



โครงการ เรื่อง **MOVIE STREAMING**

จัดทำโดย

นายณัฐดนัย วัฒนาไชย 65070075

นายธนภัทร สหบรรเทิงศิลป์ 65070095

นายรามิล ธรรมจินดา 65070198

เสนอ

ผศ.ดร. พัฒนพงษ์ ฉันทมิตรโอภาส

รายงานเล่มนี้เป็นส่วนหนึ่งของวิชา

06066300 แนวคิดระบบฐานข้อมูล

หลักสูตรวิทยาศาสตรบัณฑิต สาขาวิชาวิทยาการข้อมูลและการวิเคราะห์เชิงธุรกิจ

ภาคเรียนที่ 1 ปีการศึกษา 2566 คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

COPYRIGHT 2023

FACULTY OF INFORMATION TECHNOLOGY

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

ใบรับรองโครงการ (PROJECT)

เรื่อง

MOVIE STREAMING

นายณัฐดนัย วัฒนาไชย 65070075

นายธนภัทร สหบรรเทิงศิลป์ 65070095

นายรามิล ธรรมจินดา 65070198

ขอรับรองว่ารายงานฉบับนี้ ข้าพเจ้าไม่ได้คัดลอกมาจากที่ใด
รายงานฉบับนี้ได้รับการตรวจสอบและอนุมัติให้เป็นส่วนหนึ่งของ

การศึกษาวិชาแนวคิดระบบฐานข้อมูล

คณะเทคโนโลยีสารสนเทศ

ภาคเรียนที่ 1 ปีการศึกษา 2566

.....
(นาย.....)
(นาย.....)
(นาย.....)

หัวข้อโครงการ (ภาพยนตร์สตรีมมิ่ง)

นักศึกษา นายณัฐดนัย วัฒนาไชย 65070075

นายธนภัทร สหบรรเทิงศิลป์ 65070095

นายรามิล ธรรมจินดา 65070198

สาขาวิชา วิทยาการข้อมูลและการวิเคราะห์เชิงธุรกิจ

ปีการศึกษา 1/2566

บทคัดย่อ

ระบบ MOVIE STREAMING คือ ระบบที่ให้บริการการสตรีมวิดีโอออนไลน์ โดยเน้นการจัดการข้อมูลที่มีในฐานข้อมูลอย่างมีประสิทธิภาพและมีประโยชน์ที่สำคัญสำหรับระบบนี้ ประโยชน์ของฐานข้อมูลในระบบ MOVIE STREAMING มีหลายด้าน รวมถึงการเก็บรวบรวมข้อมูลเกี่ยวกับวิดีโอทั้งหมดที่ถูกสตรีม รายละเอียดเกี่ยวกับผู้ใช้ และข้อมูลการใช้งานที่เป็นประโยชน์สำหรับการวิเคราะห์และการปรับปรุงบริการ ทำให้ระบบสามารถพัฒนาและปรับปรุงให้มีประสิทธิภาพมากยิ่งขึ้นตามความต้องการของผู้ใช้ ทั้งนี้เพื่อให้การให้บริการสตรีมวิดีโอมีประสิทธิภาพรวดเร็ว และทันเทคโนโลยีล่าสุด โดยมีฐานข้อมูลเป็นส่วนสำคัญที่ช่วยให้ระบบดำเนินงานได้อย่างมีประสิทธิภาพและเป็นระบบที่มีประสิทธิภาพในการบริหารจัดการข้อมูลที่ใช้ในการสตรีมวิดีโอออนไลน์นี้อย่างเหมาะสม

กิตติกรรมประกาศ

เนื่องด้วยความต้องการพัฒนาระบบการสตรีมวิดีโอออนไลน์ที่มีประสิทธิภาพและมีประโยชน์โดยใช้ฐานข้อมูลเพื่อการจัดการข้อมูลอย่างเหมาะสม ซึ่งเป็นส่วนสำคัญที่ช่วยในการจัดเก็บ การเข้าถึงข้อมูล และการประมวลผลข้อมูลอย่างมีประสิทธิภาพสำหรับระบบ MOVIE STREAMING นี้ได้สำเร็จลุล่วงทุกประการตามจุดประสงค์ของโครงการ

ทางผู้จัดทำขอขอบพระคุณ ผศ.ดร. พัฒนพงษ์ ฉันทมิตรโอภาส และ ผศ.ดร.กนกวรรณ อัจฉริยะชาญวณิช ที่ได้ให้เกียรติเป็นที่ปรึกษาโครงการ ได้คอยให้ความช่วยเหลือ ให้คำแนะนำ และขอขอบพระคุณคณาจารย์ในคณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังทุกท่าน ที่คอยให้คำแนะนำในการทำโครงการครั้งนี้ด้วย

นายณัฐดนัย วัฒนาไชย 65070075

นายธนภัทร สหบรรเทิงศิลป์ 65070095

นายรามิล ธรรมจินดา 65070198

สารบัญ

หน้า

บทคัดย่อภาษาไทย	I
กิตติกรรมประกาศ.....	II
สารบัญ	III
สารบัญตาราง.....	IV
สารบัญรูป.....	V
บทที่	
1. บทนำ.....	1
1.1 ความเป็นมาของปัญหา.....	1
1.2 วัตถุประสงค์.....	2
1.3 ขอบเขตของระบบ.....	2
1.4 ประโยชน์ที่คาดว่าจะได้รับ.....	2
1.5 โครงสร้างของรายงาน.....	2
2. ทฤษฎีที่เกี่ยวข้องและเทคโนโลยีที่ได้นำมาประกอบการศึกษาและพัฒนาาระบบ.....	3
2.1 สถาปัตยกรรมฐานข้อมูล.....	3
2.2 ระบบการทำงานของระบบฐานข้อมูล	6
2.3 ฐานข้อมูลเชิงสัมพันธ์.....	8
2.4 Normalization.....	10

3. ขั้นตอนการดำเนินโครงการ.....	14
3.1 ความต้องการของระบบ.....	14
3.2 การวิเคราะห์และออกแบบฐานข้อมูล.....	15
3.3 หลักการทำงานของระบบ.....	31
4. ผลการดำเนินงาน.....	37
4.1 ส่วนของผู้ดูแลระบบ.....	37
4.2 ส่วนของผู้ใช้ระบบ.....	40
5. สรุปผลการดำเนินงานของโครงการ.....	43
5.1 สรุปผล.....	43
5.2 ประโยชน์ที่ได้รับ.....	43
5.3 ปัญหาและอุปสรรคในการดำเนินงาน.....	43
5.4 ข้อเสนอแนะ.....	43

สารบัญรูป

หน้า

รูปที่

1 แผนภาพ Entity Relationship Diagram.....	15
2 นอร์มัลฟอร์มระดับที่ 1 รีเลชั่น USERS.....	16
3 นอร์มัลฟอร์มระดับที่ 1 รีเลชั่น DEVICES.....	16
4 นอร์มัลฟอร์มระดับที่ 1 รีเลชั่น DEVICE_TYPES.....	16
5 นอร์มัลฟอร์มระดับที่ 1 รีเลชั่น PLAYLISTS.....	17
6 นอร์มัลฟอร์มระดับที่ 1 รีเลชั่น PLAYLIST_MOVIES.....	17
7 นอร์มัลฟอร์มระดับที่ 1 รีเลชั่น MOVIES.....	17
8 นอร์มัลฟอร์มระดับที่ 1 รีเลชั่น RATINGS.....	18
9 นอร์มัลฟอร์มระดับที่ 1 รีเลชั่น GENRES.....	18
10 นอร์มัลฟอร์มระดับที่ 1 รีเลชั่น MOVIE_GENRES.....	18
11 นอร์มัลฟอร์มระดับที่ 2 รีเลชั่น USERS.....	19
12 นอร์มัลฟอร์มระดับที่ 2 รีเลชั่น DEVICE_TYPES.....	19
13 นอร์มัลฟอร์มระดับที่ 2 รีเลชั่น PLAYLISTS.....	20
14 นอร์มัลฟอร์มระดับที่ 2 รีเลชั่น MOVIES.....	21
15 นอร์มัลฟอร์มระดับที่ 2 รีเลชั่น GENRES.....	22
16 นอร์มัลฟอร์มระดับที่ 3 รีเลชั่น USERS.....	23
17 นอร์มัลฟอร์มระดับที่ 3 รีเลชั่น DEVICE_TYPES.....	24

รูปที่

18 นอร์มัลฟอร์มระดับที่ 3 รีเลชั่น PLAYLISTS.....	25
19 นอร์มัลฟอร์มระดับที่ 3 รีเลชั่น MOVIES.....	26
20 นอร์มัลฟอร์มระดับที่ 3 รีเลชั่น GENRES.....	27
21 การใช้ phpMyAdmin สร้างฐานข้อมูล.....	31
22 การใช้ phpMyAdmin สร้างฐานข้อมูล.....	32
23 การใช้ phpMyAdmin เลือกฐานข้อมูล.....	32
24 การใช้ phpMyAdmin จัดการฐานข้อมูล.....	33
25 การใช้ phpMyAdmin จัดการฐานข้อมูล.....	33
26 การใช้ phpMyAdmin จัดการฐานข้อมูล.....	34
27 ฐานข้อมูลหนึ่งและรูปภาพ.....	35
28 โฟล์เดอร์ที่ใช้เก็บหนังและโปสเตอร์.....	36
29 การกำหนด path ตอนดึงข้อมูล.....	36
30 ตัวอย่างการแสดงผลลัพธ์.....	36
31 การสร้างตาราง MOVIES.....	37
32 การสร้างตาราง USERS.....	37
33 การสร้างตาราง PLAYLISTS.....	38
34 การสร้างตาราง MOVIE_GENRES.....	38
35 การสร้าง USER ชื่อ ramin.....	38
36 การกำหนดสิทธิ์ให้ USER ramin สามารถ SELECT ตาราง MOVIES.....	39

รูปที่

37 การกำหนดสิทธิ์ให้ USER ramin สามารถ SELECT, INSERT, UPDATE, DELETE ตาราง PLAYLISTS.....	39
38 การกำหนดสิทธิ์ให้ USER ramin สามารถ SELECT คอลัมน์ Genre จากตาราง GENRES.....	39
39 การกำหนดสิทธิ์ให้ USER Thanapat สามารถทำได้ทุกอย่างกับฐานข้อมูล movies_project.....	39
40 การยกเลิกสิทธิ์ USER Thanapat ไม่สามารถทำการใดๆกับตาราง ratings.....	39
41 การใช้คำสั่ง SELECT แสดงหนังที่ผู้ใช้งานกดถูกใจมากที่สุด 3 อันดับแรก.....	40
42 ผลลัพธ์จากคำสั่งรูปที่ 41.....	40
43 การใช้คำสั่ง SELECT แสดง Genre ที่มากที่สุด 3 อันดับแรก.....	40
44 ผลลัพธ์จากคำสั่งรูปที่ 43.....	41
45 การใช้คำสั่ง SELECT แสดงจำนวน Device ในแต่ละ subscriptionType.....	41
46 ผลลัพธ์จากคำสั่งรูปที่ 45.....	41
47 การใช้คำสั่ง SELECT แสดงหนังหมวด Action, Sci-fi.....	42
48 ผลลัพธ์จากคำสั่งรูปที่ 47.....	42
49 การใช้คำสั่ง INSERT เพิ่มข้อมูลประเภทหนังในตาราง Genres.....	42
50 ผลลัพธ์จากคำสั่งรูปที่ 49.....	42

สารบัญตาราง

หน้า

ตารางที่

1 ตารางข้อมูลในระบบ.....	28
2 พจนานุกรมข้อมูลตาราง USERS.....	28
3 พจนานุกรมข้อมูลตาราง MOVIES.....	29
4 พจนานุกรมข้อมูลตาราง MOVIE_GENRES.....	29
5 พจนานุกรมข้อมูลตาราง GENRES.....	29
6 พจนานุกรมข้อมูลตาราง DEVICES.....	29
7 พจนานุกรมข้อมูลตาราง DEVICE_TYPES.....	30
8 พจนานุกรมข้อมูลตาราง PLAYLISTS.....	30
9 พจนานุกรมข้อมูลตาราง PLAYLISTS_MOVIES.....	30
10 พจนานุกรมข้อมูลตาราง RATINGS.....	30

บทที่ 1

บทนำ

1.1 ความเป็นมาของปัญหา

ในการพัฒนาระบบสารสนเทศเรื่อง “MOVIES STREAMING” นี้ ได้พบปัญหาหลายประการที่มีผลกระทบต่อประสิทธิภาพและประโยชน์ของระบบ ซึ่งปัญหาเหล่านี้มาจากความต้องการในการจัดการข้อมูลและการเข้าถึงข้อมูลของระบบ MOVIES STREAMING ร่วมกับการทำงานของฐานข้อมูลที่สำคัญในการดำเนินงานของระบบดังนี้

ปัญหาการจัดการข้อมูล ฐานข้อมูลที่ใช้ในระบบ MOVIES STREAMING ไม่ได้ถูกออกแบบให้สามารถจัดการข้อมูลวิดีโออย่างมีประสิทธิภาพ การจัดเก็บและเรียกดูข้อมูลมีความซับซ้อนและช้า ซึ่งเป็นอุปสรรคในการให้บริการที่รวดเร็วและตอบสนองต่อความต้องการของผู้ใช้

ปัญหาความปลอดภัยของข้อมูล ระบบฐานข้อมูลในโครงการนี้มีข้อบกพร่องในเรื่องความปลอดภัยของข้อมูล ทำให้ข้อมูลที่ถูกจัดเก็บนั้นอาจจะเสี่ยงต่อการนำข้อมูลไปใช้ไม่ถูกวิธี ซึ่งเป็นปัญหาสำคัญที่ต้องแก้ไขเพื่อให้ข้อมูลในระบบนี้มีความปลอดภัยมากยิ่งขึ้น

ปัญหาการปรับปรุงและพัฒนาระบบ ฐานข้อมูลในระบบนี้มีข้อจำกัดในการปรับปรุงและพัฒนาระบบให้ทันสมัย การปรับปรุงหรือเพิ่มเติมความสามารถของระบบมีความซับซ้อนและยุ่งยาก เนื่องจากโครงสร้างและองค์ประกอบของฐานข้อมูลไม่ได้ถูกวางแผนให้เหมาะสมสำหรับการพัฒนาระบบใหม่ ทำให้การพัฒนาต้องใช้เวลาและทรัพยากรมากมาย

การแก้ไขปัญหาดังกล่าวจะเป็นสิ่งสำคัญในการพัฒนาระบบ MOVIES STREAMING ให้สามารถทำงานได้อย่างมีประสิทธิภาพและมีประโยชน์สำหรับผู้ใช้และองค์กรที่ใช้งานระบบนี้

1.2 วัตถุประสงค์

ในการพัฒนาระบบสารสนเทศเรื่อง “MOVIES STREAMING” มีวัตถุประสงค์เพื่อ มุ่งเน้นพัฒนาระบบการสตรีมวิดีโอให้มีประสิทธิภาพและมีฐานข้อมูลที่มีประโยชน์อย่างมากสำหรับผู้ใช้และองค์กร และมีเป้าหมายที่ชัดเจนในการเพิ่มประสิทธิภาพในการจัดการข้อมูล เพิ่มระบบความปลอดภัยของข้อมูล พัฒนาให้บริการมีประสิทธิภาพ ปรับปรุงและพัฒนาระบบให้ทันสมัยและทำให้ระบบเป็นที่ยอมรับตามความต้องการของผู้ใช้และองค์กรทั้งในระยะยาวและระยะสั้น

1.3 ขอบเขตของระบบ

ขอบเขตของรายงานเกี่ยวกับโครงการ “VIDEO STREAMING SYSTEM” ถูกกำหนดเพื่อให้ความชัดเจนและเน้นไปที่ประเด็นหลักที่จะถูกนำเสนอในรายงานนี้ โดยมีขอบเขตดังนี้

1. การอธิบายฐานข้อมูล: รายงานจะอธิบายโครงสร้างและการทำงานของฐานข้อมูลที่ใช้ในระบบ MOVIES STREAMING รวมถึงการจัดเก็บข้อมูลและความสัมพันธ์ของข้อมูล
2. วิธีการดำเนินการและเทคโนโลยีที่ใช้: รายงานจะอธิบายวิธีการดำเนินการและเทคโนโลยีที่ถูกนำมาใช้ในการพัฒนาระบบ MOVIES STREAMING พร้อมฐานข้อมูล

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1. การเพิ่มประสิทธิภาพในการบริหารจัดการข้อมูล
2. ปรับปรุงประสิทธิภาพการให้บริการ
3. การปรับปรุงและพัฒนาระบบให้ทันสมัย
4. การประหยัดเวลาและทำงานได้มีประสิทธิภาพสูงขึ้น

1.5 โครงสร้างของรายงาน

รายงานฉบับนี้ มีโครงสร้างของรายงานดังต่อไปนี้

บทที่ 1 กล่าวถึงความเป็นมาและความสำคัญของปัญหา วัตถุประสงค์ ขอบเขตของระบบ และประโยชน์ที่คาดว่าจะได้รับ

บทที่ 2 ทฤษฎีที่เกี่ยวข้องและเทคโนโลยีที่ได้นำมาใช้ประกอบการศึกษาและพัฒนาแอปพลิเคชัน

บทที่ 3 ขั้นตอนการดำเนินโครงการ

บทที่ 4 ผลการดำเนินงาน

บทที่ 5 สรุปผลการดำเนินงานของโครงการ

บทที่ 2

ทฤษฎีที่เกี่ยวข้องและเทคโนโลยีที่ได้นำมาใช้ประกอบการศึกษาและพัฒนา ระบบ

2.1 สถาปัตยกรรมข้อมูล

คือ การออกแบบและวางระบบการจัดการข้อมูลในทุกกระบวนการ หรือหมายถึงการวาง workflow การไหลของข้อมูลภายในองค์กร ซึ่งอาจรวมทั้งงานฮาร์ดแวร์ ซอฟต์แวร์ รวมไปถึงหน้าที่ในการจัดการข้อมูลด้วย คล้ายๆ กับ ‘สถาปนิก’ ที่ออกแบบอาคาร วางประตู กำหนดห้องต่างๆ ตามหน้าที่ (function) ให้สะดวกและเป็นระบบ โดย Data Architecture มีหน้าที่ในการกำหนดโครงสร้างหรือ infrastructure สำหรับจัดการกับข้อมูลตั้งแต่ต้นน้ำยันปลายน้ำว่า ข้อมูลจะได้มาจากแหล่งไหน ถูกเก็บไว้ที่ใดและในรูปแบบใด จัดการกับข้อมูลอย่างไร จนไปถึงปลายน้ำคือการใช้ข้อมูล

สถาปัตยกรรมฐานข้อมูล เป็นมุมมองแนวความคิด ที่ใช้ในการอธิบายถึงรูปแบบและโครงสร้าง ของข้อมูลในระบบฐาน ฐานข้อมูลโดยไม่ขึ้นอยู่กับโครงสร้างจริงของระบบฐานข้อมูลนั้น ๆ ผู้ใช้ฐานข้อมูลจะมองข้อมูลนี้ในแง่มุมมองหรือวิธีที่แตกต่างกันไปตามจุดประสงค์ของการประยุกต์ใช้งาน โดยไม่จำเป็นต้องสนใจว่ามีลักษณะการจัดเก็บข้อมูลแท้จริงเป็นเช่นไร ระบบฐานข้อมูลจะทำการซ่อนรายละเอียดไว้ ฐานข้อมูลมีจุดเด่นเหนือแฟ้มข้อมูลทั่วไปตรงที่ความเป็นอิสระข้อมูลเพราะโปรแกรมประยุกต์ที่ผู้ใช้เขียนขึ้นไม่ขึ้นกับโครงสร้างของข้อมูลที่เปลี่ยนไป ด้วยเหตุนี้ได้มีการกำหนดมาตรฐานขึ้นโดย The Standards Planning and Requirements Committee (SPARC) ของ American National Standards institute (ANSI) ที่เรียกว่า ANSI/SPARC จะถูกแบ่งออกเป็น 3 ระดับ เรียกว่าสถาปัตยกรรมสามระดับ (Three-level Architecture) ประกอบด้วย

- 1) ระดับภายใน (Internal Level)
- 2) ระดับความคิด (Concept Level)
- 3) ระดับภายนอก (External Level)

2.1.1 ระดับภายใน (Internal level)

เป็นระดับที่มองถึงวิธีการจัดเก็บข้อมูลในระดับฟิสิกคอลว่ามีรูปแบบและโครงสร้างการจัดเก็บข้อมูลอย่างไร ซึ่งมีหน้าที่ ในการจัดเก็บข้อมูลจริง ๆ ในหน่วยความจำ เช่น ดิสก์ ว่าอยู่ตำแหน่งใด รวมทั้งที่เกี่ยวกับดัชนี (Index) ซึ่งในระดับนี้จะเป็นหน้าที่ของผู้ดูแลระบบตัดสินใจในระดับกายภาพว่าจะเก็บข้อมูลด้วยหน่วยความจำแบบใด จะมีการดูแลรักษาอย่างไร ซึ่ง โดยมากผู้ใช้ทั่วไปไม่มีสิทธิ์เข้ามาอยู่ในระดับนี้

อาจจะกล่าวได้ว่า สคีมาระดับภายใน (Internal Schemas Level) พิจารณาการจัดการระบบการเก็บข้อมูลจริง อธิบาย ฐานข้อมูลในการเก็บทางกายภาพจริง ๆ มองข้อมูลโดยมุมมองของระบบจัดการฐานข้อมูล ใช้โครงสร้างข้อมูล (Data Structure) และการจัดระเบียบแฟ้ม (File Organization) ในการอธิบาย และจะทำงานร่วมกับระบบปฏิบัติการ (Operating System) ในการเก็บข้อมูลลงที่หน่วยเก็บสำรอง (Secondary Storage) เราเรียกโครงสร้างระดับภายในว่า physical schema ซึ่งได้จากการ map โครงสร้างข้อมูลระดับแนวคิดลงสู่ข้อมูล เช่น ดิสก์ เป็นต้น ส่วนใหญ่โครงสร้างจะเป็น index หรือ tree

2.1.2. ระดับความคิด (Conceptual level)

เป็นระดับที่อยู่ถัดขึ้นมาได้แก่ระดับของการมองความสัมพันธ์ของข้อมูลในระบบฐานข้อมูลว่ามีการจัดเก็บข้อมูลอย่างไร มีความสัมพันธ์กับข้อมูลอื่นอย่างไรบ้าง รวมทั้งกฎเกณฑ์ต่าง ๆ เกี่ยวกับข้อมูล เช่น กฎเกณฑ์ของตัวข้อมูลความปลอดภัย (Security) และความคงสภาพของข้อมูล (Integrity) ซึ่งยังไม่ใช้รูปแบบหรือวิธีการจัดเก็บจริงในในสื่อบันทึกข้อมูลเป็นเพียงแค่มุมมองในแนวความคิดว่า ข้อมูลจะมีการจัดเก็บอย่างไรเท่านั้น ผู้ที่มีสิทธิ์จะใช้ข้อมูลในระดับนี้คือโปรแกรมเมอร์ (Programmer) หรือผู้บริหารฐานข้อมูล (Database Administrator) เริ่มจากผู้ออกแบบ หรือผู้ดูแลระบบเริ่มวิเคราะห์ความต้องการข้อมูลขององค์กรว่าประเภทใดบ้างที่จะจัดเก็บ ควรจะเก็บอะไรบ้างและระหว่างข้อมูลควรมีความสัมพันธ์อย่างไรบ้างจากนั้นจึงนำมาเขียนเป็นสคีมา (Schema) ซึ่งถือเป็นจุดสิ้นสุด ของระดับแนวความคิด สคีมา (Schema) สามารถอธิบายว่าฐานข้อมูลนั้น ๆ สร้างขึ้นมาประกอบด้วยอะไรบ้างแต่ละเอนทิตี ประกอบด้วยข้อมูลอะไรบ้าง มีลักษณะเป็นอย่างไรและมีความสัมพันธ์อย่างไร อาจจะกล่าวได้ว่าสคีมาระดับเชิงมโนภาพ (Conceptual Schemas Level) จะเป็นตัวที่ใช้เชื่อมระหว่างสคีมาระดับภายนอกกับสคีมาระดับภายใน อธิบายฐานข้อมูลในรายละเอียดโดยรวมทั้งหมด เพื่อเชื่อมกับสิ่งที่ผู้ใช้มองรูปแบบข้อมูล ความสัมพันธ์เงื่อนไขต่าง ๆ รวมถึงความมั่นคงและความถูกต้องของข้อมูลจะถูกเก็บไว้ด้วยแต่จะไม่ลงลึกถึงการเก็บในเครื่อง เราจะเรียกในส่วนนี้ว่า logical schema หรือ schema ซึ่งเป็นโครงสร้างข้อมูลที่น่าเสนอตามรูปแบบของ data model ถ้าเป็น relational database ข้อมูลในระดับ create table จะหมายถึงตารางทุกตารางใน model

2.1.3. ระดับภายนอก (External level)

คือ หน้าต่างหรือวิว(View) ที่ผู้ใช้งานนอกมีสิทธิเข้าไปใช้ได้วิว(View) คือ ส่วนของข้อมูลที่ใช้ทั่วไปมีความสนใจแล มีสิทธิที่เข้านามาใช้ได้จาก สคีมา (Concept Schema) เราสามารถเรียกสิ่งที่ใช้อธิบายวิวข้อมูลที่ถูกดึงมาจากฐานข้อมูลที่อยู่ใน ระดับแนวคิด (Conceptual) นี้ว่า External Schema หรือ Subschema หรือ view ซึ่งในระดับนี้โปรแกรมจะเห็นเพียงโครงสร้างข้อมูลบางส่วน เช่น กรณีของ relational database โปรแกรมจะเห็น view เป็นเพียงบางส่วนของตาราง หรือ บางส่วนของตารางหลายตารางมารวมกันเป็นหนึ่ง view โดยจะต้องไม่ copy หรือ duplicate data จาก table มาไว้ที่ view ระดับภายนอกเป็นระดับที่ใกล้กับผู้ใช้มากที่สุด เป็นสิ่งที่ผู้ใช้คิดเกี่ยวกับข้อมูลจะอธิบายถึงวิว(View) ที่ผู้ใช้สนใจข้อมูลที่ เก็บจริงอาจมีมากกว่า ที่ผู้ใช้ต้องการและข้อมูลตัวเดียวกันผู้ใช้อาจมองไม่เหมือนกัน เช่น ข้อมูลวันที่ (ผู้ใช้คนหนึ่งอาจมอง เป็น วัน/ เดือน ปีอีกคนมองเป็น เดือน/ วัน/ปีก็ได้) นอกจากนั้นสิ่งที่ผู้ใช้มองเห็นอาจไม่ได้เก็บจริงในเครื่องแต่ได้จากการคำนวณออกมาส่วนนี้เองจะถูกแปลโดยระบบจัดการฐานข้อมูลเก็บไว้ในพจนานุกรมข้อมูล(Data dictionary) นอกจากนั้นสถาปัตยกรรมทั้งสามระดับดังกล่าวยังมีความเป็นอิสระของข้อมูลที่เห็นได้คือ สคีมาระดับภายนอกจะมีรูปแบบที่คงที่ไม่เปลี่ยนแปลงไป แม้จะมีการเปลี่ยนสคีมาระดับเชิงมโนภาพ เช่น มีการเพิ่มรูปแบบข้อมูลใหม่หรือมีความสัมพันธ์ใหม่ๆ เพิ่มขึ้น เรียกว่า ความไม่พึ่งพิงทางตรรกะ(Logical Data Independence) และสคีมาระดับเชิงมโนภาพ ก็เช่นกันจะมีรูปแบบที่คงที่แม้จะมีการเปลี่ยนแปลงทางกายภาพ เช่น เปลี่ยนวิธีเข้าถึงข้อมูล (Access Method) หรือลำดับ ของข้อมูลที่เก็บอยู่จริงเปลี่ยนไป เรียกว่าความไม่พึ่งพิงทางกายภาพ (Physical Data Independence) จุดประสงค์หลักของ สถาปัตยกรรมทั้งสามระดับ คือการที่ต้องการให้ข้อมูลเป็น อิสระจากซอฟต์แวร์ทั้งที่เป็นของผู้ใช้และระบบ กล่าวคือ การเปลี่ยนแปลงใด ๆ ต่อทั้งฮาร์ดแวร์และซอฟต์แวร์ไม่ควรจะมีผลต่อข้อมูลนั้น ๆ

