

ออกแบบฐานข้อมูลสำหรับร้านเช่าแผ่นหนังภาพยนตร์

จัดทำโดย

นายธนภัทร จำเริญ	6510503417
นายบุรินทร์ จารีไฟบูลย์	6510503484
นางสาวพิชญา เอี่ยมกล้าหาญ	6510503620
นายภัทรพล ณ เนตร	6510503654
นายสมพล อันหลำ	6510503841

เสนอ

รศ.ดร. อินธิราภรณ์ มูลศาสตร์

รายงานฉบับนี้เป็นส่วนหนึ่งของรายวิชาระบบฐานข้อมูล (Database

Systems) (01204351)

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์

มหาวิทยาลัยเกษตรศาสตร์

ปีการศึกษา 2566 ภาคปลาย

สารบัญ

1. บทนำ	3
1.1 ความเป็นมา	3
1.2 วัตถุประสงค์	3
1.3 ขอบเขตการศึกษา	3
1.4 ผลที่คาดว่าจะได้รับ	4
2. ศึกษาความต้องการจัดเก็บข้อมูล	5
2.1 รายละเอียดการเก็บข้อมูลแบบเก่า	5
2.2 รายละเอียดการเก็บข้อมูลแบบใหม่ที่คณะกรรมการออกแบบ	6
3. การออกแบบฐานข้อมูล	8
3.1 Entity-Relation Diagram (ER Diagram)	8
3.2 คำอธิบาย Entities	8
4. Normalization	9
4.1 Normalization Table ต่างๆ	9
1. Table Renting	9
2. Table Customer	10
3. Table Contact	11
4. Table Employee	12
5. Table Duty	13
6. Table Category	13
7. Table Movie	14
8. Table Director	15
4.2 Relational Diagram	16
4.3 Data Dictionary	16
Renting ตารางเก็บข้อมูลการเช่าหนัง	16
Renting_Movie ตารางเก็บข้อมูลรายชื่อหนังที่ถูกเช่าในแต่ละครั้ง	17
Movie ตารางเก็บข้อมูลหนัง	17
Customer ตารางเก็บข้อมูลลูกค้า	18
Contact ตารางเก็บข้อมูลการติดต่อ	19
Employee ตารางเก็บข้อมูลพนักงาน	19
Duty ตารางเก็บข้อมูลหน้าที่ในร้าน	20
Employee_duty ตารางเก็บข้อมูลหน้าที่ของพนักงานแต่ละคน	20
Catagory ตารางเก็บข้อมูลประเภทหนัง	21
Director ตารางเก็บข้อมูลผู้กำกับ	21
TamBon ตารางเก็บข้อมูลตำบล	21
Postal ตารางเก็บข้อมูลรหัสไปรษณีย์	22
Amphoe ตารางเก็บข้อมูลอำเภอ	22
Province ตารางเก็บข้อมูลจังหวัด	23
5. Query Design & Development	24
5.1 คำสั่งการสร้างตารางและข้อมูลในแต่ละตาราง	24
1) Renting	24

2) Renting_Movie	25
3) Movie	25
4) Category	26
5) Director	27
6) Customer	28
7) Employee	28
8) Employee_duty	29
9) Duty	29
10) Contact	30
11) Tambon	30
12) Postal	31
13) Amphoe	31
14) Province	32
5.2 Queries คำสั่งสอบถามข้อมูล	33
1) ค้นหาจำนวนลูกค้าในแต่ละจังหวัด	33
2) ดูข้อมูลพนักงานชื่อจริง นามสกุล หน้าที่และรายได้ของพนักงานร้าน	33
3) หารายชื่อนักแสดง Animation ที่ถูกเข้าโดยนาย Nathan Aroon	34
4) แสดงข้อมูลที่อยู่ของนาย Jattis Jaroen	34
5) แสดงรายชื่อนักแสดงร่วมรันที่ถูกเข้าในปี 2022	35
6) หนังที่สร้างในยุค 90s พร้อมแสดงรันที่ฉายและคะแนน IMDB โดยต้องมากกว่าหรือเท่ากับ 8 และมี Sompon Zeroone เป็นผู้กำกับ	35
7) หารายชื่อนักแสดง Family หรือ Comedy	36
8) หาผลรวมรายได้ที่หนังประเภท Sci-Fi, Animation, Action ทำได้	36
9) หาข้อมูลผู้กำกับหนังและชื่อนักที่กำกับของหนังหมวดหมู่ Games and Horror	37
10) แสดงข้อมูลที่อยู่ของพนักงานที่อยู่ในจังหวัดปทุมธานีหรือเชียงใหม่	37
6. การบริหารจัดการโครงการ	38
6.1 การวางแผนโครงการ	38
6.2 หน้าที่และความรับผิดชอบของแต่ละสมาชิก	38

1. บทนำ

1.1 ความเป็นมา

ในปัจจุบัน กิจการพาณิตรส่วนใหญ่ในไทยไม่รุ่งเรืองมากนัก เนื่องจากมีการลักลอบขายหนังเถื่อนจำนวนมากในโลกอินเทอร์เน็ต ไม่ว่าจะเป็นในเว็บหนัง เถื่อน ในเฟสบุ๊คหรือแม้แต่ในยูทูป ทำให้ผู้กำกับและนักแสดงหนังต่างๆ อาจไม่ดีเท่าที่ควร อย่างมาก นอกจากนี้การมาของแอปพลิเคชันดูหนังต่างๆ อาจไม่ดีเท่าที่ควร เนื่องจากมีราคาแพงและต้องจ่ายเงินรายเดือน ดังนั้นร้านเช่าหนังพาณิตรจึง เป็นอีกทางเลือกสำหรับคนที่ต้องการเพียงดูหนังเฉพาะเจาะจงและไม่เสียเงินมาก นัก ถึงแม้ว่าในประเทศไทยนั้นกิจกรรมร้านเช่าพาณิตรส่วนใหญ่นั้นจะหายไป จากหน้าสังคมไทยแล้ว แต่การมีการจัดการข้อมูลการยืมพาณิตรที่ดีขึ้น อาจ ทำให้กิจกรรมร้านเช่าหนังพาณิตรไทยกลับมาอีกรอบนึง

ทางคณะผู้จัดทำจึงขอเสนอโครงการออกแบบฐานข้อมูลสำหรับร้านเช่า แผ่นหนังพาณิตร เพื่อสนับสนุนกิจกรรมพาณิตรไทย โดยเป็นประโยชน์ต่อผู้ ใช้งานประกอบด้วยเจ้าของกิจการ พนักงานและลูกค้าของร้าน นอกจากนี้ยังเป็น ประโยชน์ต่อองค์กรกับสร้างพาณิตรและคนที่ต้องการดูหนังถูกกฎหมายของไทย

1.2 วัตถุประสงค์

- 1) เพื่อออกรูปแบบและสร้างฐานข้อมูลสำหรับร้านเช่าหนังพาณิตร
- 2) เพื่อช่วยให้กิจกรรมร้านเช่าหนังพาณิตร มีฐานข้อมูลที่ดีขึ้น
- 3) เพื่อช่วยให้กองกำกับสร้างพาณิตรได้รายได้เพิ่มขึ้นจากการที่ผู้คนส่วน ใหญ่หันมาเช่าหนังแทนการดูหนังผิดกฎหมาย
- 4) เพื่อให้ผู้จัดทำได้ทดลองใช้ความรู้ในการเรียนวิชาฐานข้อมูลมาปฏิบัติใช้ จริงในชีวิต

1.3 ขอบเขตการศึกษา

พนักงานร้านเช่าหนัง

1. สามารถจัดเก็บข้อมูลหนังได้อย่างเป็นระเบียบ
2. สามารถเช็คข้อมูลการยืมหนังของลูกค้าได้ง่าย

ลูกค้า

1. สามารถค้นหาหนังได้เร็วขึ้น
2. สามารถรับหนังผ่านระบบไปรษณีย์ได้
3. สามารถเช่าหนังดูที่ร้านได้
4. สามารถค้นหาหนังที่เฉพาะเจาะจงมากขึ้น เช่น ค้นหาด้วยชื่อ

ผู้กำกับหนัง

กองกำกับสร้างภาพยนตร์

1. สามารถตรวจสอบข้อมูลหนังที่ลูกค้าซื้อชื่อ
2. ลดการดูหนังผิดลิขสิทธิ์ได้มีระบบเช่าหนังที่ดี

1.4 ผลที่คาดว่าจะได้รับ

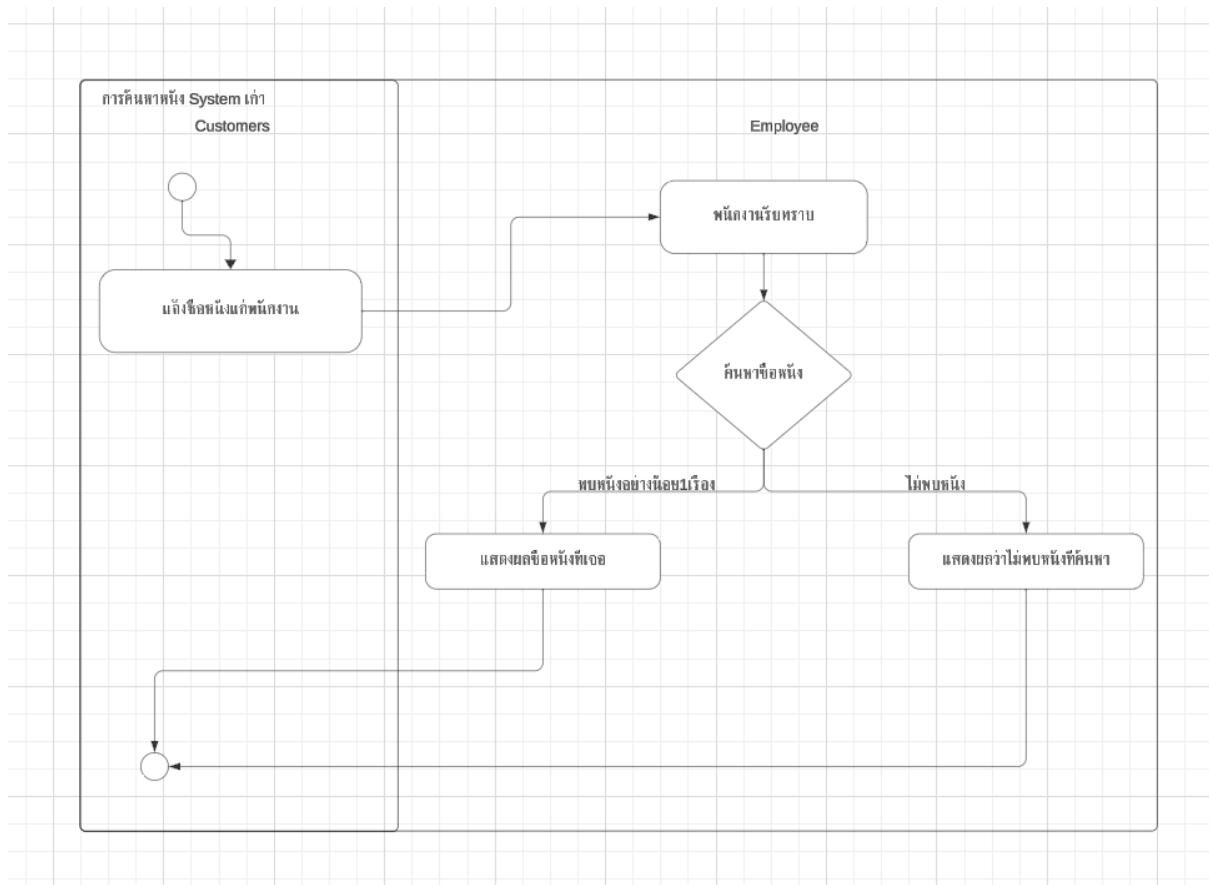
- 1) มีระบบฐานข้อมูลที่เก็บข้อมูลของร้านเช่าหนังภาพยนตร์ที่ดีขึ้น
- 2) ผู้จัดทำได้รับความรู้เกี่ยวกับการพัฒนาฐานข้อมูล

2. ศึกษาความต้องการจัดเก็บข้อมูล

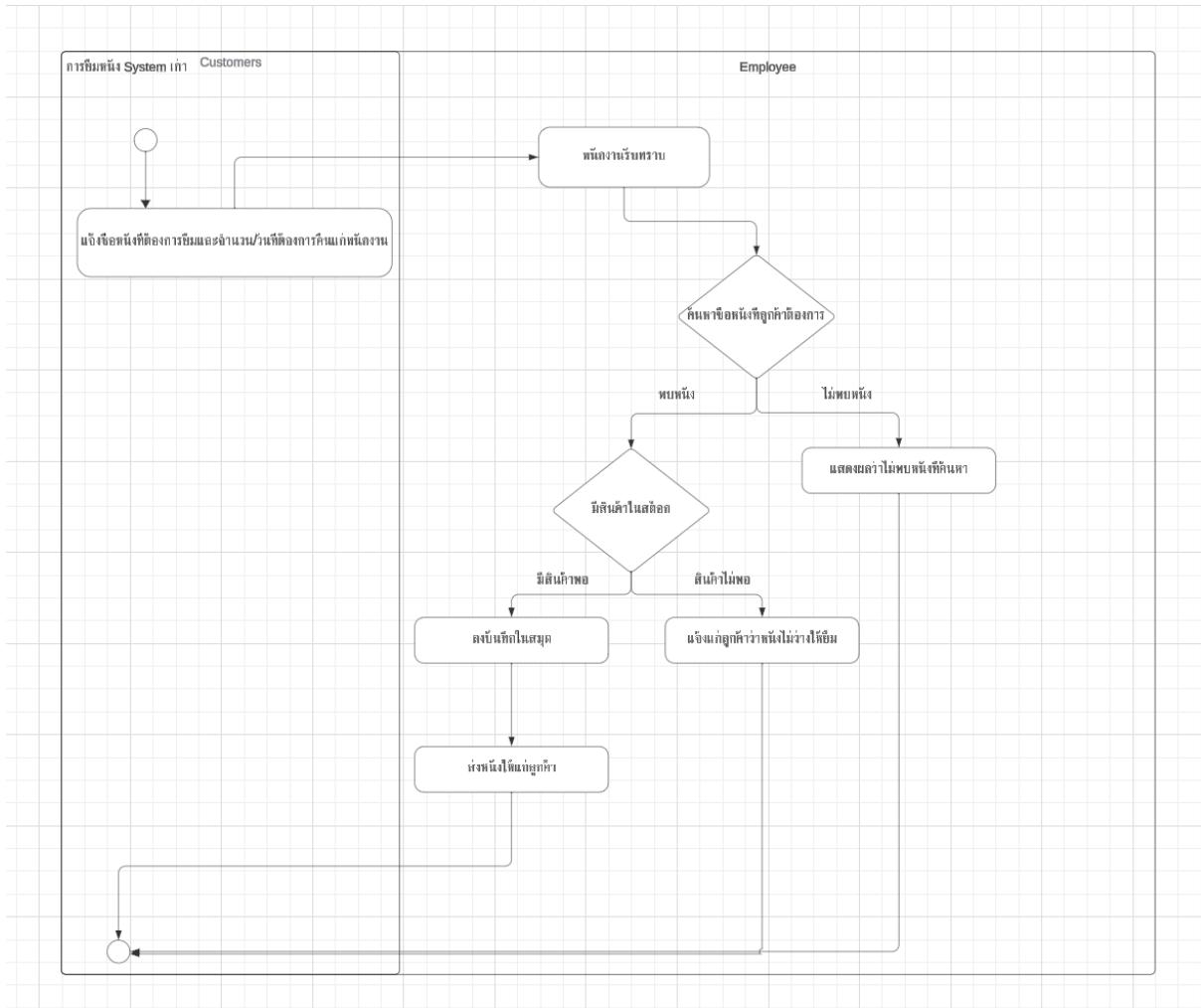
2.1 รายละเอียดการเก็บข้อมูลแบบเก่า

ในสมัยที่ร้านเช่าภาคยนตร์ยังรุ่งเรือง วิธีที่ใช้ในการเก็บข้อมูลคือ เก็บไว้ในสมุดกระดาษที่มีตัวเลขรหัสแผ่นหนัง พร้อมกับสมุดบันทึกรายการ การเช่าแผ่นหนังของลูกค้า ซึ่งวิธีแบบเก่านั้นทำให้เข้าถึงข้อมูลยากเมื่อมี ลูกค้าเข้ามาในร้านและบอกชื่อหนังที่ต้องการ ผู้ขายก็จะต้องไปนั่งดูรายชื่อ รายการหนังตั้งแต่หน้าแรกเพื่อที่จะดูว่าตัวเองเก็บแผ่นหนังไว้ที่ไหน นอกเหนื่อนนั้น เมื่อหาเจอแล้วจะทำการลงสมุดบันทึกประกอบด้วย ซึ่งผู้ยืม ชื่อหนัง วิธีติดต่อและวันครบกำหนด ข้อมูลทั้งหมดถูกบันทึกลงสมุด กระดาษ ไม่มี Backup ถ้าสมุดหาย สมุดฉบับเดียว ก็ไม่สามารถตามหาคนที่ เช่าแผ่นหนังได้อีก

Dataflow การค้นหาหนังแบบเก่า



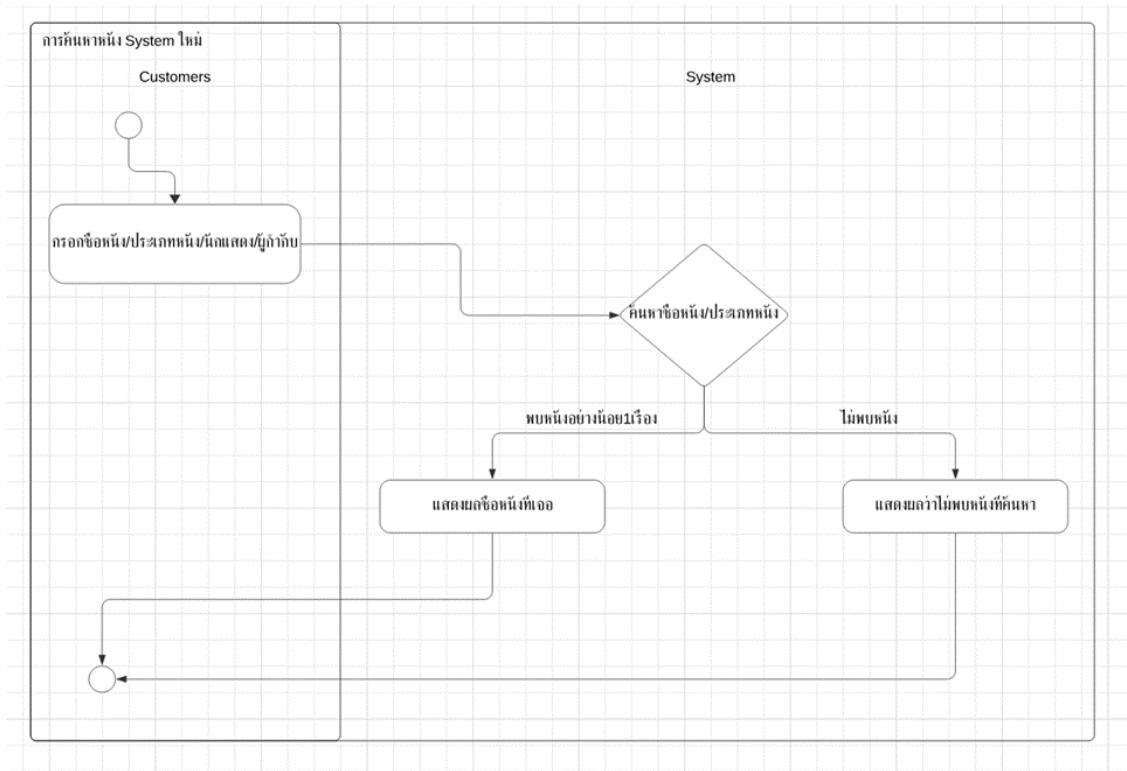
Dataflow การยืนหนังแบบเก่า



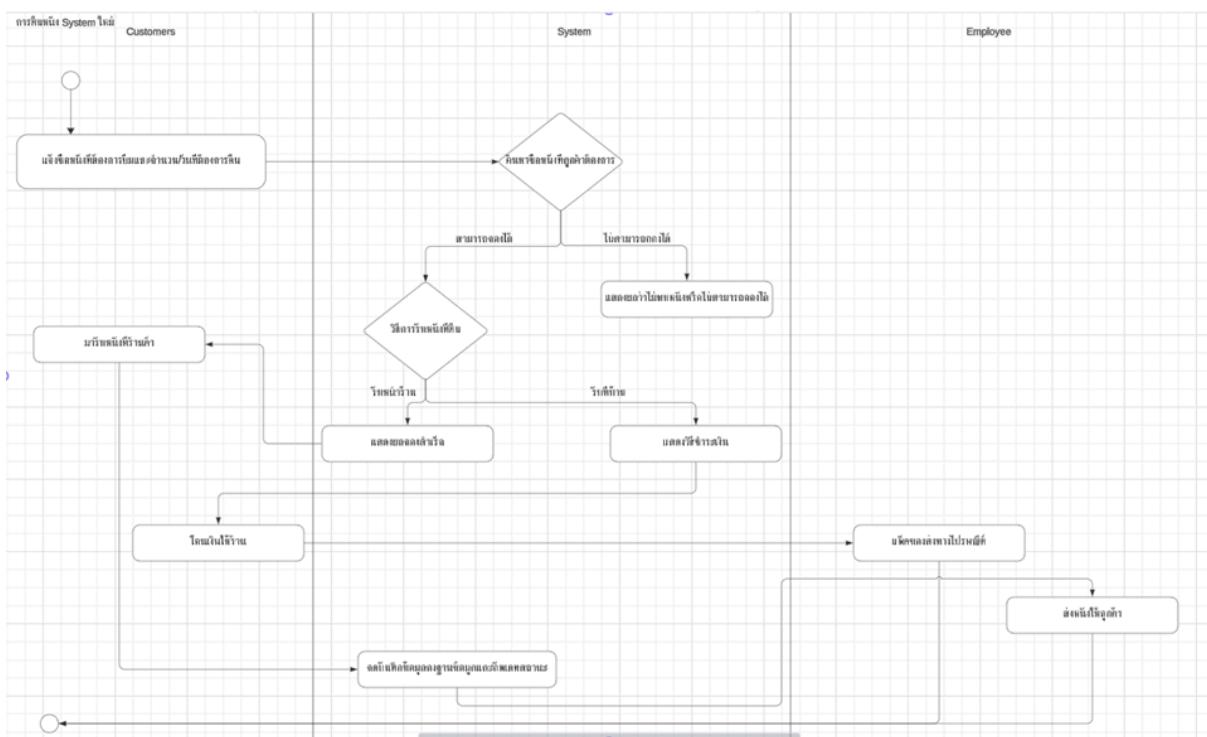
2.2 รายละเอียดการเก็บข้อมูลแบบใหม่ที่คณะผู้จัดทำ ออกแบบ

ข้อมูลของแผ่นหนังภาพนิทรรศรวมทั้งข้อมูลลูกค้า ข้อมูลพนักงานหรือข้อมูลการเช่าทั้งหมดถูกเก็บไว้ในฐานข้อมูลของคอมพิวเตอร์ เมื่อต้องมีการเรียกใช้สามารถทำได้โดยการเขียนโปรแกรมคำสั่งหาแผ่นหนังภาพนิทรรศที่ต้องการรวมทั้งยังสามารถ Insert ข้อมูลการเช่า และมีการแจ้งเตือนใน Application ในกรณีที่ลูกค้าไม่ได้คืนแผ่นหนังก่อน Deadline การเช่าอีกด้วย

Dataflow การค้นหาหนังแบบใหม่

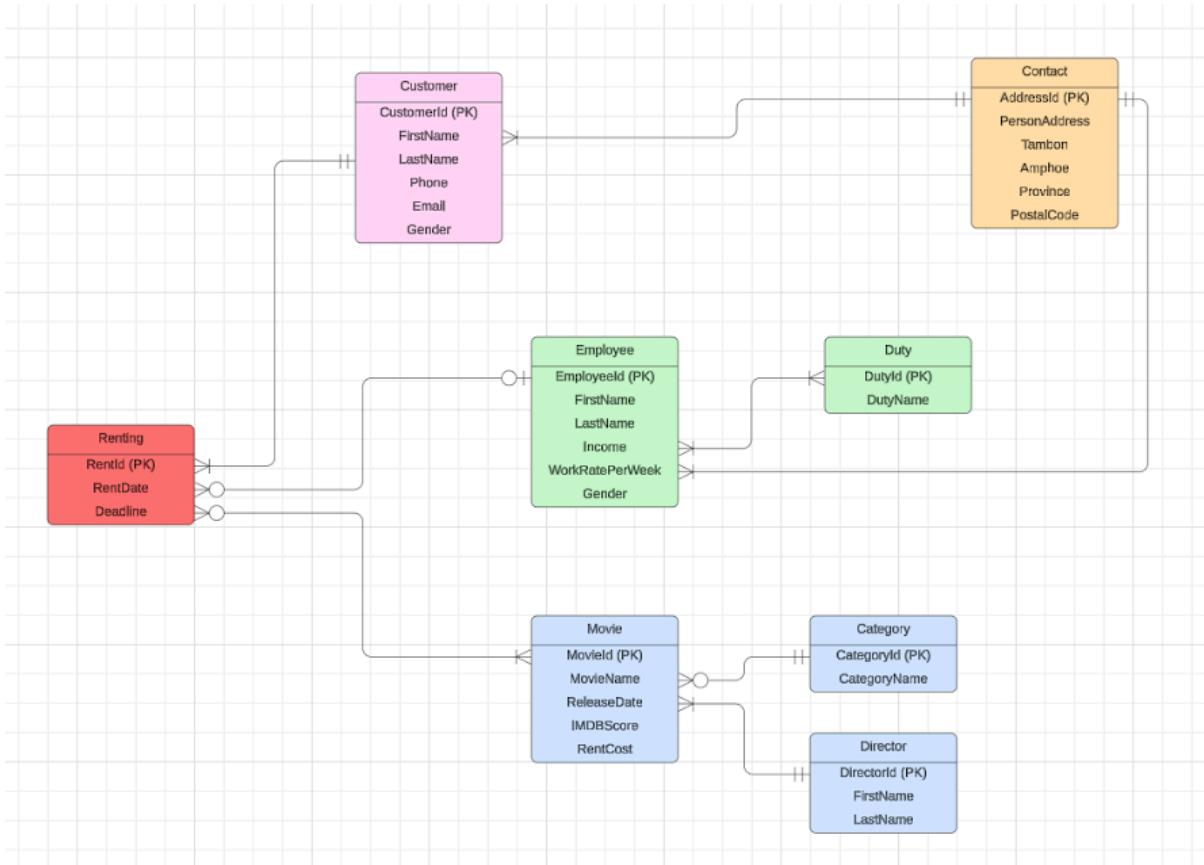


Dataflow การยืมหนังแบบใหม่



3. การออกแบบฐานข้อมูล

3.1 Entity-Relation Diagram (ER Diagram)



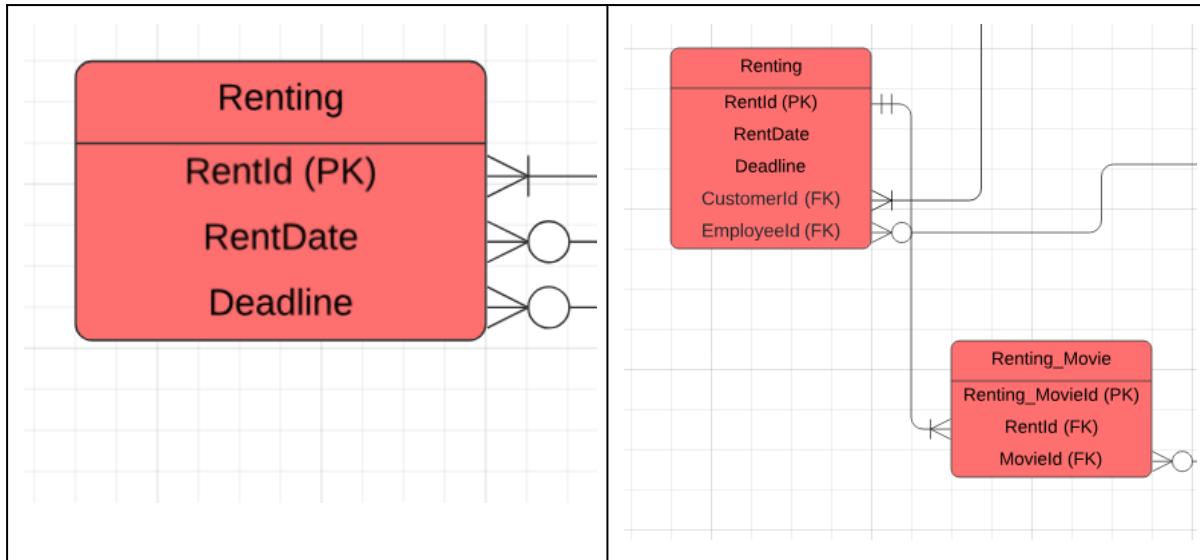
3.2 คำอธิบาย Entities

- Renting คือ ตารางเก็บข้อมูลการเช่าภาพยนต์
- Employee คือ ตารางเก็บข้อมูลส่วนตัวพนักงาน
- Duty คือ ตารางเก็บข้อมูลหน้าที่พนักงาน
- Movie คือ ตารางเก็บข้อมูลภาพยนต์
- Category คือ ตารางเก็บข้อมูลประเภทภาพยนต์
- Director คือ ตารางเก็บข้อมูลผู้กำกับ
- Customer คือ ตารางเก็บข้อมูลลูกค้า
- AddressId คือ ตารางเก็บข้อมูลที่อยู่ของผู้ที่เกี่ยวข้องกับร้านทั้งหมด

