UTM

UNIVERSITI TEKNOLOGI MALAYSIA

SECD2525 - DATABASE

SEMESTER I (2023/2024)

PROPOSAL PHASE 3

DATABASE LOGICAL DESIGN

LECTURER'S NAME :

DR. IZYAN IZZATI BINTI KAMSANI

SECTION :

02

GROUP NAME :

JAVA 21

GROUP MEMBERS:

NAME	MATRIC NO.
MUHAMMAD ARIF BIN AHMAD TERMIZI	B23CS0051
NUR AIMI AFIQAH BINTI RAWI	B23CS0065
NUR BALQIS BATRISYIA BINTI SAIDIN SYAZLI	B23CS0066
WAN MUHAMMAD ADAM ZIKRI BIN IWAN SETIAWAN	B23CS0077

TABLE OF CONTENT

1.0 Introduction

In the ever-expanding realm of cinematic entertainment, the vast array of movies available can be both a blessing and a challenge for enthusiasts seeking their next cinematic experience. As the volume of films continues to grow, the need for an efficient and comprehensive Movie Search Engine becomes increasingly imperative. This database project aims to address this demand by developing a sophisticated and user-friendly Movie Search Engine that empowers users to explore and discover films based on various criteria.

The Movie Search Engine is designed to be a centralized hub for cinephiles, offering a seamless and intuitive platform to search for movies, access relevant information, and make informed viewing choices. By harnessing the power of a robust database, this project seeks to organize and index an extensive collection of movies, allowing users to navigate through genres, directors, actors, release years, and more, with ease.

The primary objective of this endeavor is to provide users with a tailored and personalized movie-watching experience. Through advanced search algorithms and user-friendly interfaces, the Movie Search Engine aims to deliver precise and relevant results, ensuring that users can quickly identify movies that match their preferences. Additionally, the project aims to enhance the overall user experience by incorporating features such as reviews, ratings, and recommendations, fostering a sense of community and interaction among movie enthusiasts.

As technology continues to evolve, so too does the landscape of the film industry. This Movie Search Engine project is positioned at the intersection of cinema and information technology, offering a dynamic and responsive solution to the challenges posed by the ever-expanding cinematic universe. By amalgamating the richness of film data with cutting-edge search functionalities, this project aspires to redefine the way users engage with and discover movies, ushering in a new era of cinematic exploration.

2.0 Overview of project

In the dynamic and ever-evolving realm of entertainment, the need for efficient and comprehensive movie search engines has become paramount. As cinema continues to produce an abundance of content, organizing and accessing this vast repository of information is a challenging task. The Movie Search Engine Database Project aims to address this challenge by implementing a robust and sophisticated database system that facilitates seamless exploration and retrieval of movie-related data.

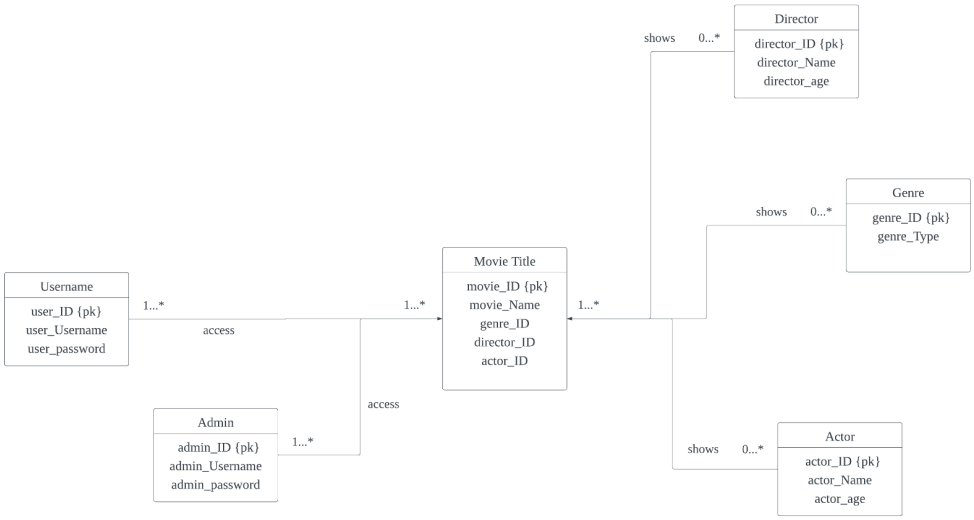
The Movie Search Engine Database Project is driven by the desire to offer users a centralized platform where they can effortlessly search, filter, and discover movies based on various criteria such as genre, release year, director, actor, and user ratings. The project is poised to revolutionize the way users interact with movie databases, providing a sophisticated and intuitive interface for movie enthusiasts.

3.0 Database conceptual design

3.1 Updated business rule

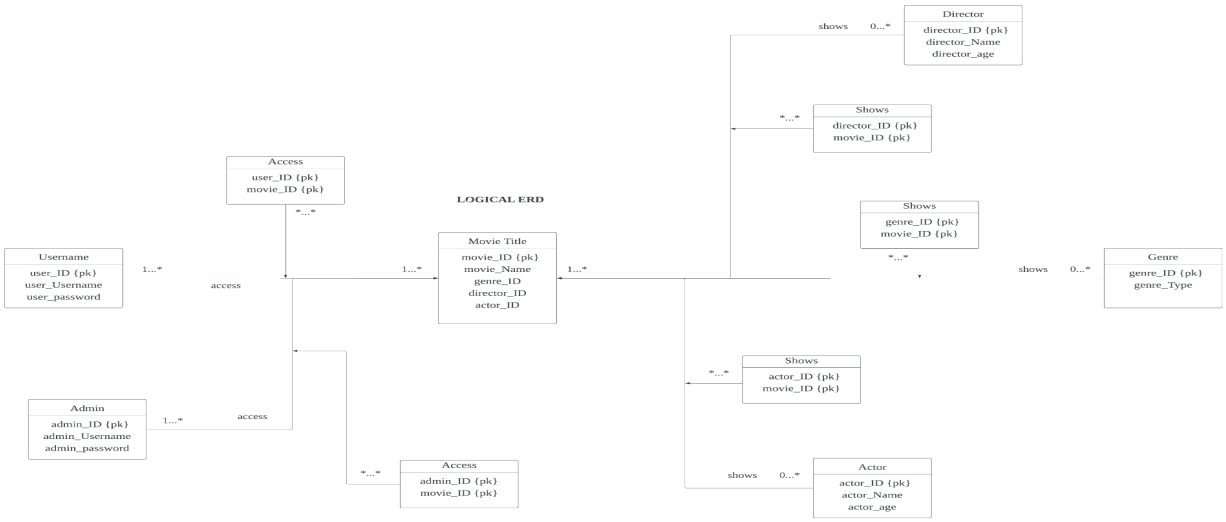
- A user can access many movie title.
- A movie title can be accessed by a user.
- An admin can access many movie title.
- A movie title can be accessed by one admin.
- A director can shows many movie title.
- A movie title may or may not shows many director.
- A genre can shows many movie title.
- A movie title may or may not shows many genre.
- An actor can shows many movie title.
- A movie title may or may not shows many actor.

3.2 Conceptual ERD



4.0 DB logical design

4.1 Logical ERD



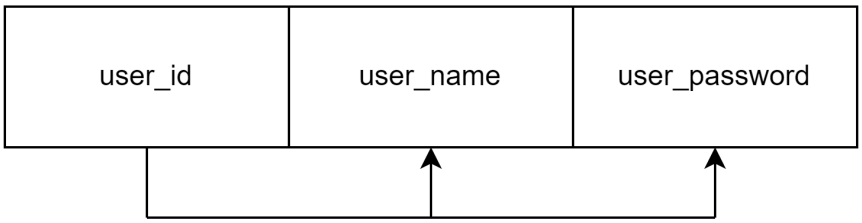
4.2 Updated Data Dictionary

Entity Name	Attributes	Description	Data Type & Length	Nulls	Multi-valued
Username	user_ID user_Name user_password	• Uniquely identifies a user • Name of the user account • Password of the user account	• 20 Number • 250 Variable characters • 250 Variable characters	• No • No • No	• No • No • No
Admin	admin_ID admin_Username admin_password	• Uniquely identifies a admin • Name of the admin account • Password of the admin account	• 20 Number • 250 Variable characters • 250 Variable characters	• No • No • No	• No • No • No
Movie	movie_ID movie_Name genre_ID director_ID actor_ID	• Uniquely identifies a movie • Name of the movie • Uniquely identifies a genre • Uniquely identifies a director • Uniquely identifies a actor	• 20 Number • 250 Variable characters • 250 Number • 250 Number • 250 Number	• No • No • No • No • No	• No • No • No • No • No
Director	director_ID director_Name director_age	• Uniquely identifies a director • Name of the director • Age of the director	• 20 Number • 250 Variable characters • 250 Variable characters	• No • No • No	• No • No • No
Genre	genre_ID genre_Type	• Uniquely identifies a genre	• 20 Number • 250 Variable	• No • No	• No • No

		• Type of genre	characters		
Actor	actor_ID actor_Name actor_age	• Uniquely identifies a actor • Name of the actor • Age of the actor	• 20 Number • 250 Variable characters • 250 Variable characters	• No • No • No	• No • No • No
Action	genre_Type	• Type of genre	• 10 Variable characters	• No	• No
Horror	genre_Type	• Type of genre	• 10 Variable characters	• No	• No
Science Fiction	genre_Type	• Type of genre	• 10 Variable characters	• No	• No
Adventure	genre_Type	• Type of genre	• 10 Variable characters	• No	• No
Comedy	genre_Type	• Type of genre	• 10 Variable characters	• No	• No
Animation	genre_Type	• Type of genre	• 10 Variable characters	• No	• No

4.3 Normalization

1. Username



USERNAME(user_id{pk},user_name,user_password)

PK : user_id

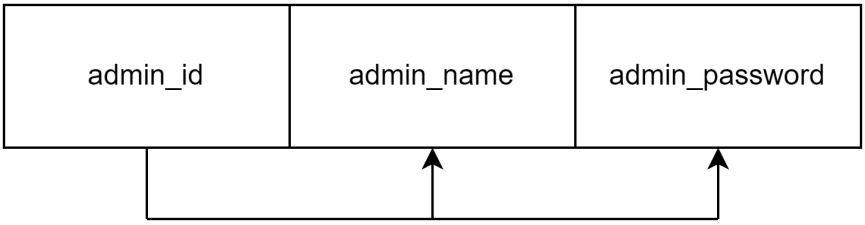
FK: -

FD1:user_id →user_name,user_password

1NF,2NF,3NF & BCNF:

USERNAME(***user_id{pk}***,user_name,user_password)

2. Admin



ADMIN(admin_id{pk},admin_name,admin_password)

PK : admin_id

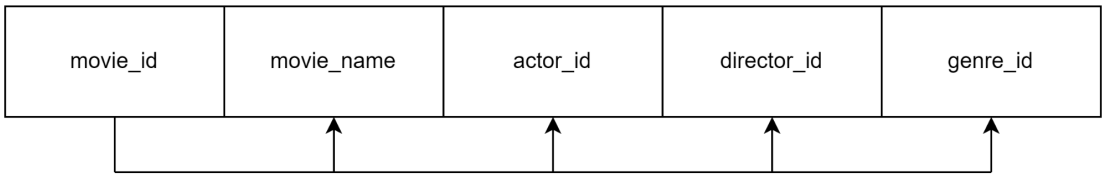
FK: -

FD1:admin_id →admin_name,admin_password

1NF,2NF,3NF & BCNF:

ADMIN(***admin_id{pk}***,admin_name,admin_password)

3. Movie Title



MOVIE_TITLE(movie_id{pk},movie_name,actor_id,director_id,genre_id)

PK : movie_id

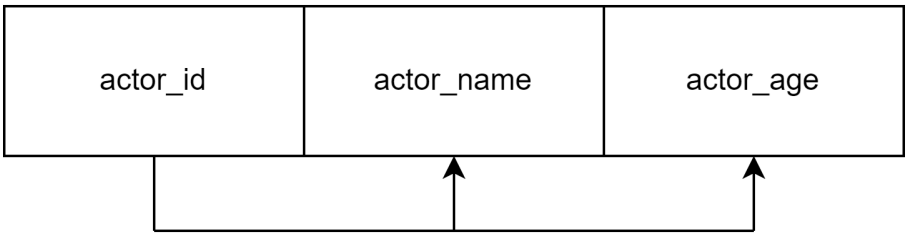
FK: actor_id,director_id,genre_id

FD1:movie_id →movie_name,actor_id,director_id,genre_id

1NF,2NF,3NF & BCNF:

MOVIE_TITLE(***movie_id{pk}***,movie_name,actor_id,director_id,genre_id)

4. Actor



ACTOR(actor_id{pk},actor_name,actor_age)

PK : actor_id

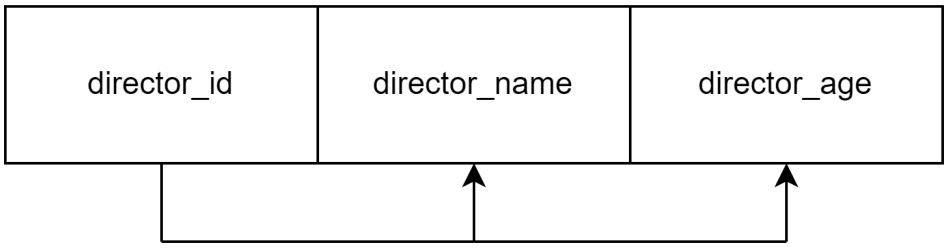
FK: -

FD1:actor_id →actor_name,actor_age

1NF,2NF,3NF & BCNF:

ACTOR(***actor_id{pk}***,actor_name,actor_age)

5. Director



DIRECTOR(director_id{pk},director_name,director_age)

PK : director_id

FK: -

FD1:director_id →director_name,director_age

1NF,2NF,3NF & BCNF:

DIRECTOR(***director_id{pk}***,director_name,director_age)

6. Genre

GENRE(genre_id{pk},genre_type)

PK : genre_id

FK: -

FD1:genre_id →genre_type

1NF,2NF,3NF & BCNF:

GENRE(***genre_id{pk}***,genre_type)

5.0 Relational DB Schemas (after normalization)

Username(user_ID, user_Name, user_password)

PK = user_ID

Admin(admin_ID, admin_Username, admin_password)

PK = admin_ID

Movie(movie_ID, movie_Name, genre_ID, director_ID, actor_ID)

PK = movie_ID

FK = genre_ID, director_ID, actor_ID

Director(director_ID, director_Name, director_age)

PK = director_ID

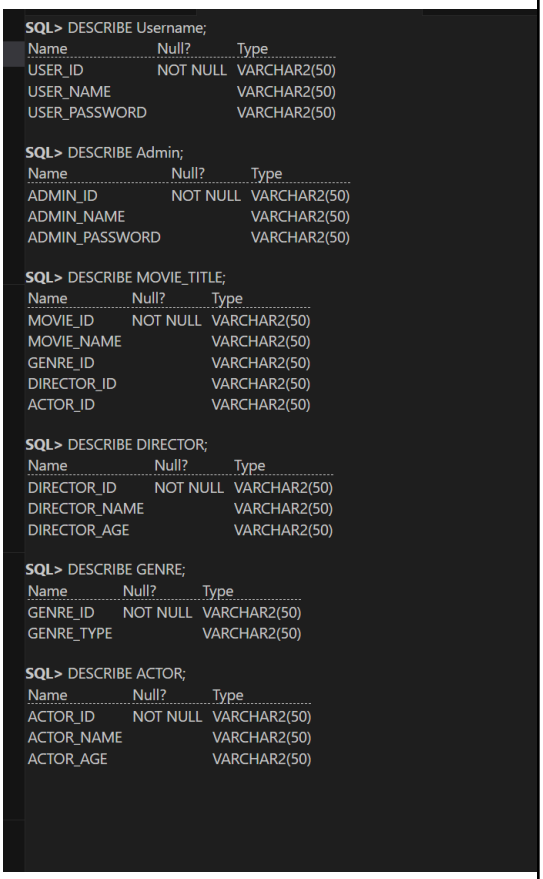
Genre(genre_ID, genre_Type)

PK = genre_ID

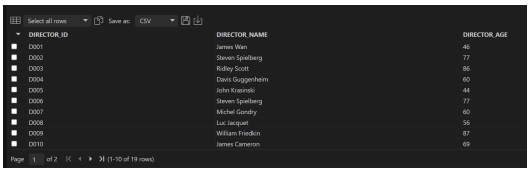
Actor(actor_ID, actor_Name, actor_age)

PK = actor_ID

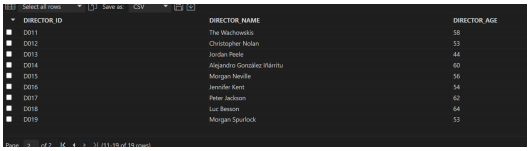
6.0 SQL Statements (DDL & DML)

	DDL	OUTPUT
1.0	**CREATE TABLE** Username (User_ID NUMBER, User_Name VARCHAR2(**250**), User_Password VARCHAR2(**250**), **CONSTRAINT** PK_User **PRIMARY KEY** (User_ID)); **CREATE TABLE ADMIN**(ADMIN_ID NUMBER(**20**), ADMIN_NAME VARCHAR2(**250**), ADMIN_PASSWORD VARCHAR2(**250**), **CONSTRAINT** PK_ADMIN **PRIMARY KEY** (ADMIN_ID)); **CREATE TABLE** MOVIE_TITLE(MOVIE_ID NUMBER(**20**), MOVIE_NAME VARCHAR2(**250**), GENRE_ID NUMBER(**250**), DIRECTOR_ID NUMBER(**250**), ACTOR_ID NUMBER(**250**), **CONSTRAINT** PK_MOVIETITLE **PRIMARY KEY** (MOVIE_ID)); **CREATE TABLE** DIRECTOR(DIRECTOR_ID NUMBER(**20**), DIRECTOR_NAME VARCHAR2(**250**), DIRECTOR_AGE VARCHAR2(**250**), **CONSTRAINT** PK_DIRECTOR **PRIMARY KEY** (DIRECTOR_ID)); **CREATE TABLE** GENRE(GENRE_ID NUMBER(**20**), GENRE_TYPE VARCHAR2(**250**), **CONSTRAINT** PK_GENRE **PRIMARY KEY** (GENRE_ID)); **CREATE TABLE** ACTOR(ACTOR_ID NUMBER(**250**), ACTOR_NAME VARCHAR2(**250**), ACTOR_AGE VARCHAR2(**250**), **CONSTRAINT** PK_ACTOR **PRIMARY KEY** (ACTOR_ID));	
	DML	**OUTPUT**
1.0	**INSERT INTO** Director(DIRECTOR_ID,DIRECTOR_NAME,DIRECTOR_AGE) **VALUES** ('D001', 'James Wan', '46'); **INSERT INTO** Director(DIRECTOR_ID,DIRECTOR_NAME,DIRECTOR_AGE)	

```
VALUES ('D002', 'Steven Spielberg', '77');

INSERT INTO
Director(DIRECTOR_ID,DIRECTOR_NAME,DIRECTOR_AGE)
VALUES ('D003', 'Ridley Scott', '86');

INSERT INTO
Director(DIRECTOR_ID,DIRECTOR_NAME,DIRECTOR_AGE)
VALUES ('D004', 'Davis Guggenheim', '60');

INSERT INTO
Director(DIRECTOR_ID,DIRECTOR_NAME,DIRECTOR_AGE)
VALUES ('D005', 'John Krasinski', '44');

INSERT INTO
Director(DIRECTOR_ID,DIRECTOR_NAME,DIRECTOR_AGE)
VALUES ('D006', 'Steven Spielberg', '77');

INSERT INTO
Director(DIRECTOR_ID,DIRECTOR_NAME,DIRECTOR_AGE)
VALUES ('D007', 'Michel Gondry', '60');

INSERT INTO
Director(DIRECTOR_ID,DIRECTOR_NAME,DIRECTOR_AGE)
VALUES ('D008', 'Luc Jacquet', '56');

INSERT INTO
Director(DIRECTOR_ID,DIRECTOR_NAME,DIRECTOR_AGE)
VALUES ('D009', 'William Friedkin', '87');

INSERT INTO
Director(DIRECTOR_ID,DIRECTOR_NAME,DIRECTOR_AGE)
VALUES ('D010', 'James Cameron', '69');

INSERT INTO
Director(DIRECTOR_ID,DIRECTOR_NAME,DIRECTOR_AGE)
VALUES ('D011', 'The Wachowskis', '58');

INSERT INTO
Director(DIRECTOR_ID,DIRECTOR_NAME,DIRECTOR_AGE)
VALUES ('D012', 'Christopher Nolan', '53');

INSERT INTO
Director(DIRECTOR_ID,DIRECTOR_NAME,DIRECTOR_AGE)
VALUES ('D013', 'Jordan Peele', '44');

INSERT INTO
Director(DIRECTOR_ID,DIRECTOR_NAME,DIRECTOR_AGE)
VALUES ('D014', 'Alejandro González Iñárritu', '60');

INSERT INTO
Director(DIRECTOR_ID,DIRECTOR_NAME,DIRECTOR_AGE)
VALUES ('D015', 'Morgan Neville', '56');

INSERT INTO
Director(DIRECTOR_ID,DIRECTOR_NAME,DIRECTOR_AGE)
VALUES ('D016', 'Jennifer Kent', '54');

INSERT INTO
Director(DIRECTOR_ID,DIRECTOR_NAME,DIRECTOR_AGE)
VALUES ('D017', 'Peter Jackson', '62');
```

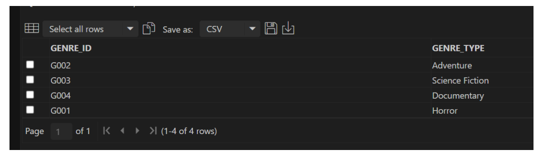
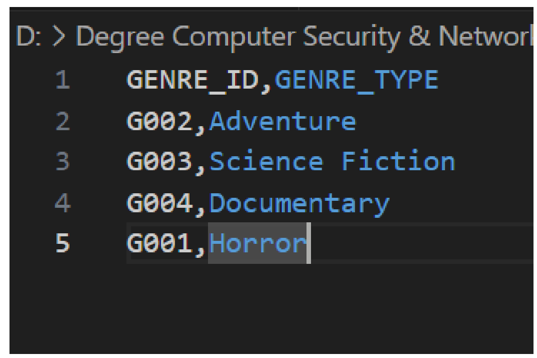


DIRECTOR_ID	DIRECTOR_NAME	DIRECTOR_AGE
D001	James Wan	46
D002	Steven Spielberg	77
D003	Ridley Scott	86
D004	Davis Guggenheim	60
D005	John Krasinski	44
D006	Steven Spielberg	77
D007	Michel Gondry	60
D008	Luc Jacquet	56
D009	William Friedkin	87
D010	James Cameron	69



DIRECTOR_ID	DIRECTOR_NAME	DIRECTOR_AGE
D011	The Wachowskis	58
D012	Christopher Nolan	53
D013	Jordan Peele	44
D014	Alejandro González Iñárritu	60
D015	Morgan Neville	56
D016	Jennifer Kent	54
D017	Peter Jackson	62
D018	Luc Besson	64
D019	Morgan Spurlock	53

```
DIRECTOR_ID,DIRECTOR_NAME,DIRECTOR_AGE
D001,James Wan,46
D002,Steven Spielberg,77
D003,Ridley Scott,86
D004,Davis Guggenheim,60
D005,John Krasinski,44
D006,Steven Spielberg,77
D007,Michel Gondry,60
D008,Luc Jacquet,56
D009,William Friedkin,87
D010,James Cameron,69
D011,The Wachowskis,58
D012,Christopher Nolan,53
D013,Jordan Peele,44
D014,Alejandro González Iñárritu,60
D015,Morgan Neville,56
D016,Jennifer Kent,54
D017,Peter Jackson,62
D018,Luc Besson,64
D019,Morgan Spurlock,53
```

	INSERT INTO Director(DIRECTOR_ID,DIRECTOR_NAME,DIRECTOR_AGE) **VALUES** ('D018', 'Luc Besson', '64'); **INSERT INTO** Director(DIRECTOR_ID,DIRECTOR_NAME,DIRECTOR_AGE) **VALUES** ('D019', 'Morgan Spurlock', '53');	
2.0	**INSERT INTO** Genre(GENRE_ID,GENRE_TYPE) **VALUES** ('G001', 'Horror'); **INSERT INTO** Genre(GENRE_ID,GENRE_TYPE) **VALUES** ('G002', 'Adventure'); **INSERT INTO** Genre(GENRE_ID,GENRE_TYPE) **VALUES** ('G003', 'Science Fiction'); **INSERT INTO** Genre(GENRE_ID,GENRE_TYPE) **VALUES** ('G004', 'Documentary'); **INSERT INTO** Genre(GENRE_ID,GENRE_TYPE) **VALUES** ('G001', 'Horror'); **INSERT INTO** Genre(GENRE_ID,GENRE_TYPE) **VALUES** ('G002', 'Adventurer'); **INSERT INTO** Genre(GENRE_ID,GENRE_TYPE) **VALUES** ('G003', 'Science Fiction'); **INSERT INTO** Genre(GENRE_ID,GENRE_TYPE) **VALUES** ('G004', 'Documentary'); **INSERT INTO** Genre(GENRE_ID,GENRE_TYPE) **VALUES** ('G001', 'Horror'); **INSERT INTO** Genre(GENRE_ID,GENRE_TYPE) **VALUES** ('G002', 'Adventurer'); **INSERT INTO** Genre(GENRE_ID,GENRE_TYPE) **VALUES** ('G003', 'Science Fiction'); **INSERT INTO** Genre(GENRE_ID,GENRE_TYPE) **VALUES** ('G004', 'Documentary'); **INSERT INTO** Genre(GENRE_ID,GENRE_TYPE) **VALUES** ('G001', 'Horror'); **INSERT INTO** Genre(GENRE_ID,GENRE_TYPE) **VALUES** ('G002', 'Adventurer'); **INSERT INTO** Genre(GENRE_ID,GENRE_TYPE) **VALUES** ('G003', 'Science Fiction'); **INSERT INTO** Genre(GENRE_ID,GENRE_TYPE) **VALUES** ('G004', 'Documentary'); **INSERT INTO** Genre(GENRE_ID,GENRE_TYPE) **VALUES** ('G001', 'Horror');	 

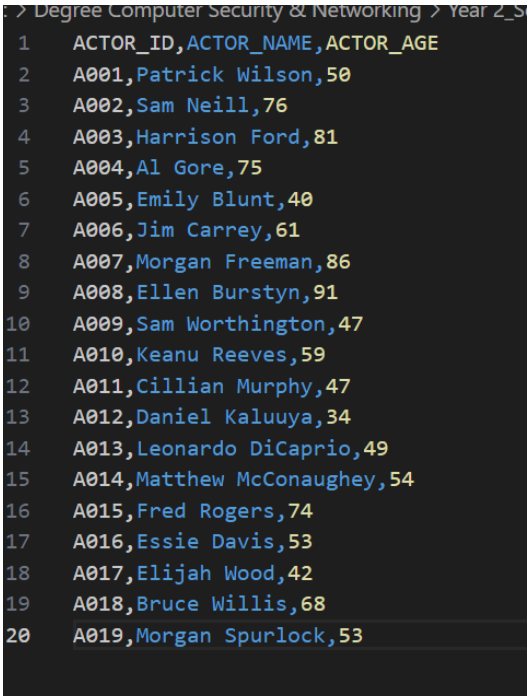
```
INSERT INTO Genre(GENRE_ID,GENRE_TYPE)
VALUES ('G002', 'Adventurer');

INSERT INTO Genre(GENRE_ID,GENRE_TYPE)
VALUES ('G003', 'Science Fiction');

INSERT INTO Genre(GENRE_ID,GENRE_TYPE)
VALUES ('G004', 'Documentary');
```

3.0

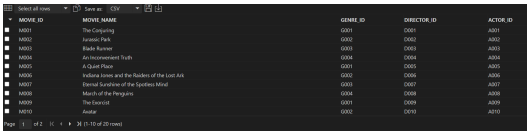
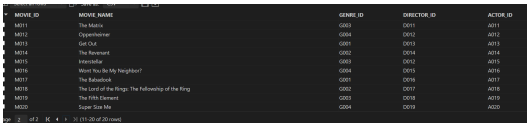
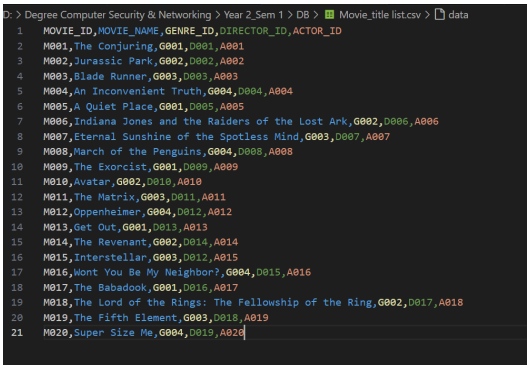
```
INSERT INTO Actor(ACTOR_ID,ACTOR_NAME,ACTOR_AGE)
VALUES ('A001', 'Patrick Wilson','50');

INSERT INTO Actor(ACTOR_ID,ACTOR_NAME,ACTOR_AGE)
VALUES ('A002', 'Sam Neill','76');

INSERT INTO Actor(ACTOR_ID,ACTOR_NAME,ACTOR_AGE)
VALUES ('A003', 'Harrison Ford','81');

INSERT INTO Actor(ACTOR_ID,ACTOR_NAME,ACTOR_AGE)
VALUES ('A004', 'Al Gore','75');

INSERT INTO Actor(ACTOR_ID,ACTOR_NAME,ACTOR_AGE)
VALUES ('A005', 'Emily Blunt','40');

INSERT INTO Actor(ACTOR_ID,ACTOR_NAME,ACTOR_AGE)
VALUES ('A006', 'Jim Carrey','61');

INSERT INTO Actor(ACTOR_ID,ACTOR_NAME,ACTOR_AGE)
VALUES ('A007', 'Morgan Freeman','86');

INSERT INTO Actor(ACTOR_ID,ACTOR_NAME,ACTOR_AGE)
VALUES ('A008', 'Ellen Burstyn','91');

INSERT INTO Actor(ACTOR_ID,ACTOR_NAME,ACTOR_AGE)
VALUES ('A009', 'Sam Worthington','47');

INSERT INTO Actor(ACTOR_ID,ACTOR_NAME,ACTOR_AGE)
VALUES ('A010', 'Keanu Reeves','59');

INSERT INTO Actor(ACTOR_ID,ACTOR_NAME,ACTOR_AGE)
VALUES ('A011', 'Cillian Murphy','47');

INSERT INTO Actor(ACTOR_ID,ACTOR_NAME,ACTOR_AGE)
VALUES ('A012', 'Daniel Kaluuya','34');

INSERT INTO Actor(ACTOR_ID,ACTOR_NAME,ACTOR_AGE)
VALUES ('A013', 'Leonardo DiCaprio','49');

INSERT INTO Actor(ACTOR_ID,ACTOR_NAME,ACTOR_AGE)
VALUES ('A014', 'Matthew McConaughey','54');

INSERT INTO Actor(ACTOR_ID,ACTOR_NAME,ACTOR_AGE)
VALUES ('A015', 'Fred Rogers','74');

INSERT INTO Actor(ACTOR_ID,ACTOR_NAME,ACTOR_AGE)
VALUES ('A016', 'Essie Davis','53');

INSERT INTO Actor(ACTOR_ID,ACTOR_NAME,ACTOR_AGE)
VALUES ('A017', 'Elijah Wood','42');
```

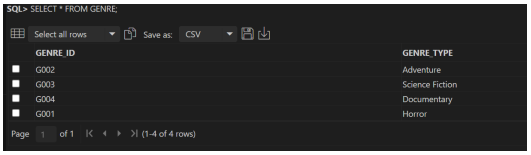
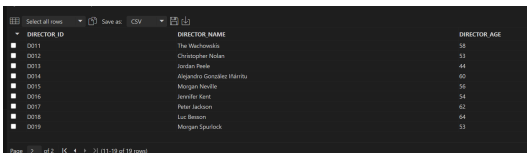
ACTOR_ID	ACTOR_NAME	ACTOR_AGE
A001	Patrick Wilson	50
A002	Sam Neill	76
A003	Harrison Ford	81
A004	Al Gore	75
A005	Emily Blunt	40
A006	Jim Carrey	61
A007	Morgan Freeman	86
A008	Ellen Burstyn	91
A009	Sam Worthington	47
A010	Keanu Reeves	59

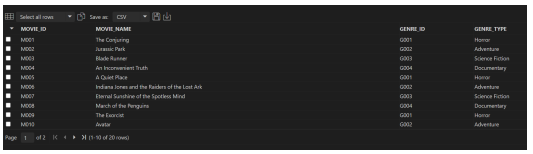
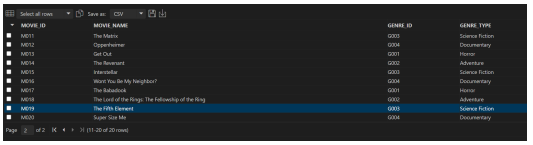
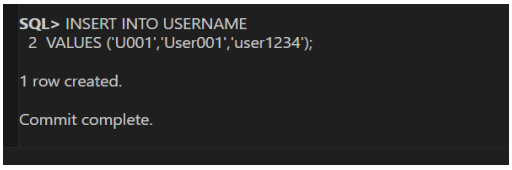
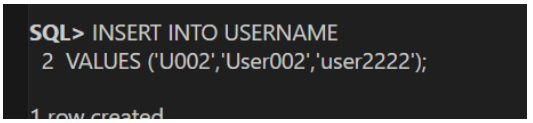
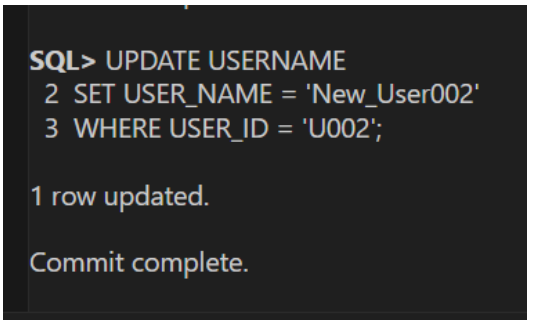
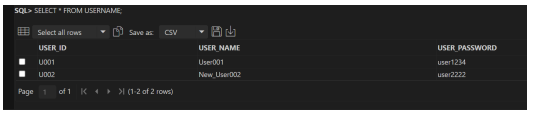
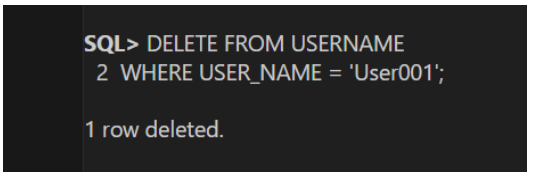
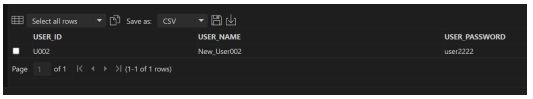
ACTOR_ID	ACTOR_NAME	ACTOR_AGE
A011	Cillian Murphy	47
A012	Daniel Kaluuya	34
A013	Leonardo DiCaprio	49
A014	Matthew McConaughey	54
A015	Fred Rogers	74
A016	Essie Davis	53
A017	Elijah Wood	42
A018	Bruce Willis	68
A019	Morgan Spurlock	53



```
ACTOR_ID,ACTOR_NAME,ACTOR_AGE
A001,Patrick Wilson,50
A002,Sam Neill,76
A003,Harrison Ford,81
A004,Al Gore,75
A005,Emily Blunt,40
A006,Jim Carrey,61
A007,Morgan Freeman,86
A008,Ellen Burstyn,91
A009,Sam Worthington,47
A010,Keanu Reeves,59
A011,Cillian Murphy,47
A012,Daniel Kaluuya,34
A013,Leonardo DiCaprio,49
A014,Matthew McConaughey,54
A015,Fred Rogers,74
A016,Essie Davis,53
A017,Elijah Wood,42
A018,Bruce Willis,68
A019,Morgan Spurlock,53
```

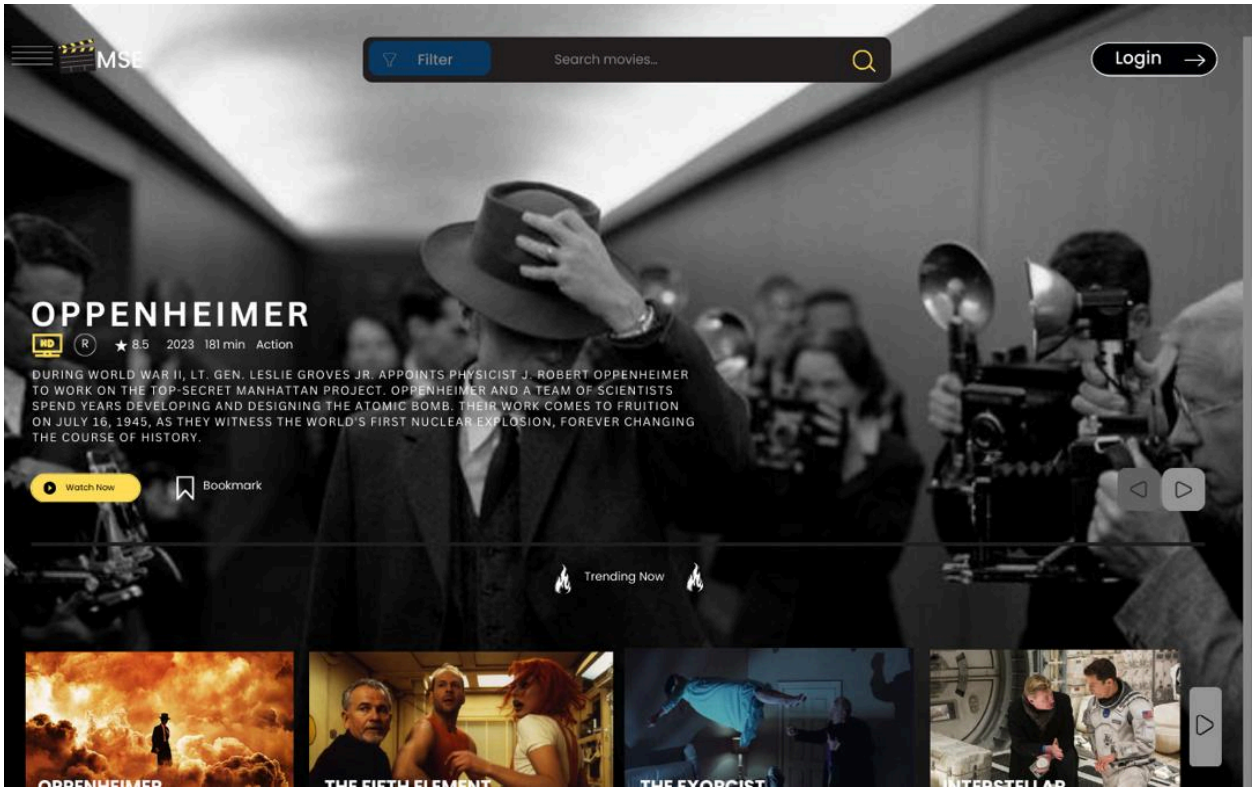
	INSERT INTO Actor(ACTOR_ID,ACTOR_NAME,ACTOR_AGE) **VALUES ('A018', 'Bruce Willis','68');** **INSERT INTO** Actor(ACTOR_ID,ACTOR_NAME,ACTOR_AGE) **VALUES ('A019', 'Morgan Spurlock','53');**	
4.0	**INSERT INTO** