

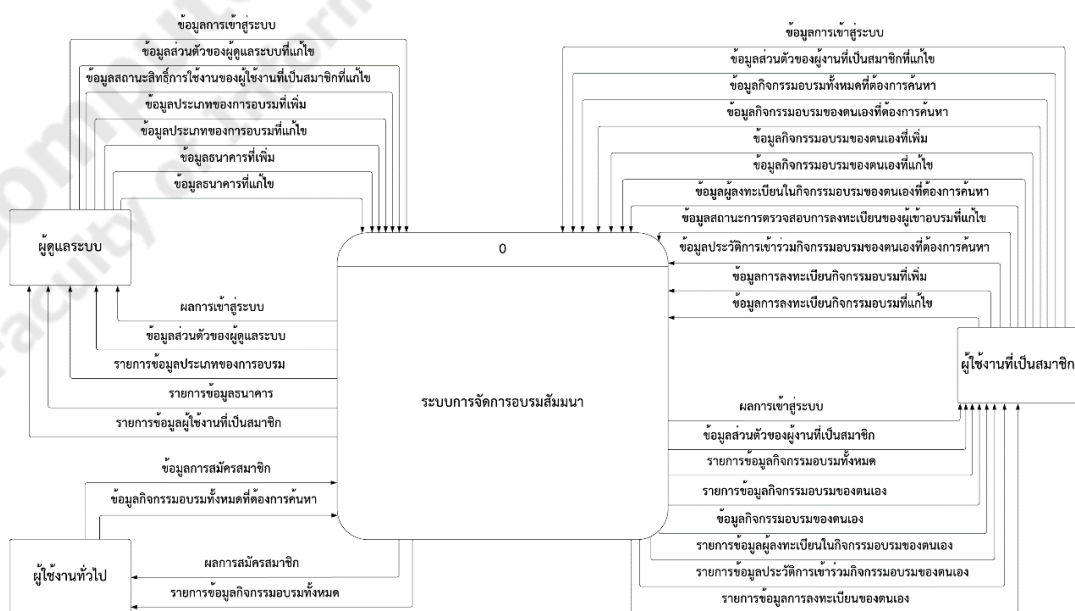
บทที่ 3

ขั้นตอนการดำเนินงาน

สำหรับในบทนี้จะกล่าวถึงขั้นตอนในการดำเนินงานของโครงการปริญญาโทซึ่งจะทำให้ทราบถึงการวิเคราะห์และการออกแบบโดยละเอียด ว่ามีแนวทางในการทำงานและมีขั้นตอนในการทำงานอย่างไรบ้างโดยขั้นตอนในการทำงานมีรายละเอียดดังนี้

- 3.1 แผนภาพบริบท (Context Diagram)
- 3.2 แผนภาพการไหลของข้อมูล (Data Flow Diagram)
- 3.3 พจนานุกรมข้อมูล (Data Dictionaries)
 - 3.3.1 External Entity Description
 - 3.3.2 Data flow (Data flow Description and Data Structure of Data Flow)
 - 3.3.3 Data Store (Data flow Description and Data Structure of Data Store)
 - 3.3.4 คำอธิบายการประมวลผล (Process Description and Specification)
- 3.4 แผนภาพ Entity Relationship Diagram (ER - Diagram)
- 3.5 การออกแบบฐานข้อมูล (Database Design)
- 3.6 การพัฒนาระบบ

3.1 แผนภาพบริบท (Context Diagram)

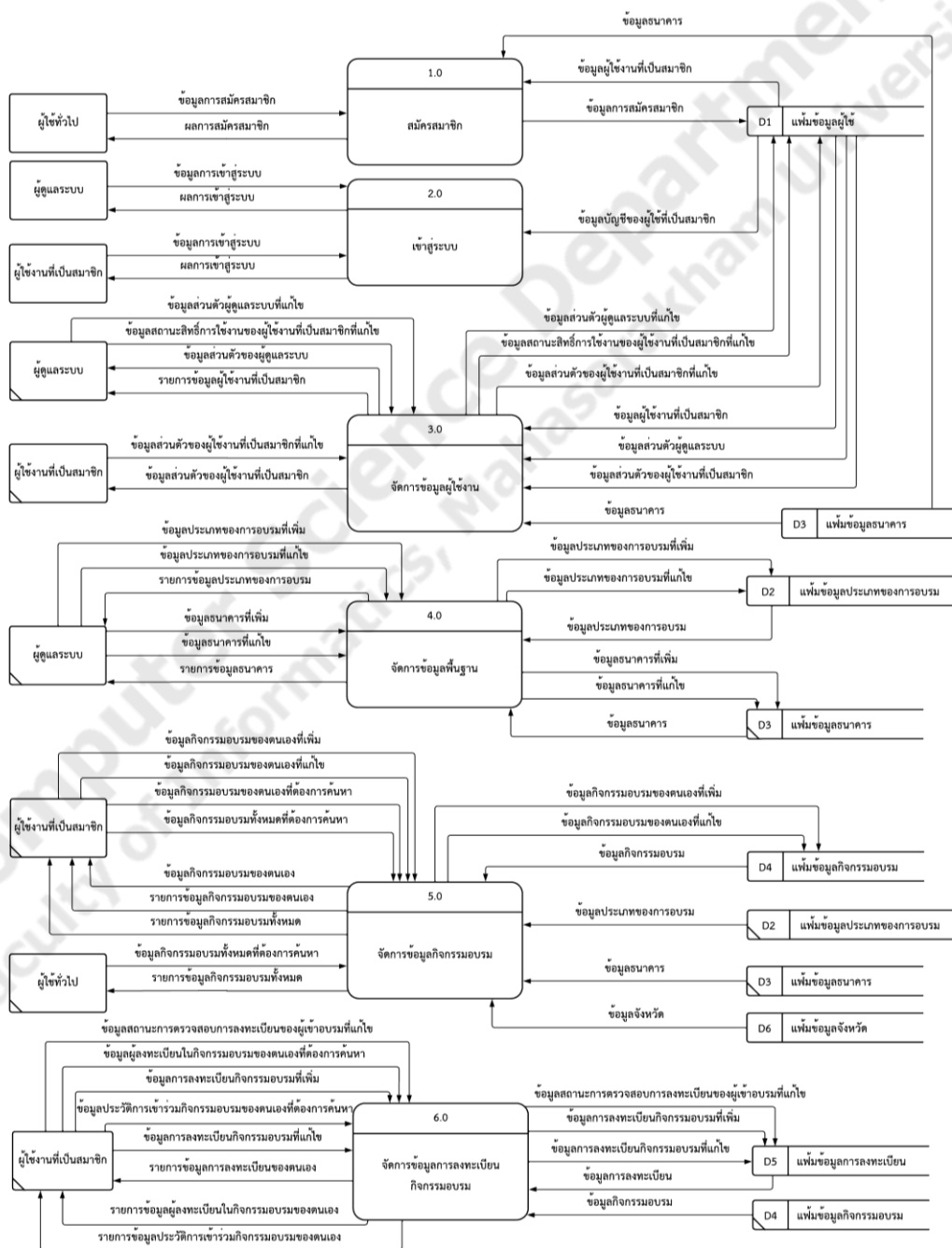


ภาพประกอบที่ 3.1 Context Diagram

3.2 แผนภาพการไหลของข้อมูล (Data Flow Diagram)

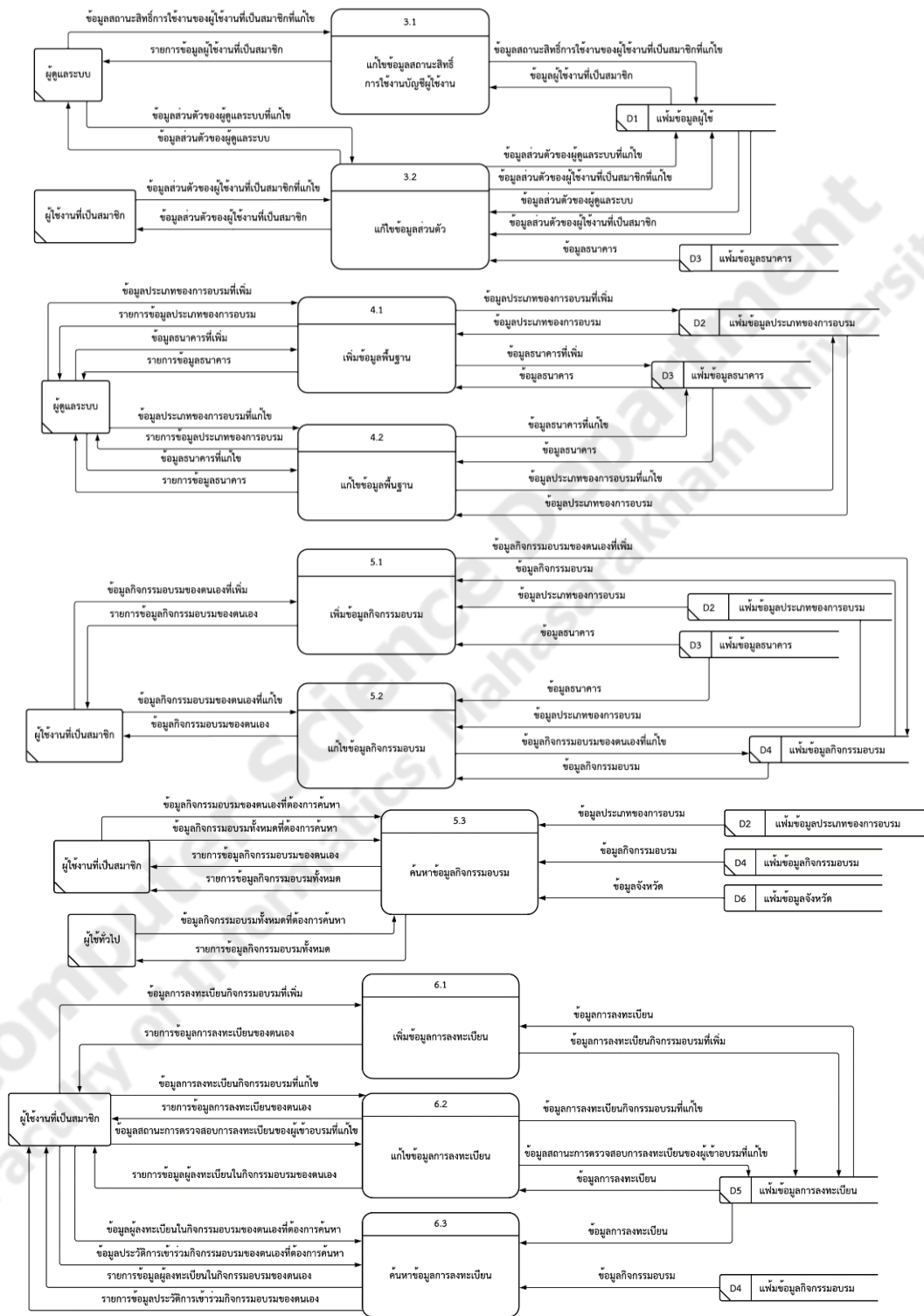
แผนภาพกระแสข้อมูลที่มีการวิเคราะห์ในเชิงโครงสร้าง ซึ่งเป็นแผนภาพที่บอกรายละเอียดของระบบโดยเฉพาะข้อมูล ที่ทราบถึงการรับและส่งข้อมูล การประสานงานในระหว่างกิจกรรมต่าง ๆ ในการดำเนินงานของระบบ แสดงถึงการไหลของข้อมูลทั้ง Input และ Output ระหว่างระบบกับผู้ใช้งาน

3.2.1 Data Flow Diagram Level 1



ภาพประกอบที่ 3.2 Data Flow Diagram Level 1

3.2.2 Data Flow Diagram Level 2



ภาพประกอบที่ 3.3 Data Flow Diagram Level 2

3.3 พจนานุกรมข้อมูล (Data Dictionaries)

พจนานุกรมข้อมูล เป็นส่วนที่ใช้ในการอธิบายถึงกระแสข้อมูลหรือแหล่งเก็บข้อมูลที่ปรากฏอยู่ในแผนภาพแสดงการไหลของกระแสข้อมูล

3.3.1 External Entity Description

ตารางที่ 3.1 External Entity Description

Name	Description	Input Data Flow	Output Data Flow
ผู้ใช้งานทั่วไป	บุคคลที่ไม่ได้เป็นสมาชิกของระบบ	<ul style="list-style-type: none"> - ผลการสมัครสมาชิก - รายการข้อมูลกิจกรรมอบรมทั้งหมด 	<ul style="list-style-type: none"> - ข้อมูลการสมัครสมาชิก - ข้อมูลกิจกรรมอบรมทั้งหมดที่ต้องการค้นหา
ผู้ดูแลระบบ	ผู้ที่ดูแลจัดการการใช้งานของเว็บ	<ul style="list-style-type: none"> - ผลการเข้าสู่ระบบ - ข้อมูลส่วนตัวของผู้ดูแลระบบ - รายการข้อมูลประเภทของการอบรม - รายการข้อมูลธนาคาร - รายการข้อมูลผู้ใช้งานที่เป็นสมาชิก 	<ul style="list-style-type: none"> - ข้อมูลการเข้าสู่ระบบ - ข้อมูลส่วนตัวของผู้ดูแลระบบที่แก้ไข - ข้อมูลประเภทของการอบรมที่เพิ่ม - ข้อมูลประเภทของการอบรมที่แก้ไข - ข้อมูลธนาคารที่เพิ่ม - ข้อมูลธนาคารที่แก้ไข - ข้อมูลสถานะสิทธิ์ การใช้งานของผู้ใช้งานที่เป็นสมาชิกที่แก้ไข
ผู้ใช้งานที่เป็นสมาชิก	ผู้ที่สามารถจัดการข้อมูลของกิจกรรมอบรม	<ul style="list-style-type: none"> - ผลการเข้าสู่ระบบ - ข้อมูลส่วนตัวของพนักงานที่เป็นสมาชิก - รายการข้อมูลกิจกรรมอบรมทั้งหมด - รายการข้อมูลกิจกรรมอบรมของตนเอง - ข้อมูลกิจกรรมอบรมของตนเอง - รายการข้อมูลผู้ลงทะเบียนในกิจกรรมอบรมของตนเอง - รายการข้อมูลการลง 	<ul style="list-style-type: none"> - ข้อมูลการเข้าสู่ระบบ - ข้อมูลส่วนตัวของพนักงานที่เป็นสมาชิกที่แก้ไข - ข้อมูลกิจกรรมอบรมทั้งหมดที่ต้องการค้นหา - ข้อมูลกิจกรรมอบรมของตนเองที่ต้องการค้นหา - ข้อมูลกิจกรรมอบรมของตนเองที่เพิ่ม - ข้อมูลกิจกรรมอบรมของตนเองที่แก้ไข - ข้อมูลผู้ลงทะเบียนในกิจกรรมอบรมของตนเองที่ต้องการค้นหา - ข้อมูลสถานะการตรวจสอบการลง

ตารางที่ 3.1 External Entity Description (ต่อ)

Name	Description	Input Data Flow	Output Data Flow
		ทะเบียนของตนเอง - รายการข้อมูลประวัติ การเข้าร่วมกิจกรรม อบรมของตนเอง	ทะเบียนของผู้เข้าอบรมที่แก้ไข - ข้อมูลการลงทะเบียนกิจกรรมอบรม ที่เพิ่ม - ข้อมูลการลงทะเบียนกิจกรรมอบรม ที่แก้ไข - ข้อมูลประวัติการเข้าร่วมกิจกรรม อบรมของตนเองที่ต้องการค้นหา

3.3.2 Data flow (Data flow Description and Data Structure of Data Flow)

เป็นขั้นตอนการทำงานของระบบซึ่งทำให้เราทราบถึงการรับ-ส่งข้อมูลแสดงถึงการไหลของข้อมูลทั้งข้อมูลเข้า (Input) และข้อมูลส่งออก (Output) ระหว่างข้อมูลต้นทางถึงข้อมูลปลายทางโดยอธิบายข้อมูลและขั้นตอนในการดำเนินงานดังต่อไปนี้

ตารางที่ 3.2 Data flow (Data flow Description and Data Structure of Data Flow)

Name	Description	Source	Destination	Data Structure
ข้อมูลการสมัครสมาชิก	รายละเอียดข้อมูลที่ใช้ทั่วไป ต้องการออกเพื่อสมัครเป็นสมาชิกของเว็บ	ผู้ใช้ทั่วไป	Process 1.0 สมัครสมาชิก	รหัสสมาชิก+คำนำหน้า ชื่อ=[นาย นาง นางสาว]+ ชื่อ+นามสกุล+(วันเดือนปี เกิด)+เบอร์ติดต่อ+อีเมล+ (ที่อยู่=(ถนน)+(ตำบล)+ (อำเภอ)+(จังหวัด)+(เลข ไปรษณีย์))+(รูปโปรไฟล์)+ (รหัสธนาคาร)+(เลขที่ บัญชี)+(ชื่อเจ้าของบัญชี)+ รหัสผ่าน+ประเภทผู้ใช้งาน+ สถานะการใช้งาน
		Process 1.0 สมัครสมาชิก	D1 เพิ่มข้อมูล ผู้ใช้	
ข้อมูลผู้ใช้งานที่เป็นสมาชิก	ข้อมูลผู้ใช้งานที่เป็นสมาชิก	D1 เพิ่มข้อมูล ผู้ใช้	Process 1.0 สมัครสมาชิก	รหัสสมาชิก+คำนำหน้าชื่อ+ ชื่อ+นามสกุล+วันเดือนปี เกิด+เบอร์ติดต่อ+อีเมล+ที่ อยู่+รูปโปรไฟล์+รหัสสนา
			Process 3.1 แก้ไขข้อมูล	

ตารางที่ 3.2 Data flow (Data flow Description and Data Structure of Data Flow) (ต่อ)

Name	Description	Source	Destination	Data Structure
			สถานะสิทธิ์การใช้งาน การจ้างงาน บัญชีผู้ใช้งาน Process 3.2 แก้ไขข้อมูลส่วนตัว	การ+เลขที่บัญชี+ชื่อเจ้าของบัญชี+รหัสผ่าน+ประเภทผู้ใช้งาน+สถานะการใช้งาน
ผลการสมัครสมาชิก	ผลการสมัครสมาชิก	Process 1.0 สมัครสมาชิก	ผู้ใช้ทั่วไป	[สมัครสมาชิกสำเร็จ สมัครสมาชิกไม่สำเร็จ]
ข้อมูลการเข้าสู่ระบบ	ข้อมูลการเข้าสู่ระบบ	ผู้ดูแลระบบ ผู้ใช้งานที่เป็นสมาชิก	Process 2.0 เข้าสู่ระบบ	อีเมล+รหัสผ่าน
ข้อมูลบัญชีของผู้ใช้ที่เป็นสมาชิก	ข้อมูลบัญชีของผู้ใช้ที่เป็นสมาชิก	D1 เพิ่มข้อมูลผู้ใช้	Process 2.0 เข้าสู่ระบบ	อีเมล+รหัสผ่าน+สถานะการใช้งาน
ผลการเข้าสู่ระบบ	ผลจากการที่ผู้ใช้ทำการเข้าสู่ระบบ	Process 2.0 เข้าสู่ระบบ	ผู้ดูแลระบบ ผู้ใช้งานที่เป็นสมาชิก Process 3.2 แก้ไขข้อมูลส่วนตัว Process 5.1 เพิ่มข้อมูลกิจกรรมอบรม Process 5.2 แก้ไขข้อมูลกิจกรรมอบรม	[เข้าสู่ระบบสำเร็จ เข้าสู่ระบบไม่สำเร็จ]
ข้อมูลสถานะสิทธิ์การใช้งานบัญชีผู้ใช้งานที่	ข้อมูลสถานะสิทธิ์การใช้งานบัญชีผู้ใช้งานที่	ผู้ดูแลระบบ	Process 3.1 แก้ไขข้อมูลสถานะสิทธิ์	สถานะการใช้งาน=[0 1]

ตารางที่ 3.2 Data flow (Data flow Description and Data Structure of Data Flow) (ต่อ)

Name	Description	Source	Destination	Data Structure
เป็นสมาชิกที่ แก้ไข	เป็นสมาชิกที่ แก้ไข		การใช้งาน บัญชีผู้ใช้งาน	
		Process 3.1 แก้ไขข้อมูล สถานะสิทธิ์ การใช้งาน บัญชีผู้ใช้งาน	D1 เพิ่มข้อมูล ผู้ใช้	
รายการข้อมูล ผู้ใช้งานที่เป็น สมาชิก	รายการข้อมูล ผู้ใช้งานที่เป็น สมาชิก	Process 3.1 แก้ไขข้อมูล สถานะสิทธิ์ การใช้งาน บัญชีผู้ใช้งาน	ผู้ดูแลระบบ	{คำนำหน้าชื่อ+ชื่อ+ นามสกุล+วันเดือนปีเกิด+ เบอร์ติดต่อ+อีเมล+ที่อยู่+ รูปโปรไฟล์+ธนาคาร+เลขที่ บัญชีธนาคาร+ชื่อเจ้าของ บัญชีธนาคาร+สถานะการ ใช้งาน}
ข้อมูลส่วนตัว ของผู้ดูแลระบบ ที่แก้ไข	ข้อมูลส่วนตัว ของผู้ดูแลระบบ ที่แก้ไขแล้ว	ผู้ดูแลระบบ	Process 3.2 แก้ไขข้อมูล ส่วนตัว	คำนำหน้าชื่อ=[นาย นาง นางสาว]+ชื่อ+นามสกุล+ (วันเดือนปีเกิด)+เบอร์ ติดต่อ+(ที่อยู่=(ถนน)+ (ตำบล)+(อำเภอ)+(จังหวัด)+ (เลขไปรษณีย์))+(รูป โปรไฟล์)+(รหัสธนาคาร)+ (เลขที่บัญชีธนาคาร)+(ชื่อ เจ้าของบัญชีธนาคาร)+ รหัสผ่าน
ข้อมูลส่วนตัว ของผู้ใช้งานที่ เป็นสมาชิกที่ แก้ไข	ข้อมูลส่วนตัว ของผู้ใช้งานที่ เป็นสมาชิกที่ แก้ไขแล้ว	ผู้ใช้งานที่เป็น สมาชิก	Process 3.2 แก้ไขข้อมูล ส่วนตัว	
		Process 3.2 แก้ไขข้อมูล ส่วนตัว	D1 เพิ่มข้อมูล ผู้ใช้	
ข้อมูลส่วนตัว ของผู้ดูแลระบบ	รายละเอียดขอ มูลส่วนตัวของ ผู้ดูแลระบบ	Process 3.2 แก้ไขข้อมูล ส่วนตัว	ผู้ดูแลระบบ D1 เพิ่มข้อมูล ผู้ใช้	คำนำหน้าชื่อ+ชื่อ+นาม สกุล+วันเดือนปีเกิด+เบอร์ ติดต่อ+อีเมล+ที่อยู่+รูปโปร

ตารางที่ 3.2 Data flow (Data flow Description and Data Structure of Data Flow) (ต่อ)

Name	Description	Source	Destination	Data Structure
ข้อมูลส่วนตัว ของผู้ใช้งานที่ เป็นสมาชิก	รายละเอียด ข้อมูลส่วนตัว ของผู้ใช้งานที่ เป็นสมาชิก	Process 3.2 แก้ไขข้อมูล ส่วนตัว	ผู้ใช้งานที่เป็น สมาชิก	ไฟล์+ธนาคาร+เลขที่บัญชี ธนาคาร+ชื่อเจ้าของบัญชี ธนาคาร+ประเภทผู้ใช้งาน
			D1 เพิ่มข้อมูล ผู้ใช้	
ข้อมูลประเภท ของการอบรม	ข้อมูลประเภท ของการอบรม	D2 เพิ่มข้อมูล ประเภทของ การอบรม	Process 4.1 เพิ่มข้อมูลพื้น ฐาน	รหัสประเภทของการ อบรม+ชื่อประเภทของการ อบรม+สถานะการใช้งาน
			Process 4.2 แก้ไขข้อมูล พื้นฐาน	
			Process 5.1 เพิ่มข้อมูลกิจ กรรมอบรม	
			Process 5.2 แก้ไขข้อมูล กิจกรรมอบรม	
ข้อมูลประเภท ของการอบรมที่ เพิ่ม	ข้อมูลประเภท ของการอบรมที่ ต้องการเพิ่ม	ผู้ดูแลระบบ	Process 4.1 เพิ่มข้อมูลพื้น ฐาน	รหัสประเภทของการ อบรม+ชื่อประเภทของการ อบรม+สถานะการใช้งาน
			Process 4.1 เพิ่มข้อมูลพื้น ฐาน	
ข้อมูลประเภท ของการอบรมที่ แก้ไข	ข้อมูลประเภท ของการอบรมที่ แก้ไขแล้ว	ผู้ดูแลระบบ	Process 4.2 แก้ไขข้อมูล พื้นฐาน	ชื่อประเภทของการอบรม+ สถานะการใช้งาน
			Process 4.2 แก้ไขข้อมูล พื้นฐาน	
รายการข้อมูล	รายการข้อมูล	Process 4.1	ผู้ดูแลระบบ	{ชื่อประเภทของการอบรม+

ตารางที่ 3.2 Data flow (Data flow Description and Data Structure of Data Flow) (ต่อ)

Name	Description	Source	Destination	Data Structure
ประเภทของการอบรม	ประเภทของการอบรม	เพิ่มข้อมูลพื้นฐาน		สถานะการใช้งาน}
		Process 4.2 แก้ไขข้อมูลพื้นฐาน		
ข้อมูลธนาคาร	ข้อมูลธนาคาร	D3 เพิ่มข้อมูลธนาคาร	Process 1.0 สมัครสมาชิก	รหัสธนาคาร+ชื่อธนาคาร+สถานะการใช้งาน
			Process 3.2 แก้ไขข้อมูลส่วนตัว	
			Process 4.1 เพิ่มข้อมูลพื้นฐาน	
			Process 4.2 แก้ไขข้อมูลพื้นฐาน	
			Process 5.1 เพิ่มข้อมูลกิจกรรมอบรม	
			Process 5.2 แก้ไขข้อมูลกิจกรรมอบรม	
ข้อมูลธนาคารที่เพิ่ม	ข้อมูลธนาคารที่ต้องการเพิ่ม	ผู้ดูแลระบบ	Process 4.1 เพิ่มข้อมูลพื้นฐาน	รหัสธนาคาร+ชื่อธนาคาร+สถานะการใช้งาน
		Process 4.1 เพิ่มข้อมูลพื้นฐาน	D3 เพิ่มข้อมูลธนาคาร	
ข้อมูลธนาคารที่	ข้อมูลธนาคารที่	ผู้ดูแลระบบ	Process 4.2	ชื่อธนาคาร+สถานะการใช้

ตารางที่ 3.2 Data flow (Data flow Description and Data Structure of Data Flow) (ต่อ)

Name	Description	Source	Destination	Data Structure
แก้ไข	แก้ไขแล้ว		แก้ไขข้อมูลพื้นฐาน	งาน
		Process 4.2 แก้ไขข้อมูลพื้นฐาน	D3 เพิ่มข้อมูลธนาคาร	
รายการข้อมูลธนาคาร	รายการข้อมูลธนาคาร	Process 4.1 เพิ่มข้อมูลพื้นฐาน	ผู้ดูแลระบบ	{ชื่อธนาคาร+สถานะการใช้งาน}
		Process 4.2 แก้ไขข้อมูลพื้นฐาน	ผู้ดูแลระบบ	
ข้อมูลกิจกรรมอบรม	ข้อมูลกิจกรรมอบรม	D8 เพิ่มข้อมูลกิจกรรมอบรม	Process 5.1 เพิ่มข้อมูลกิจกรรมอบรม	รหัสกิจกรรมอบรม+ชื่อกิจกรรมอบรม+รหัสประเภทของการอบรม+รายละเอียดกิจกรรมอบรม+ชื่อ-นามสกุลวิทยากร+วัตถุประสงค์ของกิจกรรมอบรม+ภาพโปรมิตกิจกรรมอบรม+ไฟล์เอกสารกำหนดการในการดำเนินการกิจกรรม+ช่วงวันที่จัดกิจกรรมอบรม+ช่วงวันที่ลงทะเบียน+ระยะเวลาการจัดกิจกรรม+สถานที่จัดกิจกรรมอบรม+ค่าใช้จ่ายในการเข้าอบรม+รหัสธนาคาร+เลขที่บัญชีธนาคาร+ชื่อเจ้าของบัญชีธนาคาร+จำนวนผู้เข้าร่วมกิจกรรมอบรม
			Process 5.2 แก้ไขข้อมูลกิจกรรมอบรม	
			Process 5.3 ค้นหาข้อมูลกิจกรรมอบรม	
			Process 6.1 เพิ่มข้อมูลการลงทะเบียน	
			Process 6.2 เพิ่มข้อมูลการลงทะเบียน	
			Process 6.3 เพิ่มข้อมูลการลงทะเบียน	

ตารางที่ 3.2 Data flow (Data flow Description and Data Structure of Data Flow) (ต่อ)

Name	Description	Source	Destination	Data Structure
			Process 7.0 เพิ่มข้อมูลการ ลงทะเบียน	รม+รหัสผู้ใช้งานที่เป็น ผู้สร้าง+วันที่ และเวลาสร้าง กิจกรรมอบรม+สถานะการ ดำเนินการ
ข้อมูลกิจกรรม อบรมของตน เองที่เพิ่ม	ข้อมูลกิจกรรม อบรมของตน เองที่ต้องการ เพิ่ม	ผู้ใช้งานที่เป็น สมาชิก	Process 5.1 เพิ่มข้อมูลกิจกรรมอบรม	รหัสกิจกรรมอบรม+ชื่อ กิจกรรมอบรม+(รหัส ประเภทของการอบรม)+ (รายละเอียดกิจกรรม อบรม)+(ชื่อ-นามสกุล วิทยากร)+(วัตถุประสงค์ ของกิจกรรมอบรม)+(ภาพ โปรโมตกิจกรรมอบรม)+ (ไฟล์เอกสารกำหนดการใน การดำเนินกิจกรรม)+ช่วง วันที่จัดกิจกรรมอบรม+ช่วง วันที่ลงทะเบียน+ระยะเวลา การจัดกิจกรรม+สถานที่จัด กิจกรรมอบรม+ค่าใช้จ่ายใน การเข้าอบรม+(รหัส ธนาคาร)+(เลขที่ บัญชี ธนาคาร)+(ชื่อเจ้าของบัญชี ธนาคาร)+จำนวนผู้เข้าร่วม กิจกรรมอบรม+รหัส ผู้ใช้งานที่เป็นผู้สร้าง+วันที่ และเวลาสร้างกิจกรรม อบรม+สถานะการดำเนินการ
		Process 5.1 เพิ่มข้อมูลกิจกรรมอบรม	D8 เพิ่มข้อมูล กิจกรรมอบรม	
ข้อมูลกิจกรรม อบรมของ	ข้อมูลกิจกรรม อบรมของตน	ผู้ใช้งานที่เป็น สมาชิก	Process 5.2 แก้ไขข้อมูลกิจกรรม	ชื่อกิจกรรมอบรม+(รหัส ประเภทของการอบรม)+

ตารางที่ 3.2 Data flow (Data flow Description and Data Structure of Data Flow) (ต่อ)

Name	Description	Source	Destination	Data Structure
ตนเองที่แก้ไข	เองที่แก้ไขแล้ว		กรรมอบรม	(รายละเอียดกิจกรรมอบรม)+(ชื่อ-นามสกุลวิทยา
		Process 5.2 แก้ไขข้อมูลกิจกรรมอบรม	D8 เพิ่มข้อมูลกิจกรรมอบรม	กร)+(วัตถุประสงค์ของกิจกรรมอบรม)+(ภาพโปรโมตกิจกรรมอบรม)+(ไฟล์เอกสารกำหนดการในการดำเนินกิจกรรม)+(ช่วงวันที่จัดกิจกรรมอบรม+ช่วงวันที่ลงทะเบียน+ระยะเวลาการจัดกิจกรรม+(สถานที่จัดกิจกรรมอบรม=(ถนน)+(ตำบล)+(อำเภอ)+(จังหวัด)+(เลขไปรษณีย์))+ค่าใช้จ่ายในการเข้าอบรม+(รหัสธนาคาร)+(เลขที่บัญชีธนาคาร)+(ชื่อเจ้าของบัญชีธนาคาร)+จำนวนผู้เข้าร่วมกิจกรรมอบรม+สถานะการดำเนินการชื่อกิจกรรมอบรม+{ประเภทของการอบรม}+รายละเอียดกิจกรรมอบรม+ชื่อ-นามสกุลวิทยากร+วัตถุประสงค์ของกิจกรรมอบรม+ภาพโปรโมตกิจกรรมอบรม+ไฟล์เอกสารกำหนดการในการดำเนินกิจกรรม+(ช่วงวันที่จัดกิจกรรมอบรม+ช่วงวันที่ลงทะเบียน+ระยะเวลาการ
ข้อมูลกิจกรรมอบรมของตนเอง	ข้อมูลกิจกรรมอบรมของตนเอง	Process 5.2 แก้ไขข้อมูลกิจกรรมอบรม	ผู้ใช้งานที่เป็นสมาชิก	

ตารางที่ 3.2 Data flow (Data flow Description and Data Structure of Data Flow) (ต่อ)

Name	Description	Source	Destination	Data Structure
				จัดกิจกรรม+สถานที่จัดกิจกรรมอบรม+ค่าใช้จ่ายในการเข้าอบรม+รหัสธนาคาร+เลขที่บัญชีธนาคาร+ชื่อเจ้าของบัญชีธนาคาร+จำนวนผู้เข้าร่วมกิจกรรมอบรม+สถานะการดำเนินการ
ข้อมูลกิจกรรมอบรมของตนเองที่ต้องการค้นหา	ข้อมูลที่ใช้ในการสืบค้นข้อมูลเกี่ยวกับกิจกรรมอบรมของตนเอง	ผู้ใช้งานที่เป็นสมาชิก	Process 5.3 ค้นหาข้อมูลกิจกรรมอบรม	(ชื่อกิจกรรมอบรม)+(รหัสประเภทของการอบรม)+(วันที่เริ่มต้นและเสร็จสิ้นกิจกรรมอบรม)+(วันที่ลงทะเบียน)+(จังหวัดที่จัดกิจกรรมอบรม)
ข้อมูลกิจกรรมอบรมทั้งหมดที่ต้องการค้นหา	ข้อมูลที่ใช้ในการสืบค้นข้อมูลเกี่ยวกับกิจกรรมอบรมทั้งหมด	ผู้ใช้ทั่วไป	Process 5.3 ค้นหาข้อมูลกิจกรรมอบรม	
		ผู้ใช้งานที่เป็นสมาชิก	Process 5.3 ค้นหาข้อมูลกิจกรรมอบรม	
รายการข้อมูลกิจกรรมอบรมของตนเอง	รายการข้อมูลกิจกรรมอบรมของตนเอง	Process 5.1 เพิ่มข้อมูลกิจกรรมอบรม	ผู้ใช้งานที่เป็นสมาชิก	{ชื่อกิจกรรมอบรม+{ประเภทของการอบรม}+รายละเอียดกิจกรรมอบรม+ชื่อนามสกุลวิทยากร+วัตถุประสงค์ของกิจกรรมอบรม+ภาพโปรมตกิจกรรมอบรม+ไฟล์เอกสารกำหนด
		Process 5.3 ค้นหาข้อมูลกิจกรรมอบรม		
รายการข้อมูลกิจกรรมอบรมทั้งหมด	รายการข้อมูลกิจกรรมอบรมทั้งหมด	Process 5.3 ค้นหาข้อมูลกิจกรรมอบรม	ผู้ใช้งานที่เป็นสมาชิก	การในการดำเนินกิจกรรม+ช่วงวันที่จัดกิจกรรมอบรม+ช่วงวันที่ลงทะเบียน+ระยะ
			ผู้ใช้ทั่วไป	

ตารางที่ 3.2 Data flow (Data flow Description and Data Structure of Data Flow) (ต่อ)

Name	Description	Source	Destination	Data Structure
				เวลาการจัดกิจกรรม+สถานที่จัดกิจกรรม+ค่าใช้จ่ายในการเข้าอบรม+ชื่อธนาคาร+เลขที่บัญชีธนาคาร+ชื่อเจ้าของบัญชีธนาคาร+จำนวนผู้ลงทะเบียน+จำนวนผู้เข้าร่วมกิจกรรมอบรม+วันที่ และเวลาสร้างกิจกรรมอบรม+สถานะการดำเนินการ}
ข้อมูลการลงทะเบียน	ข้อมูลการลงทะเบียน	D9 เพิ่มข้อมูลการลงทะเบียน	Process 5.3 ค้นหาข้อมูลกิจกรรมอบรม	<u>รหัสการลงทะเบียน+รหัสกิจกรรมอบรม+รหัสผู้ใช้ที่ลงทะเบียน+วัน และเวลาที่ลงทะเบียน+ภาพหลักฐานการชำระเงิน+วันที่ และเวลาชำระเงิน+เบอร์ติดต่อผู้เข้าร่วม+สถานะการตรวจสอบ+หมายเหตุ</u>
			Process 6.1 เพิ่มข้อมูลการลงทะเบียน	
			Process 6.2 แก้ไขข้อมูลการลงทะเบียน	
			Process 6.3 ค้นหาข้อมูลการลงทะเบียน	
ข้อมูลการลงทะเบียน	ข้อมูลการลงทะเบียน	ผู้ใช้งานที่เป็นสมาชิก	Process 6.1 เพิ่มข้อมูลการลงทะเบียน	<u>รหัสการลงทะเบียน+รหัสกิจกรรมอบรม+รหัสผู้ใช้ที่ลงทะเบียน+วัน และเวลาที่ลงทะเบียน+ภาพหลักฐาน</u>

ตารางที่ 3.2 Data flow (Data flow Description and Data Structure of Data Flow) (ต่อ)