4. Normalization

4.1 Normalization Table ต่างๆ

1. Table Renting



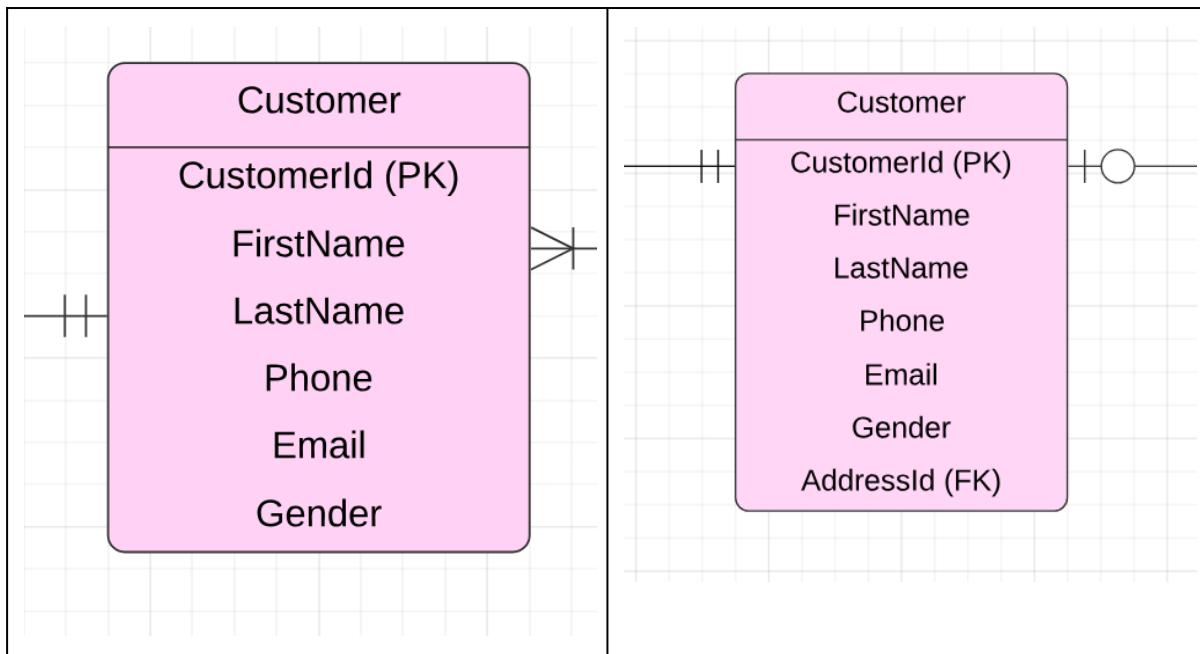
Functional dependencies

Rentid \rightarrow Rentdate,Deadline,Movield

1NF	เป็น atomic
2NF	NKA ขึ้นกับทุกส่วนของคีย์
3NF	ไม่มี transitive dependency
BCNF	ไม่มี $A \rightarrow B$ และไม่ได้เป็นคีย์ของตาราง

มีความสัมพันธ์ many to many กับ movie จึงมี weak entities table Renting Movie มาเชื่อมใน Relational

2. Table Customer

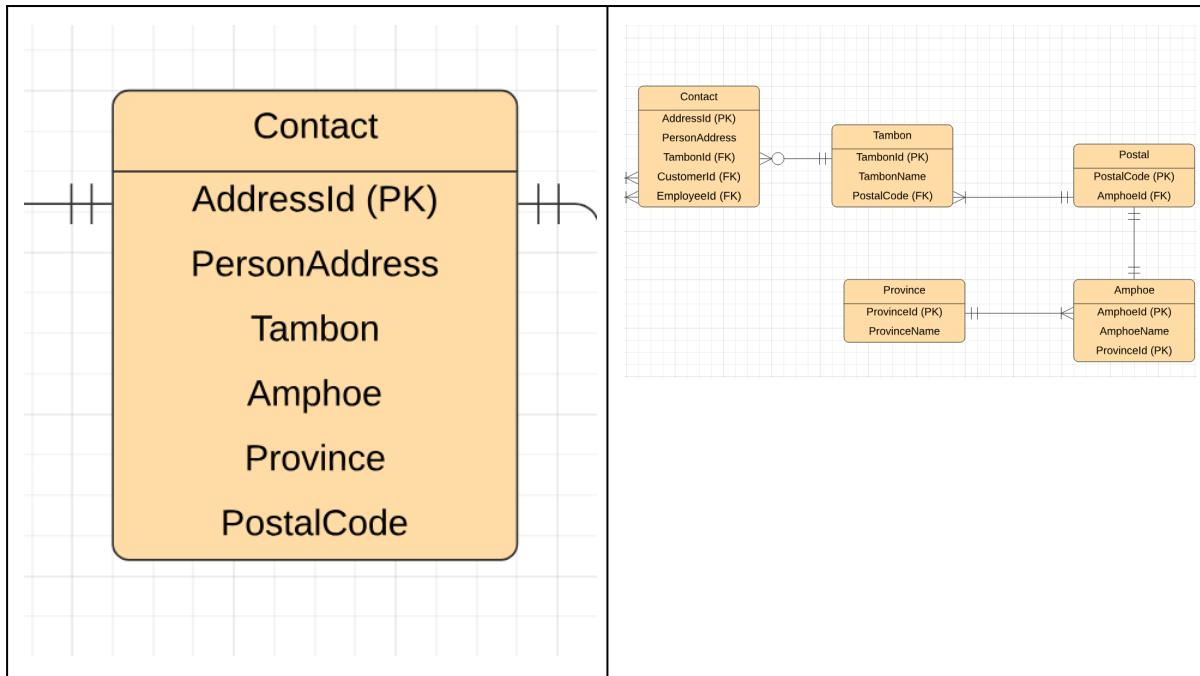


Functional dependencies

Customerid → Firstname,Lastname,Phone,Email,Gender

1NF	เป็น atomic
2NF	NKA ขึ้นกับทุกส่วนของคีย์
3NF	ไม่มี transitive dependency
BCNF	ไม่มี A->B และไม่ได้เป็นคีย์ของตาราง

3. Table Contact



Functional dependencies

AddressId → PersonAddress, Tambon, Amphoe, Province, PostalCode

Tambon → PostalCode → Amphoe → Province

1NF	เป็น atomic
2NF	NKA ขึ้นกับทุกส่วนของคีย์
3NF	มี transitive dependency
BCNF	มี A->B และไม่ได้เป็นคีย์ของตาราง

แยก Table ออกมาจึงได้ดังรูปในช่องขวาซึ่งมีดังนี้

Table Tambon, Postal, Amphoe, Province

Contact เชื่อมกับ Tambon ก่อน เพราะมีหลายตำบลที่มีรหัสซ้ำได้เช่น

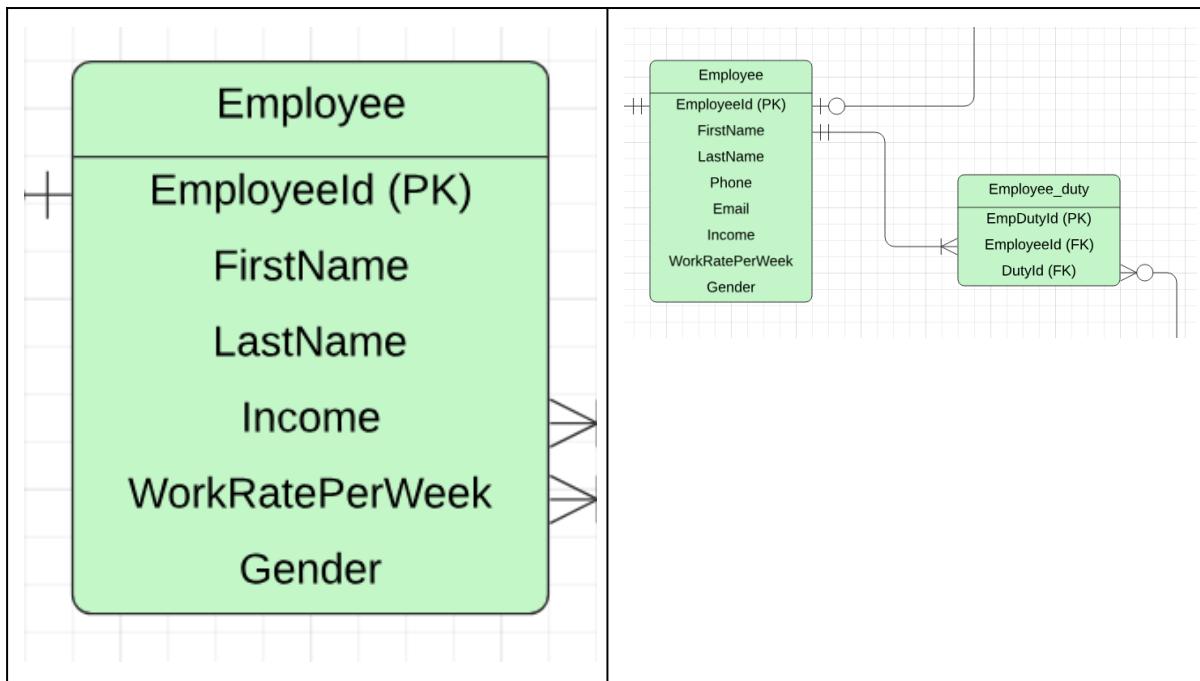
12120 คลองเจ็ด , 12120 คลองสาม

Functional dependencies

{TambonId → TambonName, PostalCode} , {PostalCode → AmphoeId} ,

{AmphoeId → AmphoeName, ProvincId}, {ProvincId → ProvinceName}

4. Table Employee



Functional dependencies

EmployeeId → FirstName, LastName, Income, WorkRatePerWeek, Gender

1NF	เป็น atomic
2NF	NKA ขึ้นกับทุกส่วนของคีย์
3NF	ไม่มี transitive dependency
BCNF	ไม่มี A->B แล้วไม่ได้เป็นคีย์ของตาราง

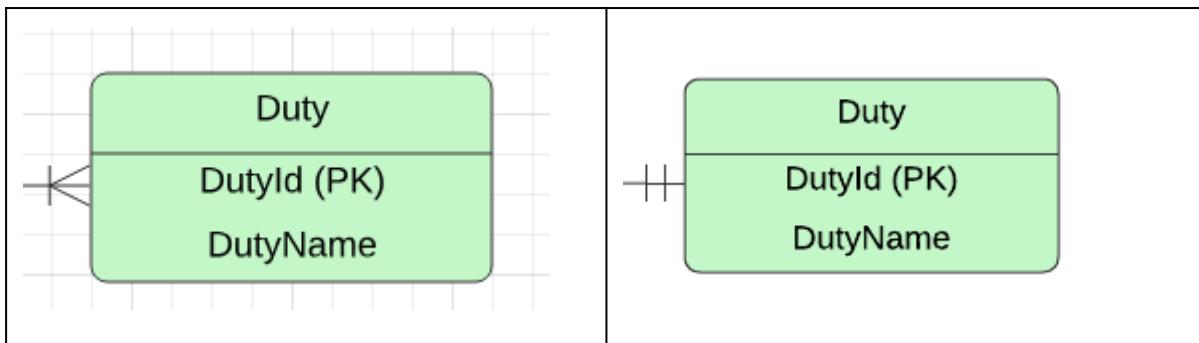
มีความสัมพันธ์ many to many กับ Duty จึงมี weak entities table
Employee_duty มาเชื่อมใน Relational