2.2 ระบบการทำงานของระบบฐานข้อมูล

ระบบการทำงานของระบบฐานข้อมูล หรือ DBMS คือ ระบบการจัดการฐานข้อมูล หรือซอฟต์แวร์ที่ดูแลจัดการเกี่ยวกับฐานข้อมูล โดยอำนวยความสะดวกให้แก่ผู้ใช้ทั้งในด้านการสร้าง การปรับปรุงแก้ไข การเข้าถึงข้อมูล และการจัดการเกี่ยวกับระบบแฟ้มข้อมูลทางกายภาพ ภายในฐานข้อมูลซึ่งต่างไปจากระบบแฟ้มข้อมูลคือ หน้าที่เหล่านี้จะเป็นของโปรแกรมเมอร์ ในการติดต่อฐานข้อมูลไม่ว่าจะด้วยการใช้คำสั่งในกลุ่ม DML หรือ DDL หรือ จะด้วยโปรแกรมต่างๆ ทุกคำสั่งที่ใช้กระทำกับฐานข้อมูลจะถูกโปรแกรม DBMS นำมาแปล (Compile) เป็นการกระทำต่างๆ ภายใต้อำนาจคำสั่งนั้นๆ เพื่อนำไปกระทำกับตัวข้อมูลในฐานข้อมูลต่อไป

DBMS ถูกพัฒนาขึ้นเพื่อแก้ไขปัญหาด้าน Data Independence ที่ไม่มีในระบบแฟ้มข้อมูล ทำให้มีความเป็นอิสระจากทั้งส่วนของฮาร์ดแวร์ และข้อมูลภายในฐานข้อมูลกล่าวคือโปรแกรม DBMS นี้จะมีการทำงานที่ไม่ขึ้นอยู่กับรูปแบบ (Platform) ของตัวฮาร์ดแวร์ ที่นำมาใช้กับระบบฐานข้อมูลรวมทั้งมีรูปแบบในการอ้างถึงข้อมูลที่ไม่ขึ้นอยู่กับโครงสร้างทางกายภาพของข้อมูลด้วยการใช้ Query Language ในการติดต่อกับข้อมูลในฐานข้อมูลแทนคำสั่งภาษาคอมพิวเตอร์ในยุคที่ 3 ส่งผลให้ผู้ใช้สามารถเรียกใช้ข้อมูลจากฐานข้อมูลได้โดยไม่ต้องทราบถึงประเภทหรือขนาดของข้อมูลนั้นหรือสามารถกำหนดลำดับที่ของฟิลด์ ในการกำหนดการแสดงผลได้โดยไม่ต้องคำนึงถึงลำดับที่จริงของฟิลด์นั้น

สำหรับส่วนการทำงานต่างๆ ภายในดีบีเอ็มเอสที่ทำหน้าที่แปลคำสั่งไปเป็นการปฏิบัติการต่าง ๆ กับข้อมูลนั้น ประกอบด้วยส่วนการปฏิบัติการดังนี้

- 1) ตัวจัดการฐานข้อมูล (Database Manager) : เป็นส่วนที่ทำหน้าที่กำหนดการกระทำต่าง ๆ ให้กับส่วน File Manager เพื่อไปกระทำกับข้อมูลที่เก็บอยู่ในฐานข้อมูล (ตัวจัดการไฟล์ เป็นส่วนที่ทำหน้าที่บริหารจัดการกับข้อมูลที่เก็บอยู่ในฐานข้อมูลในระดับกายภาพ)
- 2) ตัวประมวลผลสอบถาม (Query Processor) : เป็นส่วนที่ทำหน้าที่แปลงกำหนดคำสั่งของ ภาษาสอบถาม (Query Language) ให้อยู่ในรูปแบบของคำสั่งที่ตัวจัดการฐานข้อมูลเข้าใจ
- 3) ตัวแปลภาษาจัดดำเนินการข้อมูลล่วงหน้า (Data Manipulation Language Precompiler) : เป็นส่วนที่ทำหน้าที่แปลประโยคคำสั่งของกลุ่มคำสั่งในดีเอ็มแอล ให้อยู่ในรูปแบบที่ส่วนรหัสเชิงวัตถุของโปรแกรมแอปพลิเคชัน ใช้นำเข้าเพื่อส่งต่อไปยังส่วนตัวจัดการฐานข้อมูลในการแปลประโยคคำสั่งของกลุ่มคำสั่งของดีเอ็มแอล ของส่วนตัวแปลภาษาจัดดำเนินการข้อมูลล่วงหน้าจะต้องทำงานร่วมกับส่วนตัวประมวลผลข้อคำถาม
- 4) ตัวแปลภาษานิยามข้อมูลล่วงหน้า (Data Definition Language Precompiler) : เป็นส่วนที่ทำหน้าที่แปลประโยคคำสั่งของกลุ่มคำสั่งในภาษานิยามข้อมูล ให้อยู่ในรูปแบบของเมตาเดตา (MetaData) ที่เก็บอยู่ในส่วนพจนานุกรมข้อมูล (Data Dictionary) ของฐานข้อมูล (เมตาเดตา คือ รายละเอียดที่บอกถึงโครงสร้างต่าง ๆ ของข้อมูล)

- 5) รหัสจุดหมายของโปรแกรมแอปพลิเคชัน (Application Programs Object Code) : เป็นส่วนที่ทำหน้าที่แปลงคำสั่งต่าง ๆ ของโปรแกรม รวมทั้งคำสั่งในกลุ่มคำสั่งภาษาจัดดำเนินการข้อมูล หรือ ดีเอ็มแอลที่ส่งต่อมาจากส่วนตัวแปลภาษาจัดดำเนินการข้อมูลล่วงหน้าให้อยู่ในรูปแบบของรหัสจุดหมาย (Object Code) ที่จะส่งต่อไปให้ตัวจัดการฐานข้อมูลเพื่อกระทำกับข้อมูลในฐานข้อมูล

6)

หน้าที่ของ DBMS

- 1) ทำหน้าที่แปลงคำสั่งที่ใช้จัดการกับข้อมูลภายในฐานข้อมูลให้อยู่ในรูปแบบที่ข้อมูลเข้าใจ
- 2) ทำหน้าที่ในการนำคำสั่งต่างๆ ซึ่งได้รับการแปลแล้วไปส่งให้ฐานข้อมูลทำงาน เช่น การเรียกใช้ข้อมูล (Retrieve) การจัดเก็บข้อมูล (Update) การลบข้อมูล (Delete) หรือ การเพิ่มข้อมูลเป็นต้น (Add) ฯลฯ
- 3) ทำหน้าที่ป้องกันความเสียหายที่จะเกิดขึ้นกับข้อมูลภายในฐานข้อมูล โดยจะคอยตรวจสอบว่าคำสั่งใดที่สามารถทำงานได้และคำสั่งใดที่ไม่สามารถทำได้
- 4) ทำหน้าที่รักษาความสัมพันธ์ของข้อมูลภายในฐานข้อมูลให้มีความถูกต้องอยู่เสมอ
- 5) ทำหน้าที่เก็บรายละเอียดต่าง ๆ ที่เกี่ยวข้องกับข้อมูลภายในฐานข้อมูลไว้ใน data dictionary ซึ่งรายละเอียดเหล่านี้มักจะถูกเรียกว่า "ข้อมูลของข้อมูล" (Meta Data)
- 6) ทำหน้าที่ควบคุมให้ฐานข้อมูลทำงานได้อย่างถูกต้องและมีประสิทธิภาพ

2.3 ฐานข้อมูลเชิงสัมพันธ์

ฐานข้อมูลเชิงสัมพันธ์ หมายความว่า จะมีการจัดเก็บข้อมูลในลักษณะที่เป็นกลุ่มของข้อมูลที่มีความสัมพันธ์กัน ในฐานข้อมูลหนึ่งๆ สามารถที่จะมีตารางตั้งแต่ 1 ตารางเป็นต้นไป และในแต่ละตารางนั้นก็อาจมีได้หลายคอลัมน์ (Column) หลายแถว (Row) ตัวอย่างเช่น เราต้องการเก็บข้อมูลพนักงาน ในตารางของข้อมูลพนักงานก็จะประกอบด้วยคอลัมน์ ที่อธิบายชื่อ นามสกุล ที่อยู่ เงินเดือน แผนกที่สังกัด

เป็นต้น และในตารางนั้น ก็สามารถที่จะมีข้อมูลพนักงานได้มากกว่า 1 คน (Row) และตารางข้อมูลพนักงานนั้นอาจจะมีความสัมพันธ์กับตารางอื่น เช่น ตารางที่เก็บชื่อและจำนวนบุตรของพนักงาน

ฐานข้อมูลเชิงสัมพันธ์ถูกออกแบบมาเพื่อลดความซ้ำซ้อนของการเก็บข้อมูล และสามารถเรียกใช้ข้อมูลได้อย่างมีประสิทธิภาพ โดยมีหลักดังนี้

- 1) ตารางจะต้องมีชื่อไม่ซ้ำกัน
- 2) แต่ละฟิลด์จะบรรจุประเภทข้อมูลเพียงชนิดเดียวเท่านั้นแน่นอน
- 3) ข้อมูลในแต่ละเรคอร์ดจะต้องไม่ซ้ำกัน

นอกจากนี้แต่ละตารางยังสามารถเรียกได้อีกอย่างว่ารีเลชัน (Relation) แถวแต่ละแถวภายในตารางเรียกว่าทูเปิล (Tuple) และคอลัมน์เรียกว่าแอททริบิวต์ (Attribute)

จุดเด่นของข้อมูลเชิงสัมพันธ์

- 1) ง่ายต่อการเรียนรู้ และการนำไปใช้งาน ทำให้เห็นภาพข้อมูลชัดเจน
- 2) ภาษาที่ใช้จัดการข้อมูลเป็นแบบซีเควล ซึ่งมีประสิทธิภาพสูงเข้าใจง่าย
- 3) การออกแบบระบบมีทฤษฎีรองรับ สามารถลดความซ้ำซ้อนของข้อมูลได้

กฎที่เกี่ยวข้องกับคีย์ในฐานข้อมูลเชิงสัมพันธ์

- 1) กฎความบูรณภาพของเอนทิตี (The Entity Integrity Rule)

กฎนี้ระบุไว้ว่าแอททริบิวต์ใดที่เป็นคีย์หลัก ข้อมูลในแอททริบิวต์นั้นจะเป็นค่าว่าง (Null) ไม่ได้ ความหมายของการเป็นค่าว่างไม่ได้ (Not Null) หมายความว่า ข้อมูลของแอททริบิวต์ที่เป็นคีย์หลัก จะไม่ทราบค่าที่แน่นอนหรือไม่มีค่าไม่ได้

- 2) กฎความบูรณภาพของการอ้างอิง (The Referential Integrity Rule)

การอ้างอิงข้อมูลระหว่างรีเลชันในฐานข้อมูลเชิงสัมพันธ์จะใช้คีย์นอกของรีเลชันหนึ่งไปตรวจสอบกับค่าของแอททริบิวต์ที่เป็นคีย์หลักของอีกรีเลชันหนึ่งเพื่อเรียกดูข้อมูลอื่นๆที่เกี่ยวข้องหรือค่าของคีย์นอกจะต้องอ้างอิงให้ตรงกับค่าของคีย์หลักได้จึงจะสามารถเชื่อมโยงข้อมูลระหว่างสองรีเลชันได้ สำหรับคีย์นอกจะมีค่าว่างได้หรือไม่ขึ้นอยู่กับกฎเกณฑ์การออกแบบฐานข้อมูล เช่น ในกรณีที่มีรีเลชันพนักงานมี Depno เป็นคีย์นอกอาจจะถูกระบุว่าต้องทราบค่าแต่ในกรณีพนักงานทดลองงานอาจยังไม่มีค่า Depno เพราะยังไม่ได้ถูกบรรจุในกรณีที่มีการลบหรือแก้ไขข้อมูลของแอททริบิวต์ที่เป็นคีย์

หลักซึ่งมีคีย์นอก จากอีกรีเลชันหนึ่งอ้างอิงถึง จะทำการลบหรือแก้ไขข้อมูลได้หรือไม่ ขึ้นอยู่กับการออกแบบฐานข้อมูล ว่าได้ระบุให้แอตทริบิวต์มีคุณสมบัติอย่างไร ซึ่งมีโอกาสเป็นไปได้ 4 ทางเลือก

- a. การลบหรือแก้ไขข้อมูลแบบมีข้อจำกัด (Restrict) การลบหรือแก้ไขข้อมูลจะกระทำได้ เมื่อข้อมูลของคีย์หลักในรีเลชันหนึ่งไม่มีข้อมูลที่ถูกอ้างอิง โดยคีย์นอกของอีกรีเลชันหนึ่งเช่น รหัสแผนก Depno ในรีเลชันDepจะถูกแก้ไขหรือลบทิ้งต่อเมื่อไม่มีพนักงานคนใดสังกัดอยู่ในแผนกนั้น
- b. การลบหรือแก้ไขข้อมูลแบบต่อเรียง (Cascade) การลบหรือการแก้ไขข้อมูล จะทำแบบเป็นลูกโซ่ คือ หากมีการแก้ไขหรือลบข้อมูลของคีย์หลักในรีเลชันหนึ่งระบบจะทำการลบหรือแก้ไขข้อมูลของคีย์นอกในอีกรีเลชันหนึ่งที่ยังอ้างอิงถึงข้อมูลของคีย์หลักที่ถูกลบไปด้วย เช่น ในกรณีที่ยกเลิกแผนก 9 ในEntityแผนก ข้อมูลของพนักงานที่อยู่แผนก 10 ในEntityพนักงาน จะถูกลบออกไปด้วย
- c. การลบหรือแก้ไขข้อมูลโดยเปลี่ยนเป็นค่าว่าง (Nullify) การลบหรือแก้ไขข้อมูลจะทำได้เมื่อมีการเปลี่ยนค่าของคีย์นอกในข้อมูลที่ถูกอ้างอิงให้เป็นค่าว่างเสียก่อน เช่น พนักงานที่อยู่ในแผนกที่ 9 จะถูกเปลี่ยนค่าเป็นค่าว่างก่อนหลังจากนั้น การลบข้อมูลของแผนกที่มีรหัส 9 จะถูกลบทิ้งหรือแก้ไขทันที ภายใน Entity แผนก
- d. การลบหรือแก้ไขข้อมูลแบบใช้ค่าโดยปริยาย (Default) การลบหรือแก้ไขข้อมูลของคีย์หลักสามารถทำได้โดยถ้าหากมีคีย์นอกที่ยังอ้างอิงถึงคีย์หลักที่ถูกลบหรือแก้ไข ก็จะทำให้การปรับค่าของคีย์นอกนั้นโดยปริยาย (Default Value) ที่ถูกกำหนดขึ้นเช่น ในกรณีที่ยกเลิกแผนก 9 ใน Entity แผนก ข้อมูลของพนักงานที่อยู่แผนก 9 ใน Entity พนักงานจะถูกเปลี่ยนค่าเป็น 00 ซึ่งเป็นค่าโดยปริยาย ที่หมายความว่า ไม่ได้สังกัดแผนกใด ไม่เข้าใจกลับไปอ่านใหม่