Name	Description	Source	Destination	Data Structure
กิจกรรมอบรมที่เพิ่ม	กิจกรรมอบรมที่ต้องการเพิ่ม	Process 6.1 เพิ่มข้อมูลการลงทะเบียน	D9 เพิ่มข้อมูล การลงทะเบียน	การชำระเงิน+วันที่ และ เวลาชำระเงิน+เบอร์ติดต่อ ผู้เข้าร่วม+สถานะการ
ข้อมูลการลงทะเบียนกิจกรรมอบรมที่แก้ไข	ข้อมูลการลงทะเบียนกิจกรรมอบรมที่แก้ไขแล้ว	Process 6.2 ผู้ใช้งานที่เป็นสมาชิก	Process 6.2 แก้ไขข้อมูล การลงทะเบียน	ตรวจสอบ+หมายเหตุ+ เบอร์ติดต่อผู้เข้าร่วม+ สถานะการตรวจสอบ
		Process 6.2 แก้ไขข้อมูล การลงทะเบียน	D9 เพิ่มข้อมูล การลงทะเบียน	
รายการข้อมูลการลงทะเบียนของตนเอง	รายการข้อมูลการลงทะเบียนของตนเอง	Process 6.1 เพิ่มข้อมูลการลงทะเบียน	ผู้ใช้งานที่เป็นสมาชิก	{ชื่อกิจกรรมอบรม+ชื่อ- นามสกุลผู้ใช้ที่ลงทะเบียน+ วัน และเวลาที่ลงทะเบียน+ ภาพหลักฐานการชำระเงิน+ วันที่ และเวลาชำระเงิน+ เบอร์ติดต่อผู้เข้าร่วม+ สถานะการตรวจสอบ+ หมายเหตุ}
		Process 6.2 แก้ไขข้อมูล การลงทะเบียน		
ข้อมูลสถานะการตรวจสอบการลงทะเบียนของผู้เข้าอบรมที่แก้ไข	ข้อมูลสถานะการตรวจสอบการลงทะเบียนของผู้เข้าอบรมที่แก้ไขแล้ว	Process 6.2 ผู้ใช้งานที่เป็นสมาชิก	Process 6.2 แก้ไขข้อมูล การลงทะเบียน	สถานการณ์ตรวจสอบ =[ตรวจสอบผ่าน ตรวจสอบ ไม่ผ่าน]+(หมายเหตุ)
		Process 6.2 แก้ไขข้อมูล การลงทะเบียน	D9 เพิ่มข้อมูล การลงทะเบียน	
รายการข้อมูลผู้ลงทะเบียนใน	รายการข้อมูลผู้ลงทะเบียนใน	Process 6.2 แก้ไขข้อมูล	ผู้ใช้งานที่เป็นสมาชิก	{ชื่อ-นามสกุลผู้ใช้ที่ลงทะเบียน+วัน และเวลาที่ลงทะเบียน

ตารางที่ 3.2 Data flow (Data flow Description and Data Structure of Data Flow) (ต่อ)

Name	Description	Source	Destination	Data Structure
กิจกรรมอบรม	กิจกรรมอบรม	ก า ร ล ง ท ะ เป็ ย น		ทะเบียน+ภาพหลักฐานการ ชำระเงิน+วันที่ และเวลา ชำระเงิน+เบอร์ติดต่อ+ สถานะการตรวจสอบ+ หมายเหตุ}
ข้อมูลผู้ลงทะเบียนในกิจกรรมอบรมของตนเองที่ต้องการค้นหา	ข้อมูลที่ใช้ในการสืบค้นข้อมูลเกี่ยวกับผู้ลงทะเบียนในกิจกรรมอบรมของตนเอง	ผู้ใช้งานที่เป็นสมาชิก	Process 6.3 ค้นหาข้อมูล ก า ร ล ง ท ะ เป็ ย น	(ชื่อกิจกรรมอบรม)+(รหัสประเภทของการอบรม)+(สถานะการตรวจสอบ)+(ชื่อผู้ลงทะเบียน)+(วันที่ชำระ)
รายการข้อมูลผู้ลงทะเบียนในกิจกรรมอบรมของตนเอง	รายการข้อมูลผู้ลงทะเบียนในกิจกรรมอบรมของตนเอง	Process 6.3 ค้นหาข้อมูล ก า ร ล ง ท ะ เป็ ย น	ผู้ใช้งานที่เป็นสมาชิก	{ชื่อ-นามสกุลผู้ใช้ที่ลงทะเบียน+วัน และเวลาที่ลงทะเบียน+ภาพหลักฐานการชำระเงิน+วันที่ และเวลาชำระเงิน+สถานะการตรวจสอบ+หมายเหตุ}
ข้อมูลประวัติการเข้าร่วมกิจกรรมอบรมของตนเองที่ต้องการค้นหา	ข้อมูลที่ใช้ในการสืบค้นข้อมูลเกี่ยวกับประวัติการเข้าร่วมกิจกรรมอบรมของตนเอง	ผู้ใช้งานที่เป็นสมาชิก	Process 6.3 ค้นหาข้อมูล ก า ร ล ง ท ะ เป็ ย น	(ชื่อกิจกรรมอบรม)+(รหัสประเภทของการอบรม)
รายการข้อมูลประวัติการเข้าร่วมกิจกรรมอบรมของตนเอง	รายการข้อมูลประวัติการเข้าร่วมกิจกรรมอบรมของตนเอง	Process 6.3 ค้นหาข้อมูล ก า ร ล ง ท ะ เป็ ย น	ผู้ใช้งานที่เป็นสมาชิก	{ชื่อกิจกรรมอบรม+{ประเภทของการอบรม)+รายละเอียดกิจกรรมอบรม+ชื่อ-นามสกุลวิทยากร+วัตถุประสงค์ของกิจกรรมอบรม+ภาพโปรโมตกิจกรรม

ตารางที่ 3.2 Data flow (Data flow Description and Data Structure of Data Flow) (ต่อ)

Name	Description	Source	Destination	Data Structure
				อบรม+ไฟล์เอกสารกำหนด การในการดำเนินกิจกรรม+ ช่วงวันที่จัดกิจกรรมอบรม+ ระยะเวลาการจัดกิจกรรม+ สถานที่จัดกิจกรรมอบรม+ ค่าใช้จ่ายในการเข้าอบรม+ จำนวนผู้เข้าร่วมกิจกรรม อบรม + สถานที่ การ ดำเนินการ+วัน และเวลาที่ ลงทะเบียน+ภาพหลักฐาน การชำระเงิน+วันที่ และ เวลาชำระเงิน+สถานะการ ตรวจสอบ+หมายเหตุ+ชื่อ- นามสกุลผู้ใช้งานที่เป็น ผู้สร้าง+เบอร์ผู้ใช้งานที่เป็น ผู้สร้าง+อีเมลผู้ใช้งานที่เป็น ผู้สร้าง}

3.3.3 Data Store (Data flow Description and Data Structure of Data Store)

Data Store คือ การนำข้อมูลเข้าไปจัดการในฐานข้อมูล โดยมีการแยกออกเป็นแฟ้มข้อมูล
 ดังต่อไปนี้

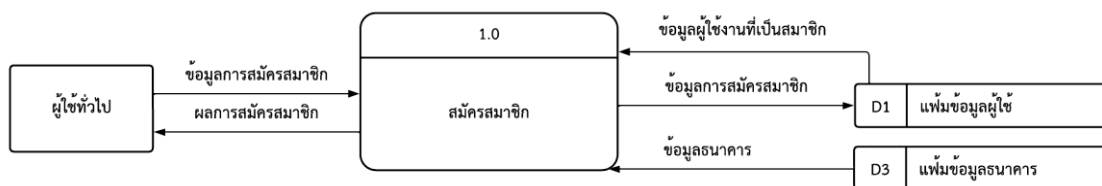
ตารางที่ 3.3 Data Store (Data flow Description and Data Structure of Data Store)

ID	Data Store	Description	Data Structure
D1	แฟ้มข้อมูลผู้ใช้	เก็บข้อมูลผู้ใช้งาน ในระบบ	รหัสสมาชิก+คำนำหน้าชื่อ+ชื่อ+นามสกุล+ วันเดือนปีเกิด+เบอร์ติดต่อ+อีเมล+ที่อยู่+ รูปโปรไฟล์+รหัสธนาคาร+เลขที่บัญชี ธนาคาร+ชื่อเจ้าของบัญชีธนาคาร+ รหัสผ่าน+ประเภทผู้ใช้งาน+สถานะการใช้ งาน

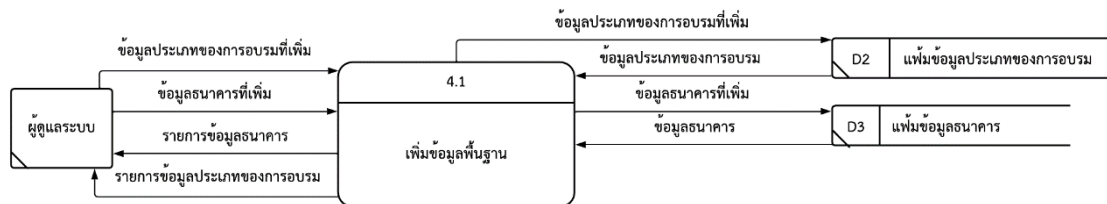
ตารางที่ 3.3 Data Store (Data flow Description and Data Structure of Data Store) (ต่อ)

ID	Data Store	Description	Data Structure
D2	เพิ่มข้อมูลประเภทของการอบรม	เก็บข้อมูลประเภทของการอบรม	รหัสประเภทของการอบรม+ชื่อประเภทของการอบรม+สถานะการใช้งาน
D3	เพิ่มข้อมูลธนาคาร	เก็บข้อมูลธนาคาร	รหัสธนาคาร+ชื่อธนาคาร+สถานะการใช้งาน
D4	เพิ่มข้อมูลกิจกรรมอบรม	เก็บข้อมูลกิจกรรมอบรม	รหัสกิจกรรมอบรม+ชื่อกิจกรรมอบรม+รหัสประเภทของการอบรม+รายละเอียดกิจกรรมอบรม+ชื่อ-นามสกุลวิทยากร+วัตถุประสงค์ของกิจกรรมอบรม+ภาพโปรโมตกิจกรรมอบรม+ไฟล์เอกสารกำหนดการในการดำเนินกิจกรรม+ช่วงวันที่จัดกิจกรรม+ช่วงวันที่ลงทะเบียน+วันที่ และเวลาที่สร้างกิจกรรมอบรม+ระยะเวลาการจัดกิจกรรม+สถานที่จัดกิจกรรมอบรม + ค่าใช้จ่ายในการเข้าอบรม+รหัสธนาคาร+เลขที่บัญชีธนาคาร+ชื่อเจ้าของบัญชีธนาคาร+จำนวนผู้เข้าร่วมกิจกรรมอบรม+รหัสผู้ใช้งานที่เป็นผู้สร้าง+สถานะการดำเนินการ
D5	เพิ่มข้อมูลการลงทะเบียน	เก็บข้อมูลการลงทะเบียน	รหัสกิจกรรมอบรม+รหัสผู้ใช้ที่ลงทะเบียน+รหัสการลงทะเบียน+วัน และเวลาที่ลงทะเบียน+เบอร์ติดต่อผู้เข้าร่วม+ภาพหลักฐานการชำระเงิน+วันที่ และเวลาชำระเงิน+สถานะการตรวจสอบ+หมายเหตุ

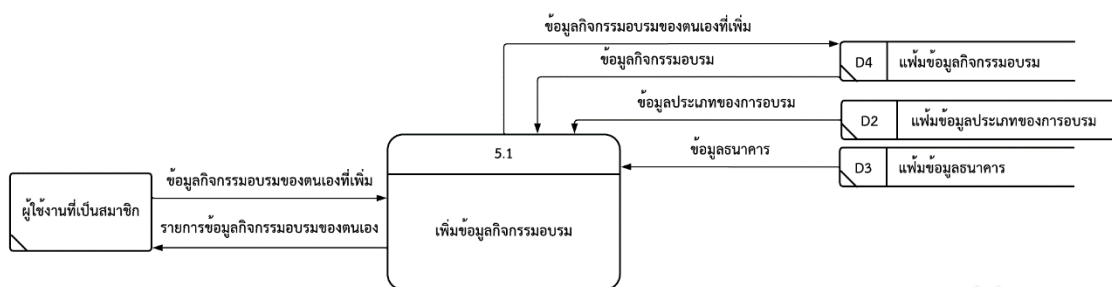
3.3.4 คำอธิบายการประมวลผล (Process Description and Specification)



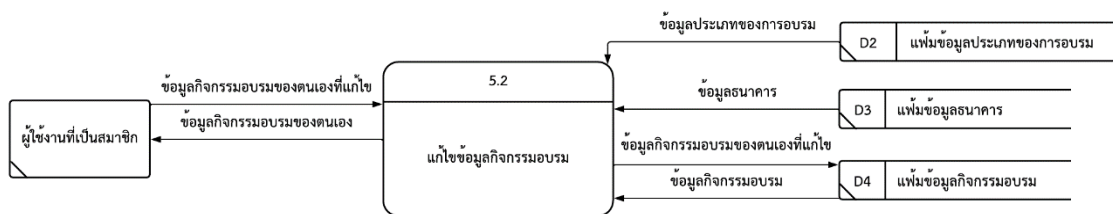
Number	1.0
Name	สมัครสมาชิก
Description	การสมัครสมาชิก
Input Data Flow	<ul style="list-style-type: none"> - ข้อมูลการสมัครสมาชิก - ข้อมูลธนาคาร - ข้อมูลผู้ใช้งานที่เป็นสมาชิก
Output Data Flow	<ul style="list-style-type: none"> - ข้อมูลการสมัครสมาชิก - ผลการสมัครสมาชิก
Process Logic	<p>เริ่ม</p> <ol style="list-style-type: none"> 1. รับข้อมูลการสมัครสมาชิก 2. ตรวจสอบข้อมูลการสมัครสมาชิก <ol style="list-style-type: none"> 2.1 ถ้าข้อมูลถูกต้องและครบถ้วน บันทึกข้อมูลลงในฐานข้อมูล 2.2 ถ้าข้อมูลไม่ถูกต้องหรือไม่ครบถ้วน <p style="text-align: center;">ทำการแจ้งเตือนว่า “ข้อมูลไม่ถูกต้องหรือไม่ครบถ้วน” แล้ว</p> <p>กลับไปข้อ 2.</p> <p>จบการทำงาน</p>



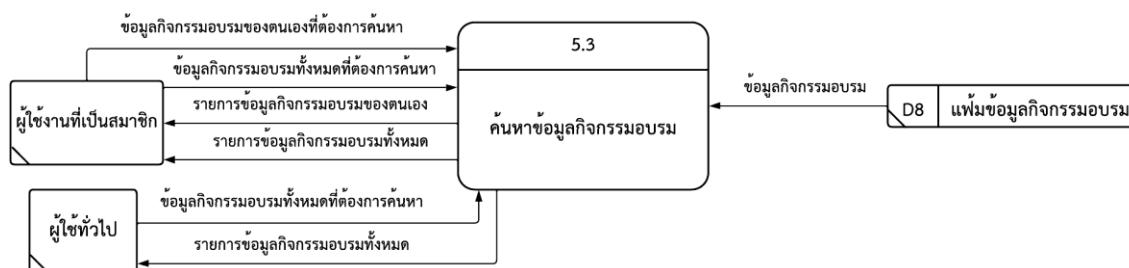
Number	4.1
Name	เพิ่มข้อมูลพื้นฐาน
Description	การเพิ่มข้อมูลพื้นฐาน เช่น ข้อมูลประเภทของการอบรม ข้อมูลธนาคาร และข้อมูลหัวข้อการประเมินอบรม
Input Data Flow	<ul style="list-style-type: none"> - ข้อมูลประเภทของการอบรมที่เพิ่ม - ข้อมูลธนาคารที่เพิ่ม - ข้อมูลประเภทของการอบรม - ข้อมูลธนาคาร
Output Data Flow	<ul style="list-style-type: none"> - ข้อมูลประเภทของการอบรมที่เพิ่ม - ข้อมูลธนาคารที่เพิ่ม - รายการข้อมูลธนาคาร - รายการข้อมูลประเภทของการอบรม
Process Logic	<p>เริ่ม</p> <ol style="list-style-type: none"> 1. รับข้อมูลพื้นฐานที่ต้องการเพิ่ม 2. ตรวจสอบข้อมูล <ol style="list-style-type: none"> 2.1 ถ้าข้อมูลถูกต้องและครบถ้วน ไปข้อ 3. 2.2 ถ้าข้อมูลไม่ถูกต้องหรือไม่ครบถ้วน <p>แสดงข้อความ “ข้อมูลไม่ถูกต้องหรือข้อมูลไม่ครบถ้วน” ไปข้อ 1.</p> 3. ตรวจสอบข้อมูลในเพิ่มข้อมูลพื้นฐาน <ol style="list-style-type: none"> 3.1 ถ้ามีข้อมูลพื้นฐานแล้ว แสดงข้อความ “มีข้อมูลในระบบแล้ว” ไปข้อ 1. 3.2 ถ้าไม่มีข้อมูลพื้นฐานในเพิ่มข้อมูล ให้บันทึกข้อมูลพื้นฐานลงในเพิ่มพื้นฐาน <p>จบการทำงาน</p>



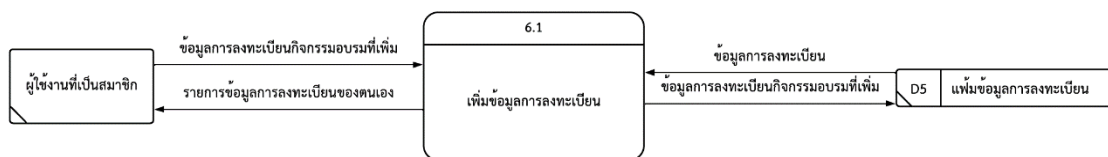
Number	5.1
Name	เพิ่มข้อมูลกิจกรรมอบรม
Description	การเพิ่มข้อมูลกิจกรรมอบรมของตนเอง
Input Data Flow	<ul style="list-style-type: none"> - ข้อมูลกิจกรรมอบรมของตนเองที่เพิ่ม - ข้อมูลกิจกรรมอบรม - ข้อมูลประเภทการอบรม - ข้อมูลธนาคาร
Output Data Flow	<ul style="list-style-type: none"> - ข้อมูลกิจกรรมอบรมของตนเองที่เพิ่ม - รายการข้อมูลกิจกรรมอบรมของตนเอง
Process Logic	<p>เริ่ม</p> <ol style="list-style-type: none"> 1. รับข้อมูล กิจกรรมอบรมของตนเอง ที่ต้องการเพิ่ม 2. ตรวจสอบข้อมูล <ol style="list-style-type: none"> 2.1 ถ้าข้อมูลถูกต้องและครบถ้วน บันทึกข้อมูลลงในฐานข้อมูล 2.2 ถ้าข้อมูลไม่ถูกต้องหรือไม่ครบถ้วน แสดงข้อความ “ข้อมูลไม่ถูกต้องหรือข้อมูลไม่ครบถ้วน” ไปข้อ 1. <p>จบการทำงาน</p>



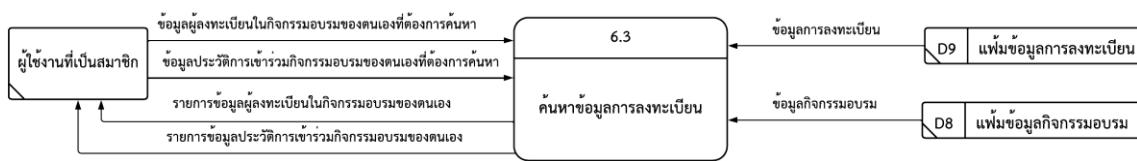
Number	5.2
Name	แก้ไขข้อมูลกิจกรรมอบรม
Description	การแก้ไขข้อมูลกิจกรรมอบรมของตนเอง
Input Data Flow	<ul style="list-style-type: none"> - ข้อมูลกิจกรรมอบรมของตนเองที่แก้ไข - ข้อมูลกิจกรรมอบรม - ข้อมูลประเภทของการอบรม - ข้อมูลธนาคาร
Output Data Flow	<ul style="list-style-type: none"> - ข้อมูลกิจกรรมอบรมของตนเองที่แก้ไข - ข้อมูลกิจกรรมอบรมของตนเอง
Process Logic	<p>เริ่ม</p> <ol style="list-style-type: none"> 1. แสดงข้อมูลกิจกรรมอบรมของตนเองที่ต้องการแก้ไข 2. รับข้อมูลกิจกรรมอบรมที่ต้องการแก้ไข 3. ตรวจสอบข้อมูล <ol style="list-style-type: none"> 3.1 ถ้าข้อมูลถูกต้องและครบถ้วน อัปเดตข้อมูลลงในฐานข้อมูล 3.2 ถ้าข้อมูลไม่ถูกต้องหรือไม่ครบถ้วน ทำการแจ้งเตือนว่า “ข้อมูลไม่ถูกต้องหรือไม่ครบถ้วน” แล้วกลับไปข้อ 2. <p>จบการทำงาน</p>



Number	5.3
Name	ค้นหาข้อมูลกิจกรรมอบรม
Description	การค้นหาข้อมูลกิจกรรมอบรม
Input Data Flow	<ul style="list-style-type: none"> - ข้อมูลกิจกรรมอบรมของตนเองที่ต้องการค้นหา - ข้อมูลกิจกรรมอบรมทั้งหมดที่ต้องการค้นหา - ข้อมูลกิจกรรมอบรม
Output Data Flow	<ul style="list-style-type: none"> - รายการข้อมูลกิจกรรมอบรมของตนเอง - รายการข้อมูลกิจกรรมอบรมทั้งหมด
Process Logic	<p>เริ่ม</p> <ol style="list-style-type: none"> 1. รับ ชื่อกิจกรรมอบรม(ถ้ามี) และ ประเภทของการอบรม(ถ้ามี) และ ช่วงวันที่จัดกิจกรรมอบรม(ถ้ามี) และ วันที่ลงทะเบียน(ถ้ามี) และ จังหวัดที่จัดกิจกรรมอบรม(ถ้ามี) ที่ต้องการค้นหา 2. ค้นหาข้อมูลกิจกรรมอบรมเพิ่มข้อมูลกิจกรรมอบรม ตามข้อมูลที่ได้รับเข้ามา 3. ตรวจสอบ <ol style="list-style-type: none"> 3.1 ถ้ามีข้อมูล แสดงรายการข้อมูลกิจกรรมอบรม ซึ่งประกอบด้วย ชื่อกิจกรรมอบรม ภาพโปรโมตกิจกรรมอบรม ชื่อ-นามสกุลวิทยากร ค่าใช้จ่ายในการเข้าอบรม จำนวนผู้เข้าร่วมกิจกรรมอบรม 3.2 ถ้าไม่มีข้อมูล แสดงข้อความ “ไม่พบรายการข้อมูลกิจกรรมอบรมที่ต้องการค้นหา” <p>จบการทำงาน</p>