Table Employee_duty

Functional dependencies

EmpDutyId → EmployeeId, DutyId

5. Table Duty

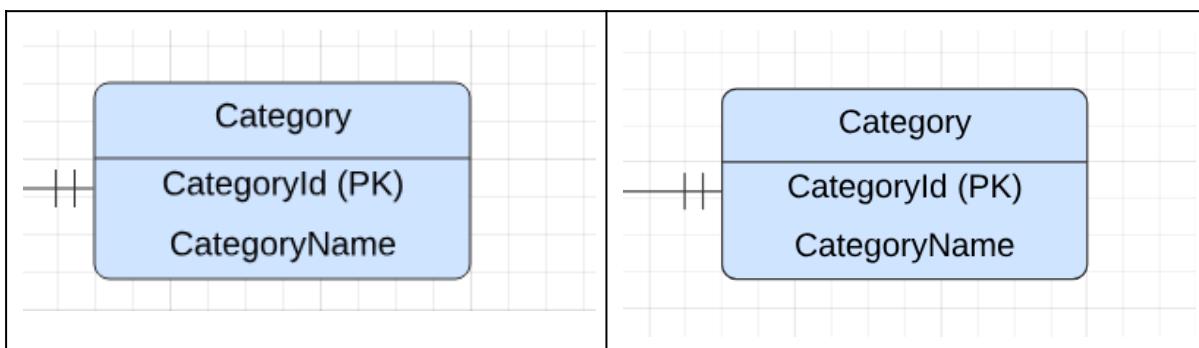


Functional dependencies

DutyId → DutyName

1NF	เป็น atomic
2NF	NKA ขึ้นกับทุกส่วนของคีย์
3NF	ไม่มี transitive dependency
BCNF	ไม่มี $A \rightarrow B$ และไม่ได้เป็นคีย์ของตาราง

6. Table Category

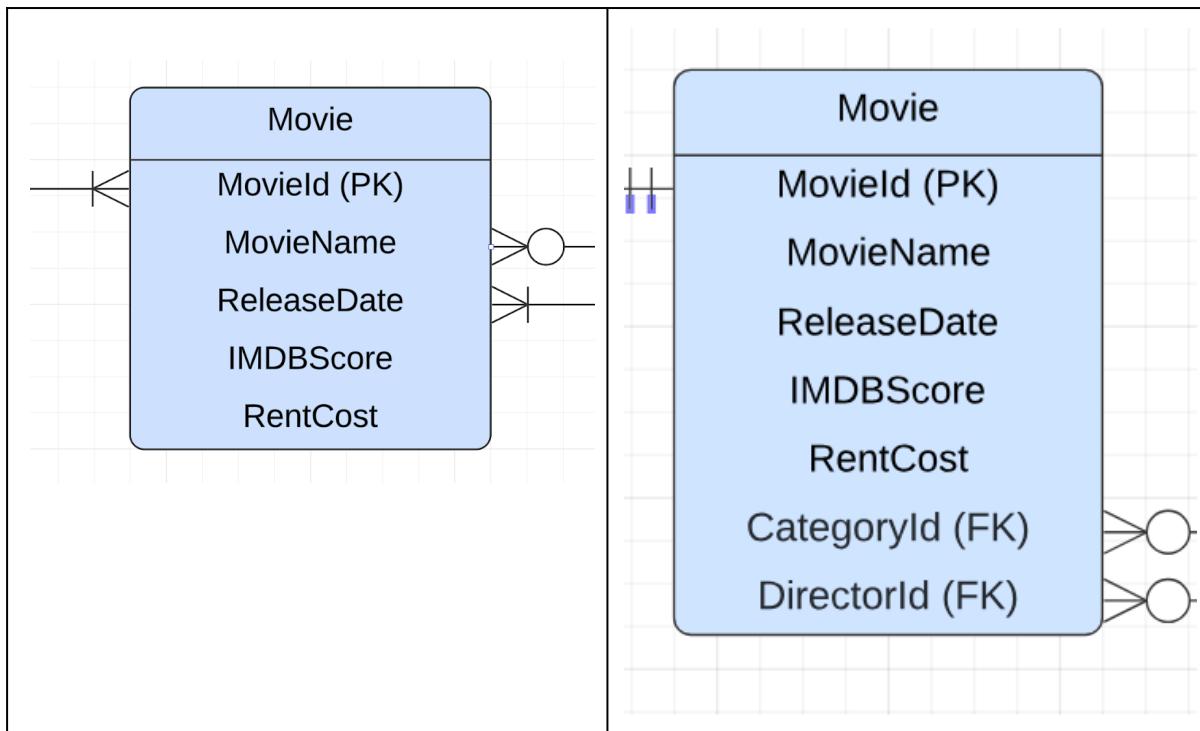


Functional dependencies

CategoryId → CategoryName

1NF	เป็น atomic
2NF	NKA ขึ้นกับทุกส่วนของคีย์
3NF	ไม่มี transitive dependency
BCNF	ไม่มี $A \rightarrow B$ และไม่ได้เป็นคีย์ของตาราง

7.Table Movie

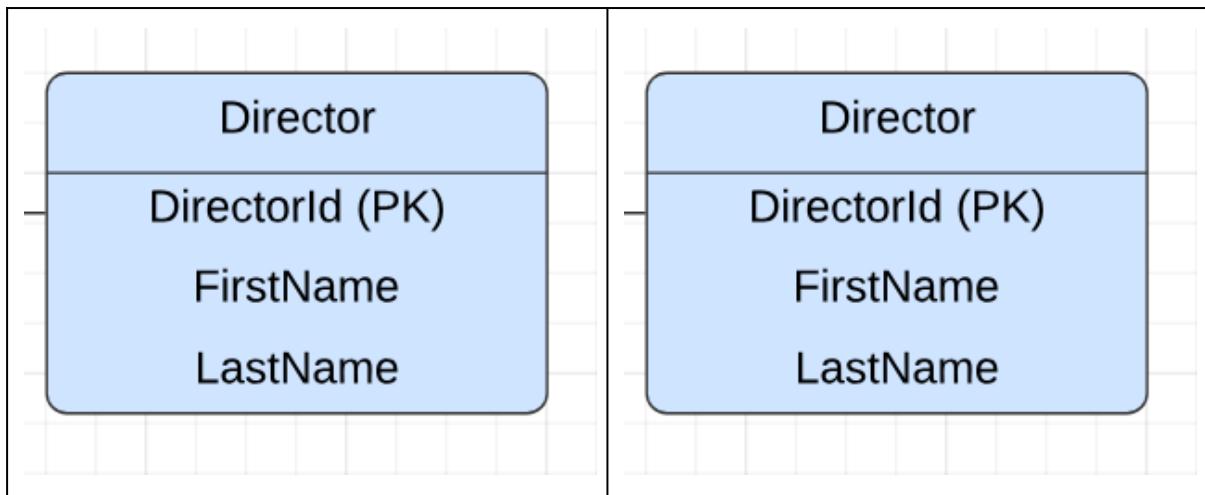


Functional dependencies

$\text{Movield} \rightarrow \text{MovieName}, \text{ReleaseDate}, \text{IMDBScore}, \text{RentCost}$

1NF	เป็น atomic
2NF	NKA ขึ้นกับทุกส่วนของคีย์
3NF	ไม่มี transitive dependency
BCNF	ไม่มี $A \rightarrow B$ และไม่ได้เป็นคีย์ของตาราง

8.Table Director

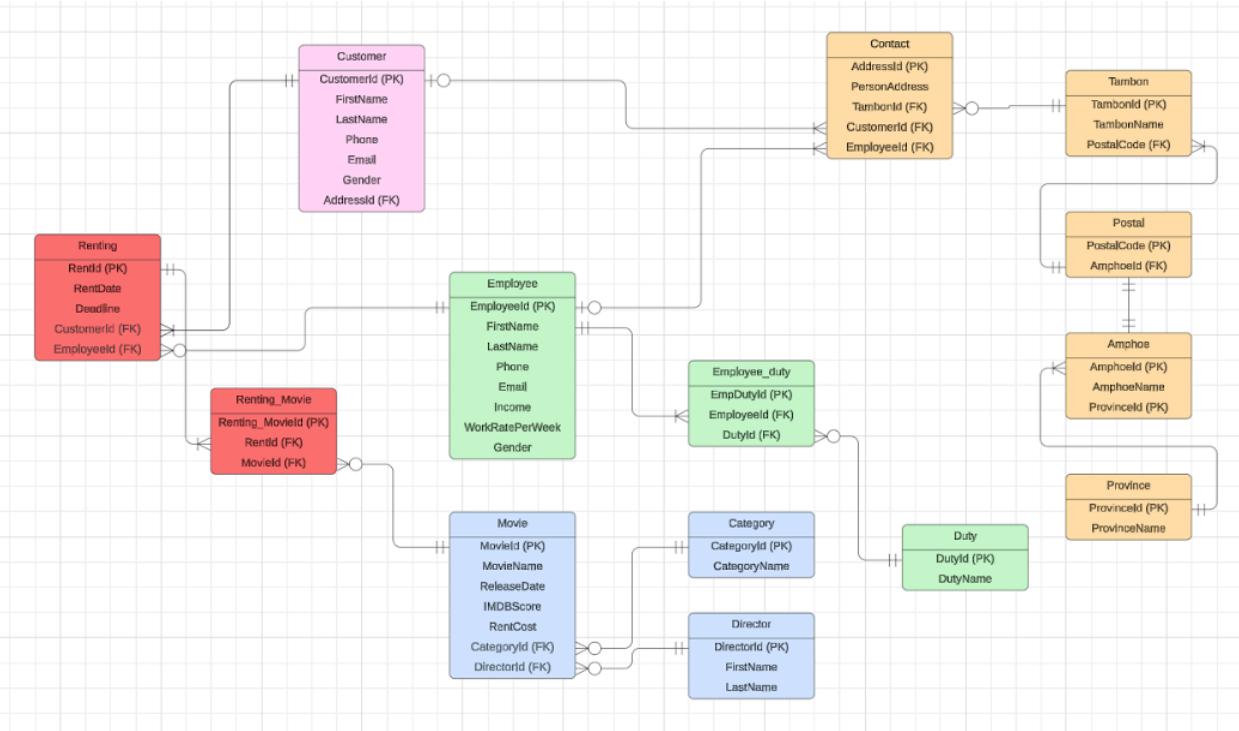


Functional dependencies

$\text{DirectorId} \rightarrow \text{FirstName}, \text{LastName}$

1NF	เป็น atomic
2NF	NKA ขึ้นกับทุกส่วนของคีย์
3NF	ไม่มี transitive dependency
BCNF	ไม่มี $A \rightarrow B$ และไม่ได้เป็นคีย์ของตาราง