2.4 Normalization

Normalization (นอร์มัลไลเซชัน) คือ วิธีการหนึ่งที่ช่วยลดความซ้ำซ้อนของข้อมูลจาก Database ซึ่งเป็นวิธีของการกำหนดแอตทริบิวต์ให้กับแต่ละเอนทิตี เพื่อให้ได้โครงสร้างของตารางที่ดี สามารถควบคุมความซ้ำซ้อนของข้อมูลและหลีกเลี่ยงความผิดปกติของข้อมูล ซึ่งการทำนอร์มัลไลเซชันจะทำให้จำนวนข้อมูลนั้นลดน้อยลง แต่จำนวนตารางนั้นจะเพิ่มมากขึ้น

การทำนอร์มัลไลเซชันจะประกอบไปด้วย Normal Form (นอร์มัลฟอร์ม) แบบต่างๆ ที่มีเงื่อนไขการทำให้อยู่ในรูปแบบนอร์มัลฟอร์มที่แตกต่างกันไป ขึ้นอยู่กับผู้ออกแบบฐานข้อมูลว่าต้องการลดความซ้ำซ้อนของข้อมูลให้อยู่ในระดับใด ซึ่งนอร์มัลฟอร์มมีรูปแบบต่างๆ ดังต่อไปนี้

- First Normal Form : 1NF (เฟิร์ส นอร์มัลฟอร์ม) หรือนอร์มัลฟอร์มที่ 1
- Second Normal Form : 2NF (เซค'เคินดู นอร์มัลฟอร์ม) นอร์มัลฟอร์มที่ 2
- Third Normal Form : 3NF (เธิร์ด นอร์มัลฟอร์ม) นอร์มัลฟอร์มที่ 3
- Boyce-Codd Normal Form : BCNF (บอยซ์คอดด์นอร์มัลฟอร์ม)
- Fourth Normal Form : 4NF (ฟอर्थ นอร์มัลฟอร์ม) นอร์มัลฟอร์มที่ 4
- Fifth Normal Form : 5NF (ฟิฟท์ นอร์มัลฟอร์ม) นอร์มัลฟอร์มที่ 5

2.4.1 รูปแบบนอร์มอล ระดับที่ 1 (First Normal Form : 1NF)

เป็นระดับที่ใช้สำหรับการปรับโครงสร้างของข้อมูลของรีเลชันให้มีคุณสมบัติตามนิยาม ดังนี้ คือ นิยาม "รีเลชันใด ๆ จะมีคุณสมบัติอยู่ในรูปแบบนอร์มอล ระดับที่ 1 ก็ต่อเมื่อ ทุกแอตทริบิวต์ในแต่ ละทูเพิลมีค่าของข้อมูลเพียงค่าเดียว คือต้องไม่มีค่ากลุ่มข้อมูลที่ซ้ำกัน (Repeating Group)"

2.4.2 รูปแบบนอร์มอล ระดับที่ 2 (Second Normal Form : 2NF)

เป็นการแก้ไขปัญหาความซ้ำซ้อนของข้อมูล ที่ปรากฏในรีเลชันที่ผ่านการนอร์มอลไลซ์ ระดับที่ 1 โดยรูปแบบนอร์มอล ระดับที่ 2 จะต้องมียุทธศาสตร์ดังนี้ คือ นิยาม "รีเลชันใด ๆ จะมีคุณสมบัติอยู่ในรูปแบบนอร์มอล ระดับที่ 2 ก็ต่อเมื่อ

- 1) รีเลชันนั้นมีคุณสมบัติอยู่ในรูปแบบนอร์มอล ระดับที่ 1

- 2) ทุกแอตทริบิวต์ที่ไม่ใช่คีย์หลัก ต้องมีความสัมพันธ์กับแอตทริบิวต์ที่เป็นคีย์หลัก หรือทุกแอตทริบิวต์ที่ประกอบกันเป็นคีย์หลัก มีใช้ส่วนหนึ่งส่วนใดของคีย์หลัก (เป็นกรณีที่คีย์หลักนั้น มีคีย์ร่วม (Composition Key) คือ มีหลายแอตทริบิวต์ประกอบกันเป็นคีย์หลัก)"

2.4.3 รูปแบบนอร์มอล ระดับที่ 3 (Third Normal Form : 3NF)

จากรีเลชันนักศึกษาจะเห็นว่าโครงสร้างเป็นไปตามนิยามของรูปแบบนอร์มอล ระดับที่ 2 จะเห็นว่ายังปรากฏข้อมูลซ้ำกัน "Transitive Dependency" ดังนั้นรีเลชันนี้จึงไม่อยู่ในรูปแบบ นอร์มอลระดับที่ 3 และอาจก่อให้เกิดปัญหาความผิดพลาด จึงต้องแยกแอตทริบิวต์ที่ไม่ใช่คีย์หลัก หรือส่วนใดส่วนหนึ่งของคีย์หลัก ที่ก่อให้เกิดปัญหาออกมา โดยใช้นิยามของรูปแบบนอร์มอล ระดับที่ 3 ดังนี้ นิยาม "รีเลชันใด ๆ จะมีคุณสมบัติอยู่ในรูปแบบนอร์มอล ระดับที่ 3 ก็ต่อเมื่อ

- 1) รีเลชันนั้นมีคุณสมบัติอยู่ในรูปแบบนอร์มอล ระดับที่ 2 แล้ว
- 2) ทุกแอตทริบิวต์ที่ไม่ใช่คีย์หลักในรีเลชัน จะต้องไม่ขึ้นกับแอตทริบิวต์อื่น ๆ ที่ไม่ใช่คีย์หลัก หรืออาจกล่าวว่า ทุกแอตทริบิวต์ที่ไม่ใช่คีย์หลักไม่มีสิทธิในการระบุค่าของแอตทริบิวต์อื่น ที่ไม่ใช่คีย์หลัก หรืออาจกล่าวอีกนัยหนึ่งได้ว่า รีเลชันนั้นต้องไม่มีความสัมพันธ์ระหว่างแอตทริบิวต์ เป็นแบบ Transitive เกิดขึ้น"

2.4.4 รูปแบบนอร์มอล ของบอยส์และคอดด์ (Boyce/Codd Normal Form : BCNF)

ในทางปฏิบัติการจัดทำรีเลชันจนกระทั่งคุณสมบัติอยู่ในรูปแบบนอร์มอล ระดับที่ 3 ก็สามารถ จัดปัญหาความซ้ำซ้อนของข้อมูลลงได้จนเกือบหมดแล้ว แต่ก็อาจมีปัญหาคือความซ้ำซ้อนของ ข้อมูลได้อีก แม้จะพบได้ค่อนข้างน้อยมากก็ตาม ดังนั้น เพื่อให้รีเลชันมีความถูกต้อง ชัดเจน และรัดกุมมากยิ่งขึ้น จึงจำเป็นต้องทำให้รีเลชัน อยู่ในรูปแบบนอร์มอลของบอยส์และคอดด์(Boyce/Codd Normal Form: BCNF) ตามนิยาม ดังนี้ นิยาม "รีเลชันใด ๆ จะมีคุณสมบัติอยู่ในรูปแบบนอร์มอลของบอยส์และคอดด์ ก็ต่อเมื่อ

- 1) ทุกแอตทริบิวต์ที่เป็นตัวระบุค่า (Determinant) ในรีเลชันนั้น ต้องเป็นคีย์หลัก

- 2) ต้องไม่มีแอตทริบิวต์ใดในรีเลชัน ที่ใช้ระบุค่าของแอตทริบิวต์ที่เป็นคีย์หลัก หรือ ส่วนหนึ่ง ส่วนใดของแอตทริบิวต์ที่ประกอบกันเป็นคีย์หลักได้"

โดยทั่วไปรูปแบบนอร์มอลของบอยส์และคอตต์ จะอยู่ในรูปแบบนอร์มอล ระดับที่ 3 แต่รูปแบบนอร์มอล ระดับที่ 3 ไม่จำเป็นต้องเป็นแบบนอร์มอลของบอยส์และคอตต์เสมอไป เนื่องจากเป็นรูปแบบที่ได้ขยายเพิ่มเติมจากรูปแบบนอร์มอล ระดับที่ 3 ให้มีความเหมาะสมมากยิ่งขึ้น โดยทั่วไปรูปแบบที่ไม่เป็นรูปแบบนอร์มอลของบอยส์และคอตต์มักจะมีคุณสมบัติ ดังนี้"เป็นรีเลชันที่มีคีย์คู่แข่งได้หลาย ๆ คีย์โดยที่คีย์คู่แข่งเป็นคีย์ผสม (Composite Key) หรือ คีย์คู่แข่งนั้นมีบางส่วนซ้ำซ้อนกัน (คือมีแอตทริบิวต์บางตัวรวมกันอยู่)"

2.4.5 รูปแบบนอร์มอล ระดับที่ 4 (Fourth Normal Form : 4NF)

ปัญหาที่เกิดขึ้นที่ไม่สามารถแก้ไขด้วยรูปแบบนอร์มอล ระดับที่ 3 และรูปแบบนอร์มอลของบอยส์และคอตต์ ก็คือปัญหาที่เกิดจากความสัมพันธ์ระหว่างแอตทริบิวต์แบบเชิงกลุ่ม ซึ่งปัญหาดังกล่าวนี้นำให้เกิดความซ้ำซ้อนของข้อมูลได้เช่นเดียวกับที่พบในรูปแบบ นอร์มอลที่ ผ่านมา ดังนั้น เราจึงใช้รูปแบบนอร์มอล ระดับที่ 4 (Fourth Normal Form: 4NF) เพื่อแก้ปัญหา ความซ้ำซ้อนของข้อมูลที่พบในรีเลชัน ดังนี้ นิยาม "รีเลชันใด ๆ จะมีคุณสมบัติอยู่ในรูปแบบนอร์มอล ระดับที่ 4 ก็ต่อเมื่อ

- 1) รีเลชันนั้นมีคุณสมบัติอยู่ในรูปแบบนอร์มอล ของบอยส์และคอตต์แล้ว
- 2) รีเลชันนั้นต้องไม่มีการขึ้นต่อกันเป็นเชิงกลุ่ม (Multivalued Dependency) หรืออาจกล่าวได้ว่า เป็นรีเลชันที่ไม่มีความสัมพันธ์ในการระบุค่าของแอตทริบิวต์เป็นแบบหลายค่า โดยที่แอตทริบิวต์ที่ถูกระบุค่าเหล่านี้ไม่มีความสัมพันธ์กัน (Independently Multivalued Dependency)"

2.4.6 รูปแบบนอร์มอล ระดับที่ 5 (Fifth Normal Form : 5NF)

นิยาม "รีเลชันใด ๆ จะมีคุณสมบัติอยู่ในรูปแบบนอร์มอล ระดับที่ 5 ก็ต่อเมื่อ

- 1) รีเลชันนั้นมีคุณสมบัติอยู่ในรูปแบบนอร์มอล ระดับที่ 4 แล้ว
- 2) ต้องไม่มีการขึ้นต่อกันเชิงร่วม (Join Dependency)"

บทที่ 3

ขั้นตอนการดำเนินโครงการ

3.1 ความต้องการระบบ

3.1.1 Functional Requirements

- 1) สามารถเข้าถึงการใช้ระบบฐานข้อมูล
- 2) สามารถสร้างตารางเก็บข้อมูลผ่านการใช้คำสั่ง SQL ได้
- 3) สามารถแสดงรายการข้อมูลต่างๆ ที่ต้องการผ่านการใช้คำสั่ง SQL ในการแสดงผลได้
- 4) สามารถแก้ไข เพิ่ม หรือลบข้อมูลที่ต้องการผ่านการใช้คำสั่ง SQL ได้

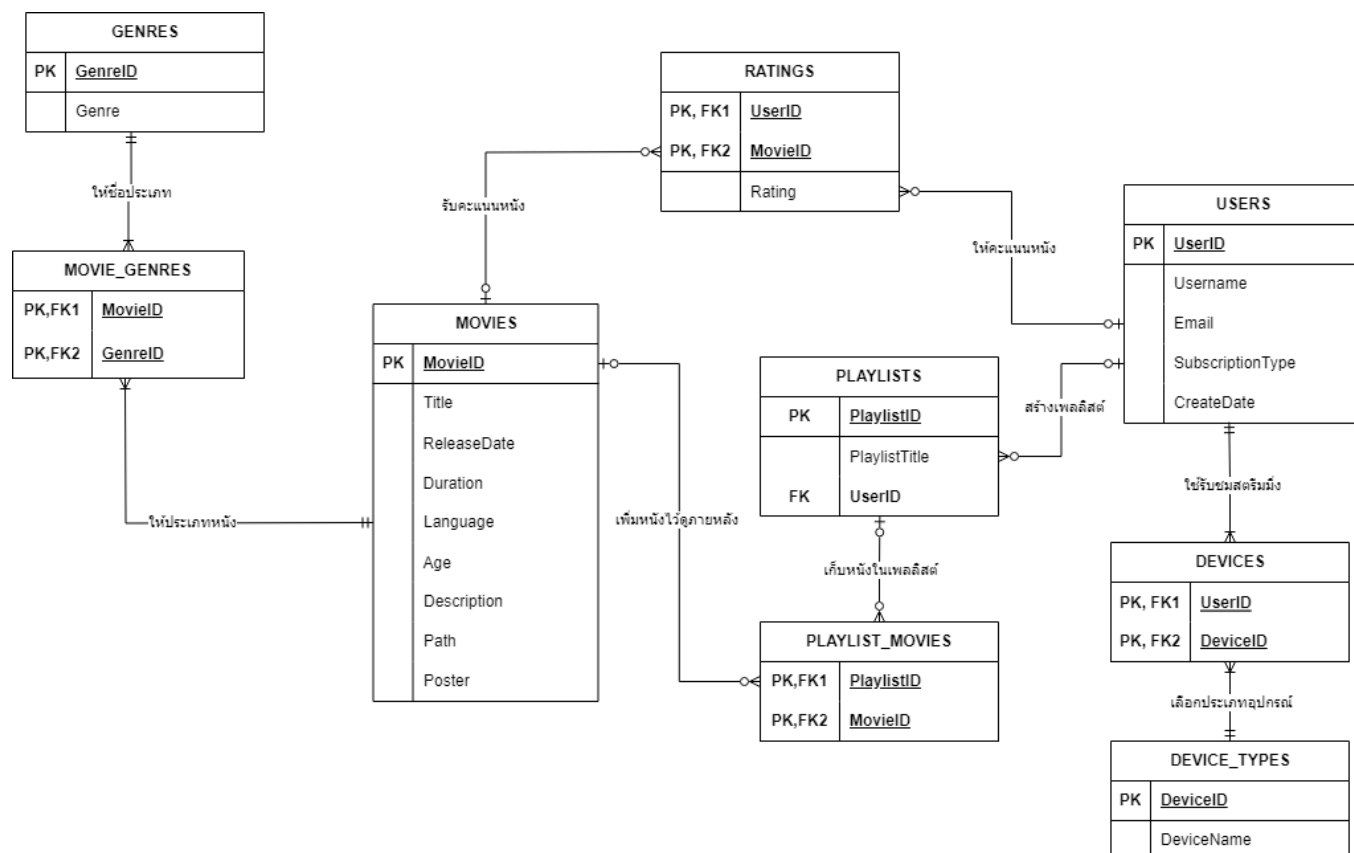
5) สามารถจัดการสิทธิในการเข้าถึงฐานข้อมูลผ่านการใช้คำสั่ง SQL ได้

3.1.2 Data Requirements

- 1) ตารางข้อมูลผู้ใช้งาน ประกอบไปด้วยข้อมูล รหัสผู้ใช้งาน, ชื่อผู้ใช้งาน, อีเมล, สถานะสมัครสมาชิก, วันที่สร้างบัญชี
- 2) ตารางข้อมูลหนังสือ ประกอบไปด้วยข้อมูล รหัสหนังสือ, ชื่อหนังสือ, วันที่ฉาย, ระยะเวลา, ภาษา, การจำกัดอายุ, คำอธิบาย, ชื่อไฟล์, ชื่อโปสเตอร์
- 3) ตารางข้อมูลประเภทหนังสือ ประกอบไปด้วยข้อมูล รหัสประเภทหนังสือ, ประเภทหนังสือ
- 4) ตารางข้อมูลเพลลิสต์ ประกอบไปด้วยข้อมูล รหัสเพลลิสต์, ชื่อเพลลิสต์, รหัสผู้ใช้งาน
- 5) ตารางข้อมูลการให้ความชอบหนังสือ ประกอบไปด้วยข้อมูล รหัสการให้ความชอบ, ความชอบ, รหัสผู้ใช้งาน, รหัสหนังสือ
- 6) ตารางข้อมูลประเภทอุปกรณ์รับชม ประกอบไปด้วยข้อมูล รหัสอุปกรณ์, ชื่ออุปกรณ์
- 7) ตารางข้อมูลอุปกรณ์รับชม ประกอบไปด้วยข้อมูล รหัสผู้ใช้งาน, รหัสอุปกรณ์
- 8) ตารางข้อมูลเพลลิสต์หนังสือ ประกอบไปด้วยข้อมูล รหัสเพลลิสต์, รหัสหนังสือ
- 9) ตารางข้อมูลประเภทของหนังสือ ประกอบไปด้วยข้อมูล รหัสหนังสือ, รหัสประเภทหนังสือ

3.2 การวิเคราะห์และออกแบบฐานข้อมูล

3.2.1 แผนภาพ Entity Relationship Diagram(ERD)



รูปที่ 1 แผนภาพ Entity Relationship Diagram

3.2.2 Normalization

นอร์มัลฟอร์มระดับที่ 1 (First Normal Form : 1NF)

- รีเลชั่น USERS

UserID	Username	Email	SubscriptionType	CreateDate
1	Ramin	rmain2546@gmail.com	Premium	2022-03-02
2	John	john123@gmail.com	Basic	2022-03-03
3	Alice	alice456@gmail.com	Standard	2022-03-04

รูปที่ 2 นอร์มัลฟอร์มระดับที่ 1 รีเลชั่น USERS

แต่ละแอททริบิวต์ไม่เกิดMultivalued Attribute ดังนั้นรีเลชันนี้อยู่ใน 1NF

- รีเลชั่น DEVICES

UserID	DeviceID
1	2
1	3
1	4
2	5

รูปที่ 3 นอร์มัลฟอร์มระดับที่ 1 รีเลชั่น DEVICES

แต่ละแอททริบิวต์ไม่เกิดMultivalued Attribute ดังนั้นรีเลชันนี้อยู่ใน 1NF เป็น Bridge Entity

- รีเลชั่น DEVICE_TYPES

DeviceID	DeviceName
1	iPhone
2	iPad
3	TV

รูปที่ 4 นอร์มัลฟอร์มระดับที่ 1 รีเลชั่น DEVICE_TYPES

แต่ละแอททริบิวต์ไม่เกิดMultivalued Attribute ดังนั้นรีเลชันนี้อยู่ใน 1NF

- รีเลชั่น PLAYLISTS

PlaylistID	PlaylistTitle	UserID
1	Doo Action	1
2	Doo Adventure	1
3	My Drama	3

รูปที่ 5 นอร์มัลฟอร์มระดับที่ 1 รีเลชัน PLAYLISTS

แต่ละแอททริบิวต์ไม่เกิดMultivalued Attribute ดังนั้นรีเลชันนี้อยู่ใน 1NF

- รีเลชัน PLAYLIST_MOVIES

PlaylistID	MovieID
1	2
1	3
1	6

รูปที่ 6 นอร์มัลฟอร์มระดับที่ 1 รีเลชัน PLAYLIST_MOVIES

แต่ละแอททริบิวต์ไม่เกิดMultivalued Attribute ดังนั้นรีเลชันนี้อยู่ใน 1NF เป็น Bridge Entity

- รีเลชัน MOVIES

MovieID	Title	ReleaseDate	Duration	Language	Age	Description	path	poster
1	A Beautiful Mind	2001-12-13	02:15:00	English	PG-13	After John Nash, a brilliant but asocial mathematici...	A_Beautiful_Mind.mp4	A_Beautiful_Mind.jpg
2	Along With the Gods: The Last 49 Days	2018-08-01	02:22:00	Korean	Not Rated	In the afterlife, one guardian helps a man through...	Along_With_The_Gods.mp4	Along_With_The_Gods.jpg
3	I, Robot	2004-07-16	01:55:00	English	PG-13	In 2035, a technophobic cop investigates a crime t...	I_Robot.mp4	I_Robot.jpg

รูปที่ 7 นอร์มัลฟอร์มระดับที่ 1 รีเลชัน MOVIES

แต่ละแอททริบิวต์ไม่เกิดMultivalued Attribute ดังนั้นรีเลชันนี้อยู่ใน 1NF

- รีเลชัน RATINGS

UserID	MovieID	Rating
1	1	Like
1	14	Like
1	11	Dislike

รูปที่ 8 นอร์มัลฟอร์มระดับที่ 1 รีเลชัน RATINGS

แต่ละแอททริบิวต์ไม่เกิดMultivalued Attribute ดังนั้นรีเลชันนี้อยู่ใน 1NF เป็น Bridge Entity

- รีเลชัน GENRES

GenreID	Genre
1	Action
2	Adventure
3	Drama

รูปที่ 9 นอร์มัลฟอร์มระดับที่ 1 รีเลชัน GENRES

แต่ละแอททริบิวต์ไม่เกิดMultivalued Attribute ดังนั้นรีเลชันนี้อยู่ใน 1NF

- รีเลชัน MOVIE_GENRES

MovieID	GenreID
1	3
1	15
1	16

รูปที่ 10 นอร์มัลฟอร์มระดับที่ 1 รีเลชัน MOVIE_GENRES

แต่ละแอททริบิวต์ไม่เกิดMultivalued Attribute ดังนั้นรีเลชันนี้อยู่ใน 1NF เป็น Bridge Entity

นอร์มัลฟอร์มระดับที่ 2 (Second Normal Form : 2NF)

- รีเลชั่น USERS

UserID	Username	Email	SubscriptionType	CreateDate
1	Ramin	rmain2546@gmail.com	Premium	2022-03-02
2	John	john123@gmail.com	Basic	2022-03-03
3	Alice	alice456@gmail.com	Standard	2022-03-04

รูปที่ 11 นอร์มัลฟอร์มระดับที่ 2 รีเลชั่น USERS

ขั้นตอนที่ 1 : Identify All Key Components

UserID

ขั้นตอนที่ 2 : Identify the Dependent Attributes

FD1: UserID → Username, Email, SubscriptionType, CreateDate

แอททริบิวต์ UserID ทำหน้าที่เป็น Key ที่บอกถึงแอททริบิวต์อื่นทุกตัวเป็น Full Functional Dependency จึงไม่มี Partial Dependency ดังนั้นรีเลชันนี้อยู่ใน 2NF

- รีเลชั่น DEVICE_TYPES

DeviceID	1	DeviceName
1		iPhone
2		iPad
3		TV

รูปที่ 12 นอร์มัลฟอร์มระดับที่ 2 รีเลชั่น DEVICE_TYPES

ขั้นตอนที่ 1 : Identify All Key Components

DeviceID

ขั้นตอนที่ 2 : Identify the Dependent Attributes

FD1: DeviceID → DeviceName

แอททริบิวต์ DeviceID ทำหน้าที่เป็น Key ที่บอกถึงแอททริบิวต์อื่นทุกตัวเป็น Full Functional Dependency จึงไม่มี Partial Dependency ดังนั้นรีเลชันนี้อยู่ใน 2NF

- รีเลชัน PLAYLISTS

PlaylistID	PlaylistTitle	UserID
1	Doo Action	1
2	Doo Adventure	1
3	My Drama	3

รูปที่ 13 นอร์มัลฟอร์มระดับที่ 2 รีเลชัน PLAYLISTS

ขั้นตอนที่ 1 : Identify All Key Components

PlaylistID

ขั้นตอนที่ 2 : Identify the Dependent Attributes

FD1: PlaylistID → PlaylistTitle, UserID

แอททริบิวต์ PlaylistID ทำหน้าที่เป็น Key ที่บอกถึงแอททริบิวต์อื่นทุกตัวเป็น Full Functional Dependency จึงไม่มี Partial Dependency ดังนั้นรีเลชันนี้อยู่ใน 2NF

- รีเลชัน MOVIES

MovieID	Title	ReleaseDate	Duration	Language	Age	Description	path	poster
1	A Beautiful Mind	2001-12-13	02:15:00	English	PG-13	After John Nash, a brilliant but asocial mathematici...	A_Beautiful_Mind.mp4	A_Beautiful_Mind.jpg
2	Along With the Gods: The Last 49 Days	2018-08-01	02:22:00	Korean	Not Rated	In the afterlife, one guardian helps a man through...	Along_With_The_Gods.mp4	Along_With_The_Gods.jpg
3	I, Robot	2004-07-16	01:55:00	English	PG-13	In 2035, a technophobic cop investigates a crime t...	I_Robot.mp4	I_Robot.jpg

รูปที่ 14 นอร์มัลฟอร์มระดับที่ 2 รีเลชัน MOVIES

ขั้นตอนที่ 1 : Identify All Key Components

MovieID

ขั้นตอนที่ 2 : Identify the Dependent Attributes

FD1: MovieID → Title, ReleaseDate, Duration, Language, Age, Description, path, poster

แอททริบิวต์ MovieID ทำหน้าที่เป็น Key ที่บอกถึงแอททริบิวต์อื่นทุกตัวเป็น Full Functional Dependency จึงไม่มี Partial Dependency ดังนั้นรีเลชันนี้อยู่ใน 2NF

- รีเลชัน GENRES

GenreID	Genre
1	Action
2	Adventure
3	Drama

รูปที่ 15 นอร์มัลฟอร์มระดับที่ 2 รีเลชัน GENRE

ขั้นตอนที่ 1 : Identify All Key Components

GenreID

ขั้นตอนที่ 2 : Identify the Dependent Attributes

FD1: GenreID → Genre

แอททริบิวต์ GenreID ทำหน้าที่เป็น Key ที่บอกถึงแอททริบิวต์อื่นทุกตัวเป็น Full Functional Dependency จึงไม่มี Partial Dependency ดังนั้นรีเลชันนี้อยู่ใน 2NF

นอร์มัลฟอร์มระดับที่ 3 (Third Normal Form : 3NF)

- รีเลชั่น USERS

UserID	Username	Email	SubscriptionType	CreateDate
1	Ramin	rmain2546@gmail.com	Premium	2022-03-02
2	John	john123@gmail.com	Basic	2022-03-03
3	Alice	alice456@gmail.com	Standard	2022-03-04

รูปที่ 16 นอร์มัลฟอร์มระดับที่ 3 รีเลชั่น USERS

ผลลัพธ์จากรีเลชั่น USERS ที่อยู่ใน 2NF และมี FD ดังนี้

FD1: UserID \rightarrow Username, Email, subscriptionType, CreateDate

พิจารณา

FD1: UserID \rightarrow Username, Email, subscriptionType, CreateDate

แอททริบิวต์ที่เป็น Non-key คือ Username Email subscriptionType CreateDate ไม่สามารถทำหน้าที่เป็น Determinant เพราะฉะนั้น รีเลชั่นนี้ไม่มี Transitive Dependency ดังนั้นรีเลชั่นนี้อยู่ใน 3NF

- รีเลชั่น DEVICE_TYPES

DeviceID	DeviceName
1	iPhone
2	iPad
3	TV

รูปที่ 17 นอร์มัลฟอร์มระดับที่ 3 รีเลชัน DEVICE_TYPES

ผลลัพธ์จากรีเลชัน DEVICE_TYPES ที่อยู่ใน 2NF และมี FD ดังนี้

FD1: DeviceID → DeviceName

พิจารณา

FD1: DeviceID → DeviceName

แอททริบิวต์ที่เป็น Non-key คือ DeviceName ไม่สามารถทำหน้าที่เป็น Determinant เพราะฉะนั้น รีเลชันนี้ไม่มี Transitive Dependency ดังนั้นรีเลชันนี้อยู่ใน 3NF

- รีเลชัน PLAYLISTS

PlaylistID	PlaylistTitle	UserID
1	Doo Action	1
2	Doo Adventure	1
3	My Drama	3

รูปที่ 18 นอร์มัลฟอร์มระดับที่ 3 รีเลชัน PLAYLISTS

ผลลัพธ์จากรีเลชัน PLAYLISTS ที่อยู่ใน 2NF และมี FD ดังนี้

FD1: PlaylistID → PlaylistTitle, UserID

พิจารณา

FD1: PlaylistID → PlaylistTitle, UserID

แอททริบิวต์ที่เป็น Non-key คือ PlaylistTitle UserID ไม่สามารถทำหน้าที่เป็น Determinant เพราะฉะนั้น รีเลชันนี้ไม่มี Transitive Dependency ดังนั้นรีเลชันนี้อยู่ใน 3NF

- รีเลชัน MOVIES

MovieID	Title	ReleaseDate	Duration	Language	Age	Description	path	poster
1	A Beautiful Mind	2001-12-13	02:15:00	English	PG-13	After John Nash, a brilliant but asocial mathematici...	A_Beautiful_Mind.mp4	A_Beautiful_Mind.jpg
2	Along With the Gods: The Last 49 Days	2018-08-01	02:22:00	Korean	Not Rated	In the afterlife, one guardian helps a man through...	Along_With_The_Gods.mp4	Along_With_The_Gods.jpg
3	I, Robot	2004-07-16	01:55:00	English	PG-13	In 2035, a technophobic cop investigates a crime t...	I_Robot.mp4	I_Robot.jpg

รูปที่ 19 นอร์มัลฟอร์มระดับที่ 3 รีเลชัน MOVIES

ผลลัพธ์จากรีเลชัน MOVIES ที่อยู่ใน 2NF และมี FD ดังนี้

FD1: MovieID → Title, ReleaseDate, Duration, Language, Age, Description, path, poster

พิจารณา

FD1: MovieID → Title, ReleaseDate, Duration, Language, Age, Description, path, poster

แอททริบิวต์ที่เป็น Non-key คือ Title ReleaseDate Duration Language Age Description path poster ไม่สามารถทำหน้าที่เป็น Determinant เพราะฉะนั้น รีเลชันนี้ไม่มี Transitive Dependency ดังนั้น รีเลชันนี้อยู่ใน 3NF

- รีเลชัน GENRES

GenreID	Genre
1	Action
2	Adventure
3	Drama

รูปที่ 20 นอร์มัลฟอร์มระดับที่ 3 รีเลชัน GENRES

ผลลัพธ์จากรีเลชัน GENRES ที่อยู่ใน 2NF และมี FD ดังนี้

FD1: GenreID \rightarrow Genre

พิจารณา

FD1: GenreID \rightarrow Genre

แอททริบิวต์ที่เป็น Non-key คือ Genre ไม่สามารถทำหน้าที่เป็น Determinant เพราะฉะนั้น รีเลชันนี้ไม่มี Transitive Dependency ดังนั้นรีเลชันนี้อยู่ใน 3NF

พจนานุกรมข้อมูล (Data Dictionary)

1. ตารางข้อมูลในระบบ

Table Name	Description
USERS	จัดเก็บข้อมูลเกี่ยวกับผู้ใช้งาน
MOVIES	จัดเก็บข้อมูลเกี่ยวกับหนัง
MOVIE_GENRES	จัดเก็บข้อมูลเกี่ยวกับการจัดประเภทของหนัง
GENRES	จัดเก็บข้อมูลเกี่ยวกับประเภทของหนัง
DEVICES	จัดเก็บข้อมูลเกี่ยวกับอุปกรณ์รับชมของผู้ใช้งาน
DEVICE_TYPES	จัดเก็บข้อมูลเกี่ยวกับประเภทอุปกรณ์รับชม
PLAYLISTS	จัดเก็บข้อมูลเกี่ยวกับเพลย์ลิสต์ของผู้ใช้งาน
PLAYLISTS_MOVIES	จัดเก็บข้อมูลเกี่ยวกับหนังที่อยู่ในเพลย์ลิสต์
RATINGS	จัดเก็บข้อมูลเกี่ยวกับการให้คะแนนของผู้ใช้งาน

ตารางที่ 1 ตารางข้อมูลในระบบ

2. พจนานุกรมของตาราง

1) ตาราง USERS

Attribute Name	Description	Type	Null	Key	FK Reference
UserID	รหัสผู้ใช้	INT(6)	NO	PK	
Username	ชื่อผู้ใช้	VARCHAR(50)	NO		
Email	อีเมลของผู้ใช้	VARCHAR(50)	NO		
SubscriptionType	ประเภทการสมัครใช้งานของผู้ใช้	VARCHAR(15)	NO		
CreateDate	วันที่ผู้ใช้ทำการสมัครการใช้งาน	DATE	NO		

ตารางที่ 2 พจนานุกรมข้อมูลตาราง USERS

2) ตาราง MOVIES

Attribute Name	Description	Type	Null	Key	FK Reference
MovieID	รหัสหนัง	INT(6)	NO	PK	
Title	ชื่อหนัง	VARCHAR(255)	NO		
ReleaseDate	วันที่หนังออกฉาย	DATE	NO		
Duration	ความยาวของหนัง	TIME	NO		
Language	ภาษาต้นฉบับ	VARCHAR(30)	NO		
Age	เรทอายุผู้ชมที่สามารถรับชมได้	VARCAHR(30)	NO		
Description	คำอธิบายเกี่ยวกับหนัง	VARCHAR(255)	NO		
path	ช่องทางการดูหนัง	VARCHAR(100)	NO		
Poster	โปสเตอร์ของหนัง	VARCHAR(100)	NO		

ตารางที่ 3 พจนานุกรมข้อมูลตาราง MOVIES

3) ตาราง MOVIE_GENRES

Attribute Name	Description	Type	Null	Key	FK Reference
MovieID	รหัสหนัง	INT(6)	NO	PK,FK1	movies.MovieID
GenreID	รหัสประเภทหนัง	INT(6)	NO	PK,FK2	genres.GenreID

ตารางที่ 4 พจนานุกรมข้อมูลตาราง MOVIE_GENRES

4) ตาราง GENRES

Attribute Name	Description	Type	Null	Key	FK Reference
GenreID	รหัสประเภทหนัง	INT(6)	NO	PK	
Genre	ประเภทหนัง	VARCHAR(100))			

ตารางที่ 5 พจนานุกรมข้อมูลตาราง GENRES

5) ตาราง DEVICES

Attribute Name	Description	Type	Null	Key	FK Reference
UserID	รหัสผู้ใช้	INT(6)	NO	PK,FK1	users.UserID
DeviceID	รหัสอุปกรณ์รับชม	INT(6)	NO	PK,FK2	devices.DeviceID

ตารางที่ 6 พจนานุกรมข้อมูลตาราง DEVICES

6) ตาราง **DEVICE_TYPES**

Attribute Name	Description	Type	Null	Key	FK Reference
DeviceID	รหัสอุปกรณ์รับชม	INT(6)	NO	PK	
DeviceName	ชื่ออุปกรณ์รับชม	VARCHAR(30)	NO		

ตารางที่ 7 พจนานุกรมข้อมูลตาราง **DEVICE_TYPES**

7) ตาราง **PLAYLISTS**

Attribute Name	Description	Type	Null	Key	FK Reference
PlaylistID	รหัสเพลย์ลิสต์	INT(6)	NO	PK	
PlaylistTitle	ชื่อเพลย์ลิสต์	VARCHAR(255)	NO		
UserID	รหัสผู้ใช้	INT(6)	NO	FK	users.UserID

ตารางที่ 8 พจนานุกรมข้อมูลตาราง **PLAYLISTS**

8) ตาราง **PLAYLISTS_MOVIES**

Attribute Name	Description	Type	Null	Key	FK Reference
PlaylistID	รหัสเพลย์ลิสต์	INT(6)	NO	PK,FK 1	playlist.PlaylistID
MovieID	รหัสหนัง	INT(6)	NO	PK,FK 2	movies.MovieID

ตารางที่ 9 พจนานุกรมข้อมูลตาราง **PLAYLISTS_MOVIES**

9) ตาราง **RATINGS**

Attribute Name	Description	Type	Null	Key	FK Reference
UserID	รหัสผู้ใช้	INT(6)	NO	PK,FK1	users.UserID
MovieID	รหัสหนัง	INT(6)	NO	PK,FK2	movies.MovieID
Rating	การให้คะแนน	VARCHAR(20)	NO		

ตารางที่ 10 พจนานุกรมข้อมูลตาราง **RATINGS**

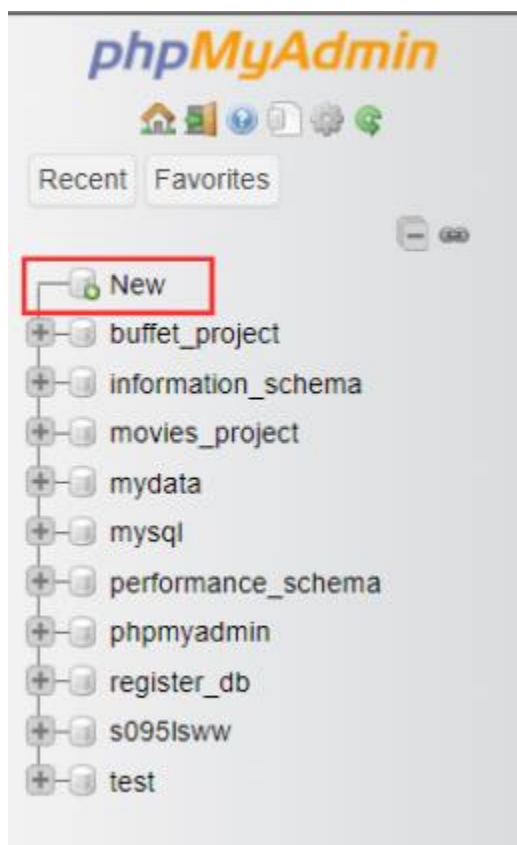
3.3 หลักการทำงานของระบบ

3.3.1 ซอฟต์แวร์ที่เลือกใช้

- 1) MariaDB ซอฟต์แวร์ DBMS
- 2) phpMyAdmin ใช้จัดการฐานข้อมูล
- 3) BitTorrent ใช้โหลดหนัง

3.3.2 ขั้นตอนการสร้างฐานข้อมูล

- 1) เลือกโปรแกรมจัดการฐานข้อมูล โดยเลือกเป็น phpMyAdmin เนื่องจากมีความสะดวกในการ สร้าง แก้ไข เพิ่มหรือลบ ข้อมูล
- 2) เมื่อเข้ามาในหน้าโปรแกรมกตสร้างฐานข้อมูลโดยกด New



รูปที่ 21 การใช้ phpMyAdmin สร้างฐานข้อมูล

3) ใส่ชื่อฐานข้อมูลแล้วกดปุ่ม Create



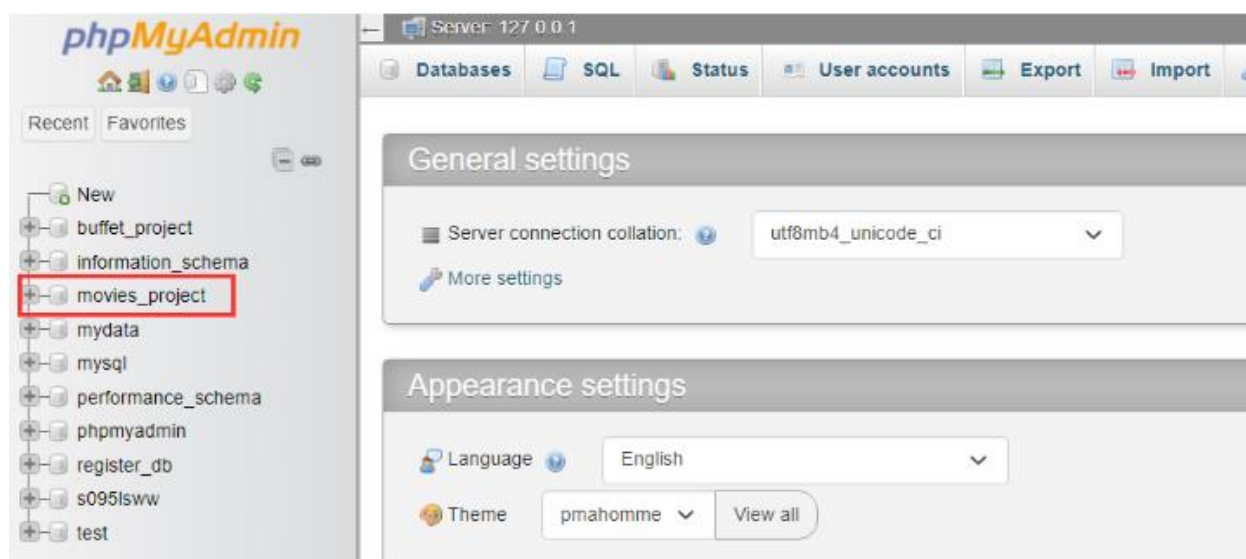
Create database

Database name: utf8mb4_general_ci

Create

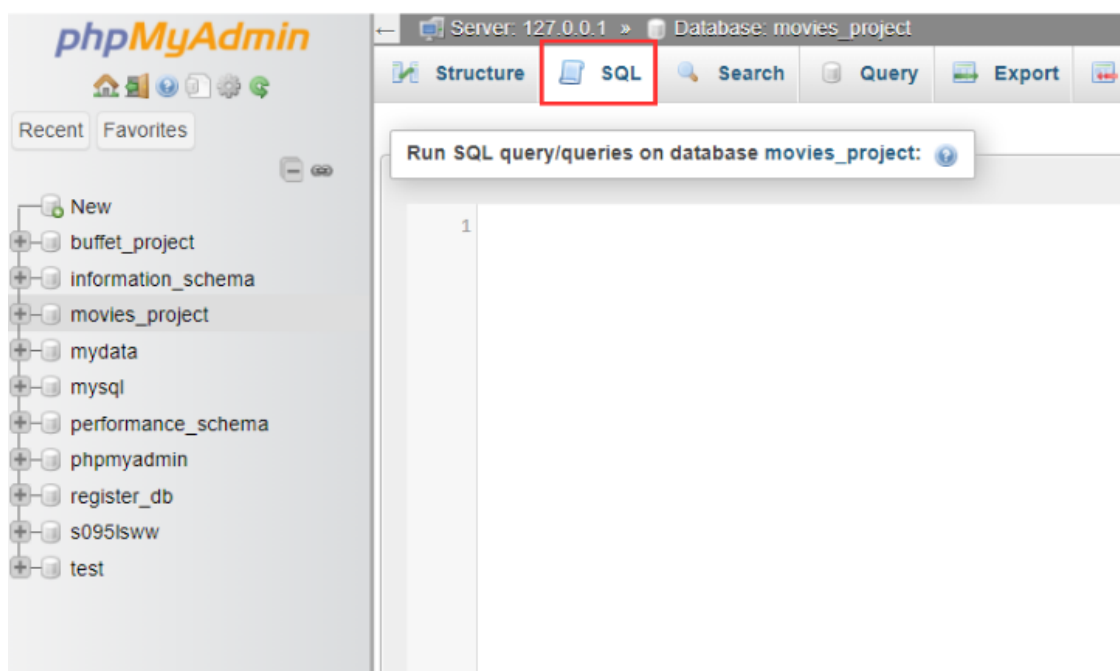
รูปที่ 22 การใช้ phpMyAdmin สร้างฐานข้อมูล

4) เมื่อสร้างฐานข้อมูลเสร็จแล้วกดไปที่ฐานข้อมูลที่สร้างขึ้น



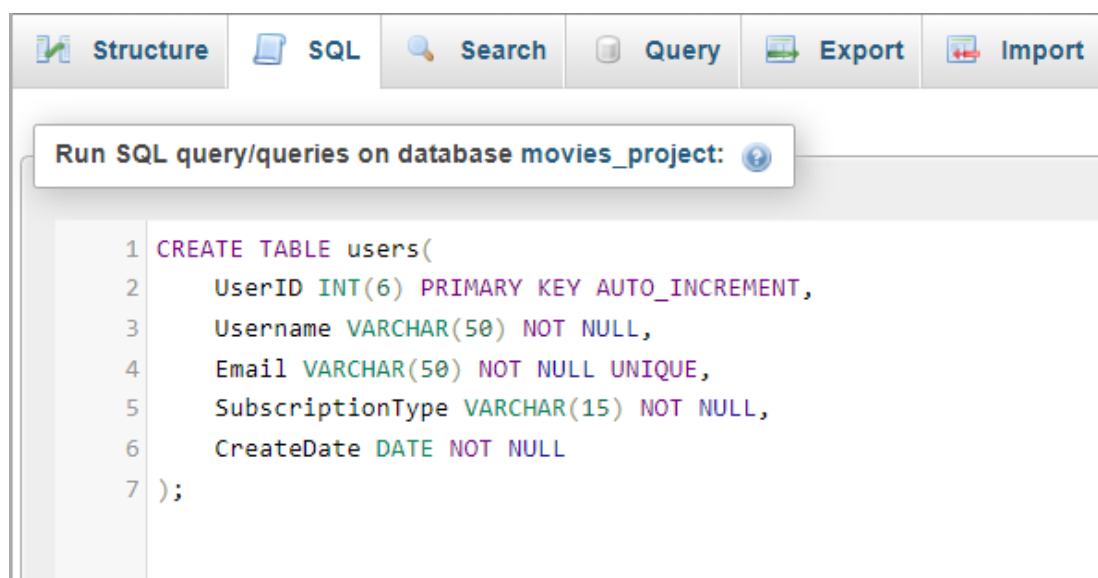
รูปที่ 23 การใช้ phpMyAdmin เลือกฐานข้อมูล

5) กดเลือกหน้า SQL เพื่อพิมพ์คำสั่ง SQL



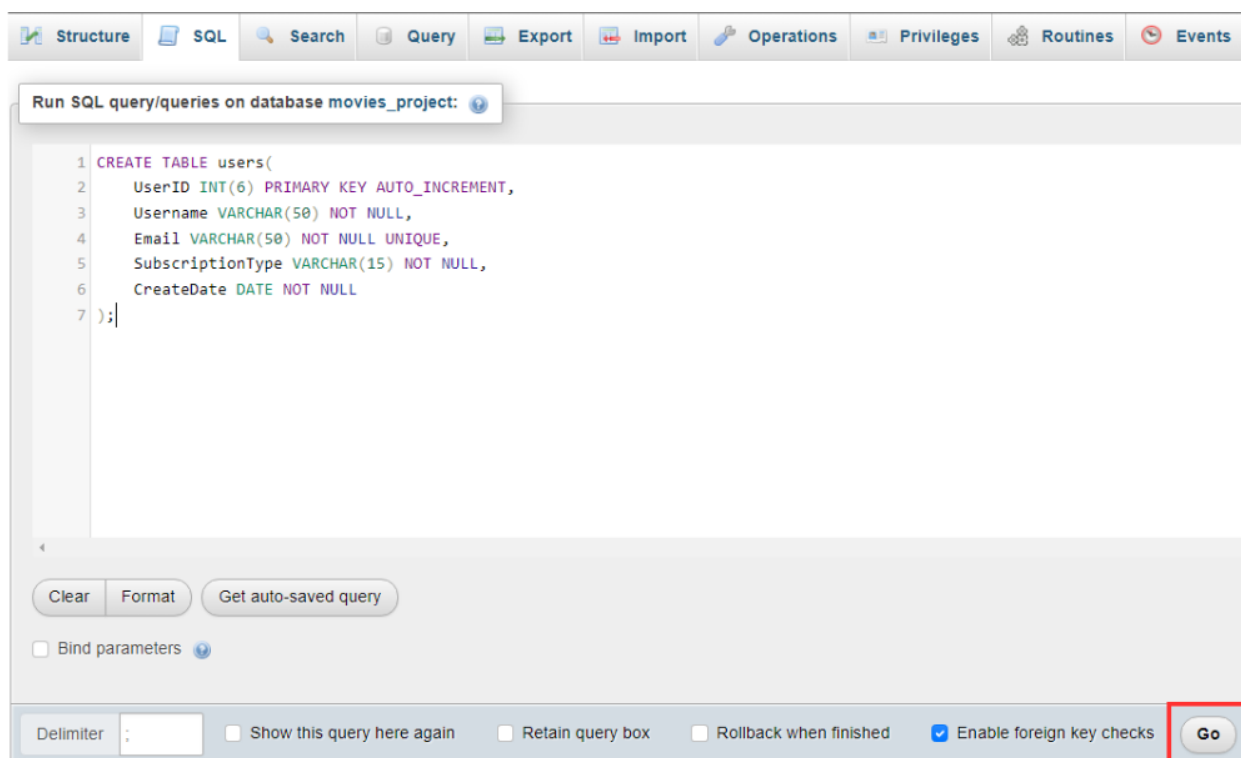
รูปที่ 24 การใช้ phpMyAdmin จัดการฐานข้อมูล

6) พิมพ์คำสั่ง SQL ตามรายละเอียดของ Data Dictionary



รูปที่ 25 การใช้ phpMyAdmin จัดการฐานข้อมูล

7) กด GO เพื่อรันโปรแกรม



รูปที่ 26 การใช้ phpMyAdmin จัดการฐานข้อมูล

- 8) สร้างตารางอื่นๆได้แก่ MOVIES, GENRES, PLAYLISTS, RATINGS, DEVICE_TYPES, MOVIE_GENRES, PLAYLIST_MOVIES, DEVICES

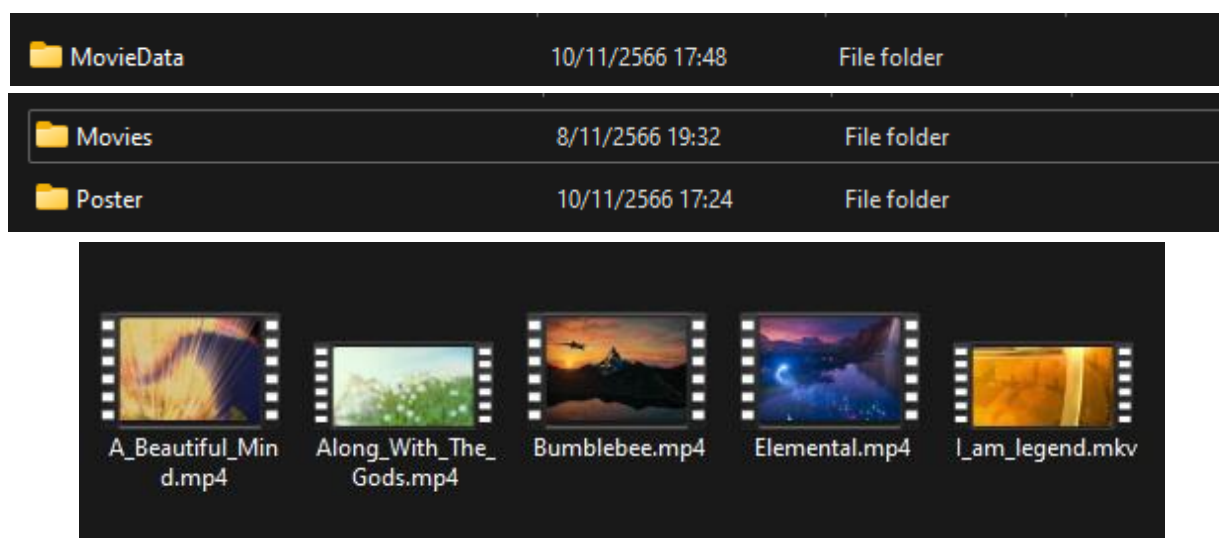
3.3.3 กระบวนการทำงาน

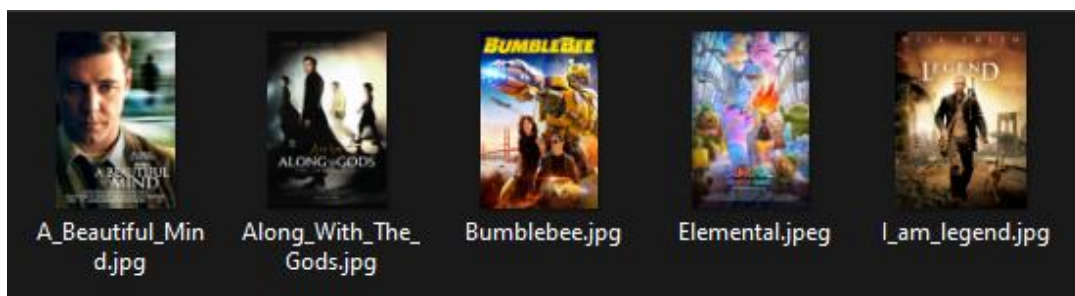
- 1) ข้อมูลที่เป็นหนังและรูปภาพจะเก็บเป็นชื่อไฟล์ลงในฐานข้อมูล

path	poster
A_Beautiful_Mind.mp4	A_Beautiful_Mind.jpg
Along_With_The_Gods.mp4	Along_With_The_Gods.jpg
I_Robot.mp4	I_Robot.jpg
Bumblebee.mp4	Bumblebee.jpg
Elemental.mp4	Elemental.jpeg
I_am_legend.mkv	I_am_legend.jpg
Train_To_Busan.mp4	Train_To_Busan.jpg
MayWho.mp4	MayWho.jpg
SuckSeed.mp4	SuckSeed.jpg
The_Matrix_Reloaded.mp4	The_Matrix_Reloaded.jpg
Transformers_The_Last_Knight.mp4	Transformers_The_Last_Knight.jpg
Interstellar.mp4	Interstellar.jpg
Iron_Man.mp4	Iron_Man.jpg
John_Wick.mp4	John_Wick.jpg

รูปที่ 27 ฐานข้อมูลหนังและรูปภาพ

- 2) โดยข้อมูลหนังและรูปภาพจะอยู่ในโฟลเดอร์ MovieData ภายในโฟลเดอร์ประกอบไปด้วยโฟลเดอร์ Movies ที่ใช้เก็บหนัง และ Poster ใช้เก็บรูปโปสเตอร์หนัง





รูปที่ 28 โพล์เดอร์ที่ใช้เก็บหนังและโปสเตอร์

- 3) เมื่อนำข้อมูลมาใช้ให้กำหนด path ตอนดึงข้อมูลให้ตรงกับที่เก็บโพลเดอร์

```
<video width="300px" height="400px" controls poster=<?php echo "../MovieData/Poster/" . $row["poster"] . ".jpg" ?>
src=<?php echo "../MovieData/Movies/" . $row["path"] . ".mp4" ?>
</video>
```

รูปที่ 29 การกำหนด path ตอนดึงข้อมูล

- 4) ก็จะสามารถแสดงผลได้อย่างถูกต้อง



รูปที่ 30 ตัวอย่างการแสดงผล

บทที่ 4

ผลการดำเนินงาน

4.1 ส่วนของผู้ดูแลระบบ

4.1.1 คำสั่ง DDL (Data Definition Language)