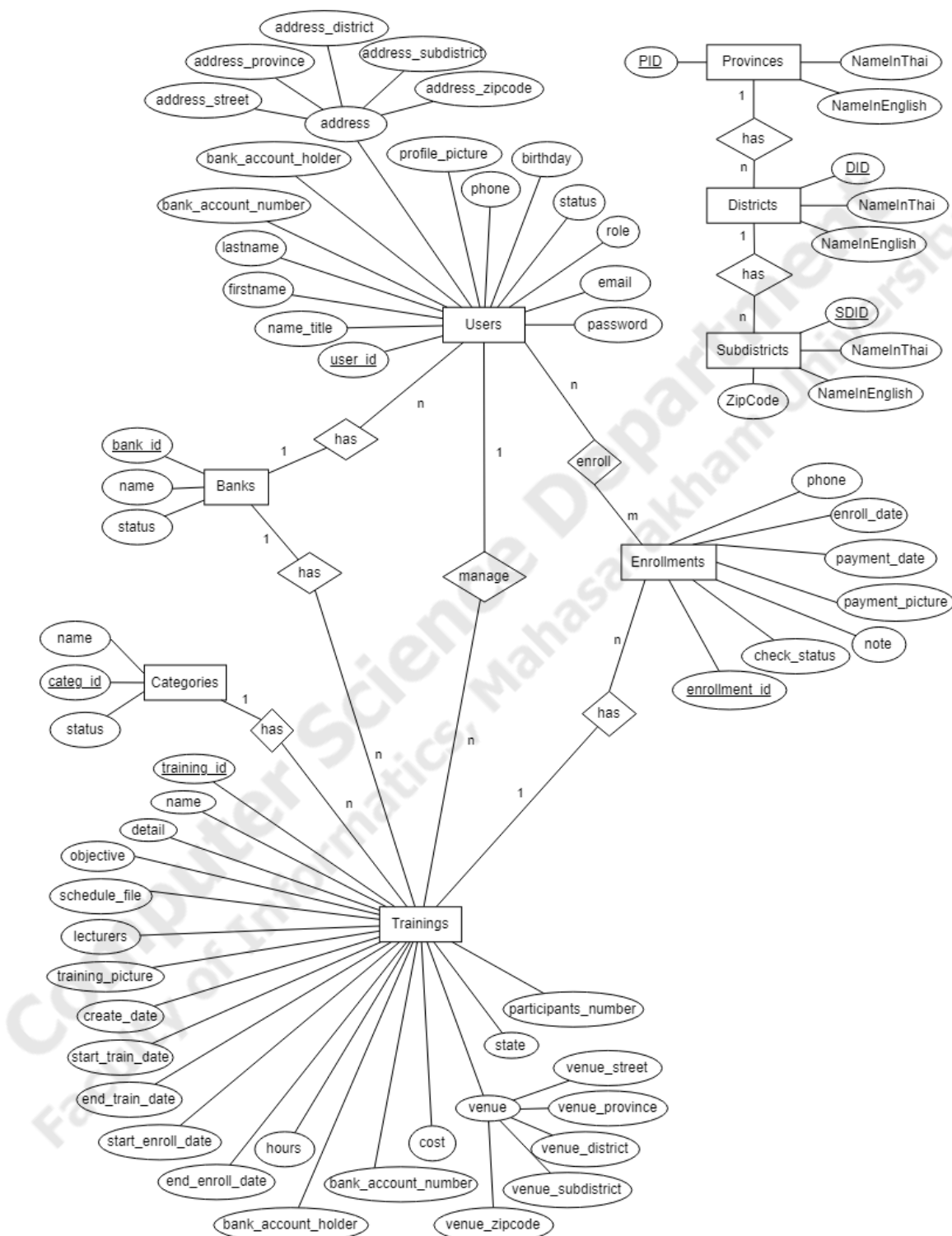


Number	6.1
Name	เพิ่มข้อมูลการลงทะเบียน
Description	การเพิ่มข้อมูลการลงทะเบียน
Input Data Flow	- ข้อมูลการลงทะเบียนกิจกรรมอบรมที่เพิ่ม - ข้อมูลการลงทะเบียน
Output Data Flow	- ข้อมูลการลงทะเบียนกิจกรรมอบรมที่เพิ่ม - รายการข้อมูลการลงทะเบียนของตนเอง
Process Logic	เริ่ม <ol style="list-style-type: none"> 1. รับข้อมูลการลงทะเบียน ในกิจกรรมอบรมที่ต้องการเพิ่ม 2. ตรวจสอบข้อมูล <ol style="list-style-type: none"> 2.1 ถ้าข้อมูลถูกต้องและครบถ้วน บันทึกข้อมูลลงในฐานข้อมูล 2.2 ถ้าข้อมูลไม่ถูกต้องหรือไม่ครบถ้วน แสดงข้อความ “ข้อมูลไม่ถูกต้องหรือข้อมูลไม่ครบถ้วน” ไปข้อ 1. จบการทำงาน



Number	6.3
Name	ค้นหาข้อมูลการลงทะเบียน
Description	การค้นหาข้อมูลการลงทะเบียน
Input Data Flow	<ul style="list-style-type: none"> - ข้อมูลผู้ลงทะเบียนในกิจกรรมอบรมของตนเองที่ต้องการค้นหา - ข้อมูลประวัติการเข้าร่วมกิจกรรมอบรมของตนเองที่ต้องการค้นหา - ข้อมูลการลงทะเบียน - ข้อมูลกิจกรรมอบรม
Output Data Flow	<ul style="list-style-type: none"> - รายการข้อมูลผู้ลงทะเบียนในกิจกรรมอบรมของตนเอง - รายการข้อมูลประวัติการเข้าร่วมกิจกรรมอบรมของตนเอง
Process Logic	<p>เริ่ม</p> <ol style="list-style-type: none"> 1. รับ ชื่อกิจกรรมอบรม(ถ้ามี) และ ประเภทของการอบรม(ถ้ามี) และ สถานะการตรวจสอบ(ถ้ามี) และ ชื่อผู้ลงทะเบียน(ถ้ามี) และ วันที่ชำระค่าลงทะเบียน(ถ้ามี) ที่ต้องการค้นหา 2. ค้นหาข้อมูลการลงทะเบียน จากเพิ่มข้อมูลการลงทะเบียน ตามข้อมูลที่รับเข้ามา 3. ตรวจสอบ <ol style="list-style-type: none"> 3.1 ถ้ามีข้อมูล แสดงรายการข้อมูลการลงทะเบียน ซึ่งประกอบด้วย <ul style="list-style-type: none"> ชื่อผู้ลงทะเบียน เบอร์ติดต่อ ภาพหลักฐานการชำระค่าอบรม และสถานะการตรวจสอบ 3.2 ถ้าไม่มีข้อมูล แสดงข้อความ “ไม่พบรายการข้อมูลการลงทะเบียนที่ต้องการค้นหา” <p>จบการทำงาน</p>

3.4 แผนภาพ Entity Relationship Diagram (ER - Diagram)



ภาพประกอบที่ 3.4 แผนภาพ Entity Relationship Diagram

3.5 การออกแบบฐานข้อมูล (Database Design)

ตารางที่ 3.4 ตารางผู้ใช้ (Users)

ID	Column	Type	Description	Example Data	Constraint
1	user_id	varchar(450)	รหัสสมาชิก ผู้ใช้งาน	WTq8zYcZfaW VvMncigHqwQ	PK
2	name_title	int(4)	ค่านำหน้าชื่อ	0 = นาย	
3	firstname	varchar(256)	ชื่อจริง	อารียาพร	
4	lastname	varchar(256)	นามสกุล	ศรเกษตริน	
5	profile_picture	varchar(256)	รูปโปรไฟล์	Profile2.png	
6	phone	varchar(100)	เบอร์ติดต่อ	0855411789	
7	birthday	date	วันเดือนปีเกิด	1998-04-05	
8	address_street	varchar(256)	รายละเอียด ของ ที่อยู่	45/103 หมู่ 8 ถ. นาเนียน	
9	address_province	varchar(256)	จังหวัดของที่ อยู่	สุราษฎร์ธานี	
10	address_district	varchar(256)	อำเภอของที่ อยู่	เมืองสุราษฎร์ธานี	
11	address_subdistrict	varchar(256)	ตำบลของที่ อยู่	มะขามเตี้ย	
12	address_zipcode	varchar(256)	เลขไปรษณีย์ ของที่อยู่	84000	
13	email	varchar(450)	อีเมล	mdmd@gmail.c om	Not null
14	password	varchar(450)	รหัสผ่าน	Tdlddn3GTsd 4Ah6bTV4es	Not null
15	role	int(4)	ป ร ะ เ ก ท ผู้ใช้งาน	0 = Admin, 1 = Member	
16	user_bank_id	int(4)	รหัสธนาคาร	1	Foreign key from Banks
17	bank_account_num	varchar(256)	เลขบัญชี	1235647890	

ตารางที่ 3.4 ตารางผู้ใช้ (Users) (ต่อ)

	ber		ธนาคาร		
18	bank_account_holder	varchar(450)	ชื่อเจ้าของบัญชี	นางน้ำค้าง ทนวยั่งยืน	
19	status	int(4)	สถานะการใช้งาน	0 = ใช้งาน , 1 = ไม่ใช้งาน	

ตารางที่ 3.5 ตารางประเภทของการอบรม (Categories)

ID	Column	Type	Description	Example Data	Constraint
1	categ_id	int(4)	รหัสประเภทของการอบรม	1	PK
2	name	varchar(256)	ชื่อประเภทของการอบรม	การเงินและการบัญชี	
3	status	int(4)	สถานะการใช้งาน	0 = ใช้งาน , 1 = ไม่ใช้งาน	

ตารางที่ 3.6 ตารางธนาคาร (Banks)

ID	Column	Type	Description	Example Data	Constraint
1	bank_id	int(4)	รหัสธนาคาร	1	PK
2	name	varchar(256)	ชื่อธนาคาร	กสิกร	
3	status	int(4)	สถานะการใช้งาน	0 = ใช้งาน , 1 = ไม่ใช้งาน	

ตารางที่ 3.7 ตารางจังหวัด (Provinces)

ID	Column	Type	Description	Example Data	Constraint
1	PID	int(4)	รหัสจังหวัด	1	PK
2	NameInThai	varchar(450)	ชื่อจังหวัดภาษาไทย	กรุงเทพมหานคร	
3	NameInEnglish	varchar(256)	ชื่อจังหวัดภาษาอังกฤษ	Bangkok	

ตารางที่ 3.8 ตารางอำเภอ (Districts)

ID	Column	Type	Description	Example Data	Constraint
1	DID	int(4)	รหัสอำเภอ	1	PK
2	NameInThai	varchar(450)	ชื่ออำเภอภาษาไทย	เขต บางรัก	

ตารางที่ 3.8 ตารางอำเภอ (Districts) (ต่อ)

ID	Column	Type	Description	Example Data	Constraint
3	NameInEnglish	varchar(256)	ชื่ออำเภอภาษาอังกฤษ	Bang Rak	
4	ProvinceId	int(4)	รหัสจังหวัด	1	Foreign key from Provinces

ตารางที่ 3.9 ตารางตำบล (Subdistricts)

ID	Column	Type	Description	Example Data	Constraint
1	SDID	int(4)	รหัสตำบล	1	PK
2	NameInThai	varchar(150)	ชื่อตำบลภาษาไทย	วัดบูรพาภิรมย์	
3	NameInEnglish	varchar(150)	ชื่อตำบลภาษาอังกฤษ	Wang Burapha Phirom	
4	DistrictId	int(4)	รหัสอำเภอ	1	Foreign key from Districts
5	ZipCode	int(4)	รหัสไปรษณีย์	10200	

ตารางที่ 3.10 ตารางกิจกรรมอบรม (Trainings)

ID	Column	Type	Description	Example Data	Constraint
1	training_id	varchar(450)	รหัสกิจกรรมอบรม	FgBnx6eu7DUwfgzStzW QPA	PK
2	name	varchar(256)	ชื่อกิจกรรมอบรม	การบริหารลูกหนี้และทวงหนี้ที่ได้ผล	
3	detail	varchar(500)	รายละเอียด	วางแผนการบริหารหนี้ขององค์กรคุณ	
4	objective	varchar(500)	วัตถุประสงค์	เพื่อให้ผู้เข้าอบรมเข้าใจแนวทางการวิเคราะห์ลูกหนี้และลดความเสี่ยง	
5	training_pi	varchar(256)	รูปโปสเตอร์	picture.png	

ตารางที่ 3.10 ตารางกิจกรรมอบรม (Trainings) (ต่อ)

ID	Column	Type	Description	Example Data	Constraint
	cture				
6	schedule_file	varchar(256)	ไฟล์เอกสารกำหนดการ	Schedule.pdf	
7	start_train_date	date	วันที่เริ่มอบรม	2020-01-06	
8	end_train_date	date	วันที่สิ้นสุดการอบรม	2020-01-09	
9	hours	int(4)	ระยะเวลาการจัดกิจกรรม (ชั่วโมง)	19	
10	start_enroll_date	dateTime	วันที่และเวลาเริ่มการลงทะเบียน	2020-15-05T12:30	
11	end_enroll_date	dateTime	วันที่และเวลาสิ้นสุดการลงทะเบียน	2020-3 0 - 05T23:59	
12	venue_street	varchar(256)	รายละเอียดของสถานที่จัดกิจกรรมอบรม	360/2 หมู่ 8 ถนนบ้านเมือง	
13	venue_province	varchar(256)	จังหวัดของสถานที่จัดกิจกรรมอบรม	นครพนม	
14	venue_district	varchar(256)	อำเภอของสถานที่จัดกิจกรรมอบรม	เมืองนครพนม	
15	venue_subdistrict	varchar(256)	ตำบลของสถานที่จัดกิจกรรมอบรม	หนองแสง	
16	venue_zipcode	varchar(256)	เลขไปรษณีย์ของสถานที่จัดกิจกรรมอบรม	48000	
17	cost	Double	ค่าใช้จ่าย	2500.00	

ตารางที่ 3.10 ตารางกิจกรรมอบรม (Trainings) (ต่อ)

ID	Column	Type	Description	Example Data	Constraint
18	participants_number	int(4)	จำนวนผู้เข้าร่วม	200	
19	training_bank_id	varchar(450)	รหัสธนาคาร	1	Foreign key from Banks
20	bank_account_number	varchar(256)	เลขที่บัญชีธนาคาร	445541418414	
21	create_date	dateTime	วันที่ และเวลาที่สร้างกิจกรรมอบรม	2020-06-01T23:59	
22	lecturers	varchar(900)	ชื่อ-นามสกุลวิทยากร	ศาสตราจารย์ ปรีชา งาม, ศาสตราจารย์ ปิติ ภูมิ โชตวานิช	
23	bank_account_holder	varchar(256)	ชื่อ-นามสกุลเจ้าของบัญชีธนาคาร	ภาวดี แสงตะวัน	
24	user_owner_id	varchar(256)	รหัสผู้ใช้ที่เป็นผู้สร้างอบรม	WTq8zYcZfaW VvMncigHqwQ	Foreign key from Users
25	state	int(4)	สถานการณ์ดำเนินการ	1 = โปสต์แล้ว, 2 = สำเร็จแล้ว	

ตารางที่ 3.11 ตารางการลงทะเบียน (Enrollments)

ID	Column	Type	Description	Example Data	Constraint
1	enrollment_id	varchar(450)	รหัสการลงทะเบียน	9UGIPZwcpm2 U2wK9xpPAsR	PK
2	enrollment_training_id	varchar(450)	รหัสกิจกรรมอบรม	FgBnx6eu7DUw fgzStzWQPA	Foreign key from Trainings

ตารางที่ 3.11 ตารางการลงทะเบียน (Enrollments) (ต่อ)

ID	Column	Type	Description	Example Data	Constraint
3	enrollment_user_id	varchar(256)	รหัสผู้ใช้ที่ลงทะเบียน	SZ35DV4sWp6 DyGtHXmQ35S	Foreign key from Users
4	phone	varchar(100)	เบอร์ติดต่อ	0888451645	
5	enroll_date	datetime	วัน และเวลาที่ลงทะเบียน	2020-18-05T18:18	
6	payment_picture	varchar(256)	ภาพหลักฐานการชำระเงิน	Payment.png	
7	payment_date	datetime	วันที่ และเวลาชำระเงิน	2020-18-05T18:20	
8	check_status	varchar(256)	สถานะการตรวจสอบ	0 = ยังไม่ตรวจสอบ, 1 = ตรวจสอบผ่าน, 2 = ตรวจสอบไม่ผ่าน, 3 = ยกเลิกการลงทะเบียน	
9	note	varchar(256)	หมายเหตุ	หลักฐานการชำระเงินไม่ถูกต้อง (จำนวนเงิน วันที่ และเวลาโอนไม่ถูกต้อง)	

3.6 การพัฒนาระบบ


ระบบการจัดการอบรมสัมมนา (System of Training Management) พัฒนาด้วย Framework ASP.NET CORE MVC (Model-View-Controller) โดยโครงสร้างการทำงานหลัก ๆ ดังนี้ Model ใช้ในการเก็บข้อมูล เป็นส่วนที่ติดต่อกับฐานข้อมูล เพื่อทำการดึงและบันทึกข้อมูล View เป็นส่วนที่ใช้ในการแสดงผลให้กับผู้ใช้งาน โดยจะแสดงข้อมูลจากโมเดลไปยังหน้าเว็บที่ใช้ HTML, CSS และ JavaScript สำหรับการแสดงผล และ Controller ที่เป็นส่วนควบคุมกระบวนการทำงานของ View กับ Model

3.6.1 สมัครงานสมาชิก

1. ตัวอย่างหน้าสมัครงานสมาชิก

สมัครงานสมาชิก

***โปรดกรอกข้อมูล**



คำนำหน้าชื่อ
--เลือกคำนำหน้าชื่อ--

ชื่อจริง *

นามสกุล *

เบอร์ติดต่อ *

อัปโหลดภาพโปรไฟล์
Choose File No file chosen

ธนาคาร
-- เลือกธนาคาร --

อีเมล *

name@example.com

เลขที่บัญชีธนาคาร

รหัสผ่าน *

รหัสผ่าน

ชื่อเจ้าของบัญชี

กรอกรหัสผ่านอีกครั้ง *

รหัสผ่าน

[เคยสมัครงานสมาชิกแล้ว ?](#)

สมัครงานสมาชิก

เข้าสู่ระบบ

ภาพประกอบที่ 3.5 ตัวอย่างหน้าสมัครงานสมาชิก

2. โมเดลที่ใช้เก็บข้อมูลสำหรับสมัครงานสมาชิก

```

84 public class InputModel
85 {
86     [Display(Name = "คำนำหน้าชื่อ")]
87     public NameTitle? Name_title { get; set; }
88     [Required(ErrorMessage = "โปรดกรอกข้อมูล")]
89     [Display(Name = "ชื่อจริง")]
90     public string FirstName { get; set; }
91     [Required(ErrorMessage = "โปรดกรอกข้อมูล")]
92     [Display(Name = "นามสกุล")]
93     public string LastName { get; set; }
94
95     [RegularExpression("[0-9]+$",
96         ErrorMessage = "เบอร์ติดต่อต้องเป็นตัวเลข")]
97     [Required(ErrorMessage = "โปรดกรอกข้อมูล")]
98     [Display(Name = "เบอร์ติดต่อ")]
99     public string Phone { get; set; }
100    [Display(Name = "ธนาคาร")]
101    public int? BankId { get; set; }
102    [RegularExpression("[0-9]+$",
103        ErrorMessage = "เลขบัญชีต้องเป็นตัวเลข")]
104    [Display(Name = "เลขบัญชี")]

```

ภาพประกอบที่ 3.6 โมเดลที่ใช้เก็บข้อมูลสำหรับสมัครงานสมาชิก

```

105     public string? Bank_account_number { get; set; }
106     [Display(Name = "ชื่อเจ้าของบัญชี")]
107     public string? Bank_account_holder { get; set; }
108     [Display(Name = "อัปโหลดภาพโปรไฟล์ ")]
109     public IFormFile? ImageFile { get; set; }
110     [Required(ErrorMessage = "โปรดกรอกข้อมูล")]
111     [EmailAddress(ErrorMessage = "รูปแบบของอีเมลไม่ถูกต้อง (ตัวอย่าง: name@example.com)")]
112     [UniqueEmail(ErrorMessage = "อีเมลนี้ผู้ใช้แล้ว โปรดกรอกอีเมลใหม่")]
113     [Display(Name = "อีเมล")]
114     public string Email { get; set; }
115     [Required(ErrorMessage = "โปรดกรอกข้อมูล")]
116     [StringLength(100, ErrorMessage = "{0} ความยาวน้อย {2} ตัว และไม่เกิน {1} ตัว", MinimumLength = 8)]
117     [DataType(DataType.Password)]
118     [Display(Name = "รหัสผ่าน")]
119     public string Password { get; set; }
120     [DataType(DataType.Password)]
121     [Display(Name = "กรอกรหัสผ่านอีกครั้ง")]
122     [Compare("Password", ErrorMessage = "รหัสผ่านไม่ตรงกัน")]
123     public string ConfirmPassword { get; set; }
124 }

```

ภาพประกอบที่ 3.6 โมเดลที่ใช้เก็บข้อมูลสำหรับสมัครสมาชิก (ต่อ)

จากภาพประกอบที่ 3.6 คลาส InputModel เป็นโมเดลที่ใช้เก็บข้อมูลสำหรับสมัครสมาชิก โดยมีเงื่อนไขต่าง ๆ เพื่อตรวจสอบความถูกต้องของข้อมูล เช่น บรรทัดที่ 88 กำหนดให้ต้องกรอกข้อมูลชื่อจริง หากไม่กรอกจะมีข้อความเตือน “โปรดกรอกข้อมูล” และใช้ [Display(Name = "ชื่อ")] เพื่อกำหนดชื่อฟิลด์ที่จะแสดงในหน้าจอให้ผู้ใช้งาน

3. ฟังก์ชัน OnPostAsync (สมัครสมาชิก) ในคลาส RegisterModel

```

136     public async Task<IActionResult> OnPostAsync(string returnUrl = null)
137     {
138         ViewData["Banks"] = new SelectList(_context.Banks
139             .Where(c => c.Status == Status.ใช้งาน).ToList()
140             .OrderBy(b => b.Name), "Id", "Name");
141         returnUrl ??= Url.Content("~/");
142         ExternalLogins = (await _signInManager.GetExternalAuthenticationSchemesAsync()).ToList();
143         if (ModelState.IsValid)
144         {
145             var user = CreateUser();
146
147             string filename = Uploadfile(Input);
148             user.Name_title = Input.Name_title;
149             user.FirstName = Input.FirstName;
150             user.LastName = Input.LastName;
151             user.PhoneNumber = Input.Phone;
152             user.BankId = Input.BankId;
153             user.Bank_account_number = Input.Bank_account_number;
154             user.Bank_account_holder = Input.Bank_account_holder;
155             user.Profile_picture = filename;
156
157             await _userStore.SetUserNameAsync(user, Input.Email, CancellationToken.None);
158             await _emailStore.SetEmailAsync(user, Input.Email, CancellationToken.None);
159             var result = await _userManager.CreateAsync(user, Input.Password);

```

ภาพประกอบที่ 3.7 ฟังก์ชัน OnPostAsync (สมัครสมาชิก) ในคลาส RegisterModel

```

161 |         if (result.Succeeded)
162 |         {
163 |             _logger.LogInformation("User created a new account with password.");
164 |
165 |             var userId = await _userManager.GetUserAsync(user);
166 |             var code = await _userManager.GenerateEmailConfirmationTokenAsync(user);
167 |             code = WebEncoders.Base64UrlEncode(Encoding.UTF8.GetBytes(code));
168 |             var callbackUrl = Url.Page(
169 |                 "/Account/ConfirmEmail",
170 |                 pageHandler: null,
171 |                 values: new { area = "Identity", userId = userId, code = code, returnUrl = returnUrl },
172 |                 protocol: Request.Scheme);
173 |
174 |             await _emailSender.SendEmailAsync(Input.Email, "Confirm your email",
175 |                 $"Please confirm your account by <a href='{HtmlEncoder.Default.Encode(callbackUrl)}'>clicking here</a>.");
176 |
177 |             if (_userManager.Options.SignIn.RequireConfirmedAccount)
178 |             {
179 |                 return RedirectToPage("RegisterConfirmation", new { email = Input.Email, returnUrl = returnUrl });
180 |             }
181 |             else
182 |             {
183 |                 await _signInManager.SignInAsync(user, isPersistent: false);
184 |                 return LocalRedirect(returnUrl);
185 |             }
186 |         }
187 |         foreach (var error in result.Errors)
188 |         {
189 |             ModelState.AddModelError(string.Empty, error.Description);
190 |         }
191 |     }
192 |
193 |     // If we got this far, something failed, redisplay form
194 |     return Page();
195 | }

```

ภาพประกอบที่ 3.7 ฟังก์ชัน OnPostAsync (สมัครสมาชิก) ในคลาส RegisterModel (ต่อ)

เมื่อผู้ใช้กรอกข้อมูลการสมัครสมาชิกแล้ว ข้อมูลที่ผู้ใช้กรอกจะถูกส่งแบบ post จาก view มาสู่ controller

บรรทัดที่ 143-191 เงื่อนไขตรวจสอบว่าข้อมูลที่ผู้ใช้กรอกเข้ามาในฟอร์มผ่านการตรวจสอบความถูกต้องตามเงื่อนไขที่กำหนดไว้ในคลาส InputModel หรือไม่

บรรทัดที่ 145 สร้างผู้ใช้ใหม่

บรรทัดที่ 147 กำหนดชื่อรูปภาพโดยใช้ฟังก์ชัน Uploadfile โดยฟังก์ชันจะทำการอัปโหลดไฟล์และส่งค่ากลับมาเป็นชื่อไฟล์

บรรทัดที่ 148-155 กำหนดค่าผู้ใช้งานตามข้อมูลที่กรอกเข้ามา

บรรทัดที่ 157 กำหนดชื่อผู้ใช้

บรรทัดที่ 158 กำหนดอีเมล

บรรทัดที่ 159 กำหนดรหัสผ่าน

บรรทัดที่ 161-186 เงื่อนไขถ้าสมัครสมาชิกสำเร็จ

บรรทัดที่ 183-184 ทำการเข้าสู่ระบบด้วยข้อมูลที่ผู้ใช้งานสมัครสมาชิกเข้ามา

บรรทัดที่ 194 ถ้าสมัครสมาชิกไม่สำเร็จระบบจะนำผู้ใช้งานกลับไปสู่หน้าสมัครสมาชิก พร้อมข้อความแสดงข้อผิดพลาดในการกรอกข้อมูล เช่น โปรตกรอกข้อมูล (ในช่องที่มี * สีแดงบนช่องกรอกข้อมูลที่เป็นข้อมูลที่จำเป็นต้องกรอก) เบอร์ติดต่อต้องเป็นตัวเลข เลขบัญชีต้องเป็นตัวเลข รูปแบบของอีเมลไม่ถูกต้อง (ตัวอย่าง: name@example.com) อีเมลนี้มีผู้ใช้แล้ว โปรตกรอกอีเมลใหม่ หรือรหัสผ่านไม่ตรงกัน เป็นต้น

4. ฟังก์ชัน Uploadfile

```

219 public string Uploadfile(InputModel input)
220 {
221     string filename = null;
222     string filePath = null;
223     if (input.ImageFile != null)
224     {
225         string uploadDir = Path.Combine(_hostEnvironment.WebRootPath, "uploads/profile");
226         filename = Guid.NewGuid().ToString() + "-" + input.ImageFile.FileName;
227         filePath = Path.Combine(uploadDir, filename);
228         using (var fileStream = new FileStream(filePath, FileMode.Create))
229         {
230             input.ImageFile.CopyTo(fileStream);
231         }
232     }
233     return filename;

```

ภาพประกอบที่ 3.8 ฟังก์ชัน Uploadfile

จากภาพประกอบที่ 3.8 คือ ฟังก์ชัน Uploadfile ที่จะใช้อัพโหลดไฟล์รูปภาพที่ผู้ใช้งานเพิ่มเข้ามาตอนสมัครสมาชิก โดยมีพารามิเตอร์เป็นอินพุตโมเดล คือ ข้อมูลที่ผู้ใช้สมัครสมาชิกเข้ามาบรรทัดที่ 223-232 เงื่อนไขถ้าข้อมูลที่ผู้ใช้สมัครเข้ามามีรูปภาพ

3.6.2 เข้าสู่ระบบ

1. ตัวอย่างหน้าเข้าสู่ระบบ

เข้าสู่ระบบ

ยังไม่เคยสมัครสมาชิก ? [สมัครสมาชิก](#)

ภาพประกอบที่ 3.9 ตัวอย่างหน้าเข้าสู่ระบบ

2. โมเดลที่ใช้เก็บข้อมูลสำหรับเข้าสู่ระบบ

```

44 | 1 reference
45 | public class InputModel
46 | {
47 |     [Required(ErrorMessage = "โปรดกรอกชื่อผู้ใช้งาน")]
48 |     [EmailAddress(ErrorMessage = "รูปแบบของอีเมลไม่ถูกต้อง (ตัวอย่าง: name@example.com)")]
49 |     public string Email { get; set; }
50 |     [Required(ErrorMessage = "โปรดกรอกชื่อผู้ใช้งาน")]
51 |     [DataType(DataType.Password)]
52 |     public string Password { get; set; }
53 |     [Display(Name = "Remember me?")]
54 |     public bool RememberMe { get; set; }

```

ภาพประกอบที่ 3.10 โมเดลที่ใช้เก็บข้อมูลสำหรับเข้าสู่ระบบ

จากภาพประกอบที่ 3.10 คลาส InputModel เป็นโมเดลที่ใช้เก็บข้อมูลสำหรับเข้าสู่ระบบ

3. ฟังก์ชัน OnPostAsync (สมัครสมาชิก) ในคลาส LoginModel

```

68 | 0 references
69 | public async Task<IActionResult> OnPostAsync(string returnUrl = null)
70 | {
71 |     returnUrl ??= Url.Content("~/");
72 |     ExternalLogins = (await _signInManager.GetExternalAuthenticationSchemesAsync()).ToList();
73 |     if (ModelState.IsValid)
74 |     {
75 |         var result = await _signInManager.PasswordSignInAsync(Input.Email, Input.Password, Input.RememberMe, lockoutOnFailure: false);
76 |         if (result.Succeeded)
77 |         {
78 |             _logger.LogInformation("User logged in.");
79 |             return LocalRedirect(returnUrl);
80 |         }
81 |         if (result.IsNotAllowed)
82 |         {
83 |             ModelState.AddModelError(string.Empty, "ผู้ใช้ไม่สามารถเข้าใช้งานได้");
84 |             return Page();
85 |         }
86 |         if (result.RequiresTwoFactor)
87 |         {
88 |             return RedirectToPage("./LoginWith2fa", new { ReturnUrl = returnUrl, RememberMe = Input.RememberMe });
89 |         }
90 |         if (result.IsLockedOut)
91 |         {
92 |             _logger.LogWarning("User account locked out.");
93 |             return RedirectToPage("./Lockout");
94 |         }
95 |         else
96 |         {
97 |             ModelState.AddModelError(string.Empty, "อีเมล หรือรหัสผ่านไม่ถูกต้อง โปรดกรอกใหม่อีกครั้ง");
98 |             return Page();
99 |         }
100 |     }
101 |     // If we got this far, something failed, redisplay form
102 |     return Page();

```

ภาพประกอบที่ 3.11 ฟังก์ชัน OnPostAsync (สมัครสมาชิก) ในคลาส LoginModel

บรรทัดที่ 75-98 เงื่อนไขตรวจสอบว่าข้อมูลที่ผู้ใช้กรอกเข้ามาเพื่อเข้าสู่ระบบผ่านการตรวจสอบความถูกต้องตามเงื่อนไขที่กำหนดไว้หรือไม่

บรรทัดที่ 74 ฟังก์ชันเข้าสู่ระบบ โดยส่งอีเมลและรหัสผ่าน

บรรทัดที่ 75-79 เงื่อนไขถ้าเข้าสู่ระบบสำเร็จ จะส่งผู้ใช้งานไปที่หน้าแรก

บรรทัดที่ 80-83 เงื่อนไขสำหรับผู้ที่ใช้ที่เข้าสู่ระบบไม่สำเร็จ เพราะถูกยกเลิกสถานะการใช้งาน จะส่งกลับไปหน้าเข้าสู่ระบบ และมีข้อความเตือนว่า “ผู้ใช้ไม่สามารถเข้าใช้งานได้”

บรรทัดที่ 94-98 เงื่อนไขสำหรับผู้ที่ใช้ที่เข้าสู่ระบบไม่สำเร็จ เพราะอีเมลหรือรหัสผ่านไม่ถูกต้อง จะส่งกลับไปหน้าเข้าสู่ระบบ และมีข้อความเตือนว่า “อีเมล หรือรหัสผ่านไม่ถูกต้อง โปรดกรอกใหม่อีกครั้ง”

3.6.3 สร้างกิจกรรมอบรม

```

318 public async Task<ActionResult> Create(Training training)
319 {
320     string picture_name = null;
321     string file_name = null;
322     if (training.ImageFile != null)
323         picture_name = Uploadfile(training.ImageFile, "img");
324     if (training.ScheduleFile != null)
325         file_name = Uploadfile(training.ScheduleFile, "file");
326     string[] date1Parts = training.Flatpickr1.Split(new string[] { " " }, StringSplitOptions.RemoveEmptyEntries);
327     string[] date2Parts = training.Flatpickr2.Split(new string[] { " " }, StringSplitOptions.RemoveEmptyEntries);
328     if ((DateTime.TryParse(date1Parts[0], out DateTime d1) && DateTime.TryParse(date1Parts[1], out DateTime d2))
329     && (DateTime.TryParse(date2Parts[0], out DateTime d3) && DateTime.TryParse(date2Parts[1], out DateTime d4)))
330     {
331         Training t = new Training()
332         {
333             Name = training.Name,
334             Description = training.Description,
335             Lecturer = training.Lecturer,
336             Objectives = training.Objectives,
337             Training_picture = picture_name,
338             Schedule_file = file_name,
339             State = State_Train.อบรม,
340             CategId = training.CategId,
341             Created_Date = DateTime.Now,
342             Start_enroll_Date = d1,
343             End_enroll_Date = d2,
344             Hours = training.Hours,
345             Start_train_Date = d3,
346             End_train_Date = d4,
347             Cost = training.Cost,
348             BankId = training.BankId,
349             Bank_account_number = training.Bank_account_number,
350             Bank_account_holder = training.Bank_account_holder,
351             NumberOfParticipants = training.NumberOfParticipants,
352             Venue_street = training.Venue_street,
353             Venue_subdistrict = training.Venue_subdistrict,
354             Venue_district = training.Venue_district,
355             Venue_province = training.Venue_province,
356             Venue_zipcode = training.Venue_zipcode,
357             OwnerId = training.OwnerId
358         };
359         _unitOfWork.Training.Insert(t);
360         _unitOfWork.Save();
361         return RedirectToAction(nameof(MyTraining));
362     }
363     return View(training);

```

ภาพประกอบที่ 3.12 ฟังก์ชันการสร้างกิจกรรมอบรม

บรรทัดที่ 322 เงื่อนไขถ้าผู้ใช้งานเพิ่มรูปภาพกิจกรรม

บรรทัดที่ 323 ทำการอัปโหลดไฟล์รูปภาพที่ไฟล์เดอร์ img และกำหนดค่าให้กับ picture_name เพื่อเก็บชื่อรูปภาพ

บรรทัดที่ 324 เงื่อนไขถ้าผู้ใช้งานเพิ่มไฟล์เอกสารที่เกี่ยวข้องกับกิจกรรม

บรรทัดที่ 325 ทำการอัปโหลดไฟล์ที่ไฟล์เดอร์ file และกำหนดค่าให้กับ file_name เพื่อเก็บชื่อไฟล์

บรรทัดที่ 326-327 ทำการแยกช่วงวันที่ที่ผู้ใช้งานเพิ่มเข้ามา โดย date1Parts จะเก็บวันที่เริ่มต้นและวันที่สิ้นสุดการลงทะเบียน และ date2Parts เก็บวันที่เริ่มและสิ้นสุดการจัดกิจกรรม

บรรทัดที่ 328-362 เงื่อนไขถ้าทำการแปลงวันที่จากชนิดข้อมูล String เป็น DateTime สำเร็จ โดย d1 คือ วันที่เริ่มการลงทะเบียน d2 คือ วันที่สิ้นสุดการลงทะเบียน d3 คือ วันที่เริ่มการจัดกิจกรรมอบรม และ d4 คือ วันที่สิ้นสุดการจัดกิจกรรมอบรม

บรรทัดที่ 331-358 เป็นการสร้างตัวแปร t เก็บข้อมูลกิจกรรมอบรม

บรรทัดที่ 359-360 บันทึกข้อมูลเข้าดาต้าเบส

บรรทัดที่ 361 ให้กลับไปหน้ากิจกรรมของฉัน

บรรทัดที่ 363 หากบันทึกไม่สำเร็จจะกลับไปหน้าเดิม

3.6.4 ฟังก์ชันบันทึกการลงทะเบียนกิจกรรม

ลงทะเบียน

ชื่อกิจกรรมอบรมที่สมัคร

Basic Flower Design

โพลดีโดย ค่าสมัคร ระยะเวลา เลขที่บัญชีธนาคาร ชื่อเจ้าของบัญชีธนาคาร

Miss A Kasetrin 500 พร้อมเพย์ นางสาวนัสรีน กุโตะ 0909854623

ชื่อ-นามสกุลผู้สมัคร *

นางสาว ศิพิน หมอนข้าง

เบอร์ติดต่อ *


0949876532

วันที่ชำระเงิน *

กรุณาเลือกวันที่ลงทะเบียน

ภาพหลักฐานการชำระ: *

Choose File No file chosen



[ยืนยัน](#)

ภาพประกอบที่ 3.13 ตัวอย่างหน้าลงทะเบียนเข้าร่วมกิจกรรม

```

421 public async Task<IActionResult> Enroll(
422     Enrollment enrollment,
423     int? Cost, string? enrollDate
424 )
425 {
426     string picture_name = null;
427     if (enrollment.PaymentImageFile != null)
428         picture_name = Uploadfile(enrollment.PaymentImageFile, "payments");
429
430     Enrollment e = new Enrollment()
431     {
432         TrainingId = enrollment.TrainingId,
433         ParticipantId = enrollment.ParticipantId,
434         Payment_picture = picture_name,
435         Phone = enrollment.Phone,
436         Enroll_date = DateTime.Now,
437     };
438
439     if (!String.IsNullOrEmpty(enrollDate))
440     {
441         if (DateTime.TryParse(enrollDate, out DateTime enroll))
442         {
443             e.Payment_date = enroll;
444         }
445     }
446
447     if (Cost == 0)
448         e.Check_status = Status_Enroll.ตรวจสอบผ่าน;
449
450     _unitofWork.Enrollment.Insert(e);
451     _unitofWork.Save();
452     return RedirectToAction("MyEnrollmentsList", "Enrollment");

```

ภาพประกอบที่ 3.14 ฟังก์ชันบันทึกการลงทะเบียนกิจกรรม

บรรทัดที่ 427-428 เงื่อนไขถ้าผู้ใช้ได้เพิ่มรูปภาพหลักฐานการชำระค่าอบรมเข้ามา แล้วจึงทำการอัปโหลดไฟล์รูปภาพ และกำหนดค่าให้กับ picture_name เพื่อเก็บชื่อรูปภาพ

บรรทัดที่ 430-437 สร้างตัวแปร e เพื่อเก็บข้อมูลการลงทะเบียน

บรรทัดที่ 439- 445 เงื่อนไขถ้ามีการเพิ่มวันที่ชำระค่าอบรมเข้ามา

บรรทัดที่ 441-444 เงื่อนไขถ้าทำการแปลงวันที่ชำระค่าอบรม จากชนิดข้อมูล String เป็น DateTime สำเร็จ แล้วจึงกำหนดให้ e.Payment_date เก็บวันที่ชำระค่าอบรม

บรรทัดที่ 447-448 ถ้ากิจกรรมอบรมไม่มีค่าใช้จ่าย จะทำการตรวจสอบสถานะผ่านเลย

บรรทัดที่ 450-451 บันทึกข้อมูลเข้าดาต้าเบส

บรรทัดที่ 452 ส่งผู้ใช้ไปที่หน้า การลงทะเบียนของฉัน