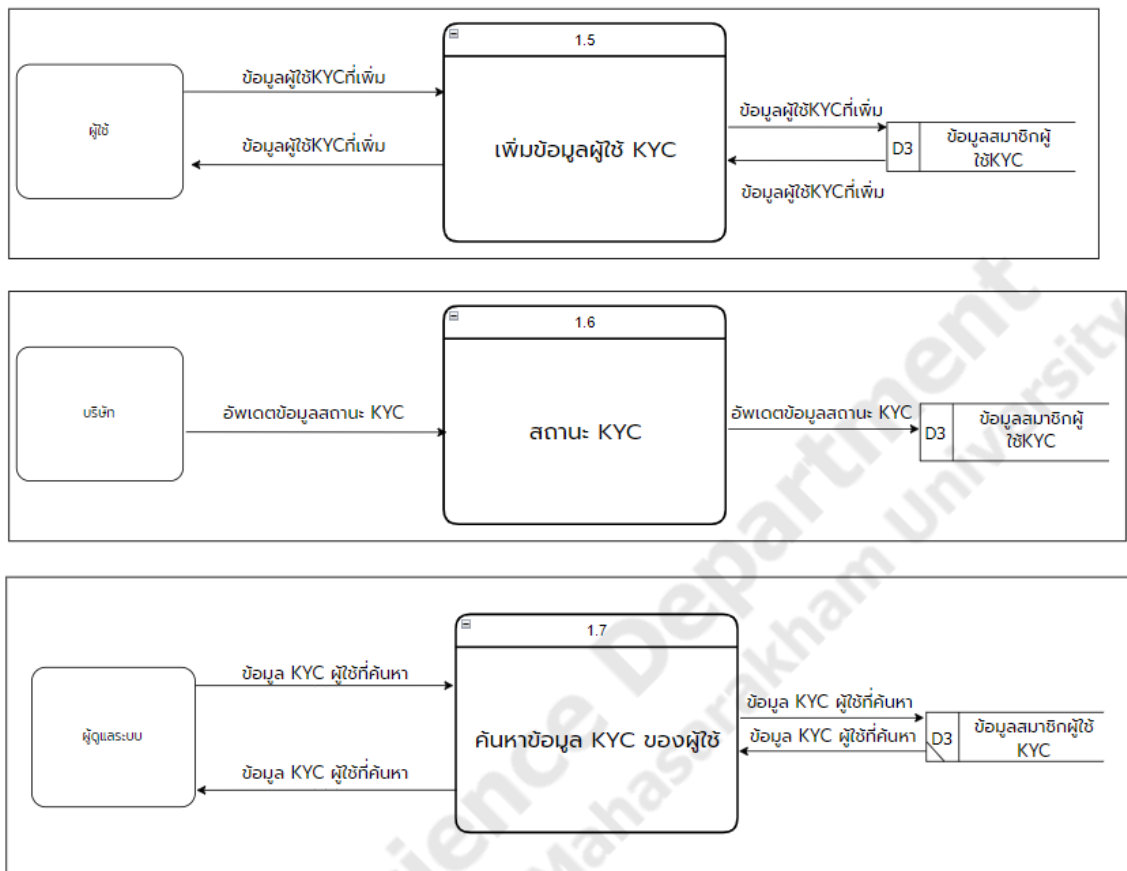
ภาพประกอบที่ 3.3 แผนภาพกระแสข้อมูลเลเวล-1 (ต่อ)

3.2.3 Data Flow Diagram Level 2

3.2.3.1 แผนภาพการไหลข้อมูลระดับ 2 กระบวนการที่ 1



ภาพประกอบที่ 3.4 แผนภาพกระแสข้อมูลเลเวล-2 กระบวนการที่ 1



ภาพประกอบที่ 3.4 แผนภาพกระแสน้ำข้อมูลเลเวล-2 กระบวนการที่ 1 (ต่อ)

3.2.3.2 แผนภาพการไหลข้อมูลระดับ 2 กระบวนการที่ 3



ภาพประกอบที่ 3.5 แผนภาพกระแสข้อมูลเลเวล-2 กระบวนการที่ 3

3.2.3.3 แผนภาพการไหลข้อมูลระดับ 2 กระบวนการที่ 5



ภาพประกอบที่ 3.6 แผนภาพกระแสข้อมูลเลเวล-2 กระบวนการที่ 5

3.3 พจนานุกรมข้อมูล (Data Dictionary)

3.3.1 อธิบายอินทิตีภายนอก (External Entity Description)

ตารางที่ 3.1 External Entity Description

Name	Description	Input Data Flow	Output Data Flow
ผู้ดูแลระบบ	เป็นผู้จัดการข้อมูลผู้ใช้และบริษัทแลกเปลี่ยนสกุลเงิน	<ul style="list-style-type: none"> - ข้อมูลเข้าสู่ระบบ - ข้อมูลผู้ใช้ที่เพิ่ม - ข้อมูลผู้ใช้ที่ลบ - ข้อมูลผู้ใช้ที่แก้ไข - ข้อมูลผู้ใช้ที่ค้นหา - ข้อมูลบริษัทแลกเปลี่ยนสกุลเงินที่เพิ่ม - ข้อมูลบริษัทแลกเปลี่ยนสกุลเงินที่ลบ 	<ul style="list-style-type: none"> - ข้อมูลประเภทสมาชิก - ข้อมูลผู้ใช้ที่เพิ่ม - ข้อมูลผู้ใช้ที่ลบ - ข้อมูลผู้ใช้ที่แก้ไข - รายการข้อมูลผู้ใช้ที่ค้นหา - ข้อมูลบริษัทแลกเปลี่ยนสกุลเงินที่เพิ่ม - ข้อมูลบริษัทแลกเปลี่ยนสกุลเงินที่ลบ

ตารางที่ 3.1 External Entity Description (ต่อ)

Name	Description	Input Data Flow	Output Data Flow
		แลกเปลี่ยนสกุลที่ลบ - ข้อมูลบริษัท แลกเปลี่ยนสกุลที่แก้ไข - ข้อมูลบริษัท แลกเปลี่ยนสกุลที่ค้นหา - ข้อมูลการอัปเดต สถานะ KYC	- ข้อมูลบริษัท แลกเปลี่ยนสกุลที่แก้ไข - รายการข้อมูลบริษัท แลกเปลี่ยนสกุลที่ค้นหา - ข้อมูลการอัปเดตสถานะ KYC
ผู้ใช้	เป็นผู้ใช้งานระบบที่มี ความสามารถในการ จองสกุลเงิน ตั้งค่า การแจ้งเตือนสกุลเงิน คำนวณอัตรา แลกเปลี่ยนสกุลเงิน และดูประวัติการจอง สกุลเงิน	- ข้อมูลเข้าสู่ระบบ - ข้อมูลผู้ใช้ที่เพิ่ม - ข้อมูลผู้ใช้ที่แก้ไข - ข้อมูลผู้ใช้ที่ค้นหา - ข้อมูลการจองสกุลเงิน เพิ่ม - ข้อมูลการแจ้งเตือน สกุลเงินที่เพิ่ม	- ข้อมูลประเภทสมาชิก - ข้อมูลผู้ใช้ที่เพิ่ม - ข้อมูลผู้ใช้ที่แก้ไข - รายการข้อมูลผู้ใช้ที่ค้นหา - ข้อมูลการจองสกุลเงินที่ เพิ่ม - ข้อมูลการแจ้งเตือนสกุล เงินที่เพิ่ม
บริษัท แลกเปลี่ยน สกุลเงิน	เป็นบริษัท แลกเปลี่ยนสกุลเงินที่ มีความสามารถในการ การอนุมัติหรือปฏิเสธ การจองสกุลเงินของ ผู้ใช้	- ข้อมูลเข้าสู่ระบบ - ข้อมูลบริษัท แลกเปลี่ยนสกุลเงินที่ เพิ่ม - ข้อมูลบริษัท แลกเปลี่ยนสกุลที่ลบ - ข้อมูลบริษัท แลกเปลี่ยนสกุลที่แก้ไข - ข้อมูลบริษัท แลกเปลี่ยนสกุลที่ค้นหา - ข้อมูลสถานะการจอง สกุลเงินที่เพิ่ม	- ข้อมูลประเภทสมาชิก - ข้อมูลบริษัท แลกเปลี่ยนสกุลเงินที่เพิ่ม - ข้อมูลบริษัท แลกเปลี่ยนสกุลที่ลบ - ข้อมูลบริษัท แลกเปลี่ยนสกุลที่แก้ไข - รายการข้อมูลบริษัท แลกเปลี่ยนสกุลที่ค้นหา - ข้อมูลสถานะการจองสกุล เงินที่เพิ่ม

ตารางที่ 3.1 External Entity Description (ต่อ)

Name	Description	Input Data Flow	Output Data Flow
		<ul style="list-style-type: none"> - ข้อมูลสถานะการจองสกุลเงินที่ลบ - ข้อมูลสถานะการจองสกุลเงินที่ค้นหา 	<ul style="list-style-type: none"> - ข้อมูลสถานะการจองสกุลเงินที่ลบ - รายการข้อมูลสถานะการจองสกุลเงินที่ค้นหา

3.3.2 อธิบายการไหลของข้อมูลและโครงสร้าง (Data Flow and Data Structure Description)

ตารางที่ 3.2 Data Flow and Data Structure Description

Name	Description	Source	Destination	Data Structure
ข้อมูลผู้ใช้ที่ค้นหา	ข้อมูลผู้ใช้ที่ต้องการค้นหาเพื่อตรวจสอบว่าผู้ใช้มีอยู่จริง	ผู้ดูแลระบบ	Process 1.1 ค้นหาข้อมูลผู้ใช้	รหัสผู้ใช้ + รหัสยืนยันตัวตน + ชื่อ
		Process 1.1 ค้นหาข้อมูลผู้ใช้	D2 ข้อมูลสมาชิกผู้ใช้	บัญชีผู้ใช้ + ชื่อ + นามสกุล
ข้อมูลผู้ใช้ที่เพิ่ม	ข้อมูลผู้ใช้ที่เพิ่มเป็นการเพิ่มข้อมูลผู้ใช้เข้าสู่ระบบ	ผู้ดูแลระบบ	Process 1.2 เพิ่มข้อมูลผู้ใช้	รหัสผู้ใช้ + รหัสยืนยันตัวตน + ชื่อ
		Process 1.2 เพิ่มข้อมูลผู้ใช้	D2 ข้อมูลสมาชิกผู้ใช้	บัญชีผู้ใช้ + อีเมล
ข้อมูลผู้ใช้ที่เพิ่ม	เป็นข้อมูลที่ผู้ใช้ทำการสมัครครั้งแรก	ผู้ใช้	Process 1.2 เพิ่มข้อมูลผู้ใช้	รหัสผู้ใช้ + รหัสยืนยันตัวตน + ชื่อ
		Process 1.2 เพิ่มข้อมูลผู้ใช้	D2 ข้อมูลสมาชิกผู้ใช้	บัญชีผู้ใช้ + อีเมล + รหัสผ่าน
ข้อมูลผู้ใช้ที่ลบ	เป็นข้อมูลผู้ใช้ที่มีอยู่แล้วที่ต้องการลบ	ผู้ดูแลระบบ	Process 1.3 ลบข้อมูลผู้ใช้	รหัสผู้ใช้ + ชื่อบัญชีผู้ใช้

ตารางที่ 3.2 Data Flow and Data Structure Description (ต่อ)

Name	Description	Source	Destination	Data Structure
		Process 1.3 ลบข้อมูล ผู้ใช้	D2 ข้อมูลสมาชิก ผู้ใช้	
ข้อมูลผู้ใช้ที่ แก้ไข	เป็นข้อมูลผู้ใช้ ที่มีอยู่แล้ว ต้องการแก้ไข	ผู้ใช้	Process 1.4 แก้ไข ข้อมูลผู้ใช้	รหัสผู้ใช้ + รหัส ยืนยันตัวตน + ชื่อ บัญชีผู้ใช้ + อีเมล + รหัสผ่าน
		Process 1.4 แก้ไข ข้อมูลผู้ใช้	D2 ข้อมูลสมาชิก ผู้ใช้	
ข้อมูลผู้ใช้ KYC ที่เพิ่ม	เป็นข้อมูลที่ ผู้ใช้ทำการ เพิ่มข้อมูล KYC	ผู้ใช้	Process 1.5 เพิ่มข้อมูลผู้ใช้ KYC	รหัสยืนยันตัวตน + ชื่อ + นามสกุล + วันเดือนปีเกิด + เลข บัตรประชาชนหรือ เลขบัตร + ใบขับขี่ + ที่อยู่ + อาชีพ + เบอร์โทรศัพท์ + สถานะยืนยันตัวตน
		Process 1.5 เพิ่มข้อมูล ผู้ใช้ KYC	D3 ข้อมูลสมาชิก ผู้ใช้ KYC	
อัปเดต ข้อมูล สถานะ KYC	เป็นข้อมูล บอก สถานะการ ยืนยันตัวตน ของผู้ใช้	บริษัท	Process 1.6 สถานะ KYC	รหัสยืนยันตัวตน + สถานะยืนยันตัวตน
		Process 1.6 สถานะ KYC	D3 ข้อมูลสมาชิก ผู้ใช้ KYC	
ข้อมูลผู้ใช้ KYC ที่ ค้นหา	เป็นการค้นหา ข้อมูลผู้ใช้KYC เพื่อยืนยัน ข้อมูล KYC	ผู้ดูแลระบบ	Process 1.7 ค้นหาข้อมูลKYC ของผู้ใช้	รหัสยืนยันตัวตน + ชื่อ + นามสกุล + วันเดือนปีเกิด + เลข บัตรประชาชนหรือ เลขบัตรใบขับขี่

ตารางที่ 3.2 Data Flow and Data Structure Description (ต่อ)

Name	Description	Source	Destination	Data Structure
		Process 1.7 ค้นหาข้อมูล KYCของผู้ใช้	D3 ข้อมูลสมาชิก ผู้ใช้ KYC	ที่อยู่ + อาชีพ + อีเมล + เบอร์ โทรศัพท์
ข้อมูลเข้าสู่ ระบบ	เป็นข้อมูลเข้า สู่ระบบของ ระบบการจอง อัตรา แลกเปลี่ยน สกุลเงินไทย ด้วยผู้ดูแล ระบบ	ผู้ดูแลระบบ	Process 2.0 เข้าสู่ระบบ	ข้อมูลผู้ใช้ + รหัสผ่าน
		Process 2. 0 เข้าสู่ระบบ ระบบ	D4 ข้อมูลสมาชิก ผู้ดูแลระบบ	
ข้อมูลเข้าสู่ ระบบ	เป็นข้อมูลเข้า สู่ระบบของ ระบบการจอง อัตรา แลกเปลี่ยน สกุลเงินไทย ด้วยผู้ใช้	ผู้ใช้	Process 2.0 เข้าสู่ระบบ	ข้อมูลผู้ใช้ + รหัสผ่าน
		Process 2. 0 เข้าสู่ระบบ ระบบ	D2 ข้อมูลสมาชิก ผู้ใช้	
ข้อมูลเข้าสู่ ระบบ	เป็นข้อมูลเข้า สู่ระบบของ ระบบการจอง อัตรา แลกเปลี่ยน สกุลเงินไทย ด้วยบริษัท แลกเปลี่ยน สกุลเงิน	บริษัท แลกเปลี่ยน เงิน	Process 2.0 เข้าสู่ระบบ	ข้อมูลผู้ใช้ + รหัสผ่าน
		Process 2. 0 เข้าสู่ระบบ ระบบ	D1 ข้อมูลสมาชิก บริษัทแลกเปลี่ยน เงิน	

ตารางที่ 3.2 Data Flow and Data Structure Description (ต่อ)

Name	Description	Source	Destination	Data Structure
ข้อมูลบริษัท แลกเปลี่ยน เงินที่เพิ่ม	ข้อมูลบริษัท แลกเปลี่ยน เงินที่เพิ่ม	ผู้ดูแลระบบ	Process 3.1 เพิ่มข้อมูลบริษัท แลกเปลี่ยนเงิน	รหัสบริษัท + ชื่อ บัญชีบริษัท + รหัสผ่าน + ชื่อบริษัท
		Process 3.1 เพิ่มข้อมูลบริษัท แลกเปลี่ยนเงิน	D1 ข้อมูล สมาชิกบริษัท แลกเปลี่ยนเงิน	ไทย+ชื่อบริษัท อังกฤษ+ผู้ให้บริการ แลกเปลี่ยน+ เว็บไซต์+ไอคอน + เบอร์โทร + อีเมล + ที่อยู่
ข้อมูลบริษัท แลกเปลี่ยน เงินที่เพิ่ม	เป็นข้อมูลที่ บริษัท แลกเปลี่ยน เงินทำการ สมัครครั้งแรก	บริษัท แลกเปลี่ยนเงิน	Process 3.1 เพิ่มข้อมูลบริษัท แลกเปลี่ยนเงิน	รหัสบริษัท + ชื่อ บัญชีบริษัท + รหัสผ่าน+ชื่อบริษัท
		Process 3.1 เพิ่มข้อมูลบริษัท แลกเปลี่ยนเงิน	D1 ข้อมูล สมาชิกบริษัท แลกเปลี่ยนเงิน	ไทย+ชื่อบริษัท อังกฤษ+ผู้ให้บริการ แลกเปลี่ยน+ เว็บไซต์+ไอคอน+ เบอร์โทร + อีเมล + ที่อยู่
ข้อมูลบริษัท แลกเปลี่ยน เงินที่ลบ	เป็นข้อมูล บริษัท แลกเปลี่ยน เงินที่มีอยู่แล้ว ต้องการลบ	ผู้ดูแลระบบ	Process 3.2 ลบข้อมูลบริษัท แลกเปลี่ยนเงิน	รหัสบริษัท + ชื่อ บัญชีบริษัท
		Process 3.2 ลบข้อมูลบริษัท แลกเปลี่ยนเงิน	D1 ข้อมูล สมาชิกบริษัท แลกเปลี่ยนเงิน	

ตารางที่ 3.2 Data Flow and Data Structure Description (ต่อ)

Name	Description	Source	Destination	Data Structure
ข้อมูลบริษัท แลกเปลี่ยน เงินที่แก้ไข	เป็นข้อมูล บริษัท แลกเปลี่ยน เงินที่มีอยู่แล้ว ต้องการแก้ไข	บริษัท แลกเปลี่ยน เงิน	Process 3.3 แก้ไขข้อมูลบริษัท แลกเปลี่ยนเงิน	ชื่อบัญชีบริษัท + รหัสผ่าน+ ชื่อบริษัท ไทย+ชื่อบริษัท อังกฤษ+ผู้ให้บริการ แลกเปลี่ยน+ เว็บไซต์+ไอคอน+ เบอร์โทร + อีเมล + ที่อยู่
		Process 3.3 แก้ไขข้อมูล บริษัท แลกเปลี่ยน เงิน	D1 ข้อมูลสมาชิก บริษัทแลกเปลี่ยน เงิน	
ข้อมูลบริษัท แลกเปลี่ยน เงินที่ค้นหา	ข้อมูลบริษัท แลกเปลี่ยน เงินที่ต้องการ ค้นหา	ผู้ดูแลระบบ	Process 3.4 ค้นหาข้อมูล บริษัทแลกเปลี่ยน เงิน	รหัสบริษัท + ชื่อ บัญชีบริษัท
		Process 3.4 ค้นหาข้อมูล บริษัท แลกเปลี่ยน เงิน	D1 ข้อมูลสมาชิก บริษัทแลกเปลี่ยน เงิน	
ข้อมูลบริษัท แลกเปลี่ยน เงินที่ค้นหา	ข้อมูลบริษัท แลกเปลี่ยน เงินที่ต้องการ ค้นหา	บริษัท แลกเปลี่ยน เงิน	Process 3.4 ค้นหาข้อมูล บริษัทแลกเปลี่ยน เงิน	รหัสบริษัท + ชื่อ บัญชีบริษัท
		Process 3.4 ค้นหาข้อมูล บริษัท แลกเปลี่ยน เงิน	D1 ข้อมูลสมาชิก บริษัทแลกเปลี่ยน เงิน	

ตารางที่ 3.2 Data Flow and Data Structure Description (ต่อ)

Name	Description	Source	Destination	Data Structure
ข้อมูลการจอง สกุลเงิน	เป็นข้อมูลที่ใช้ สำหรับการ จองสกุลเงิน ของผู้ใช้	ผู้ใช้	Process 4.0 ป้อนข้อมูลการ จองสกุลเงิน	<u>รหัสการจองสกุลเงิน</u> + ชื่อสกุลเงิน + จำนวนสกุลเงิน + วันที่ไปรับเงิน + รหัสชำระเงิน + สถานะการจองสกุล เงิน
		Process 4.0 ป้อนข้อมูลการ จองสกุลเงิน	D5 ข้อมูลการ จองสกุลเงิน	
ข้อมูล สถานะการ จองสกุลเงินที่ เพิ่ม	ข้อมูล สถานะการ จองสกุลเงินที่ ต้องการเพิ่ม	บริษัท แลกเปลี่ยนเงิน	Process 5.1 เพิ่มสถานะ ข้อมูลการจอง สกุลเงิน	<u>รหัสการจองสกุลเงิน</u> + สถานะการจอง สกุลเงิน
		Process 5.1 เพิ่มสถานะ ข้อมูลการจอง สกุลเงิน	D5 ข้อมูลการ จองสกุลเงิน	
ข้อมูล สถานะการ จองสกุลเงินที่ ลบ	เป็นข้อมูล สถานะการ จองสกุลเงินที่ ต้องการลบ	บริษัท แลกเปลี่ยนเงิน	Process 5.2 ลบสถานะ ข้อมูลการจอง สกุลเงิน	<u>รหัสการจองสกุลเงิน</u> + สถานะการจอง สกุลเงิน
		Process 5.2 ลบสถานะ ข้อมูลการจอง สกุลเงิน	D5 ข้อมูลการ จองสกุลเงิน	
ข้อมูล สถานะการ จองสกุลเงินที่ ค้นหา	ข้อมูล สถานะการ จองสกุลเงินที่ ต้องการค้นหา	บริษัท แลกเปลี่ยนเงิน	Process 5.3 ค้นหาสถานะ ข้อมูลการจอง สกุลเงิน	<u>รหัสการจองสกุลเงิน</u> + ชื่อผู้ใช้ + ชื่อสกุล เงิน

ตารางที่ 3.2 Data Flow and Data Structure Description (ต่อ)

Name	Description	Source	Destination	Data Structure
		Process 5.3 ค้นหาสถานะ ข้อมูลการจอง สกุลเงิน	D5 ข้อมูลการ จองสกุลเงิน	
ข้อมูลการแจ้ง เตือนสกุลเงิน ที่เพิ่ม	เป็นข้อมูลการ แจ้งเตือนสกุล เงินที่ผู้ใช้เพิ่ม	ผู้ใช้	Process 6.0 เพิ่มข้อมูลการ แจ้งเตือนสกุล เงิน	ชื่อสกุลเงิน + ราคา + เวลา
		Process 6.0 เพิ่มข้อมูลการ แจ้งเตือนสกุล เงิน	D6 ข้อมูล อัตราการ แลกเปลี่ยน สกุลเงิน	
ข้อมูลการแจ้ง เตือน สถานะการ จอง	เป็นข้อมูลการ แจ้งเตือน สถานะการ จองเงินจาก ผู้ใช้	บริษัท แลกเปลี่ยนเงิน	Process 7.0 แจ้งเตือน สถานะการ จอง	แสดงข้อความ
		Process 7.0 แจ้งเตือน สถานะการจอง	D5 ข้อมูลการ จองสกุลเงิน	
ข้อมูลการแจ้ง เตือน สถานะการ จอง	เป็นข้อมูลการ แจ้งเตือน สถานะการ จองเงินจาก ผู้ใช้	ผู้ดูแลระบบ	Process 7.0 แจ้งเตือน สถานะการ จอง	แสดงข้อความ
		Process 7.0 แจ้งเตือน สถานะการจอง	D5 ข้อมูลการ จองสกุลเงิน	
ข้อมูลการ คำนวณอัตรา การ	เป็นการแสดง การคำนวณ อัตราการ	ผู้ใช้	Process 8.0 การคำนวณ อัตราการ	ชื่อสกุลเงิน + ราคา

ตารางที่ 3.2 Data Flow and Data Structure Description (ต่อ)

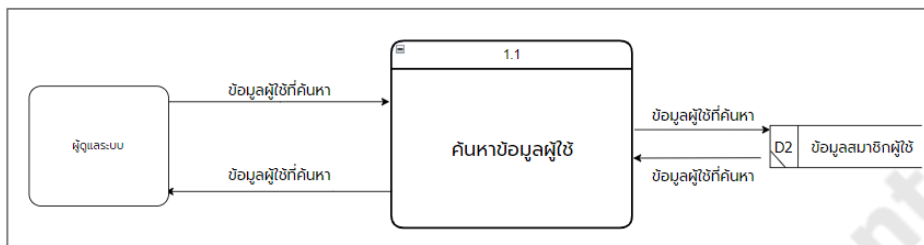
Name	Description	Source	Destination	Data Structure
แลกเปลี่ยน สกุลเงิน	แลกเปลี่ยน สกุลเงินให้ ผู้ใช้		แลกเปลี่ยน สกุลเงิน	
		Process 8.0 การคำนวณ อัตราการ แลกเปลี่ยนสกุล เงิน	D6 ข้อมูล อัตราการ แลกเปลี่ยน สกุลเงิน	
ข้อมูลประวัติ การจูงสกุล เงิน	เป็นการแสดง ข้อมูลประวัติ การจูงสกุล เงินให้ผู้ใช้	ผู้ใช้	Process 9.0 ประวัติการ จูงสกุลเงิน	ชื่อบัญชีผู้ใช้ + ชื่อ สกุลเงิน + จำนวน สกุลเงิน + ราคา + วันที่จูง + วันที่รับเงิน
		Process 9.0 ประวัติการจูง สกุลเงิน	D5 ข้อมูลการ จูงสกุลเงิน	
ข้อมูลการจูง ชำระเงิน	เป็นการแสดง ข้อมูลการแจ้ง เตือน สถานะการ จูงชำระเงิน ของผู้ใช้	ผู้ใช้	Process 10.0 แจ้งเตือน สถานะการ จูงชำระเงิน	แสดงข้อความ
		Process 10.0 แจ้งเตือน สถานะการจูง ชำระเงิน	D5 ข้อมูลการ จูงสกุลเงิน	

3.3.3 อธิบายการเก็บข้อมูลและโครงสร้าง (Data Store and Data Structure Description)

ตารางที่ 3.3 Data Store and Data Structure

ID	Data Store Name	Description	Data Structure
D1	ข้อมูลสมาชิกบริษัท แลกเปลี่ยนเงิน	เก็บข้อมูลสมาชิกบริษัท แลกเปลี่ยนเงิน	รหัสบริษัท + ชื่อบัญชีบริษัท + รหัสผ่าน + ชื่อบริษัทไทย+ชื่อ บริษัทอังกฤษ+ผู้ให้บริการ แลกเปลี่ยน+เว็บไซต์+ไอคอน+ เบอร์โทร + อีเมล + ที่อยู่
D2	ข้อมูลสมาชิกผู้ใช้	เก็บข้อมูลสมาชิกผู้ใช้	รหัสผู้ใช้+รหัสยืนยันตัวตน+ชื่อ บัญชี+ อีเมล + รหัสผ่าน
D3	ข้อมูลสมาชิกผู้ใช้KYC	เก็บข้อมูลสมาชิก KYC	รหัสยืนยันตัวตน+ชื่อ + นามสกุล + วันเดือนปี + เลข บัตรประชาชนหรือเลขบัตร ใบขับขี่+ ที่อยู่ +อาชีพ +อีเมล +เบอร์โทรศัพท์+สถานะยืนยัน ตัวตน
D4	ข้อมูลผู้ดูแลระบบ	เก็บข้อมูลผู้ดูแลระบบ	รหัสผู้ดูแลระบบ+ชื่อ + นามสกุล + ชื่อบัญชี +รหัสผ่าน
D5	ข้อมูลการจองสกุลเงิน (Blockchain)	เก็บข้อมูลการจองสกุลเงิน ต่างๆโดยใช้ตัว Blockchain ในการจัดเก็บ	รหัสการจองสกุลเงิน +รหัส ผู้ใช้+รหัสสกุลเงิน+ ชื่อสกุลเงิน + จำนวนสกุลเงิน + วันที่ไปรับเงิน + รหัสชำระเงิน +สถานการณ์จองสกุลเงิน
D6	ข้อมูลอัตราค่า แลกเปลี่ยนสกุลเงิน	เก็บข้อมูลอัตราค่า แลกเปลี่ยนสกุลเงิน	รหัสบริษัท + ชื่อบริษัท + ชื่อ สกุลเงิน + ราคา + วันที่

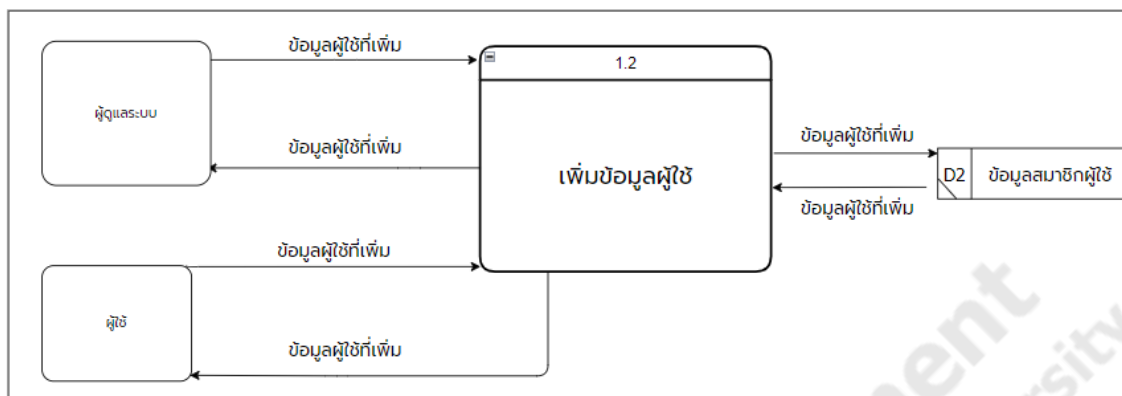
3.3.4 อธิบายการประมวลผล (Process Description)



ภาพประกอบที่ 3.7 แผนภาพการประมวลผลของการค้นหาข้อมูลผู้ใช้

ตารางที่ 3.4 อธิบายการประมวลผลการค้นหาข้อมูลผู้ใช้

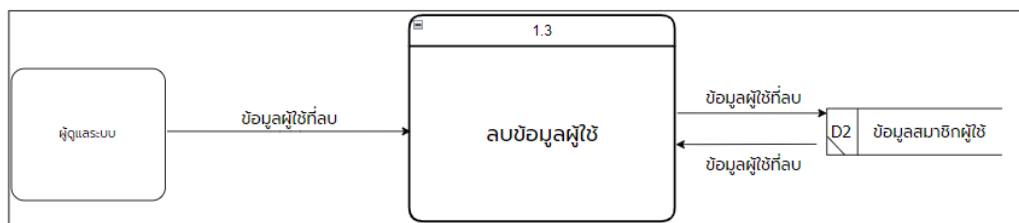
ID	1.1
NAME	ค้นหาข้อมูลผู้ใช้
DESCRIPTION	เป็นกระบวนการค้นหาข้อมูลผู้ใช้ที่มีในระบบ
INPUT DATA FLOWS	- ข้อมูลผู้ใช้ที่ค้นหา
OUTPUT DATA FLOWS	- ข้อมูลผู้ใช้ที่ค้นหา
PROCESS DESCRIPTION	<p>เริ่มต้น</p> <ol style="list-style-type: none"> รับข้อมูลผู้ใช้ (รหัสผู้ใช้+รหัสยืนยันตัวตน+ชื่อผู้ใช้ + รหัสผู้ใช้) ตรวจสอบว่ามีข้อมูลครบถ้วนและถูกต้องหรือไม่ <ol style="list-style-type: none"> ถ้าข้อมูลครบถ้วน และถูกต้องทุกรายการให้ทำข้อ 3. ต่อ ถ้าข้อมูลครบถ้วน แต่ไม่ถูกต้องให้แสดงข้อความ “กรุณาป้อนข้อมูลให้ถูกต้อง” แล้วกลับไปทำข้อที่ 1. ใหม่ ถ้าข้อมูลไม่ครบถ้วน ให้แสดงข้อความ “กรุณาป้อนข้อมูลให้ถูกต้อง” แล้วกลับไปทำข้อที่ 1. ใหม่ ตรวจสอบในข้อมูลสมาชิกว่ามีข้อมูลหรือไม่ <ol style="list-style-type: none"> ถ้ามีข้อมูลผู้ใช้ในข้อมูลสมาชิกให้ไปหน้าหลัก ถ้าข้อมูลผู้ใช้ไม่มีในข้อมูลสมาชิกให้แสดงข้อความเตือนว่า “ไม่มีข้อมูลสมาชิกนี้ในระบบกรุณากรอกข้อมูลใหม่” แล้วให้กลับไปทำข้อที่ 1 ใหม่ <p>จบการทำงาน</p>



ภาพประกอบที่ 3.8 แผนภาพการประมวลผลของการเพิ่มข้อมูลผู้ใช้

ตารางที่ 3.5 อธิบายการประมวลผลการเพิ่มข้อมูลผู้ใช้

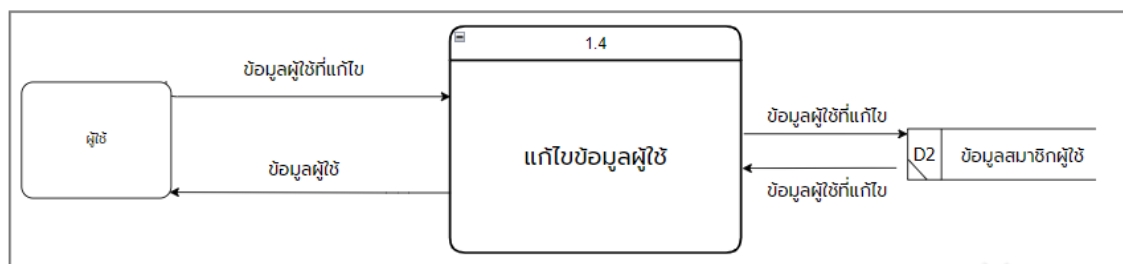
ID	1.2
NAME	เพิ่มข้อมูลผู้ใช้
DESCRIPTION	เป็นกระบวนการเพิ่มข้อมูลผู้ใช้คนใหม่ลงในระบบ
INPUT DATA FLOWS	-ข้อมูลผู้ใช้ที่เพิ่ม
OUTPUT DATA FLOWS	-ข้อมูลของผู้ใช้ที่เพิ่ม
PROCESS DESCRIPTION	<p>เริ่มต้น</p> <ol style="list-style-type: none"> รับข้อมูลผู้ใช้ (รหัสผู้ใช้+รหัสยืนยันตัวตน+ชื่อบัญชี+ อีเมล + รหัสผ่าน) ตรวจสอบว่ามีข้อมูลครบถ้วนและถูกต้องหรือไม่ <ol style="list-style-type: none"> ถ้าข้อมูลครบถ้วน และถูกต้องทุกรายการให้ทำข้อ 3. ต่อ ถ้าข้อมูลครบถ้วน แต่ไม่ถูกต้อง ให้แสดงข้อความ “กรุณาป้อนข้อมูลให้ถูกต้อง” แล้วให้กลับไปทำข้อ 1. ใหม่ ตรวจสอบข้อมูล (ชื่อบัญชี+ อีเมล + รหัสผ่าน) ในข้อมูลสมาชิกผู้ใช้ว่ามี ความถูกต้องหรือไม่ <ol style="list-style-type: none"> ถ้ามีข้อมูลผู้ใช้ (ชื่อบัญชี+ อีเมล + รหัสผ่าน) ในข้อมูลสมาชิกในระบบแสดงข้อความเตือน “มีสมาชิกนี้ในระบบแล้ว” จบการทำงาน



ภาพประกอบที่ 3.9 แผนภาพการประมวลผลของการลบข้อมูลผู้ใช้

ตารางที่ 3.6 อธิบายการประมวลผลผลการลบข้อมูลผู้ใช้

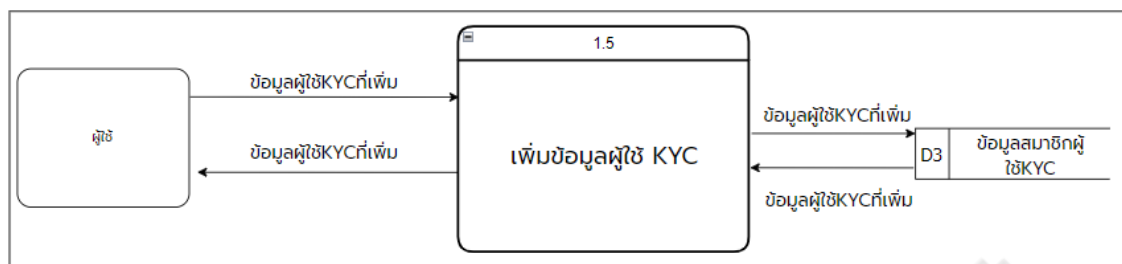
ID	1.3
NAME	ลบข้อมูลผู้ใช้
DESCRIPTION	เป็นกระบวนการลบข้อมูลผู้ใช้ที่มีในระบบ
INPUT DATA FLOWS	-ข้อมูลผู้ใช้ที่จะลบ
OUTPUT DATA FLOWS	-
PROCESS DESCRIPTION	<p>เริ่มต้น</p> <ol style="list-style-type: none"> 1. รับข้อมูลผู้ใช้ที่ต้องการจะลบ เช่น <ol style="list-style-type: none"> 1.1 รหัสผู้ใช้ 1.2 ชื่อบัญชีผู้ใช้ 1.3 ชื่อ-นามสกุล 2. ตรวจสอบว่ามีข้อมูลครบถ้วนและถูกต้องหรือไม่ <ol style="list-style-type: none"> 2.1 ถ้าข้อมูลครบถ้วน และถูกต้องทุกรายการให้ทำข้อ 3. ต่อ 2.2 ถ้าข้อมูลครบถ้วน แต่ไม่ถูกต้อง ให้แสดงข้อความ “กรุณาป้อนข้อมูลให้ถูกต้อง” แล้วให้กลับไปทำข้อ 1. ใหม่ 2.3 ถ้าข้อมูลไม่ครบถ้วน ให้แสดงข้อความ “กรุณาป้อนข้อมูลให้ถูก” แล้วกลับไปทำข้อ 1. ใหม่ 3. ตรวจสอบข้อมูล ในข้อมูลสมาชิกว่ามีข้อมูลหรือไม่ <ol style="list-style-type: none"> 3.1 ถ้ามีข้อมูลผู้ใช้ ในข้อมูลสมาชิกให้ระบบลบข้อมูลผู้ใช้นั้น และแสดงข้อความว่า “ลบบัญชีนี้แล้ว” 3.2 ถ้าข้อมูลผู้ใช้ ไม่มีในข้อมูลสมาชิกให้แสดงข้อความเตือนว่า “ไม่มีข้อมูลสมาชิกนี้ในระบบกรุณากรอกข้อมูลใหม่” แล้วให้กลับไปทำข้อ 1. ใหม่ <p>จบการทำงาน</p>



ภาพประกอบที่ 3.10 แผนภาพประมวลผลของการแก้ไขข้อมูลผู้ใช้

ตารางที่ 3.7 อธิบายการประมวลผลผลการแก้ไขข้อมูลผู้ใช้

ID	1.4
NAME	แก้ไขข้อมูลผู้ใช้
DESCRIPTION	เป็นกระบวนการแก้ไขข้อมูลผู้ใช้ที่มีในระบบ
INPUT DATA FLOWS	-ข้อมูลผู้ใช้ที่แก้ไข
OUTPUT DATA FLOWS	-ข้อมูลผู้ใช้ที่แก้ไข
PROCESS DESCRIPTION	<p>เริ่มต้น</p> <ol style="list-style-type: none"> รับข้อมูลผู้ใช้ (ชื่อบัญชี+ อีเมล + รหัสผ่าน) ตรวจสอบว่ามีข้อมูลครบถ้วนและถูกต้องหรือไม่ <ol style="list-style-type: none"> ถ้าข้อมูลครบถ้วน และถูกต้องทุกรายการให้ทำข้อ 3. ต่อ ถ้าข้อมูลครบถ้วน แต่ไม่ถูกต้อง ให้แสดงข้อความ “กรุณาป้อนข้อมูลให้ถูกต้อง” แล้วให้กลับไปทำข้อ 1. ใหม่ ถ้าข้อมูลไม่ครบถ้วน ให้แสดงข้อความ “กรุณาป้อนข้อมูลให้ถูก” แล้วกลับไปทำข้อ 1. ใหม่ ตรวจสอบข้อมูล ในข้อมูลสมาชิกว่ามีข้อมูลหรือไม่ <ol style="list-style-type: none"> ถ้ามีข้อมูลผู้ใช้ ในข้อมูลสมาชิกให้ระบบแสดงหน้ารายละเอียดข้อมูลผู้ใช้ ถ้าข้อมูลผู้ใช้ ไม่มีในข้อมูลสมาชิกให้แสดงข้อความเตือนว่า “ไม่พบข้อมูลผู้ใช้” <p>จบการทำงาน</p>



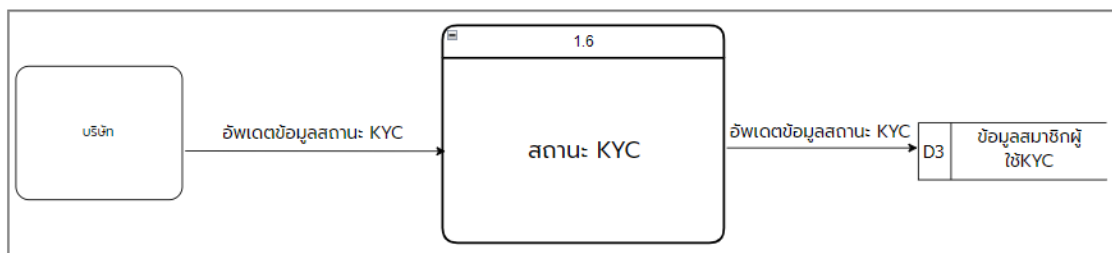
ภาพประกอบที่ 3.11 แผนภาพการประมวลผลของการเพิ่มข้อมูลผู้ใช้ KYC

ตารางที่ 3.8 อธิบายการประมวลผลการเพิ่มข้อมูลผู้ใช้ KYC

ID	1.5
NAME	เพิ่มข้อมูลผู้ใช้ KYC
DESCRIPTION	เป็นกระบวนการเพิ่มข้อมูลอื่นๆของผู้ใช้ลงในระบบหลังจากสมัครครั้งแรก
INPUT DATA FLOWS	-ข้อมูลผู้ใช้ที่เพิ่ม
OUTPUT DATA FLOWS	-ข้อมูลของผู้ใช้ที่เพิ่ม
PROCESS DESCRIPTION	เริ่มต้น 1. รับข้อมูลผู้ใช้(รหัสยืนยันตัวตน+ชื่อ + นามสกุล + วันเดือนปี + เลขบัตรประชาชนหรือเลขบัตรใบขับขี่+ ที่อยู่ + อาชีพ + เบอร์โทรศัพท์)

ตารางที่ 3.8 อธิบายการประมวลผลการเพิ่มข้อมูลผู้ใช้ KYC (ต่อ)

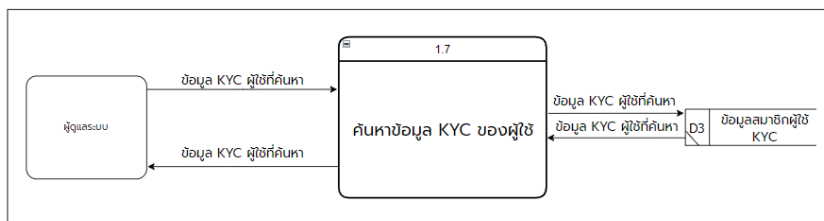
	<p>1.1 หลังจากที่ยกรอกข้อมูลในข้อ 1. เสร็จแล้วระบบ จะแจ้งเตือนถัดไปให้ผู้ใช้ ถ่ายรูปคู่กับบัตรประชาชนหรือบัตรใบขับขี่ของตน</p>  <p>2. ตรวจสอบว่ามีข้อมูลครบถ้วนและถูกต้องหรือไม่</p> <p>2.1 ถ้าข้อมูลครบถ้วน และถูกต้องทุกรายการให้ทำข้อ 3. ต่อ</p> <p>2.2 ถ้าข้อมูลครบถ้วน แต่ไม่ถูกต้อง ให้แสดงข้อความ “กรุณาป้อนข้อมูลให้ถูกต้อง” แล้วให้กลับไปทำข้อ 1. ใหม่</p> <p>3. ตรวจสอบข้อมูล (ชื่อ + นามสกุล + วันเดือนปี + เลขบัตรประชาชนหรือ เลขบัตรใบขับขี่+ ที่อยู่ + อาชีพ + อีเมล + เบอร์โทรศัพท์) ในข้อมูลสมาชิกผู้ใช้ KYC และรูปถ่ายยืนยัน ว่ามีความถูกต้องหรือไม่</p> <p>3.1 ถ้ามีข้อมูลผู้ใช้ (ชื่อ + นามสกุล + วันเดือนปี + เลขบัตรประชาชนหรือ เลขบัตรใบขับขี่+ ที่อยู่ + อาชีพ + อีเมล + เบอร์โทรศัพท์) และรูปถ่ายเพื่อยืนยันตัวตน ในข้อมูลสมาชิกในระบบแสดงข้อความเตือน “มีสมาชิกนี้ในระบบแล้ว”</p> <p>3.2 ถ้ายังไม่มีข้อมูลผู้ใช้(ชื่อ + นามสกุล + วันเดือนปี + เลขบัตรประชาชน หรือเลขบัตรใบขับขี่+ ที่อยู่ + อาชีพ + อีเมล + เบอร์โทรศัพท์) และรูปถ่ายเพื่อยืนยันตัวตน ให้ระบบบันทึกข้อมูลสมาชิกใหม่แล้วแสดงข้อความ “เพิ่มสมาชิกนี้ในระบบแล้ว”</p> <p>จบการทำงาน</p>
--	---



ภาพประกอบที่ 3.12 แผนภาพการประมวลผลของสถานะ KYC

ตารางที่ 3.9 อธิบายการประมวลผลสถานะ KYC

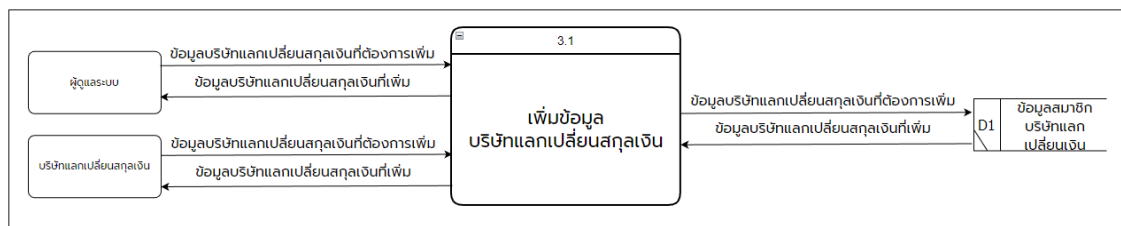
ID	1.6
NAME	สถานะ KYC
DESCRIPTION	เป็นข้อมูลบอกสถานการณียืนยันตัวตนของผู้ใช้
INPUT DATA FLOWS	-อัปเดตข้อมูลสถานะ KYC
OUTPUT DATA FLOWS	-
PROCESS DESCRIPTION	<p>เริ่ม</p> <ol style="list-style-type: none"> รับข้อมูลมาจากผู้ใช้ผ่านผู้ดูแลระบบ (รหัสยืนยันตัวตน + สถานะยืนยันตัวตน) ตรวจสอบข้อมูล (รหัสยืนยันตัวตน + สถานะยืนยันตัวตน) ดังกล่าวว่ามีข้อมูลหรือไม่ <ol style="list-style-type: none"> ถ้ามีข้อมูล (รหัสยืนยันตัวตน + สถานะยืนยันตัวตน) ดังกล่าวให้ผู้ดูแลระบบตัดสินใจเลือก “ยอมรับการยืนยันตัวตน KYC” หรือ “ปฏิเสธการยืนยันตัวตน KYC” ถ้าไม่มีข้อมูล (รหัสยืนยันตัวตน + สถานะยืนยันตัวตน) ดังกล่าวให้ระบบแสดงข้อความว่า “ไม่มีสถานะยืนยันตัวตน” ทำการบันทึกผลลัพธ์ข้อมูล 2.1 ลงฐานข้อมูลสมาชิกผู้ใช้ KYC <p>จบการทำงาน</p>



ภาพประกอบที่ 3.13 แผนภาพการประมวลผลของการค้นหาข้อมูล KYC ของผู้ใช้

ตารางที่ 3.10 อธิบายการประมวลผลการค้นหาข้อมูล KYC ของผู้ใช้

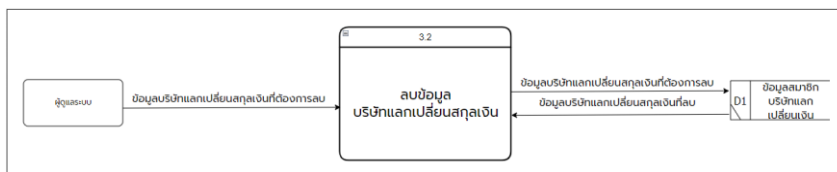
ID	1.7
NAME	ค้นหาข้อมูลผู้ใช้KYC
DESCRIPTION	เป็นกระบวนการค้นหาข้อมูลผู้ใช้ที่มีในระบบ
INPUT DATA FLOWS	- ข้อมูลผู้ใช้ที่ค้นหา KYC
OUTPUT DATA FLOWS	- ข้อมูลผู้ใช้ที่ค้นหา KYC
PROCESS DESCRIPTION	<p>เริ่มต้น</p> <ol style="list-style-type: none"> รับข้อมูลผู้ใช้ (รหัสผู้ใช้+สถานะยืนยันตัวตน+ชื่อ + นามสกุล + วันเดือนปี + เลขบัตรประชาชนหรือเลขบัตรใบขับขี่+ ที่อยู่ + อาชีพ + อีเมล + เบอร์โทรศัพท์) ตรวจสอบว่ามีข้อมูลครบถ้วนและถูกต้องหรือไม่ <ol style="list-style-type: none"> ถ้าข้อมูลครบถ้วน และถูกต้องทุกรายการให้ทำข้อ 3. ต่อ ถ้าข้อมูลครบถ้วน แต่ไม่ถูกต้องให้แสดงข้อความ “กรุณาป้อนข้อมูลให้ถูกต้อง” แล้วกลับไปทำข้อที่ 1. ใหม่ ถ้าข้อมูลไม่ครบถ้วน ให้แสดงข้อความ “กรุณาป้อนข้อมูลให้ถูกต้อง” แล้วกลับไปทำข้อที่ 1. ใหม่ ตรวจสอบในข้อมูลสมาชิกว่ามีข้อมูลหรือไม่ <ol style="list-style-type: none"> ถ้ามีข้อมูลผู้ใช้ในข้อมูลสมาชิกให้ไปหน้าหลัก ถ้าข้อมูลผู้ใช้ไม่มีในข้อมูลสมาชิกให้แสดงข้อความเตือนว่า “ไม่มีข้อมูลสมาชิกนี้ในระบบกรุณากรอกข้อมูลใหม่” แล้วให้กลับไปทำข้อที่ 1 ใหม่ <p>จบการทำงาน</p>



ภาพประกอบที่ 3.14 แผนภาพการประมวลผลของการเพิ่มข้อมูลบริษัทแลกเปลี่ยนสกุลเงิน

ตารางที่ 3.11 อธิบายการประมวลผลการเพิ่มข้อมูลบริษัทแลกเปลี่ยนสกุลเงิน

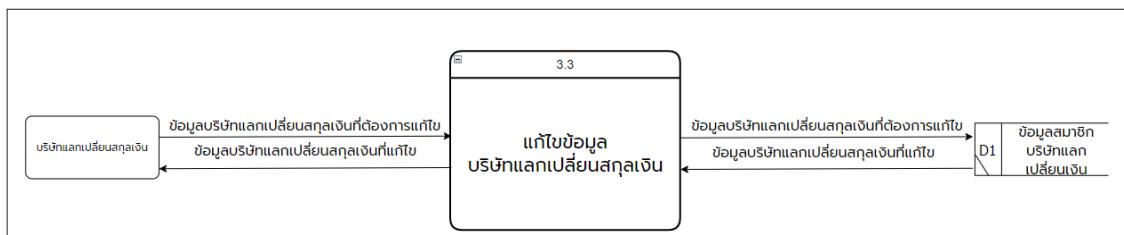
ID	3.1
NAME	เพิ่มข้อมูลบริษัทแลกเปลี่ยนสกุลเงิน
DESCRIPTION	เป็นกระบวนการเพิ่มข้อมูลบริษัทแลกเปลี่ยนสกุลเงินใหม่ลงในระบบ
INPUT DATA FLOW	ข้อมูลบริษัทแลกเปลี่ยนสกุลเงินที่ต้องการเพิ่ม
OUTPUT DATA FLOW	ข้อมูลบริษัทแลกเปลี่ยนสกุลเงินที่เพิ่ม
PROCESS DESCRIPTION	<p>เริ่ม</p> <ol style="list-style-type: none"> 1. รับข้อมูลบริษัทแลกเปลี่ยนสกุลเงิน (<u>รหัสบริษัท</u> + <u>ชื่อบัญชีบริษัท</u> + <u>รหัสผ่าน</u> + <u>ชื่อบริษัท</u> + <u>เบอร์โทร</u> + <u>อีเมล</u> + <u>ที่อยู่</u>) 2. ตรวจสอบว่ามีข้อมูลครบถ้วนและถูกต้องหรือไม่ <ol style="list-style-type: none"> 2.1 ถ้าข้อมูลครบถ้วน และถูกต้องทุกรายการให้ทำข้อ 3 2.2 ถ้าข้อมูลครบถ้วน แต่ไม่ถูกต้องให้แสดงข้อความ “กรุณาป้อนข้อมูลให้ถูกต้อง” แล้วให้กลับไปทำข้อ 1. ใหม่ 3. ตรวจสอบข้อมูล (<u>รหัสบริษัท</u> + <u>ชื่อบัญชีบริษัท</u> + <u>รหัสผ่าน</u> + <u>ชื่อบริษัท</u> + <u>เบอร์โทร</u> + <u>อีเมล</u> + <u>ที่อยู่</u>) ดังกล่าวว่ามีข้อมูลหรือไม่ <ol style="list-style-type: none"> 3.1 ถ้ามีข้อมูล (<u>รหัสบริษัท</u> + <u>ชื่อบัญชีบริษัท</u> + <u>รหัสผ่าน</u> + <u>ชื่อบริษัท</u> + <u>เบอร์โทร</u> + <u>อีเมล</u> + <u>ที่อยู่</u>) ดังกล่าวให้ระบบแสดงข้อความว่า “มีรายชื่อบริษัทนี้ในระบบแล้ว” 3.2 ถ้าไม่มีข้อมูล (<u>รหัสบริษัท</u> + <u>ชื่อบัญชีบริษัท</u> + <u>รหัสผ่าน</u> + <u>ชื่อบริษัท</u> + <u>เบอร์โทร</u> + <u>อีเมล</u> + <u>ที่อยู่</u>) ดังกล่าวให้ระบบเพิ่มข้อมูลบริษัทแลกเปลี่ยนสกุลเงินใหม่นี้ลงในฐานข้อมูลสมาชิก <p>จบการทำงาน</p>



ภาพประกอบที่ 3.15 แผนภาพการประมวลผลการลบข้อมูลบริษัทแลกเปลี่ยนสกุลเงิน

ตารางที่ 3.12 อธิบายการประมวลผลการลบข้อมูลบริษัทแลกเปลี่ยนสกุลเงิน

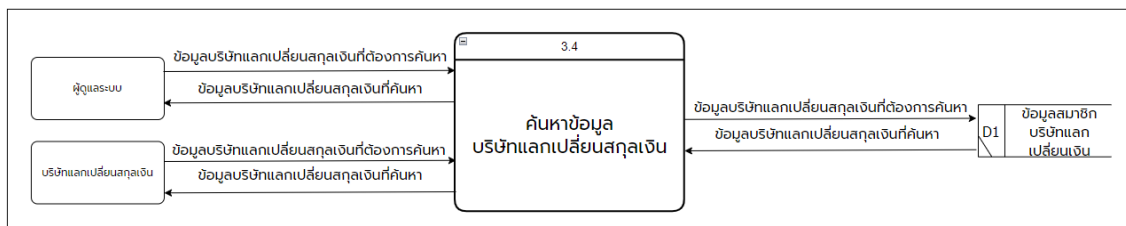
ID	3.2
NAME	ลบข้อมูลบริษัทแลกเปลี่ยนสกุลเงิน
DESCRIPTION	เป็นกระบวนการลบข้อมูลบริษัทแลกเปลี่ยนสกุลเงิน
INPUT DATA FLOWS	-ข้อมูลบริษัทแลกเปลี่ยนสกุลเงินที่จะลบ
OUTPUT DATA FLOWS	-
PROCESS DESCRIPTION	<p>เริ่มต้น</p> <ol style="list-style-type: none"> รับข้อมูลบริษัทแลกเปลี่ยนสกุลเงินที่ต้องการจะลบ เช่น <ol style="list-style-type: none"> รหัสบริษัท ชื่อบัญชีบริษัท ตรวจสอบว่ามีข้อมูลครบถ้วนและถูกต้องหรือไม่ <ol style="list-style-type: none"> ถ้าข้อมูลครบถ้วน และถูกต้องทุกรายการให้ทำข้อ 3. ต่อ ถ้าข้อมูลครบถ้วน แต่ไม่ถูกต้อง ให้แสดงข้อความ “กรุณาป้อนข้อมูลให้ถูกต้อง” แล้วให้กลับไปทำข้อ 1. ใหม่ ถ้าข้อมูลไม่ครบถ้วน ให้แสดงข้อความ “กรุณาป้อนข้อมูลให้ถูก” แล้วกลับไปทำข้อ 1. ใหม่ ตรวจสอบข้อมูล ในข้อมูลสมาชิกว่ามีข้อมูลหรือไม่ <ol style="list-style-type: none"> ถ้ามีข้อมูลบริษัทแลกเปลี่ยนสกุลเงิน ในข้อมูลสมาชิกให้ระบบลบข้อมูลบริษัทแลกเปลี่ยนสกุลเงินนั้น และแสดงข้อความว่า “ลบบัญชีนี้แล้ว” ถ้าข้อมูลบริษัทแลกเปลี่ยนสกุลเงิน ไม่มีในข้อมูลสมาชิกให้แสดงข้อความเตือนว่า “ไม่มีข้อมูลสมาชิกนี้ในระบบกรุณากรอกข้อมูลใหม่” แล้วให้กลับไปทำข้อ 1. ใหม่ <p>จบการทำงาน</p>



ภาพประกอบที่ 3.16 แผนภาพการประมวลผลของการแก้ไขข้อมูลบริษัทแลกเปลี่ยนสกุลเงิน

ตารางที่ 3.13 อธิบายการประมวลผลการแก้ไขข้อมูลบริษัทแลกเปลี่ยนสกุลเงิน

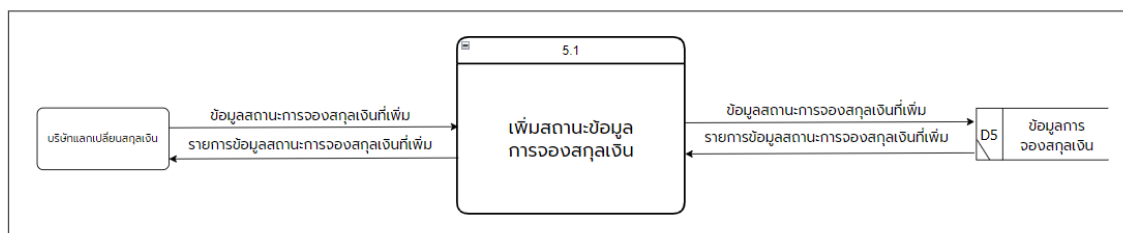
ID	3.3
NAME	แก้ไขข้อมูลบริษัทแลกเปลี่ยนสกุลเงิน
DESCRIPTION	เป็นกระบวนการแก้ไขข้อมูลบริษัทแลกเปลี่ยนสกุลเงินที่มีในระบบ
INPUT DATA FLOWS	-ข้อมูลบริษัทแลกเปลี่ยนสกุลเงินที่แก้ไข
OUTPUT DATA FLOWS	-ข้อมูลบริษัทแลกเปลี่ยนสกุลเงินที่แก้ไข
PROCESS DESCRIPTION	<p>เริ่มต้น</p> <ol style="list-style-type: none"> 1. รับข้อมูลบริษัทแลกเปลี่ยนสกุลเงิน (รหัสบริษัท + ชื่อบัญชีบริษัท + รหัสผ่าน + บริษัท + เบอร์โทร + อีเมล + ที่อยู่) <ul style="list-style-type: none"> . ตรวจสอบว่ามีข้อมูลครบถ้วนและถูกต้องหรือไม่ <ol style="list-style-type: none"> 2.1 ถ้าข้อมูลครบถ้วน และถูกต้องทุกรายการให้ทำข้อ 3. ต่อ 2.2 ถ้าข้อมูลครบถ้วน แต่ไม่ถูกต้อง ให้แสดงข้อความ “กรุณาป้อนข้อมูลให้ถูกต้อง” แล้วให้กลับไปทำข้อ 1. ใหม่ 2.3 ถ้าข้อมูลไม่ครบถ้วน ให้แสดงข้อความ “กรุณาป้อนข้อมูลให้ถูก” แล้วกลับไปทำข้อ 1. ใหม่ 3. ตรวจสอบข้อมูล ในข้อมูลสมาชิกว่ามีข้อมูลหรือไม่ <ol style="list-style-type: none"> 3.1 ถ้ามีข้อมูลบริษัทแลกเปลี่ยนสกุลเงิน ในข้อมูลสมาชิกให้ระบบแสดงหน้ารายละเอียดข้อมูลบริษัทแลกเปลี่ยนสกุลเงิน 3.2 ถ้าข้อมูลบริษัทแลกเปลี่ยนสกุลเงินไม่มีในข้อมูลสมาชิกให้แสดงข้อความเตือนว่า “ไม่พบข้อมูลบริษัทแลกเปลี่ยนสกุลเงิน” <p>จบการทำงาน</p>



ภาพประกอบที่ 3.17 อธิบายการประมวลผลการลบข้อมูลบริษัทแลกเปลี่ยนสกุลเงิน

ตารางที่ 3.14 อธิบายการประมวลผลการค้นหาข้อมูลบริษัทแลกเปลี่ยนสกุลเงิน

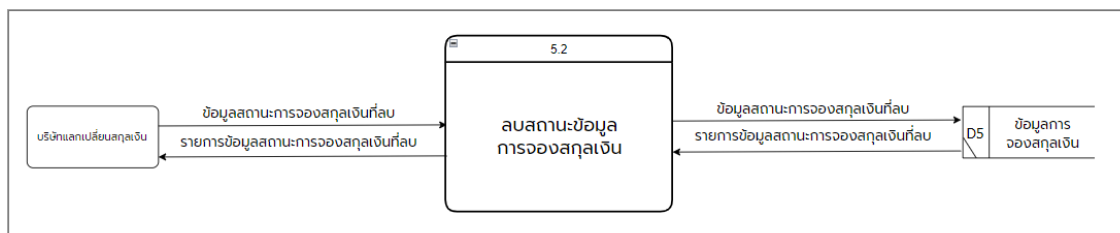
ID	3.4
NAME	ค้นหาข้อมูลบริษัทแลกเปลี่ยนสกุลเงิน
DESCRIPTION	เป็นกระบวนการค้นหาข้อมูลบริษัทแลกเปลี่ยนสกุลเงินที่มีในระบบ
INPUT DATA FLOWS	- ข้อมูลบริษัทแลกเปลี่ยนสกุลเงินที่ค้นหา
OUTPUT DATA FLOWS	- ข้อมูลบริษัทแลกเปลี่ยนสกุลเงินที่ค้นหา
PROCESS DESCRIPTION	<p>เริ่มต้น</p> <ol style="list-style-type: none"> 1. รับข้อมูลบริษัทแลกเปลี่ยนสกุลเงิน(รหัสบริษัท + ชื่อบริษัท) 2. ตรวจสอบว่ามีข้อมูลครบถ้วนและถูกต้องหรือไม่ <ol style="list-style-type: none"> 2.1 ถ้าข้อมูลครบถ้วน และถูกต้องทุกรายการให้ทำข้อ 3. ต่อ 2.2 ถ้าข้อมูลครบถ้วน แต่ไม่ถูกต้องให้แสดงข้อความ “กรุณาป้อนข้อมูลให้ถูกต้อง” แล้วกลับไปทำข้อที่ 1. ใหม่ 2.3 ถ้าข้อมูลไม่ครบถ้วน ให้แสดงข้อความ “กรุณาป้อนข้อมูลให้ถูกต้อง” แล้วกลับไปทำข้อที่ 1. ใหม่ 3. ตรวจสอบในข้อมูลสมาชิกว่ามีข้อมูลหรือไม่ <ol style="list-style-type: none"> 3.1 ถ้ามีข้อมูลบริษัทแลกเปลี่ยนสกุลเงิน ในข้อมูลสมาชิกให้ไปหน้าหลัก 3.2 ถ้าข้อมูลบริษัทแลกเปลี่ยนสกุลเงิน ไม่มีในข้อมูลสมาชิกให้แสดงข้อความเตือนว่า “ไม่มีข้อมูลสมาชิกนี้ในระบบกรุณากรอกข้อมูลใหม่” แล้วให้กลับไปทำข้อที่ 1 ใหม่ <p>จบการทำงาน</p>



ภาพประกอบที่ 3.18 แผนภาพการประมวลผลของการเพิ่มสถานะข้อมูลการขอสินเชื่อ

ตารางที่ 3.15 อธิบายการประมวลผลการเพิ่มข้อมูลบริษัทแลกเปลี่ยนสกุลเงิน

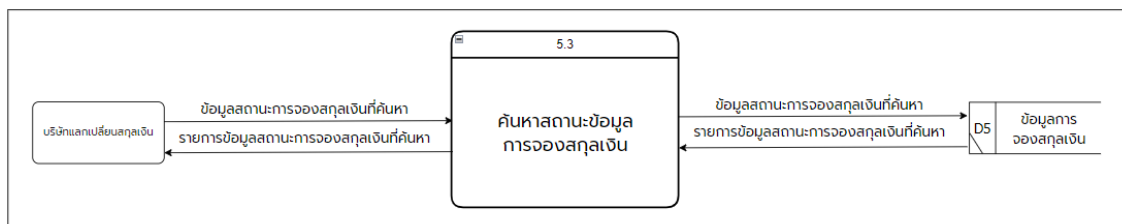
ID	5.1
NAME	เพิ่มสถานะข้อมูลการขอสินเชื่อ
DESCRIPTION	เป็นกระบวนการยืนยันสถานะการขอสินเชื่อของผู้ใช้ที่ทำการขอสินเชื่อเข้ามาในระบบ
INPUT DATA FLOWS	ข้อมูลสถานะการขอสินเชื่อที่เพิ่ม
OUTPUT DATA FLOWS	รายการข้อมูลสถานะการขอสินเชื่อที่เพิ่ม
PROCESS DESCRIPTION	<p>เริ่ม</p> <ol style="list-style-type: none"> 1. รับข้อมูลบริษัทแลกเปลี่ยนสกุลเงิน (รหัสการขอสินเชื่อ + สถานะการขอสินเชื่อ) 2. ตรวจสอบข้อมูล (รหัสการขอสินเชื่อ + สถานะการขอสินเชื่อ) ดังกล่าวว่ามีข้อมูลหรือไม่ <ol style="list-style-type: none"> 2.1 ถ้ามีข้อมูล (รหัสการขอสินเชื่อ + สถานะการขอสินเชื่อ) ดังกล่าวให้บริษัทแลกเปลี่ยนสกุลเงินตัดสินใจเลือก “ยอมรับการขอสินเชื่อ” 2.2 ถ้าไม่มีข้อมูล (รหัสการขอสินเชื่อ + สถานะการขอสินเชื่อ) ดังกล่าวให้ระบบแสดงข้อความว่า “ไม่มีข้อมูลการขอสินเชื่อ” 3. ทำการบันทึกผลลัพธ์ข้อมูล 2.1 ลงฐานข้อมูลการขอสินเชื่อ <p>จบการทำงาน</p>



ภาพประกอบที่ 3.19 แผนภาพการประมวลผลของการลบข้อมูลการจองสกุลเงิน

ตารางที่ 3.16 อธิบายการประมวลผลการลบข้อมูลบริษัทแลกเปลี่ยนสกุลเงิน

ID	5.2
NAME	ลบสถานะข้อมูลการจองสกุลเงิน
DESCRIPTION	เป็นกระบวนการปฏิเสธสถานะการจองสกุลเงินของผู้ใช้ที่ทำการจองสกุลเงินเข้ามาในระบบ
INPUT DATA FLOW	ข้อมูลสถานะการจองสกุลเงินที่ลบ
OUTPUT DATA FLOW	รายการข้อมูลสถานะการจองสกุลเงินที่ลบ
PROCESS DESCRIPTION	<p>เริ่ม</p> <ol style="list-style-type: none"> 1. รับข้อมูลบริษัทแลกเปลี่ยนสกุลเงินที่ต้องการลบ (<u>รหัสการจองสกุลเงิน</u> + สถานะการจองสกุลเงิน) 2. ตรวจสอบข้อมูล (<u>รหัสการจองสกุลเงิน</u> + สถานะการจองสกุลเงิน) ดังกล่าวว่ามีข้อมูลหรือไม่ <ol style="list-style-type: none"> 2.1 ถ้ามีข้อมูล (<u>รหัสการจองสกุลเงิน</u> + สถานะการจองสกุลเงิน) ดังกล่าวให้บริษัทแลกเปลี่ยนสกุลเงินตัดสินใจเลือก “ปฏิเสธการจองสกุลเงิน” 2.2 ถ้าไม่มีข้อมูล (<u>รหัสการจองสกุลเงิน</u> + สถานะการจองสกุลเงิน) ดังกล่าวให้ระบบแสดงข้อความว่า “ไม่มีข้อมูลการจองสกุลเงิน” 3. ทำการบันทึกผลลัพธ์ข้อมูล 2.1 ลงฐานข้อมูลการจองสกุลเงิน <p>จบการทำงาน</p>



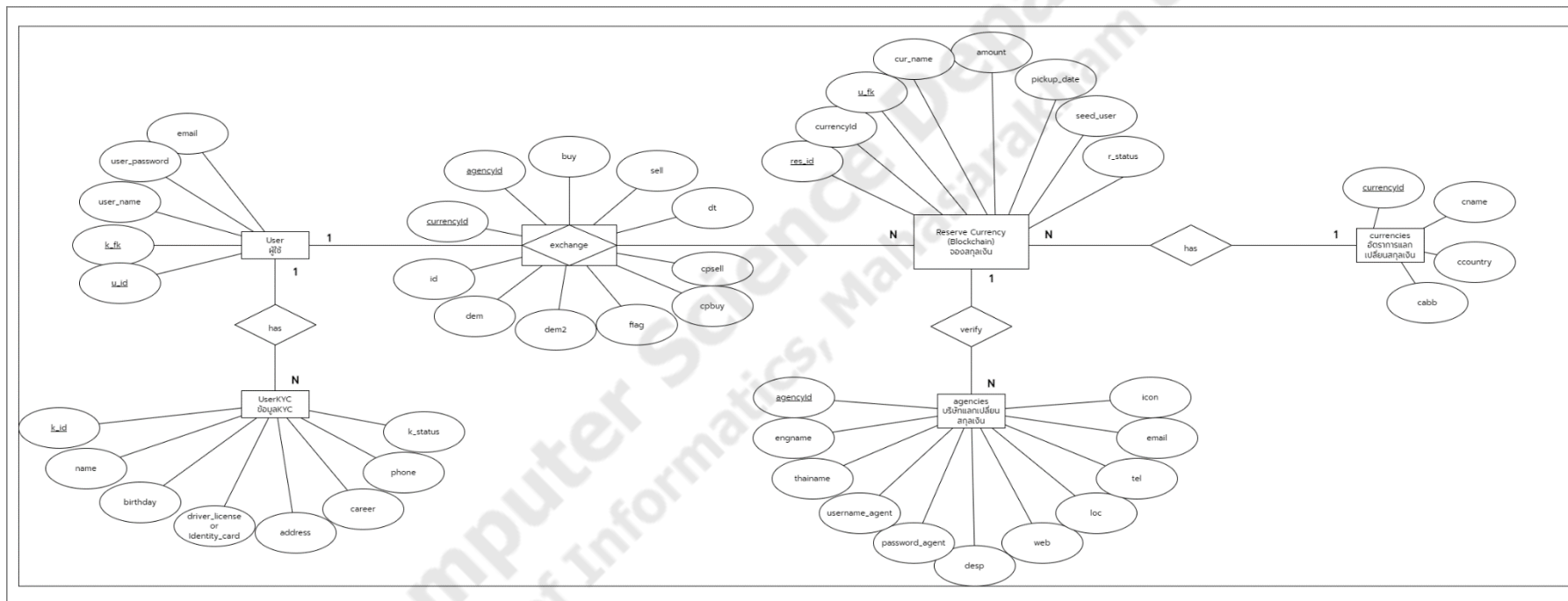
ภาพประกอบที่ 3.20 แผนภาพการประมวลผลของการค้นหาสถานะข้อมูลการจองสกุลเงิน

ตารางที่ 3.17 อธิบายการประมวลผลการค้นหาสถานะข้อมูลการจองสกุลเงิน

ID	5.3
NAME	ค้นหาสถานะข้อมูลการจองสกุลเงิน
DESCRIPTION	เป็นกระบวนการค้นหาสถานะข้อมูลการจองสกุลเงินของบริษัท
INPUT DATA FLOW	- ข้อมูลสถานะข้อมูลการจองสกุลเงินที่ค้นหา
OUTPUT DATA FLOW	- รายการข้อมูลสถานะการจองสกุลเงินที่ค้นหา
PROCESS DESCRIPTION	<p>เริ่มต้น</p> <ol style="list-style-type: none"> 1. รับข้อมูลบริษัทแลกเปลี่ยนสกุลเงิน (รหัสบริษัท + รหัสการจองสกุลเงิน + ชื่อผู้ใช้ + ชื่อสกุลเงิน) 2. ตรวจสอบว่ามีข้อมูลครบถ้วนและถูกต้องหรือไม่ <ol style="list-style-type: none"> 2.1 ถ้าข้อมูลครบถ้วน และถูกต้องทุกรายการให้ทำข้อ 3 2.2 ถ้าข้อมูลครบถ้วน แต่ไม่ถูกต้องให้แสดงข้อความ “กรุณาป้อนข้อมูลให้ถูกต้อง” แล้วให้กลับไปทำข้อ 1. ใหม่ 3. ตรวจสอบในข้อมูลการจองสกุลเงินว่ามีข้อมูลหรือไม่ <ol style="list-style-type: none"> 3.1 ถ้ามีข้อมูล (รหัสบริษัท + รหัสการจองสกุลเงิน + ชื่อผู้ใช้ + ชื่อสกุลเงิน) ในข้อมูลการจองสกุลเงินให้ไปหน้าหลัก 3.2 ถ้าไม่มีข้อมูล (รหัสบริษัท + รหัสการจองสกุลเงิน + ชื่อผู้ใช้ + ชื่อสกุลเงิน) ดังกล่าวให้แสดงข้อความ “ไม่มีข้อมูลการจองสกุลเงินในระบบกรุณากรอกข้อมูลใหม่” แล้วให้กลับไปทำข้อที่ 1. ใหม่ <p>จบการทำงาน</p>

3.4 ความสัมพันธ์(Entity Relationship Diagram)

3.4.1 Entity Relationship Diagram (ER-Diagram)



ภาพประกอบที่ 3.21 ER Diagram

3.5 การออกแบบฐานข้อมูล (Database Design)

ตารางที่ 3.18 ข้อมูลอัตราแลกเปลี่ยนสกุลเงิน (getCurrency)

ID	Column	Type	Description	Example Data	Constraint
1	AID	string	รหัสสมาชิก	s12dwA245WNvxKb25 43d19ew34w	PK
2	DataTimeUpdate	string	เวลาอัปเดตข้อมูล	06_07_2023_16_00_01	Not Null
3	agenName	string	ชื่อบริษัท	K79	Not Null
4	agency	array, map	เก็บข้อมูลอัตราแลกเปลี่ยนสกุลเงินโดยเก็บค่า ID, ราคาซื้อ, อัตราแลกเปลี่ยน, ราคาขาย	Agency: [{ Map: 0 Buy: 34.8 Cur: USD Dem1: 100 Dem2: 100 Sell: 35.1}] etc. Map: 1, 2, 3, 4	Not Null

ตารางที่ 3.19 ข้อมูลผู้ใช้ (usersPIN)

ID	Column	Type	Description	Example Data	Constraint
1	UID	string	รหัสสมาชิก	f8WNvxKbVd19 ewre	PK
2	displayName	string	ชื่อบัญชี	Dolity	Not null
3	FirstName	string	ชื่อจริง	bobo1	Not Null
4	LastName	string	นามสกุล	@yourPASS456	Not Null
5	Gender	string	เพศ	male	Not Null
6	DayofBirth	string	วันที่เกิด	5	Not Null
7	MonthofBirth	string	เดือนเกิด	February	Not Null
8	YearofBirth	string	ปีเกิด	1998	Not Null

ตารางที่ 3.19 ข้อมูลผู้ใช้ (usersPIN) (ต่อ)

ID	Column	Type	Description	Example Data	Constraint
9	IDCardNumber	string	เลขบัตรประชาชน	1486594845	Not Null
10	IDCardBackNumber	string	เลขหลังบัตรประชาชน	ME2W2Dw23dW12	Not Null
11	Phone	string	เบอร์โทรศัพท์	0896547859	Not Null
12	pin	string	รหัสรับเงิน	029456	Not Null
13	ConditionQR	bool	สถานะ TOTP	true	Not Null
14	isVerify	bool	สถานะ KYC	true	Not Null
15	ReservationStatus	bool	สถานะการจองสกุลเงิน	true	Not Null
16	DropOffStatus	bool	สถานะรับเงิน	true	Not Null
17	FCMToken	string	รหัสการแจ้งเตือน	dPE15YNeSWm89lf	Not Null

ตารางที่ 3.20 ข้อมูลผู้ดูแลระบบ (usersPIN)

ID	Column	Type	Description	Example Data	Constraint
1	UID	String	รหัสสมาชิก	f8WNvxKbVd19ewreg	PK
2	displayName	String	ชื่อบัญชี	admin1@admin.com	Not Null
3	FCMToken	String	รหัสการแจ้งเตือน	dPE15YNeSWm89lf	Not Null

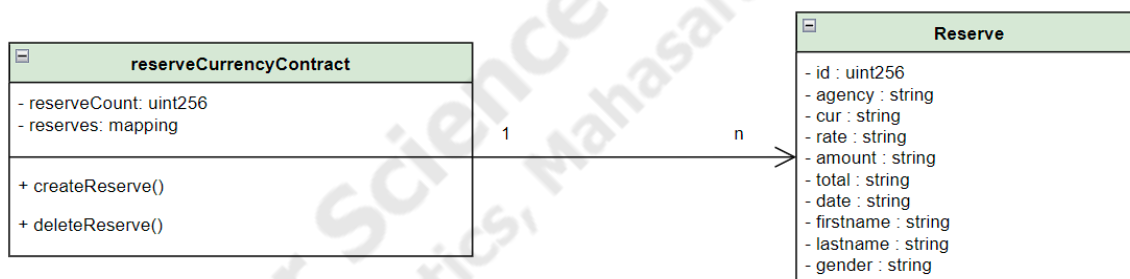
ตารางที่ 3.21 Reserve Currency (Ethereum Sepolia Testnet)

ID	Column	Type	Description	Example Data	Constraint
1	id	uint256	รหัสสมาชิก	1	PK
2	agency	String	ชื่อบริษัท	SRO	Not Null
3	cur	String	ชื่อสกุล	USD	FK
4	rate	String	ราคาอัตราแลกเปลี่ยน	0.2455	Not Null

ตารางที่ 3.21 Reserve Currency (Ethereum Sepolia Testnet) (ต่อ)

ID	Column	Type	Description	Example Data	Constraint
5	amount	String	จำนวนเงินที่ จอง	1000	Not Null
6	total	String	ราคารวมที่จะ ได้รับไทยบาท	245.50 THB	Not Null
7	date	String	วันที่ไปปรับเงิน	2023-07-26 09:00	Not Null
8	firstname	String	ชื่อจริงผู้จอง	Google	Not Null
9	lastname	String	นามสกุลผู้จอง	Login	Not Null
10	gender	String	เพศผู้จอง	Male	Not Null

3.6 การออกแบบ Class Diagram(Blockchain)



ภาพประกอบที่ 3.22 Class Diagram

ส่วนนี้จะอธิบายการนำข้อมูลที่ได้จากการจองอัตราแลกเปลี่ยนสกุลเงิน เพื่อนำข้อมูลที่ได้ไปจัดเก็บบน Blockchains โดยข้อมูลจะมีดังนี้

ตารางที่ 3.22 Class reserveCurrencyContract

Class	reserveCurrencyContract
Attribute	- reserveCount : uint256 - reserves : mapping
Methods	+ CreateReserve() + deleteReserve()

การทำงานของ Attribute และ Methods ใน Class reserveCurrencyContract จะมีดังนี้

1. Attribute reserveCount คือประกาศขึ้นใช้เป็น ID

ในการเข้าทำการจองสกุลเงินในแต่ละลำดับ

2. Attribute reserves คือประกาศให้เป็น mapping เพื่อรองรับข้อมูลที่มีโครงสร้างซับซ้อนทำให้การจองสกุลเงินได้อย่างไม่จำกัดด้วยการใช้ ID อ้างถึง
3. Methods CreateReserve() คือสร้างการจองสกุลเงินขึ้นมาตาม Structure ของ Reserve เพื่อรับข้อมูลจาก Mobile และนำขึ้น Block chain
4. Methods deleteReserve() คือการลบการจองสกุลเงินออกจาก Structure โดยจะลบ Contract และ Node ออกไปทำให้ข้อมูลนั้นหายไปเลย

ตารางที่ 3.23 Class Reserve

Class	Reserve
Attribute	<ul style="list-style-type: none"> - id : uint256 - agency : string - cur : string - rate : string - amount : string - total : string - date : string - firstname : string - lastname : string - gender : string

การทำงานของ Attribute ใน Class Reserve จะมีดังนี้

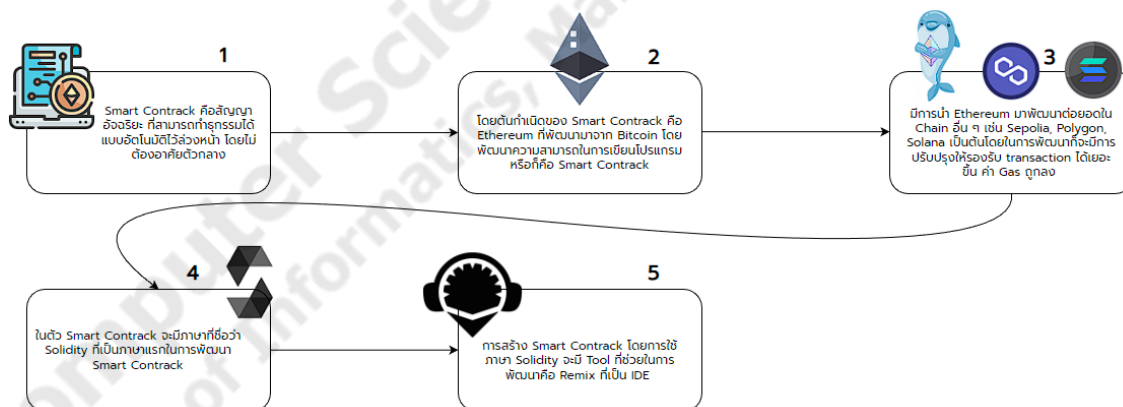
1. Attribute id คือโครงสร้าง id หรือ uid ของผู้ใช้เพื่อเป็น PK ในการเข้าถึงแบบลำดับ
2. Attribute agency คือโครงสร้างมาเพื่อเก็บข้อมูลบริษัทแลกเปลี่ยนสกุลเงินที่ผู้ใช้เลือก
3. Attribute cur คือโครงสร้างเพื่อเก็บข้อมูลสกุลเงินที่ผู้ใช้ได้ทำการจอง
4. Attribute rate คือโครงสร้างเพื่อเก็บข้อมูลอัตราแลกเปลี่ยนสกุลเงิน ณ ตอนนั้นที่ผู้ใช้ทำการจองสกุลเงิน
5. Attribute total คือโครงสร้างเพื่อเก็บข้อมูลราคาที่แลกเปลี่ยนสกุลเงินไทยบาท
6. Attribute date คือโครงสร้างเพื่อเก็บวันที่ผู้ใช้ไปรับเงินที่ทำการจองไว้
7. Attribute firstname คือโครงสร้างที่ใช้เก็บชื่อจริงของผู้ใช้
8. Attribute lastname คือโครงสร้างที่ใช้เก็บนามสกุลของผู้ใช้

9. Attribute gender คือโครงสร้างที่ใช้เก็บเพศของผู้ใช้ที่ทำการจองสกุลเงิน

3.7 การสร้างและการนำไปใช้บน Smart Contract

3.7.1 ขั้นตอนการสร้าง Smart Contract

ขั้นตอนการ Smart Contract จำเป็นต้องรู้ก่อนว่า Smart Contract คืออะไร Smart Contract หรือเรียกในชื่อไทยว่า “สัญญาอัจฉริยะ” ที่สามารถทำธุรกรรมได้แบบ อัตโนมัติโดยที่ไม่จำเป็นต้องอาศัยตัวกลาง ยกตัวอย่างเช่น ธนาคาร โดยต้นกำเนิดของ Smart Contract นั้นมาจาก Ethereum ที่ได้ทำการพัฒนามาจาก Bitcoin โดยจะมีความสามารถในการเขียนโปรแกรม หรือก็คือ ตัว Smart Contract ที่สามารถเขียนขึ้นมาเองได้โดย จะมี Open source ไว้ให้ สามารถนำไปเขียนเพิ่มเติมได้ และการนำ Ethereum มาต่อยอดใน Chain อื่นๆยกตัวอย่างเช่น Binance, Solana, Polygon เป็นต้นโดยในการพัฒนามักจะมีการปรับปรุงให้รองรับสำหรับ Transaction ที่จำนวนมากขึ้น และค่า Gas ที่ถูกลง โดยภาษาที่เอาไว้อพัฒนาตัว Smart Contract นั้นคือ ภาษา Solidity ซึ่งเป็นภาษานามสกุล .sol โดยหลักการเขียนคือจะคล้ายๆกับ Java หรือ OOP (Object Oriented Programming) ที่เอาไว้ใช้เขียน Smart Contract ซึ่งการเขียนจำเป็นต้องมี Tools ในการพัฒนา ณ ที่นี้จะใช้ ตัว Remix Solidity IDE ที่เป็นเขียนบนเว็บหรือบนวินโดว



ภาพประกอบที่ 3.23 ความเป็นมาของ Smart Contract และ Tool

ในขั้นตอนนี้คือ การเขียน Smart Contract โดยใช้ Remix IDE Etherume ในขั้นตอนนี้ที่สำคัญมากๆ จำเป็นต้องกำหนด License คือ บรรทัดแรก เป็นการกำหนด License ให้กับไฟล์ เช่น MIT, Unlicense ส่วนต่อมา ก็จะกำหนด pragma solidity คือการกำหนดตัว compiler ตั้งแต่เวอร์ชัน ตั้งแต่ $\geq 0.4.22 < 0.9.0$ ขึ้นหลังจากนั้นก็กำหนดชื่อ contract โดยจะใช้ตัวอย่างการสร้าง Contract คือ reserveCurrency ใน contract นั้นก็จะกำหนด Attribute ที่ต้องการ ในตัวอย่างดังนี้คือ

การสร้างตัวแปร uint256 noteCount ใช้สำหรับนับข้อมูลที่ถูกสร้างขึ้น

Struct Note ใช้สำหรับเก็บข้อมูลแต่ละรายการโดยจะประกอบไปด้วยโครงสร้างข้อมูลดังนี้

Uint256 id จะเอาไว้เก็บค่าของการจอง ยกตัวอย่างเช่น id: 1

String agency เก็บชื่อบริษัทที่ทำการจองสกุลเงิน ยกตัวอย่างเช่น agency: K79

String cur เก็บชื่อสกุลเงินที่ทำการจองสกุลเงิน ยกตัวอย่างเช่น cur: USD

String rate เก็บอัตราแลกเปลี่ยนสกุลเงินที่ทำการจองสกุลเงิน ยกตัวอย่างเช่น rate: 34.7

String amount เก็บจำนวนเงินที่ทำการจองสกุลเงิน ยกตัวอย่างเช่น amount: 500

String total เก็บราคารวมที่ทำการจองสกุลเงิน ยกตัวอย่างเช่น total: 17,350 THB

String firstname เก็บชื่อจริงที่ทำการจองสกุลเงิน ยกตัวอย่างเช่น firstname: tokdee

String lastname เก็บนามสกุลที่ทำการจองสกุลเงิน ยกตัวอย่างเช่น lastname: mak

String gender เก็บชื่อบริษัทที่ทำการจองสกุลเงิน ยกตัวอย่างเช่น agency: K79

Mapping เก็บคู่ของค่า id map เข้าไปกับโครงสร้างของข้อมูล uint286 => Note

ต่อมาก็จะยกตัวอย่าง function getAll() ก็จะเป็น function ที่เอาไว้เรียกค่าต่างๆที่เราได้ทำการเขียนขึ้นมาเข้ามาใส่ค่าแล้ว return ออกมาแสดงให้ดูว่า มีการเก็บข้อมูลเป็นอย่างไร

```
pragma solidity >=0.4.22 <0.9.0;

contract NotesContractV2 {
    uint256 public noteCount = 0;

    struct Note {
        uint256 id;
        string agency;
        string cur;
        string rate;
        string amount;
        string total;
        string date;
        string firstname;
        string lastname;
        string gender;
    }

    mapping(uint256 => Note) public notes;
```

ภาพประกอบที่ 3.24 การสร้าง Smart Contract ด้วย Remix Ethereum

ใช้ Event เพื่อส่งข้อมูลในโครงสร้างเป็นพารามิเตอร์เข้าไปเพื่อแสดงข้อมูลที่สร้าง Note และ Event ส่ง id เข้าไปเพื่อลบข้อมูลออก

```

event NoteCreated(
    uint256 id,
    string agency,
    string cur,
    string rate,
    string amount,
    string total,
    string date,
    string firstname,
    string lastname,
    string gender
);
event NoteDeleted(uint256 id);

```

ภาพประกอบที่ 3.24 การสร้าง Smart Contract ด้วย Remix Ethereum (ต่อ)

สร้างฟังก์ชัน crateNote เพื่อใช้สร้าง Note ขึ้นมาใหม่โดยรับค่าพารามิเตอร์ตามโครงสร้างและเก็บข้อมูล Note ใหม่ด้วยการ mapping

สร้างฟังก์ชัน deleteNote เป็นฟังก์ชันในการลบ Note ออกโดยการส่ง id เป็นพารามิเตอร์เข้าไป

```

function createNote(
    string memory _agency,
    string memory _cur,
    string memory _rate,
    string memory _amount,
    string memory _total,
    string memory _date,
    string memory _firstname,
    string memory _lastname,
    string memory _gender
) public {
    notes[noteCount] = Note(
        noteCount,
        _agency,
        _cur,
        _rate,
        _amount,
        _total,
        _date,
        _firstname,
        _lastname,
        _gender
    );
}

emit NoteCreated(
    noteCount,
    _agency,
    _cur,
    _rate,
    _amount,
    _total,
    _date,
    _firstname,
    _lastname,
    _gender
);
noteCount++;

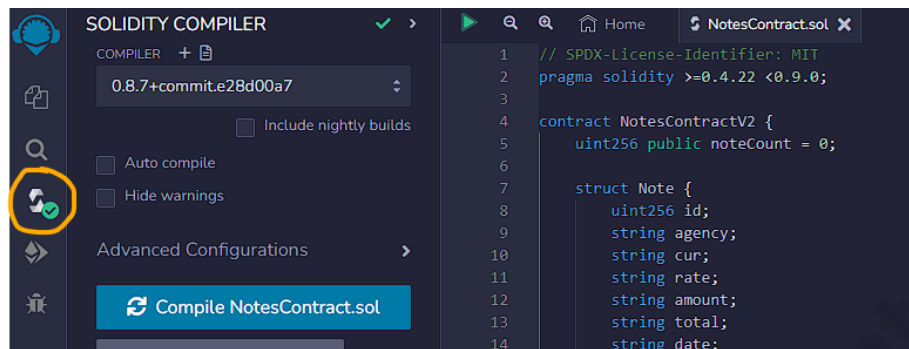
function deleteNote(uint256 _id) public {
    delete notes[_id];
    emit NoteDeleted(_id);
    noteCount--;
}

```

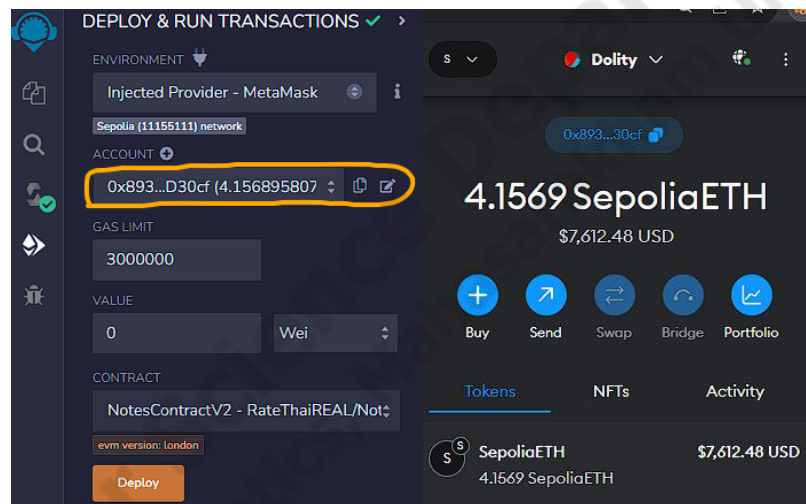
ภาพประกอบที่ 3.24 การสร้าง Smart Contract ด้วย Remix Ethereum (ต่อ)

3.7.2 ขั้นตอนการ Deploy Smart Contract

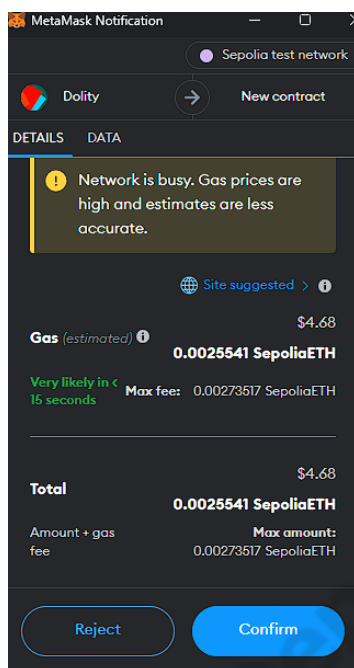
หลังจากทำการสร้าง Smart Contract เสร็จจะต้องทำการ Compile เพื่อตรวจสอบความถูกต้องของ Code เพื่อเตรียมเข้าสู่ขั้นตอน Deploy



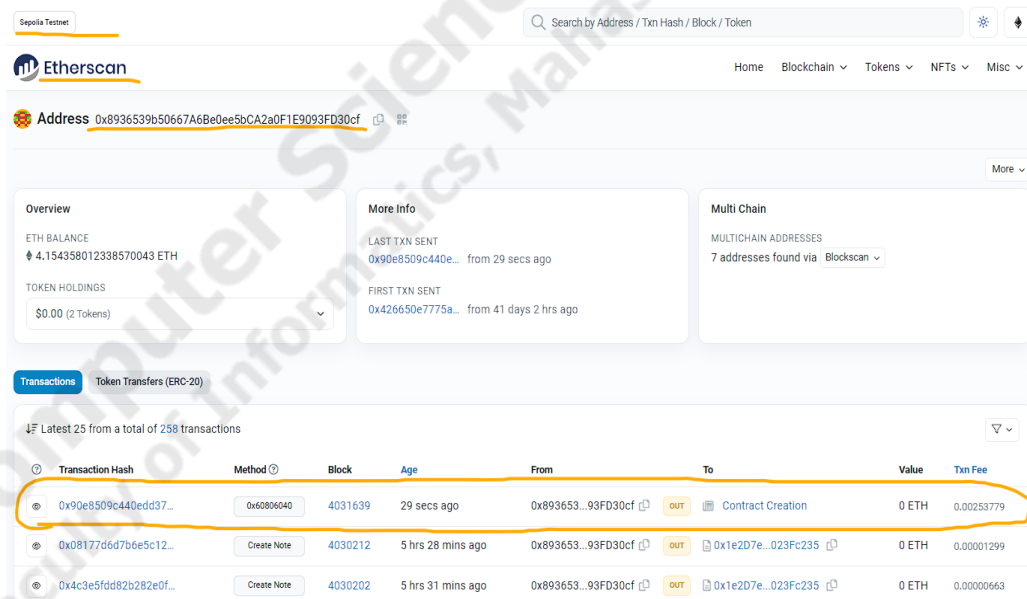
ภาพประกอบที่ 3.25 ทำการ Compile Smart Contract
เปลี่ยน Environment เป็น MetaMask เพื่อ Deploy ขึ้นสู่ Binance Test net โดยทำ
การส่งค่าเข้าไปดังภาพ



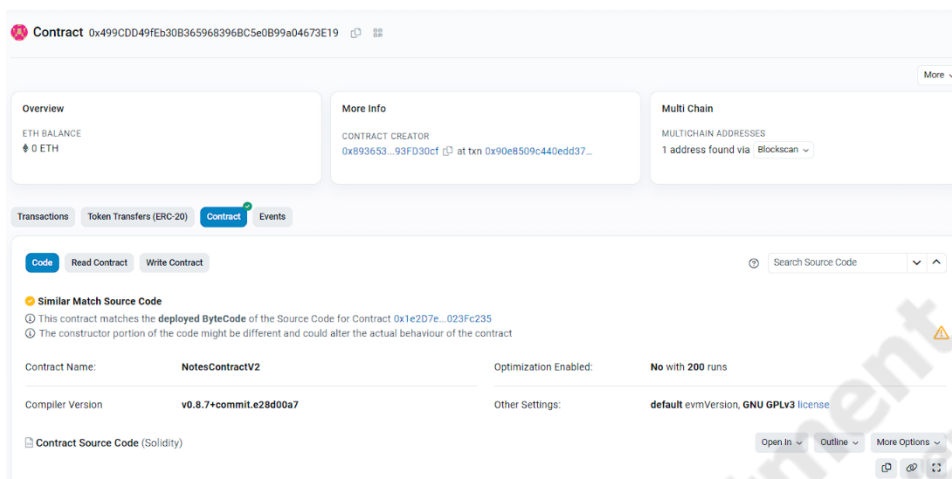
ภาพประกอบที่ 3.26 ทำการ deploy ขึ้นสู่ SepoliaETH Testnet
ในการสร้าง Smart Contract ใหม่จำเป็นต้องเสียค่า Gas ในการอัปโหลด
Smart Contract ขึ้นสู่ SepoliaETH Testnet



ภาพประกอบที่ 3.27 ค่าราคา Gas เพื่อ Deploy Smart Contract
หลังจากชำระค่า Gas ตัว Smart Contract จะถูก Deploy ขึ้นสู่ SepoliaETH Testnet

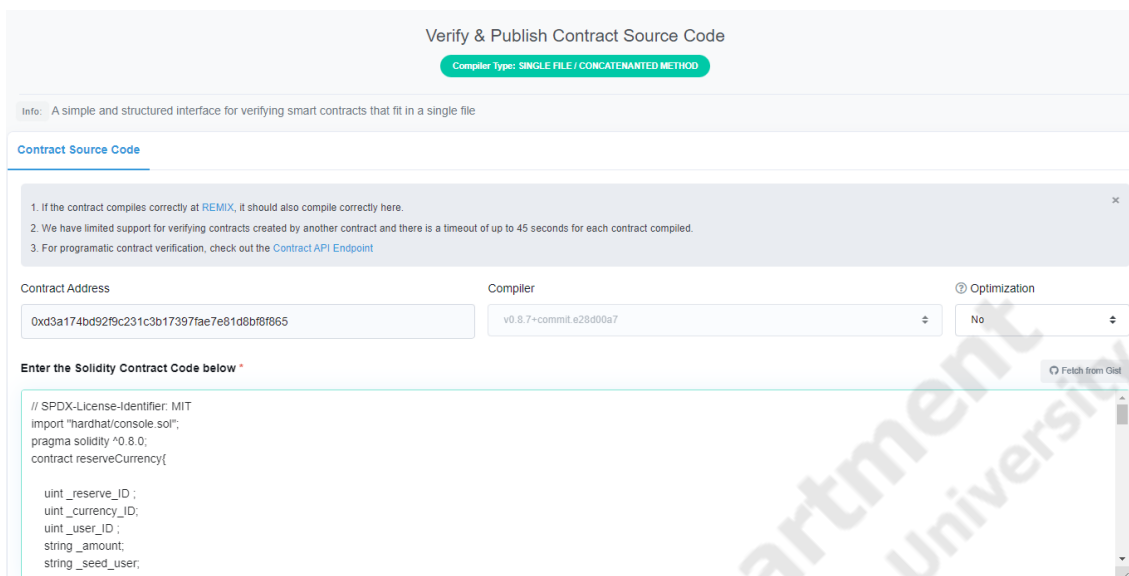


ภาพประกอบที่ 3.28 ทำการ deploy ขึ้นสู่ SepoliaETH Testnet
เมื่อ Smart Contract อยู่บน SepoliaETH Testnet เป็นที่เรียบร้อยแล้วจำเป็นต้องตรวจสอบความเป็นเจ้าของ Smart Contract นี้หรือเปิดเผย Code ในตัว Smart Contract เนื่องจากว่าถ้ายังไม่ตรวจสอบความเป็นเจ้าของหรือเปิดเผย Code ตัว Smart Contract จะเห็นในรูปแบบ Byte code



ภาพประกอบที่ 3.29 หน้าตรวจสอบความเป็นเจ้าของหรือเปิดเผย Code
 ทำการตรวจสอบความเป็นเจ้าของ Smart Contract โดยใช้ compiler Version และ
 License ที่ใช้พัฒนาตัว Smart Contract นี้

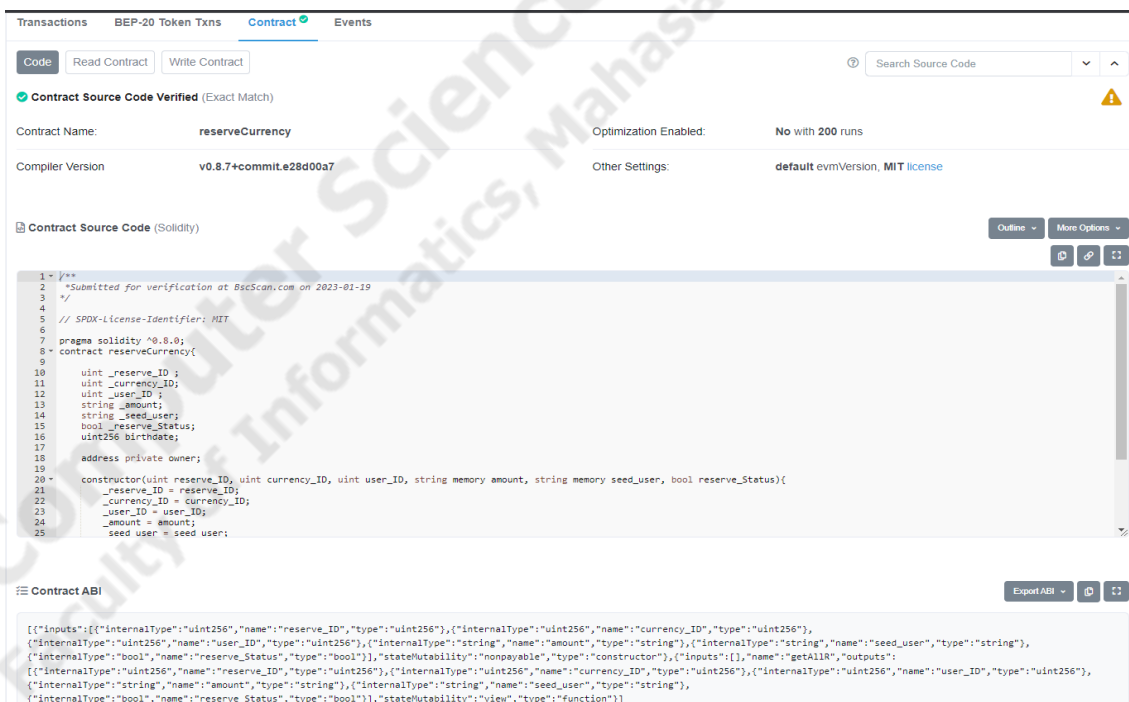
ภาพประกอบที่ 3.30 การเพิ่ม Compiler Version และ License
 ทำการเปิดเผย Code เพื่อยืนยัน Smart Contract โดยจำเป็นต้องใส่ Code ที่ทำการ
 สร้าง Smart Contract ทุกบรรทัดลงไปเพื่อตรวจสอบความเป็นเจ้าของ



ภาพประกอบที่ 3.31 การเปิดเผย Code

หลังจากเปิดเผย Code เสร็จสมบูรณ์จะได้ตัว Contract ABI มาใช้ในการพัฒนา

เชื่อมต่อกับ API ได้



ภาพประกอบที่ 3.32 ยืนยันการเปิดเผย Code และได้รับ Contract ABI

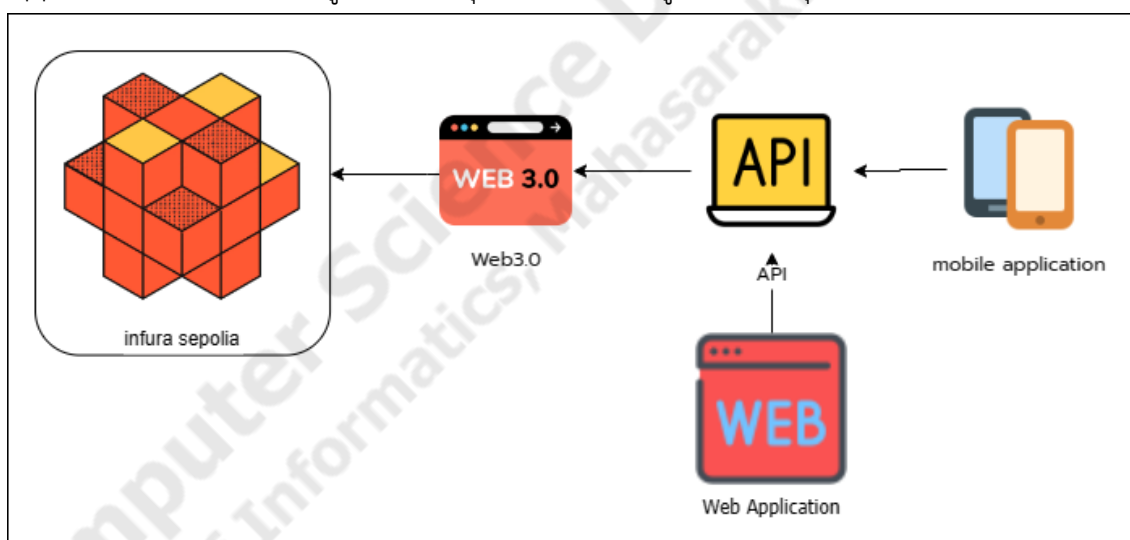
นำ Contract ABI มาพัฒนาต่อยอดด้วยการเชื่อมต่อกับ Node.js เพื่อสร้าง API ออกมาให้กับ Web Application และ Mobile Application

```
Contract ABI
Export ABI
[[{"inputs": [{"internalType": "uint256", "name": "reserve_ID", "type": "uint256"}, {"internalType": "uint256", "name": "currency_ID", "type": "uint256"}, {"internalType": "uint256", "name": "user_ID", "type": "uint256"}, {"internalType": "string", "name": "amount", "type": "string"}, {"internalType": "string", "name": "seed_user", "type": "string"}, {"internalType": "bool", "name": "reserve_Status", "type": "bool"}], "stateMutability": "nonpayable", "type": "constructor"}, {"inputs": [], "name": "getAllR", "outputs": [{"internalType": "uint256", "name": "reserve_ID", "type": "uint256"}, {"internalType": "uint256", "name": "currency_ID", "type": "uint256"}, {"internalType": "uint256", "name": "user_ID", "type": "uint256"}, {"internalType": "string", "name": "amount", "type": "string"}, {"internalType": "string", "name": "seed_user", "type": "string"}, {"internalType": "bool", "name": "reserve_Status", "type": "bool"}], "stateMutability": "view", "type": "function"}]]
```

ภาพประกอบที่ 3.33 Code Contract ABI

3.7.3 ขั้นตอนการใช้ Infura เชื่อมต่อกับ Web3

Infura คือผู้ให้บริการการเชื่อมต่อ Block Chain Ethereum หรือก็คือตัวกลางที่ให้ใช้ API เชื่อมต่อกับ Web3dart เมื่อเชื่อมต่อสำเร็จตัว Infura จะส่งคำขอ API ผ่าน URL ได้เช่น การส่งคำขอเรียกดู Smart Contract ที่อยู่บน Sepolia Testnet ในแอปพลิเคชันนี้ได้ใช้ Sepolia Testnet จึงทำให้เรียกใช้งานฟังก์ชันใน web3dart ได้เช่น ethClient ใช้ติดต่อกับ Mobile Application และใช้สร้างข้อมูลการจองสกุลเงิน และลบข้อมูลการจองสกุลเงิน บน Block Chain ได้



ภาพประกอบที่ 3.34 หลักการเรียกใช้ API โดย Web3

3.7.4 ขั้นตอนการใช้งาน Python Web Scraping

ในการทำ Web Scraping โดยจะทำการเรียก library ที่ชื่อว่า BeautifulSoup และ Selenium ในการใช้ทำ Web Scraping โดยเริ่มต้นจะกำหนด url link ของเว็บไซต์ที่ต้องการจะใช้ Scraping จากเว็บไซต์นั้นๆ

```

2 from bs4 import BeautifulSoup
3 from selenium import webdriver
4 from selenium.webdriver.common.keys import Keys
5 import time
6
7 url = "https://www.superrich1965.com/currency.php"

```

ภาพประกอบที่ 3.35 ขั้นตอนการใช้งาน Python Web Scraping เรียกใช้ library

หลังจากนั้นก็ติดตั้งตัวแปรที่เอาไว้เก็บ path เพื่อที่จะ Run driver ของตัว Selenium เพื่อที่จะให้ตัวแปร Driver นั้นเรียกใช้ url ที่เราได้ตั้งไว้ในตอนแรก โดยจะใช้คำสั่ง driver.page_source

```

9 driver = webdriver.Edge(executable_path='C:/Users/shipd/Desktop/project file/Code project/
10 driver.get(url)
11
12 time.sleep(5)
13
14 html = driver.page_source

```

ภาพประกอบที่ 3.36 ขั้นตอนการใช้งาน Python Web Scraping เรียกใช้ Url ที่กำหนด

ขั้นตอนต่อมา จะติดตั้งตัวแปร soup โดยที่เรียกใช้จาก Library BeautifulSoup โดยใส่ตัว parameter คือ soup(html, "html.parser") แล้วต่อมาก็จะสร้างตัวแปรที่ชื่อว่า all_divs ซึ่งที่เขียนเป็นคำสั่งคือ all_divs = soup.find('div', {'class': 'container'}).text ซึ่งตัวแปรนี้จะทำการเก็บค่าจากตัวแปร soup ที่ไปเก็บค่าต่างๆจาก Tag div อยู่ที่ class ที่มีชื่อว่า container โดยแปลงค่าต่างๆเป็น Text ซึ่งในตัวแปร li และ tr ก็จะทำคล้ายๆกันแต่จะใช้ Method ที่เป็น find_all

```

li = soup.find_all('div', {'class': 'table shadow'})
tr = soup.find_all('table', {'class': 'table'})

```

ทั้งสองตัวแปรนี้ จะทำการหาข้อมูลทั้งหมดที่ได้กำหนดใน parameter ก็คือ tag div แล้วเอาข้อมูลที่ class ซึ่งอยู่ที่ 'table shadow' ซึ่งอีกตัวก็จะทำคล้ายๆกันแต่จะไปดึงข้อมูลที่ Tag table อยู่ที่ class ข้อมูลที่ table

```

16 soup = BeautifulSoup(html, "html.parser")
17 all_divs = soup.find('div', {'class': 'container'}).text
18 # transcript = all_divs.find('table', class_ = 'table')
19 li = soup.find_all('div', {'class': 'table shadow'})
20 tr = soup.find_all('table', {'class': 'table'})

```

ภาพประกอบที่ 3.37 ขั้นตอนการใช้งาน Python Scraping ค้นหา Tag

หลังจากนั้นก็สร้างที่แปรที่เป็น array เพื่อที่จะเอาไว้เก็บค่าที่ได้ Web Scraping จากนั้นก็จะทำการสร้าง for loop เพื่อที่จะเก็บค่าจากการ Web Scraping ทุกๆตัวจากแต่ละ tag แต่ละ div ในเว็บไซต์ที่เราต้องการที่จะ Scraping โดยเราจะกำหนดค่า parameter ต่างๆ คือ จะนำตัวแปร tr มา for loop แล้วเก็บค่าไว้ใน row ที่นำมา for loop คือ จะให้หาตัวค่าที่ต้องการใน tag tr

โดยใน for loop จะสร้างตัวแปร td ซึ่งที่ได้เก็บค่า tr ไว้ใน row ก็จะเรียกใช้ method findAll เพื่อหาค่าที่ต้องการคือ หาค่าจาก td ดังบรรทัด 27 ตามรูปประกอบ ตัวแปร k =0 เพื่อที่จะนำไปหาค่าใน for loop ต่อไป โดยในfor นี้จะเอาไว้หาค่าสกุลเงินที่อยู่เว็บไซต์ที่ต้องการ Web Scraping ว่ามีกี่ค่าสกุลเงินทั้งหมดในเว็บไซต์ที่กำหนด

```

21 td = []
22 cur = []
23 dem = []
24 buy = []
25 sel = []
26 for row in tr:
27     td = row.findAll('td')
28     k =0
29     for t in td:
30         if(k%5 == 0 ):
31             cur.append(t.text[0:3])
32             k+=1
33     k=0
34     p=0

```

ภาพประกอบที่ 3.38 ขั้นตอนการใช้งาน Python Web Scraping กำหนด Tag ที่ต้องการหา

ต่อมาใน for loop จะทำการหาค่า dem, buy, sel และการตัดตัวอักษร ที่ติดมาจาก text html นั่นก็คือ '
' เริ่มแรกก็จะทำการสร้าง for t in td โดยที่ t จะเก็บค่าข้อมูลจาก td เริ่มแรกจะกำหนด ตัวแปร x ที่เอาไว้จัดการกับ text html ที่ติดมาด้วย โดยจะใช้ method split เพื่อทำการแยกอักษรที่ไม่ได้ต้องการ หลังจากนั้นก็จะใช้ method strip เพื่อทำการลบช่องว่าง

```

35     for t in td:
36         x = str(t).split('<br/>')
37         x[0] = x[0].split('>')[1].strip()
38         x[len(x)-1] = x[len(x)-1].split('<')[0].strip()

```

ภาพประกอบที่ 3.39 ขั้นตอนการใช้งาน Python Web Scraping ทำการลบ tag br

หลังจากนั้นก็เข้าเงื่อนไข if คือ ถ้า(len(x[0]) > 0) จะให้ทำเงื่อนไข if ต่อไปคือ ถ้า (x[0].find('<') < 0) จะให้ k +=1 โดยที่ถ้า k== 1+p ก็จะทำให้ ค่าdem ที่ได้กำหนดไว้ในตอนแรก append ไปที่ x หลังจากนั้นก็จะทำการ print ค่าที่ได้ทำการ Scraping มาดูทั้งหมด ว่าที่รายการทั้งหมด

```

39         if(len(x[0]) > 0):
40             if(x[0].find('<') < 0):
41                 k+=1
42                 if k == 1+p :
43                     dem.append(x)
44                 elif k == 2+p:
45                     buy.append(x)
46                 elif k== 3+p:
47                     sel.append(x)
48                 p+=3
49             break
50 rate = []
51
52 print(len(cur), len(sel), len(dem), len(buy))
53

```

ภาพประกอบที่ 3.40 ขั้นตอนการใช้งาน Python Web Scraping เก็บค่าไว้ในตัวแปร และแสดงผล จะเห็นได้ว่าในบรรทัดที่ 50 จะสร้างตัวแปร rate ที่เป็นarray ซึ่งจะเอาไว้เก็บค่า หน่วยเงิน(denomination) โคนจะใช้ for loop ในการหาจากค่าที่ได้จากตัวแปร dem ที่ได้เก็บมาจากตอนต้น เริ่มแรกจะทำการ for loop โดยเก็บค่า dem ไว้ที่ i แล้วสร้างตัวแปร c ที่เอาไว้เก็บค่าcur หลังจากนั้นก็จะสร้าง for loop อีกตัว โดยจะเก็บค่าdem[i]ที่ได้จาก for loop แรกซึ่งจะเก็บไว้ที่ตัวแปร j โดยจะทำการสร้างตัวแปร d ที่เอาไว้ใช้ในการเก็บค่าข้อมูล โดยบรรทัดที่ 58 นั้นจะเป็นการให้ d ที่ข้อมูลที่ชื่อว่า ['cur'] = c.strip() ก็คือให้ข้อมูลที่มีชื่อดังกล่าวนั้นลบช่องว่างออก จากนั้นก็เข้าเงื่อนไข if คือ ถ้า c == 'VND' หรือ c=='IDR' ให้ d[dem1] = dem[i][j][2:6] สามารถอธิบายได้ว่าเมื่อในตัวแปร c มีข้อมูลที่ชื่อว่า VND หรือ IDR จะให้ตัวแปร d สร้าง Objectที่ชื่อ dem1 และ dem2 นั้นหาตั้งแต่ตัวที่ i และ j จากตำแหน่งที่ 2 ถึง6

```

54 for i in range(len(dem)):
55     c = cur[i]
56     for j in range(len(dem[i])):
57         d = {}
58         d['cur'] = c.strip()
59         if c== 'VND' or c=='IDR':
60             d['dem1'] = dem[i][j][2:6]
61             d['dem2'] = dem[i][j][2:6]

```

ภาพประกอบที่ 3.41 ขั้นตอนการใช้งาน Python Web Scraping ทำการ loop เพื่อหาค่า หรือจะเข้าเงื่อนไข else ในบรรทัดที่ 63 ว่าถ้า if(dem[i][j].find('-') > 0) คือถ้าตัวแปร demตัวที่ i และ j หาเครื่องหมาย - เจอจะมีค่ามากกว่า 0 แล้วจะให้ d['dem1'] = dem[i][j].split('-')[1].strip() จะทำการให้ตัวแปร demตัวที่ i และ j แยกเครื่องหมาย - ออกจากตำแหน่งที่ 1 แล้วก็ใช้ คำสั่ง strip() เพื่อลบช่องว่างออก

`d['dem2'] = dem[i][j].split('-')[0].strip()` .ในคำสั่งนี้ก็ทำเหมือนกันแต่จะเป็นในตำแหน่งที่ 0 หรือเข้าเงื่อนไข else อีกรอบว่า

`d['dem1'] = dem[i][j].strip()` ให้ลบช่องว่างข้อมูลที่เก็บไว้ในตัวแปร `dem` ตัวที่ `i` และ `j` ออกโดยใช้คำสั่ง `strip()`

`d['dem2'] = dem[i][j].strip()` ให้ลบช่องว่างข้อมูลที่เก็บไว้ในตัวแปร `dem` ตัวที่ `i` และ `j` ออกโดยใช้คำสั่ง `strip()`

ในส่วนของ บรรทัดที่ 69-70 ให้ตัวแปร `d` สร้าง Object ชื่อ `buy` ,`sell` เพื่อให้เอาไว้เก็บค่าจากตัวแปร `buy` และ `sel` ที่เป็น array ด้วยจะให้ `buy` ตัวที่ `i` และ `j` ใช้คำสั่ง `strip()` เพื่อลบค่าช่องว่าง ในส่วนตัวแปร `sel` ก็ทำเหมือนกัน แล้วนำไปเก็บไว้ที่ `d['buy']`, `d['sell']` หลังจากนั้นก็ทำการ `append` ค่า `d` ทั้งหมดลงตัวแปร `rate`

```

62         else:
63             if(dem[i][j].find('-') > 0):
64                 d['dem1'] = dem[i][j].split('-')[1].strip()
65                 d['dem2'] = dem[i][j].split('-')[0].strip()
66             else:
67                 d['dem1'] = dem[i][j].strip()
68                 d['dem2'] = dem[i][j].strip()
69         d['buy'] = buy[i][j].strip()
70         d['sell'] = sel[i][j].strip()
71
72         rate.append(d)
73

```

ภาพประกอบที่ 3.42 ขั้นตอนการใช้งาน Python Web Scraping ทำการลบช่องว่างและขีด

ในส่วนต่อมาจะทำการนำข้อมูลที่ได้นั้น `export` ออกมาเป็นไฟล์เพื่อให้ง่ายต่อการนำไปใช้งานคืออย่างแรกจากเรียกใช้ `method datetime` เพื่อให้รู้ว่าข้อมูลที่เราได้มานั้นมาจากช่วงเวลาไหน จากนั้นก็จะสร้างตัวแปร `fn` ที่เอาไว้เก็บ `path` ที่จะ `export file` ออกไปก็กำหนด `path` ที่ต้องการ หลังจากนั้น ก็จะใช้คำสั่ง `with open(fn,'w') as data:` โดยด้านในจะมี `for loop` ที่เรียกค่าจาก ตัวแปร `rate` ที่ได้เก็บข้อมูลต่างๆจากการ `Scraping` โดยจะใช้คำสั่ง `write` แล้วก็จะใส่ `parameter` ที่ต้องการจะเขียนข้อมูลลงให้ครบหลังจากนั้นก็ `print('done')` เพื่อทำการเช็คว่าการทำงานสำเร็จ

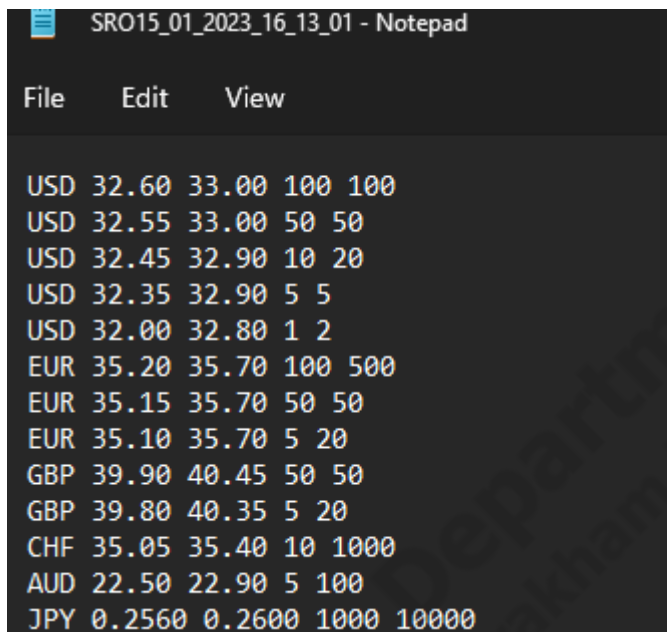
```

74 from datetime import datetime
75
76 fn = 'C:\\Users\\shipd\\Desktop\\project file\\Code project\\currency\\SRO'+str(datetime.now())
77
78 with open(fn,'w') as data:
79     for v in rate:
80         data.write('%s %s %s %s %s\n' % (v['cur'], v['buy'], v['sell'], v['dem1'], v['dem2']))
81 driver.close()
82 print('done')

```

ภาพประกอบที่ 3.43 ขั้นตอนการใช้งาน Python Web Scraping การบันทึกข้อมูล

ซึ่งข้อมูลที่ได้มานั้นจะอยู่ใน File text ในpath ที่ได้กำหนดและนี่คือตัวอย่างของข้อมูล
ที่ได้มา



File	Edit	View		
USD	32.60	33.00	100	100
USD	32.55	33.00	50	50
USD	32.45	32.90	10	20
USD	32.35	32.90	5	5
USD	32.00	32.80	1	2
EUR	35.20	35.70	100	500
EUR	35.15	35.70	50	50
EUR	35.10	35.70	5	20
GBP	39.90	40.45	50	50
GBP	39.80	40.35	5	20
CHF	35.05	35.40	10	1000
AUD	22.50	22.90	5	100
JPY	0.2560	0.2600	1000	10000

ภาพประกอบที่ 3.44 ขั้นตอนการใช้งาน Python Web Scraping แสดงผลข้อมูล

3.8 อธิบายการทำ 2 Factor Authentication

3.8.1 หลักการทำงาน 2 Factor Authentication

ระบบยืนยันตัวตน 2 ชั้น คือ เมื่อผู้ใช้ทำการเข้าสู่ระบบแอปพลิเคชันจะมีหน้าต่างแจ้งเตือนขึ้นมาเพื่อให้กรอกเลขลงไป 6 หลักโดยจะได้รับข้อความผ่าน SMS หรือเป็นระบบ Authenticator อื่น ๆ โดยจะมีหลักการทำงานคือดึงข้อมูลผู้เข้ามาทำการเข้ารหัสแบบ Base32 แล้วจึงนำรหัสนี้ไป Generate ออกมาเป็นชุดตัวเลข 6 หลัก โดยจะมีการสุ่มเลขใหม่ทุก ๆ 30 วินาที โดยหลักการที่กล่าวมาคือ TOTP ส่วนใน HOTP คือการยืนยันตัวตนครั้งเดียวที่มี Token Authentication ที่จัดเก็บ Master Key ทำให้จบการยืนยันตัวตนครั้งเดียวได้

```
// Generate TOTP code. (String versin of function incase of leading 0)
OTP.generateTOTPCodeString("JBSWY3DPEHPK3PXP", 1362302550000); // -> '505548'

// Generate HOTP Code.
OTP.generateHOTPCodeString("JBSWY3DPEHPK3PXP", 7); // -> '006676'
```

ภาพประกอบที่ 3.45 ระบบยืนยันตัวตน TOTP และ HOTP

3.8.2 ขั้นตอนการทำ 2 Factor Authentication

ก่อนอื่นต้องทำการ Import Package ดังนี้คือ บรรทัดที่ 42 base32 เพื่อเข้ารหัสแบบ base32

บรรทัดที่ 43 otp เพื่อรับค่าข้อความหรือชุดตัวเลข 6 หลักมายืนยันตัวตนและ

บรรทัดที่ 44 barcode_widget เพื่อใช้ในการ gen QRcode ขึ้นมาเพื่อให้แอป Google Authenticator สแกนเพื่อรับรหัส 2 FA

```
42   base32: ^2.1.3
43   otp: ^3.1.3
44   barcode_widget: ^2.0.3
45   pin_code_fields: ^7.4.0
```

ภาพประกอบที่ 3.46 เพิ่ม Package ใน dependencies

ต่อมาทำการ import package จาก dependencies เข้ามาใช้งานใน lib ของ drat File

```
1  import 'package:barcode_widget/barcode_widget.dart';
2  import 'package:flutter/material.dart';
3  import 'package:otp/otp.dart';
4  import 'package:base32/base32.dart';
5  import 'package:pin_code_fields/pin_code_fields.dart';
6  import 'package:testprojectbc/page/loginsuccess.dart';
```

ภาพประกอบที่ 3.47 import package

จากภาพบรรทัดที่ 30 ทำการเข้ารหัสชื่อ jakkrit ด้วย method encodeString()

บรรทัดที่ 31-38 ประกาศ Attitude ต่าง ๆ เตรียมพร้อมไว้สำหรับการทำ 2 FA

```
28  class _GooglefaPage extends State<GooglefaPage> {
29
30    final String _authKeySecret = base32.encodeString('jakkrit');
31    String? _currentCode;
32    String? title;
33
34    bool light = true;
35    bool verifyResult = false;
36
37    TextEditingController text2faController = TextEditingController();
38    String currentText = "";
```

ภาพประกอบที่ 3.48 การเข้ารหัสด้วย Base32

การสร้าง Widget ขึ้นมาเพื่อวาดเขียนแอปพลิเคชัน โดย Return เป็น Scaffold ออกไป ใน ส่วนของ AppBar ตั้งชื่อว่า 2FA จะแสดงผลข้างบนสุดในแอปพลิเคชัน มี body เป็น Center เพื่อให้ Components ต่าง ๆ จัดอยู่กลางหน้าจอ โดยจะมี SingleChildScrollView เพื่อให้เลื่อนแอปพลิเคชันขึ้นลงได้ไม่ทำให้เกิดปัญหา Pixel เกินล้นจอ สร้าง Column ขึ้นมาเพื่อให้ ลูกของ <Widget> จัดกันเป็นแถวแล้วอยู่ตรงกลางเสมอ และมีการเขียนชื่อข้างบนหน้ากระดาษเป็นตัวหนังสือหนาว่า TOTP Auth

```

41 Widget build(BuildContext context) {
42   return Scaffold(
43     appBar: AppBar(
44       title: Text("2FA"),
45     ), // AppBar
46     body: Center(
47       child: SingleChildScrollView(
48         child: Column(
49           mainAxisAlignment: MainAxisAlignment.center,
50           // crossAxisAlignment: CrossAxisAlignment.start,
51           children: <Widget>[
52             Text(
53               'TOTP Auth',
54               style: Theme.of(context).textTheme.headline4,
55             ), // Text
56           ],
57         ),
58       ),
59     ),
60   );
61 }

```

ภาพประกอบที่ 3.49 การสร้าง Widget และ child

ทำการสร้าง Padding ขึ้นมาเพื่อให้ขยับทุกส่วนเข้ามา บรรทัดที่ 74 โดยจะมีลูกเป็น PincodeTextField เป็นแป้นพิมพ์ใส่ตัวเลขสำหรับการกรอกรหัสยืนยันตัวตนใน บรรทัดที่ 75-85 จะเป็นการกำหนดธีมสีให้กับ Pincode และ บรรทัดที่ 86-88 เป็นการกำหนดกราฟฟิคและสีพื้นหลังของ Pincode บรรทัดที่ 89 จะทำการสร้าง controller ขึ้นมาเพื่อควบคุม Input ที่ใส่ลงไป controller นี้ดังภาพ

```

72 |   Padding(
73 |     padding: EdgeInsets.all(20.0),
74 |     child: PinCodeTextField(
75 |       length: 6,
76 |       obscureText: false,
77 |       animationType: AnimationType.fade,
78 |       pinTheme: PinTheme(
79 |         shape: PinCodeFieldShape.box,
80 |         borderRadius: BorderRadius.circular(5),
81 |         fieldHeight: 50,
82 |         fieldWidth: 40,
83 |         activeFillColor: Colors.white,
84 |
85 |       ), // PinTheme
86 |       animationDuration: const Duration(milliseconds: 300),
87 |       backgroundColor: Colors.blue.shade50,
88 |       enableActiveFill: true,
89 |       controller: text2faController,
90 |       onCompleted: (v) {
91 |         debugPrint("Completed");
92 |       },

```

ภาพประกอบที่ 3.50 การสร้าง PinCodeTextField และการตกแต่ง

onChanged คือ Action Listener เมื่อมีการกดจะทำงานดังนี้คือ บรรทัดที่ 95 จะทำการดึงวันเดือนปีและเวลามาทั้งหมด บรรทัดที่ 96 ทำการเรียก method generateTOTPCodeString() โดยจะมีค่า _authKeySecret ที่เป็น รหัสยืนยันตัวตนที่เข้ารหัส base32 มาแล้วดู gen ออกเป็นชุดตัวเลข 6 หลัก โดยบรรทัดที่ 98 จะดึงเอาเวลาปัจจุบัน บรรทัดที่ 99 ประกาศถึงความยาวของรหัสยืนยันตัวตน บรรทัดที่ 100 ประกาศถึงค่าที่รหัสยืนยันจะทำการรีเซ็ตใหม่ทุก ๆ 30 วินาที บรรทัดที่ 101 ใช้การเข้ารหัสแบบ SHA1 บรรทัดที่ 103 ตั้งค่าการเข้ารหัสและเวลาให้ตรงกับ Google Authenticator

```

93 |   onChanged: (value) {
94 |
95 |     int currentTime = DateTime.now().millisecondsSinceEpoch;
96 |     var timedCode = OTP.generateTOTPCodeString(
97 |       _authKeySecret,
98 |       currentTime,
99 |       length: 6,
100 |       interval: 30,
101 |       algorithm: Algorithm
102 |         .SHA1, // PLEASE change this from default/ suggested
103 |       isGoogle: true,
104 |     );
105 |     setState(() {
106 |       _currentCode = timedCode;
107 |     });

```

ภาพประกอบที่ 3.51 ตั้งค่า TOTP

ทำการกำหนดเงื่อนไขให้ pinCode ทำงานโดยการใช้ค่าที่ใส่ลง บรรทัดที่ 109 ถ้าพิมพ์เลขครบ 6 หลักถึงจะเข้าเงื่อนไขทำงานดังต่อไปนี้ บรรทัดที่ 110 ทำการเข้ารหัส jakkrit ด้วย base32

และบรรทัดที่ 111 ทำการเรียกใช้ method `constantTimeVerification` เพื่อเปรียบเทียบค่า 2 ค่า ว่าตรงกันหรือไม่ ถ้าตรงกันจะได้ผลลัพธ์ที่เก็บในตัว `verify` ออกมาเป็น `true` และถ้าไม่ตรงก็จะส่งผลลัพธ์ออกมาเป็น `false`

```

109     if(value.length == 6){
110         var secret = base32.encodeString('jakkrit');
111         var verify = OTP.constantTimeVerification(timedCode, value);

```

ภาพประกอบที่ 3.52 เงื่อนไขการทำงานของ `pinCode` การเข้ารหัสและเปรียบเทียบค่าเช็ค OTP กำหนดเงื่อนไขถ้า `verify` เป็น `true` คือ ค่า 2 ค่าที่เปรียบเทียบเท่ากันจะทำการแสดงผลว่า TOTP Pass บรรทัดที่ 115-118 จะทำการนำทางไปยังหน้า `LoginSuccessPage` ถ้า `verify` เป็น `false` จะทำการแสดงผล TOTP Not Pass และจะตั้ง State ใหม่โดยการเคลียร์ค่าใน `pinCode`

```

113     if (verify){
114         print("TOTP Passed!");
115         Navigator.pushReplacement(context,
116             MaterialPageRoute(builder: (context){
117                 return (LoginSuccessPage());
118             })); // MaterialPageRoute
119     } else {
120         print("TOTP Not Pass!");
121         setState() {
122             text2faController.clear();
123             verifyResult = true;
124         };

```

ภาพประกอบที่ 3.53 เงื่อนไขถ้า TOTP ผ่านและเงื่อนไขถ้า TOTP ไม่ผ่าน

สร้าง `Padding` ขึ้นมาเพื่อขยับค่าทุกมุมและมีลูกเป็น `Container` โดยบรรทัดที่ 139 จะมีการนำ `BarcodeWidget` มาใช้งาน บรรทัดที่ 140-141 ทำการสร้างความกว้าง ความยาวของ `QRcode` บรรทัดที่ 142 ทำการใส่ค่าที่ต้องการลงไปใน `QRcode` โดยจะมีการใช้ `Format` ของ `Google Authenticator` ตามด้วยเข้ารหัสชื่อ `jakkrit` ด้วย `base32` และบรรทัดที่ 143 ทำการ `gen QRcode` ขึ้นมา

```

136     Padding(
137         padding: EdgeInsets.all(15.0),
138         child: Container(
139             child: BarcodeWidget(
140                 width: 320,
141                 height: 320,
142                 data: "otpauth://totp/projectBC?secret=" + base32.encodeString('jakkrit') + "&issuer=Google",
143                 barcode: Barcode.qrCode(),
144             ), // BarcodeWidget
145         ), // Container
146     ), // Padding

```

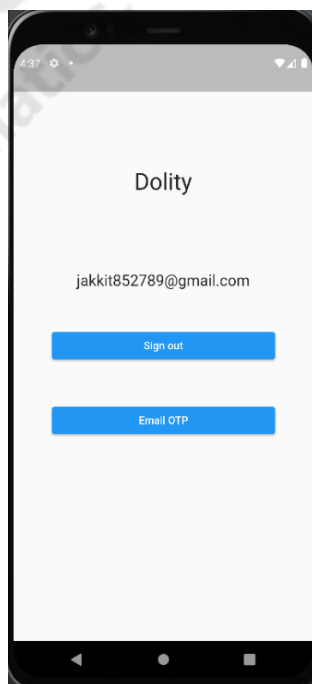
ภาพประกอบที่ 3.54 ขั้นตอนการสร้าง QRcode

ภาพของแอปพลิเคชันขั้นตอนการยืนยันตัวตนด้วย TOTP ผ่าน Google Authenticator



ภาพประกอบที่ 3.55 ขั้นตอนการยืนยันตัวตน 2 FA

หลังจากยืนยันตัวตน 2 FA สำเร็จระบบจะนำทางมายังหน้า LoginsuccessPage โดยในภาพทำการล็อกอินผ่าน google ทำให้มีการแสดงผลออกมาเป็นชื่อเล่นและอีเมลที่ลงชื่อเข้าใช้งานผ่าน google



ภาพประกอบที่ 3.56 เมื่อยืนยันตัวตนสำเร็จ