- 1) สร้างตาราง MOVIES โดยใช้คำสั่ง CREATE

```
CREATE TABLE movies(
    MovieID INT(6) PRIMARY KEY AUTO_INCREMENT,
    Title VARCHAR(255) NOT NULL,
    ReleaseDate DATE NOT NULL,
    Duration TIME NOT NULL,
    Language VARCHAR(30) NOT NULL,
    Age VARCHAR(30) NOT NULL,
    Description VARCHAR(255) NOT NULL,
    Path VARCHAR(255) NOT NULL,
    Poster VARCHAR(255) NOT NULL
);
```

รูปที่ 31 การสร้างตาราง MOVIES

- 2) สร้างตาราง USERS โดยใช้คำสั่ง CREATE

```
CREATE TABLE users(
    UserID INT(6) PRIMARY KEY AUTO_INCREMENT,
    Username VARCHAR(50) NOT NULL,
    Email VARCHAR(50) NOT NULL UNIQUE,
    SubscriptionType VARCHAR(15) NOT NULL,
    CreateDate DATE NOT NULL
);
```

รูปที่ 32 การสร้างตาราง USERS

3) สร้างตาราง PLAYLISTS โดยใช้คำสั่ง CREATE

```
CREATE TABLE PLAYLISTS (
    PlaylistID INT(6) PRIMARY KEY AUTO_INCREMENT,
    PlaylistTitle VARCHAR(255) NOT NULL,
    UserID INT(6),
    FOREIGN KEY (UserID) REFERENCES users(UserID)
    ON DELETE CASCADE
);
```

รูปที่ 33 การสร้างตาราง PLAYLISTS

4) สร้างตาราง MOVIE_GENRES โดยใช้คำสั่ง CREATE

```
CREATE TABLE MOVIE_GENRES(
    MovieID INT(6),
    GenreID INT(6),
    PRIMARY KEY(MovieID, GenreID),
    FOREIGN KEY (MovieID) REFERENCES movies(MovieID) ON DELETE CASCADE,
    FOREIGN KEY (GenreID) REFERENCES genres(GenreID) ON DELETE CASCADE
);
```

รูปที่ 34 การสร้างตาราง MOVIE_GENRES

5) สร้าง USER ชื่อ ramin โดยใช้คำสั่ง CREATE

```
1 CREATE USER 'ramin'@'localhost' IDENTIFIED BY 'matin007';
```

รูปที่ 35 การสร้าง USER ชื่อ ramin

4.1.2 คำสั่ง DCL (Data Control Language)

- 1) กำหนดสิทธิ์ให้ USER ramin สามารถ SELECT ตาราง MOVIES ได้

```
GRANT SELECT
ON movies_project.movies
TO 'ramin'@'localhost';
```

รูปที่ 36 การกำหนดสิทธิ์ให้ USER ramin สามารถ SELECT ตาราง MOVIES

- 2) กำหนดสิทธิ์ให้ USER ramin สามารถ SELECT, INSERT, UPDATE, DELETE ตาราง PLAYLISTS ได้

```
GRANT SELECT, INSERT, UPDATE, DELETE
ON movies_project.playlists
TO 'ramin'@'localhost';
```

รูปที่ 37 การกำหนดสิทธิ์ให้ USER ramin สามารถ SELECT, INSERT, UPDATE, DELETE ตาราง PLAYLISTS

- 3) กำหนดสิทธิ์ให้ USER ramin สามารถ SELECT คอลัมน์ Genre จากตาราง GENRES ได้

```
GRANT SELECT(Genre)
ON movies_project.genres
TO 'ramin'@'localhost';
```

รูปที่ 38 การกำหนดสิทธิ์ให้ USER ramin สามารถ SELECT คอลัมน์ Genre จากตาราง GENRES

- 4) กำหนดสิทธิ์ให้ USER Thanapat สามารถทำได้ทุกอย่างกับฐานข้อมูล movies_project

```
GRANT ALL
ON movies_project.*
TO 'Thanapat'@'localhost';
```

รูปที่ 39 การกำหนดสิทธิ์ให้ USER Thanapat สามารถทำได้ทุกอย่างกับฐานข้อมูล movies_project

- 5) ยกเลิกสิทธิ์ให้ USER Thanapat ไม่สามารถทำการใดๆกับตาราง ratings ได้

```
REVOKE ALL
ON movies_project.ratings
FROM Thanapat;
```

รูปที่ 40 การยกเลิกสิทธิ์ USER Thanapat ไม่สามารถทำการใดๆกับตาราง ratings

4.2 ส่วนของผู้ใช้ระบบ

4.2.1 คำสั่ง DML (Data Manipulation Language)

- 1) แสดงหนังที่ผู้ใช้งานกดถูกใจมากที่สุด 3 อันดับแรก โดยใช้ SELECT

```
select Title, Count(Rating) AS `Like`
from movies m
join ratings r on(m.MovieID=r.MovieID)
join users u on(u.UserID=r.UserID)
WHERE Rating = "Like" AND CreateDate < "2022-03-23"
GROUP BY Title
ORDER BY `Like` DESC
LIMIT 0, 3;
```

รูปที่ 41 การใช้คำสั่ง SELECT แสดงหนังที่ผู้ใช้งานกดถูกใจมากที่สุด 3 อันดับแรก

ผลลัพธ์ที่ได้มี 2 คอลัมน์ประกอบด้วย Title(ชื่อหนัง), Like(จำนวนคนที่กดถูกใจ)

Title	Like ▼ 1
John Wick	4
I, Robot	3
Elemental	2

รูปที่ 42 ผลลัพธ์จากคำสั่งรูปที่ 41

- 2) แสดง Genre ที่มากที่สุด 3 อันดับแรกจาก Playlist ของ UserID คนที่ 1 โดยใช้ SELECT

```
SELECT Genre, Count(Genre) AS `Most_Genre`
from users u
join playlists p on(u.UserID=p.UserID)
join playlist_movies pm on(p.PlaylistID=pm.PlaylistID)
join movies m on(pm.MovieID=m.MovieID)
join movie_genres mg on(m.MovieID=mg.MovieID)
join genres g on(mg.GenreID=g.GenreID)
WHERE u.UserID=1
GROUP BY Genre
ORDER BY `Most_Genre` DESC
LIMIT 3;
```

รูปที่ 43 การใช้คำสั่ง SELECT แสดง Genre ที่มากที่สุด 3 อันดับแรก

ผลลัพธ์ที่ได้มี 2 คอลัมน์ประกอบด้วย Genre(ประเภทหนัง), Most_Genre(จำนวนประเภทหนังในเพลลิสต์)

Genre	Most_Genre
Action	3
Thriller	3
Sci-fi	3

รูปที่ 44 ผลลัพธ์จากคำสั่งรูปที่ 43

3) แสดงจำนวน Device ในแต่ละ subscriptionType โดยใช้ SELECT

```
Select subscriptionType, DeviceName, COUNT(DeviceName) as `Device Number`
from users u
JOIN devices d on(u.UserID=d.UserID)
JOIN device_types dt on(dt.DeviceID=d.DeviceID)
GROUP BY subscriptionType, DeviceName;
```

รูปที่ 45 การใช้คำสั่ง SELECT แสดงจำนวน Device ในแต่ละ subscriptionType

ผลลัพธ์ที่ได้มี 3 คอลัมน์ประกอบด้วย subscriptionType(ประเภทสมาชิก), DeviceName(ชื่ออุปกรณ์), DeviceNumber(จำนวนอุปกรณ์)

subscriptionType	DeviceName	Device Number
Basic	iPad	3
Basic	iPhone	3
Basic	Samsung	3
Basic	Tablet	1
Basic	Web Browser	2
Premium	iPad	5
Premium	iPhone	1
Premium	Samsung	1
Premium	Tablet	2
Premium	TV	7
Premium	Web Browser	6
Standard	iPad	3
Standard	iPhone	3
Standard	Samsung	3
Standard	Tablet	2
Standard	Web Browser	5

รูปที่ 46 ผลลัพธ์จากคำสั่งรูปที่ 45

4) แสดงหนังหมวด Action, Sci-fi โดยใช้ SELECT

```
Select DISTINCT Title, Description, Duration, Language, Age, ReleaseDate, path, poster from movies m
join movie_genres mg on (m.MovieID=mg.MovieID)
join genres g on (g.GenreID=mg.GenreID)
WHERE Genre="Action" OR Genre="Sci-fi"
ORDER BY Title;
```

รูปที่ 47 การใช้คำสั่ง SELECT แสดงหนังหมวด Action, Sci-fi

ผลลัพธ์ที่ได้มี 8 คอลัมน์ประกอบด้วย Title(ชื่อหนัง), Description(คำอธิบายหนัง), Duration(ระยะเวลาหนัง), Language(ภาษา), Age(อายุจำกัด), ReleaseDate(วันที่ปล่อย), path(ชื่อไฟล์วิดีโอ), poster(ชื่อไฟล์โปสเตอร์)

Title	Description	Duration	Language	Age	ReleaseDate	path	poster
Along With the Gods: The Last 49 Days	In the afterlife, one guardian helps a man through...	02:22:00	Korean	Not Rated	2018-08-01	Along_With_The_Gods.mp4	Along_With_The_Gods.jpg
Bumblebee	On the run in the year 1987, Bumblebee finds refug...	01:54:00	English	PG-13	2018-12-21	Bumblebee.mp4	Bumblebee.jpg
Elemental	Follows Ember and Wade, in a city where fire-, wat...	01:41:00	English	PG	2023-06-16	Elemental.mp4	Elemental.jpg
I Am Legend	Years after a plague kills most of humanity and tr...	01:41:00	English	PG-13	2007-12-14	I_am_legend.mkv	I_am_legend.jpg
I, Robot	In 2035, a technophobic cop investigates a crime t...	01:55:00	English	PG-13	2004-07-16	I_Robot.mp4	I_Robot.jpg
Interstellar	When Earth becomes uninhabitable in the future, a...	02:49:00	English	PG-13	2014-11-05	Interstellar.mp4	Interstellar.jpg
Iron Man	After being held captive in an Afghan cave, billio...	02:06:00	English	PG-13	2008-04-30	Iron_Man.mp4	Iron_Man.jpg
John Wick	An ex-hitman comes out of retirement to track down...	01:41:00	English	R	2014-10-23	John_Wick.mp4	John_Wick.jpg
The Matrix Reloaded	Freedom fighters Neo, Trinity and Morpheus continu...	02:18:00	English	R	2003-05-15	The_Matrix_Reloaded.mp4	The_Matrix_Reloaded.jpg
Train to Busan	While a zombie virus breaks out in South Korea, pa...	01:58:00	Korean	PG-13	2016-07-20	Train_To_Busan.mp4	Train_To_Busan.jpg
Transformers: The Last Knight	A deadly threat from Earth's history reappears and...	02:34:00	English	PG-13	2017-06-21	Transformers_The_Last_Knight.mp4	Transformers_The_Last_Knight.jpg

รูปที่ 48 ผลลัพธ์จากคำสั่งรูปที่ 47

5) เพิ่มข้อมูลประเภทหนังในตาราง Genres โดยใช้ INSERT

```
INSERT INTO genres(Genre)
VALUES ("Action"), ("Adventure"), ("Drama"), ("Sci-fi"),
("War"), ("Thriller"), ("Comedy"), ("Fantasy"), ("Superhero")
```

รูปที่ 49 การใช้คำสั่ง INSERT เพิ่มข้อมูลประเภทหนังในตาราง Genres

ผลลัพธ์ที่ได้มีการเพิ่มข้อมูลประเภทหนังเข้ามาในตาราง Genres

GenreID	Genre
1	Action
2	Adventure
3	Drama
4	Sci-fi
5	War
6	Horror
7	Comedy
8	Fantasy
9	Superhero

รูปที่ 50 ผลลัพธ์จากคำสั่งรูปที่ 49

บทที่ 5

สรุปผลการดำเนินโครงการ

5.1 สรุปผล

การดำเนินโครงการ Database ที่เกี่ยวกับ Movies Streaming ได้ผลลัพธ์ที่มีความสำเร็จ โดยในการออกแบบได้มีการเพิ่มประสิทธิภาพฐานข้อมูลโดยการทำ Normalization มีการเก็บไฟล์ข้อมูลหนังและรูปโปสเตอร์ที่สามารถนำมาใช้งานได้กับฐานข้อมูลด้วยวิธีทำ path อีกทั้งยังมีการเพิ่มความแตกต่างจาก Movies Streaming เดิมโดยการสร้างตารางเพลลิสต์ที่ผู้ใช้งานสามารถสร้างเพลลิสต์ ตั้งชื่อเพลลิสต์ และเก็บหนังที่เรื่องก็ได้ที่ต้องการ โดยไม่มีความซ้ำซ้อนของข้อมูล เพื่อความสะดวกต่อผู้ใช้งานและจัดการข้อมูลได้ง่ายต่อผู้ดูแลฐานข้อมูล

5.2 ประโยชน์ที่ได้รับ

5.1.1 ได้ความรู้เกี่ยวกับแนวคิดฐานข้อมูล

5.1.2 ได้เข้าใจการออกแบบฐานข้อมูลของระบบสตรีมมิ่งหนัง

5.3 ปัญหาและอุปสรรคในการดำเนินงาน

5.2.1 ถ้าข้อมูลหนังมีมากขึ้น อาจทำให้ที่เก็บข้อมูลเต็มได้

5.4 ข้อเสนอแนะ

5.4.1 ควรมีหน้าตาโปรแกรมที่ชัดเจนเพื่อให้เห็นภาพในการดำเนินงาน

5.4.2 ควรศึกษาการทำระบบสตรีมมิ่งร่วมกับการทำหน้าตาโปรแกรม

5.4.3 ควรเก็บข้อมูลลงบน cloud เพื่อประหยัดเนื้อที่เก็บข้อมูล

บรรณานุกรม

[1]กิตติ ภัคดีวัฒนะกุล, จำลอง ครูอุตสาหะ. **คัมภีร์ระบบฐานข้อมูล** พ.ศ 2542. กรุงเทพมหานคร

:ไทยเจริญการพิมพ์. 2542 (ตัวอย่างหนังสือทั่วไป)

[2]สุจิตรา อุดลย์เกษม และคณะ. **ระบบฐานข้อมูล (DATABASE SYSTEMS) (ฉบับปรับปรุงใหม่)**พ.ศ 2560. กรุงเทพมหานคร

:สำนักพิมพ์จุฬาลงกรณ์มหาวิทยาลัย. 2560 (ตัวอย่างหนังสือทั่วไป)