4.2 Relational Diagram



4.3 Data Dictionary

Renting ตารางเก็บข้อมูลการเช่าหนัง

Order	Column Name	Data Type	Constraints	Description
1	RentId	INTEGER	Not Null,Primarykey	รหัสการเช่า
2	Rentdate	DATETIME	Not Null	วันที่เช่า
3	Deadline	DATETIME	Not Null	วันสุดท้ายของการเช่า
4	CustomerId	INTEGER	Not Null,Foreign Key	รหัสลูกค้า
5	EmployeeId	INTEGER	Not Null,Foreign Key	รหัสพนักงาน

Renting_Movie ตารางเก็บข้อมูลรายชื่อนั้งที่ถูกเช่าในแต่ละครั้ง

Order	Column Name	Data Type	Constraints	Description
1	Renting_Movield	INTEGER	Not Null,Primarykey	รหัสการเช่าหนัง
2	RentId	INTEGER	Not Null,Foreign Key	รหัสการเช่า
3	Movield	INTEGER	Not Null,Foreign Key	รหัสหนัง

Movie ตารางเก็บข้อมูลหนัง

Order	Column Name	Data Type	Constraints	Description
1	Movield	INTEGER	Not Null,Primarykey	รหัสภาพยนตร์
2	CategoryId	INTEGER	Not Null,Foreign keys	รหัสประเภทภาพยนตร์
3	DirectorId	INTEGER	Not Null,Foreign Key	รหัสผู้กำกับ
4	MovieName	NVARCHAR(120)	Not Null	ชื่อภาพยนตร์
5	ReleaseDate	DATETIME	Null	วันที่ฉายครั้งแรก
6	IMDBScore	NUMERIC	Null,RANGE(0,10)	คะแนน IDMB
7	RentCost	INTEGER	Not Null	ค่าเช่า

Customer ตารางเก็บข้อมูลลูกค้า

Order	Column Name	Data Type	Constraints	Description
1	CustomerId	INTEGER	Not Null,Primarykey	รหัสลูกค้า
2	FirstName	NVARCHAR(40)	Not Null	ชื่อจริงลูกค้า
3	LastName	NVARCHAR(40)	Not Null	นามสกุลลูกค้า
4	Phone	NVARCHAR(24)	Not Null	เบอร์โทรศัพท์ลูกค้า
5	Email	NVARCHAR(60)	Not Null	อีเมลลูกค้า
6	Gender	NVARCHAR(40)	Not Null	เพศของลูกค้า
7	AddressId	INTEGER	Not Null,Foreign Key	รหัสที่อยู่

Contact ตารางเก็บข้อมูลการติดต่อ

Order	Column Name	Data Type	Constraints	Description
1	AddressId	INTEGER	Not Null,Primarykey	รหัสที่อยู่
2	PersonAddress	NVARCHAR(60)	Not Null	ที่อยู่ลูกค้า
3	TambonId	INTEGER	Not Null,Foreign Key	รหัสตำบล
4	CustomerId	INTEGER	Foreign Key	รหัสลูกค้า
5	EmployeeId	INTEGER	Foreign Key	รหัสพนักงาน

Employee ตารางเก็บข้อมูลพนักงาน

Order	Column Name	Data Type	Constraints	Description
1	EmployeeId	INTEGER	Not Null, Primarykey	รหัสพนักงาน
2	FirstName	NVARCHAR(40)	Not Null	ชื่อจริงพนักงาน
3	LastName	NVARCHAR(40)	Not Null	นามสกุลพนักงาน
4	Phone	INVARCHAR(24)	Not Null	เบอร์โทรศัพท์พนักงาน
5	Email	NVARCHAR(60)	Not Null	อีเมลพนักงาน
6	Income	INTEGER	Not Null	รายได้พนักงาน
7	WorkRatePerWeek	INTEGER	Not Null	อัตราการทำงานต่อสัปดาห์หน่วยชั่วโมง
8	Gender	NVARCHAR(40)	Not Null	เพศของพนักงาน

Duty ตารางเก็บข้อมูลหน้าที่ในร้าน

Order	Column Name	Data Type	Constraints	Description
1	DutyId	INTEGER	Not Null,Primarykey	รหัสหน้าที่
2	DutyName	NVARCHAR(120)	Not Null	หน้าที่

Employee_duty ตารางเก็บข้อมูลหน้าที่ของพนักงานแต่ละคน

Order	Column Name	Data Type	Constraints	Description
1	EmpDutyId	INTEGER	Not Null,Primary Key	รหัสตาราง
2	DutyId	INTEGER	Not Null,Foreign Key	รหัสหน้าที่
3	EmployeeId	INTEGER	Not Null,Foreign Key	รหัสพนักงาน

Catagory ตารางเก็บข้อมูลประเภทหนัง

Order	Column Name	Data Type	Constraints	Description
1	CategoryId	INTEGER	Not Null,Primarykey	รหัสประเภท ภาพยนตร์
2	CategoryName	NVARCHAR(40)	Not Null	ประเภท ภาพยนตร์

Director ตารางเก็บข้อมูลผู้กำกับ

Order	Column Name	Data Type	Constraints	Description
1	DirectorId	INTEGER	Not Null,Primarykey	รหัสผู้กำกับ
2	FirstName	NVARCHAR(40)	Not Null	ชื่อจริงผู้กำกับ
3	LastName	NVARCHAR(40)	Not Null	นามสกุลผู้ กำกับ

TamBon ตารางเก็บข้อมูลตำบล

Order	Column Name	Data Type	Constraints	Description
1	TambonId	INTEGER	Not Null,Primarykey	รหัสตำบล
2	TambonName	NVARCHAR(120)	Not Null	ชื่อตำบล
3	Postalcode	INTEGER	Not Null,Foreign Key	รหัสไปรษณีย์

Postal ตารางเก็บข้อมูลรหัสไปรษณีย์

Order	Column Name	Data Type	Constraints	Description
1	Postalcode	INTEGER	Not Null,Primarykey	รหัสไปรษณีย์
2	Amphoeld	INTEGER	Not Null,Foreign Key	รหัสอำเภอ

Amphoe ตารางเก็บข้อมูลอำเภอ

Order	Column Name	Data Type	Constraints	Description
1	Amphoeld	INTEGER	Not Null,Primarykey	รหัสอำเภอ
2	AmphoeName	NVARCHAR(120)	Not Null	ชื่ออำเภอ
3	Provinceld	INTEGER	Not Null,Foreign Key	รหัสจังหวัด

Province ตารางเก็บข้อมูลจังหวัด

Order	Column Name	Data Type	Constraints	Description
1	ProvinceId	INTEGER	Not Null,Primarykey	รหัสจังหวัด
2	ProvinceName	NVARCHAR(120)	Not Null	ชื่อจังหวัด

5. Query Design & Development

ทางคณะผู้จัดทำได้ใช้โปรแกรม DB Browser (SQLite) ในการสร้างฐานข้อมูลสำหรับร้านเช่าหนังภาพยนตร์ขึ้นมา เนื่องจากกิจการร้านเช่าหนังภาพยนตร์ในประเทศไทยได้หายไปหมดแล้ว ดังนั้น ข้อมูลทั้งหมดที่อยู่ในฐานข้อมูลนี้ เป็นข้อมูลของร้านจำลองที่ทางคณะผู้จัดทำได้สร้างขึ้นมา

5.1 คำสั่งการสร้างตารางและข้อมูลในแต่ละตาราง

1) Renting

```
CREATE TABLE "Renting" (
    "RentId" INTEGER NOT NULL UNIQUE,
    "RentDate" DATETIME NOT NULL,
    "Deadline" DATETIME NOT NULL,
    "CustomerId" INTEGER NOT NULL,
    "EmployeeId" INTEGER NOT NULL,
    PRIMARY KEY("RentId" AUTOINCREMENT),
    FOREIGN KEY("CustomerId") REFERENCES "Customer"("CustomerId")
    FOREIGN KEY("EmployeeId") REFERENCES "Employee"("EmployeeId")
);
```

RentId	RentDate	Deadline	CustomerId	EmployeeId
Filter	Filter	Filter	Filter	Filter
1	2022-04-13 00:00:00	2022-04-23 00:00:00	1	2
2	2022-08-15 00:00:00	2022-08-25 00:00:00	1	2
3	2022-08-21 00:00:00	2022-08-31 00:00:00	2	2
4	2023-04-15 00:00:00	2023-04-25 00:00:00	3	2
5	2023-05-13 00:00:00	2023-05-23 00:00:00	2	4
6	2023-05-22 00:00:00	2023-05-31 00:00:00	2	4
7	2023-07-13 00:00:00	2023-07-23 00:00:00	1	4
8	2023-11-17 00:00:00	2023-11-27 00:00:00	4	4
9	2023-12-05 00:00:00	2023-12-15 00:00:00	2	4
10	2023-12-13 00:00:00	2023-12-23 00:00:00	2	4

2) Renting_Movie

```
CREATE TABLE "Renting_Movie" (
    "Renting_MovieId" INTEGER NOT NULL UNIQUE,
    "RentId" INTEGER NOT NULL,
    "MovieId" INTEGER NOT NULL,
    PRIMARY KEY ("Renting_MovieId" AUTOINCREMENT),
    FOREIGN KEY ("MovieId") REFERENCES "Movie"("MovieId"),
    FOREIGN KEY ("RentId") REFERENCES "Renting"("RentId")
);
```

Renting_MovieId	RentId	MovieId
Filter	Filter	Filter
1	1	14
2	2	16
3	3	6
4	4	21
5	5	12
6	6	15
7	7	7
8	8	29
9	9	13
10	10	23
11	10	1
12	10	25

3) Movie

```
CREATE TABLE "Movie" (
    "MovieId"      INTEGER NOT NULL UNIQUE,
    "MovieName"    TEXT NOT NULL,
    "ReleaseDate"   DATETIME NOT NULL,
    "IMDBScore"    NUMERIC NOT NULL,
    "RentCost"     INTEGER NOT NULL,
    "CategoryId"   INTEGER NOT NULL,
    "DirectorId"   INTEGER NOT NULL,
    FOREIGN KEY("DirectorId") REFERENCES "Director"("DirectorId"),
    FOREIGN KEY("CategoryId") REFERENCES "Category"("CategoryId"),
    PRIMARY KEY("MovieId" AUTOINCREMENT)
);
```

MovieId	MovieName	ReleaseDate	IMDbScore	RentCost	CategoryId	DirectorId
Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	BangkokZombie	1983-10-09 00:00:00	6	30	11	8
2	Stein's Door	1998-05-01 00:00:00	10	20	2	4
3	The Pool 2	2001-06-14 00:00:00	7	20	5	3
4	Re: One Two Three	1999-06-13 00:00:00	10	20	2	4
5	Going to Study by plane	2011-01-19 00:00:00	5	20	16	5
6	Arrow Art Offline	2010-02-27 00:00:00	5	20	2	4
7	KU Redemption	2012-10-05 00:00:00	4	20	4	3
8	Jungle Gump	1991-12-04 00:00:00	4	30	7	6
9	Club of Legend	2012-10-24 00:00:00	6	20	7	10
10	Yourscraft	1987-07-05 00:00:00	4	30	5	7
11	The Snake Snake Fish Fish	2020-12-26 00:00:00	6	30	8	1
12	Defend on Giant	2013-12-17 00:00:00	8	20	15	7
13	Noodle Panda 4	1973-01-22 00:00:00	4	20	2	9
14	Jattis Jaroen	1976-12-18 00:00:00	3	20	6	4
15	Rayong Revengers	1994-03-03 00:00:00	6	20	9	10
16	Chatbest : Infinite Programming	2019-01-05 00:00:00	4	30	10	1
17	Good 2018	1980-11-12 00:00:00	6	20	16	4
18	Narm vs Pythonian	2017-07-10 00:00:00	3	20	10	4
19	Hate Dead Moonlight	1987-01-15 00:00:00	7	30	6	9
20	Half Metal Enchanter	2008-10-02 00:00:00	2	20	15	8
21	BestBell Forbidden Love	1996-05-22 00:00:00	4	20	8	4
22	Diamond Man	2003-06-28 00:00:00	7	20	1	3
23	Captain Esan	2002-06-23 00:00:00	9	20	1	8
24	Jump Queen	1972-11-11 00:00:00	7	20	8	2
25	Missing Science	2012-11-13 00:00:00	10	30	11	4
26	Karate Kaisen	1984-09-05 00:00:00	5	30	16	8
27	Minor Theft Manual 6	1992-09-25 00:00:00	9	30	10	9
28	Zero Kick Woman	1984-11-28 00:00:00	9	20	7	9
29	Poker War	2013-10-10 00:00:00	6	30	16	7
30	Once I am 13 years old boy	1972-11-10 00:00:00	9	30	4	4

4) Category

```

CREATE TABLE "Category" (
    "CategoryId"      INTEGER NOT NULL UNIQUE,
    "CategoryName"   TEXT NOT NULL,
    PRIMARY KEY("CategoryId" AUTOINCREMENT)
);

```

CategoryId	CategoryName
Filter	Filter
1	Action
2	Animation
3	Children
4	Classics
5	Comedy
6	Documentary
7	Drama

8	Family
9	Foreign
10	Games
11	Horror
12	Music
13	New
14	Sci-Fi
15	Sports
16	Travel

5) Director

```
CREATE TABLE "Director" (
    "DirectorId"      INTEGER NOT NULL UNIQUE,
    "FirstName"       TEXT NOT NULL,
    "LastName"        TEXT NOT NULL,
    PRIMARY KEY("DirectorId" AUTOINCREMENT)
);
```

DirectorId	FirstName	LastName
Filter	Filter	Filter
1	Juice	Traffic
2	Bunt	Fullbrave
3	Suthamet	Goodswim
4	Sompon	Zeroone
5	Nitboon	Dadkrajui
6	Euro	Artof
7	Passa	Manukak
8	Meebun	Suasupan
9	Tawan	Threetowsan
10	Valor	Romanten

6) Customer

```
CREATE TABLE "Customer" (
    "CustomerId" INTEGER NOT NULL UNIQUE,
    "FirstName" TEXT NOT NULL,
    "LastName" TEXT NOT NULL,
    "Gender" TEXT NOT NULL,
    "Phone" TEXT NOT NULL,
    "Email" TEXT NOT NULL,
    "AddressId" INTEGER NOT NULL,
    FOREIGN KEY("AddressId") REFERENCES "Contact" ("AddressId"),
    PRIMARY KEY("CustomerId" AUTOINCREMENT)
);
```

CustomerId	FirstName	LastName	Gender	Phone	Email	AddressId
Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	Jattis	Jaroen	Male	817778888	jattis_jaroen@gmail.com	5
2	Nathan	Aroon	Male	856596325	AnimeLover@gmail.com	6
3	Pattama	Padung	Female	859565587	NaewElectric@gmail.com	7
4	Chamghat	Phoempoon	Male	236547896	PokerMaster@gmail.com	8

7) Employee

```
CREATE TABLE "Employee" (
    "EmployeeId" INTEGER NOT NULL UNIQUE,
    "FirstName" TEXT NOT NULL,
    "LastName" TEXT NOT NULL,
    "Phone" TEXT NOT NULL,
    "Email" TEXT NOT NULL,
    "Income" INTEGER NOT NULL,
    "WorkRatePerWeek" INTEGER NOT NULL,
    "Gender" TEXT NOT NULL,
    PRIMARY KEY("EmployeeId" AUTOINCREMENT)
);
```

EmployeeId	FirstName	LastName	Phone	Email	Income	WorkRatePerWeek	Gender
Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	Sunyarn	Loedvisai	815672349	SignalSunyarn@gmail.com	20000		Male
2	Sittichok	Srikhem	856423579	AAAAAAA@gmail.com	10000		Male
3	Arnold	Yamkum	894561234	MikeLabDotNet@gmail.com	10000		Male
4	Janna	Jantra	915647892	QuizzizzLover@gmail.com	10000		Female

8) Employee_duty

```
CREATE TABLE "Employee_Duty" (
    "EmpDutyId" INTEGER NOT NULL UNIQUE,
    "EmployeeId"      INTEGER NOT NULL,
    "DutyId"         INTEGER NOT NULL,
    FOREIGN KEY("EmployeeId") REFERENCES "Employee"("EmployeeId"),
    FOREIGN KEY("DutyId") REFERENCES "Duty"("DutyId"),
    PRIMARY KEY("EmpDutyId" AUTOINCREMENT)
);
```

EmpDutyId	EmployeeId	DutyId
Filter	Filter	Filter
1	1	1
2	2	2
3	3	3
4	4	2

9) Duty

```
CREATE TABLE "Duty" (
    "DutyId"      INTEGER NOT NULL UNIQUE,
    "DutyName"    TEXT NOT NULL,
    PRIMARY KEY("DutyId" AUTOINCREMENT)
);
```

DutyId	DutyName
Filter	Filter
1	Manager
2	Lender
3	Janitor

10) Contact

```
CREATE TABLE "Contact" (
    "AddressId" INTEGER NOT NULL UNIQUE,
    "PersonAddress" TEXT NOT NULL,
    "TambonId" INTEGER NOT NULL,
    "CustomerId" INTEGER NOT NULL,
    "EmployeeId" INTEGER NOT NULL,
    FOREIGN KEY("TambonId") REFERENCES "Tambon"("TambonId"),
    FOREIGN KEY("CustomerId") REFERENCES "Customer"("CustomerId"),
    FOREIGN KEY("EmployeeId") REFERENCES "Employee"("EmployeeId"),
    PRIMARY KEY("AddressId" AUTOINCREMENT)
);
```

AddressId	PersonAddress	TambonId	CustomerId	EmployeeId
Filter	Filter	Filter	Filter	Filter
1	85/22 Moo 1	2	null	1
2	1/100 Moo 8	5	null	2
3	5/100 Moo 8 Pythonia Star	6	null	3
4	96/97 Moo 9	1	null	4
5	66/76 Moo 3	3	1	null
6	420 Sakurasou Aparment	4	2	null
7	50/87 Moo 6	7	3	null
8	303 Pongneng Apartment	8	4	null

11) Tambon

```
CREATE TABLE "Tambon" (
    "TambonId" INTEGER NOT NULL UNIQUE,
    "TambonName" TEXT NOT NULL,
    "PostalCode" INTEGER NOT NULL,
    FOREIGN KEY("PostalCode") REFERENCES "Postal"("PostalCode"),
    PRIMARY KEY("TambonId" AUTOINCREMENT)
);
```

TambonId	TambonName	PostalCode
Filter	Filter	Filter
1	Doi Tao	50260
2	Khuensri	11150
3	Pha Pong	50220

4	Klong Toey	10110
5	Pracha Thipat	12130
6	Donmuang	10210
7	Lam Sai	12150
8	Klong Dan	11150

12) Postal

```
CREATE TABLE "Postal" (
    "PostalCode" INTEGER NOT NULL UNIQUE,
    "AmphoeId" INTEGER NOT NULL,
    FOREIGN KEY("AmphoeId") REFERENCES "Amphoe"("AmphoeId"),
    PRIMARY KEY("PostalCode")
);
```

PostalCode	AmphoeId
Filter	Filter
10110	4
10210	2
10550	8
11150	7

12130	5
12150	6
50220	3
50260	1

13) Amphoe

```
CREATE TABLE "Amphoe" (
    "AmphoeId" INTEGER NOT NULL UNIQUE,
    "AmphoeName" TEXT NOT NULL,
    "ProvinceId" INTEGER NOT NULL,
    FOREIGN KEY("ProvinceId") REFERENCES "Province"("ProvinceId"),
    PRIMARY KEY("AmphoeId" AUTOINCREMENT)
);
```

AmphoeId	AmphoeName	ProvinceId
Filter	Filter	Filter
1	Doi Tao	5
2	Donmuang	1
3	Doi Saket	5

4	Klong Toey	1
5	Thanyaburi	3
6	Lam Luk Ka	3
7	Sainoi	2
8	Bang Baw	4

14) Province

```
CREATE TABLE "Province" (
    "ProvinceId" INTEGER NOT NULL UNIQUE,
    "ProvinceName" TEXT NOT NULL,
    PRIMARY KEY("ProvinceId" AUTOINCREMENT)
);
```

ProvinceId	ProvinceName
Filter	Filter
1	Bangkok
2	Nonthaburi
3	Pathum Thani
4	Samut Prakan
5	Chiang Mai

5.2 Queries คำสั่งสอบถามข้อมูล

1) ค้นหาจำนวนลูกค้าในแต่ละจังหวัด

```
SELECT pn.ProvinceName, COUNT(*) as "Customer Count"
FROM Customer c, Contact ct, Tambon tb, Postal pt, Amphoe ap, Province pn
WHERE c.CustomerId = ct.CustomerId AND ap.ProvinceId = pn.ProvinceId
AND pt.AmphoeId = ap.AmphoeId AND tb.PostalCode = pt.PostalCode
AND ct.TambonId = tb.TambonId
GROUP BY pn.ProvinceName;
```

	ProvinceName	Customer Count
1	Bangkok	1
2	Chiang Mai	1
3	Nonthaburi	1
4	Pathum Thani	1

2) ดูข้อมูลพนักงานชื่อจริง นามสกุล หน้าที่และรายได้ของพนักงานร้าน

```
SELECT e.EmployeeId, e.FirstName, e.LastName, d.DutyName, e.Income
FROM Employee e, Employee_Duty ed, Duty d
WHERE e.EmployeeId = ed.EmployeeId AND ed.DutyId = d.DutyId
```

EmployeeId	FirstName	LastName	DutyName	Income
1	Sunyarn	Loedvisai	Manager	20000
2	Sittichok	Srikhem	Lender	10000
3	Arnold	Yamkum	Janitor	10000
4	Janna	Jantra	Lender	10000

3) รายชื่อหนังประเภท Animation ที่ถูกเช่าโดยนาย Nathan Aroon

```

SELECT cm.Firstname, cm.LastName, m.MovieName, c.CategoryName
FROM Customer cm
JOIN Renting r ON cm.CustomerId = r.CustomerId
JOIN Renting_Movie rm ON rm.RentId = r.RentId
JOIN Movie m ON rm.MovieId = m.MovieId
JOIN Category c ON m.CategoryId = c.CategoryId
WHERE cm.FirstName = 'Nathan' AND cm.LastName = 'Aroon'
AND c.CategoryName = 'Animation';

```

FirstName	LastName	MovieName	CategoryName
Nathan	Aroon	Arrow Art Offline	Animation
Nathan	Aroon	Noodle Panda 4	Animation

4) แสดงข้อมูลที่อยู่ของนาย Jattis Jaroen

```

SELECT cm.Firstname, ct.PersonAddress, p.ProvinceName, a AmphoeName,
t.TambonName, pt.PostalCode
FROM Customer cm
JOIN Contact ct ON cm.CustomerId = ct.CustomerId
JOIN Tambon t ON ct.TambonId = t.TambonId
JOIN Postal pt ON t.PostalCode = pt.PostalCode
JOIN Amphoe a ON pt.AmphoeId = a.AmphoeId
JOIN Province p ON a.ProvinceId = p.ProvinceId
WHERE cm.FirstName = 'Jattis' AND cm.LastName = 'Jaroen';

```

	FirstName	PersonAddress	ProvinceName	AmphoeName	TambonName	PostalCode
1	Jattis	66/76 Moo 3	Chiang Mai	Doi Saket	Pha Pong	50220

5) แสดงรายชื่อหนังพร้อมวันที่ถูกเช่าในปี 2022

```
SELECT m.MovieName, r.RentDate
FROM Movie m, Renting r, Renting_Movie rm
WHERE r.RentId = rm.RentId AND rm.MovieId = m.MovieId
AND (r.RentDate BETWEEN '2022-01-01' AND '2022-12-31')
```

MovieName	RentDate
Jattis Jaroen	2022-04-13 00:00:00
Chatbest : Infinite Programming	2022-08-15 00:00:00
Arrow Art Offline	2022-08-21 00:00:00

6) หนังที่สร้างในยุค 90s พร้อมแสดงวันที่ฉายและคะแนน IMDB โดยต้องมากกว่าหรือเท่ากับ 8 และมี Sompon Zeroone เป็นผู้กำกับ

```
SELECT m.MovieName, m.IMDBScore, m.ReleaseDate
FROM Movie m, Director d
WHERE m.IMDBScore >= 8 AND m.DirectorId = d.DirectorId
AND d.FirstName = "Sompon" AND d.LastName = "Zeroone"
AND (m.ReleaseDate BETWEEN '1990-01-01' AND '1999-12-31')
```

MovieName	IMDBScore	ReleaseDate
Stein's Door	10	1998-05-01 00:00:00
Re: One Two Three	10	1999-06-13 00:00:00

Type 1 Nested Queries

7) หารายชื่อหนังที่อยู่ในประเภท Family หรือ Comedy

```
SELECT Movie.MovieName
FROM Movie
WHERE Movie.CategoryId IN (
    SELECT CategoryId
    FROM Category
    WHERE CategoryName = 'Family' OR CategoryName = 'Comedy'
);
```

MovieName
The Pool 2
Yourscraft
The Snake Snake Fish Fish
BestBell Forbidden Love
Jump Queen

8) หาผลรวมรายได้ที่หนังประเภท Sci-Fi, Animation, Action ทำได้

```
SELECT SUM(Movie.RentCost)
FROM Movie,Category,Renting,Renting_Movie
WHERE Category.CategoryName IN ('Sci-Fi','Animation','Action')
AND Category.CategoryId=Movie.CategoryId
AND Renting_Movie.MovieId=Movie.MovieId
AND Renting.RentId=Renting_Movie.RentId
```

SUM(Movie.RentCost)

60

Type 2 Nested Queries

9) หาข้อมูลผู้กำกับหนังและชื่อหนังที่กำกับของหนังหมวด Games and Horror

```

SELECT d.DirectorId, d.FirstName, d.LastName, MovieName
FROM Director d, Movie m
WHERE d.DirectorId = m.DirectorId AND EXISTS(
    SELECT * FROM Category c
    WHERE c.CategoryId = m.CategoryId
    AND c.CategoryName IN ('Games', 'Horror')
)
ORDER BY d.DirectorId

```

DirectorId	FirstName	LastName	MovieName
1	Juice	Traffic	Chatbest : Infinite Programming
4	Sompon	Zeroone	Narm vs Pythonian
4	Sompon	Zeroone	Missing Science
8	Meebun	Suasupan	BangkokZombie
9	Tawan	Threetowsan	Minor Theft Manual 6

10) แสดงข้อมูลที่อยู่ของพนักงานที่อยู่ในจังหวัดปทุมธานีหรือเชียงใหม่

```

SELECT e.EmployeeId, e.FirstName, e.LastName,
       c.PersonAddress, t.TambonName, a.AmphoeName, pt.PostalCode
FROM Employee e
JOIN Contact c ON e.EmployeeId = c.EmployeeId
JOIN Tambon t ON c.TambonId = t.TambonId
JOIN Postal pt ON t.PostalCode = pt.PostalCode
JOIN Amphoe a ON pt.AmphoeId = a.AmphoeId
WHERE EXISTS (
    SELECT *
    FROM Province p
    WHERE p.ProvinceId = a.ProvinceId
    AND p.ProvinceName IN ('Pathum Thani', 'Chiang Mai')
)
ORDER BY e.EmployeeId;

```

6. การบริหารจัดการโครงการ

6.1 การวางแผนโครงการ

การทำงาน	ช่วงเวลา
เลือกหัวข้อโครงการ	29 ธ.ค. 2566 - 31 ธ.ค. 2566
รวบรวมข้อมูล	22 ม.ค. 2567 - 23 ม.ค. 2567
ทำรูปเล่มรายงานช่วงที่ 1	24 ม.ค. 2567 - 2 ก.พ. 2567
ทำรูปเล่มรายงานช่วงที่ 2	13 ก.พ. 2567 - 27 ก.พ. 2567
ทำสไลด์นำเสนอโครงการ	28 ก.พ. 2567 - 3 มี.ค. 2567
แก้ไขรูปเล่มรายงานให้เสร็จสมบูรณ์	5 มี.ค. 2567 - 15 มี.ค. 2567

6.2 หน้าที่และความรับผิดชอบของแต่ละสมาชิก

1) นายธนภัทร จำเริญ 6510503417

- ทำสารบัญ
- หาและเขียนข้อมูลในส่วนของรูปแบบการเก็บข้อมูลของร้านเช่าหนัง ภาพยนต์แบบเก่า
- ทำ Dataflow การค้นหาหนังทั้งแบบเก่าและแบบใหม่
- Normalize ฐานข้อมูล เขียน Functional dependencies
- ใส่ข้อมูล Data Dictionary ของแต่ละตาราง
- คิดคำสั่งค้นหาข้อมูล Query คำสั่งที่ 3 และ 4

2) นายบุรินทร์ จารีไฟบูลย์ 6510503484

- คิดหัวข้อตารางและ Attribute ของตาราง
- ทำ Entity Relationship Diagram
- คิดคำสั่งสร้างตาราง
- ใส่ข้อมูล Data Dictionary ของแต่ละตาราง
- คิดคำสั่งค้นหาข้อมูล Query คำสั่งที่ 9 และ 10 (Type 2)
- จัดทำสไลด์นำเสนอ

3) นางสาวพิชญา เอี่ยมกล้าหาญ 6510503620

- รวบรวมหัวข้อโครงงานเพื่อให้สมาชิกในกลุ่มเลือก
- คิดหัวข้อตารางและ Attribute ของตาราง
- ทำ Entity Relationship Diagram
- คิดคำสั่งสร้างตาราง
- ใส่ข้อมูล Data Dictionary ของแต่ละตาราง
- คิดคำสั่งค้นหาข้อมูล Query คำสั่งที่ 7 และ 8 (Type 1)
- จัดทำสไลเดอร์นำเสนอ

4) นายภัทรพล ณ เนตร 6510503654

- เขียนข้อมูลวัตถุประสงค์โครงงาน
- ทำ Relational Diagram
- ออกแบบข้อมูลในฐานข้อมูลของร้านเช่าหนังภาพยนตร์จำลอง โดยการใส่ข้อมูลใน csv file
- ใส่ข้อมูล Data Dictionary ของแต่ละตาราง
- คิดคำสั่งค้นหาข้อมูล Query คำสั่งที่ 5 และ 6
- เขียนข้อมูลการบริหารจัดการโครงงาน
- จัดทำสไลเดอร์นำเสนอ

5) นายสมพล อันหลา 6510503841

- ทำ Dataflow การยึดหนังทั้งแบบเก่าและแบบใหม่
- Normalizeฐานข้อมูล และเขียนรายละเอียดการ Normalize ในส่วนหัวข้อการ Normalization
- ทำ Relational Diagram
- ออกแบบข้อมูลในฐานข้อมูลของร้านเช่าหนังภาพยนตร์จำลอง โดยการใส่ข้อมูลใน csv file
- เขียนโค้ด Python เพื่อ Import ข้อมูลจาก csv file เข้าโปรแกรม DB Browser (SQLite)
- คิดคำสั่งค้นหาข้อมูล Query คำสั่งที่ 1 และ 2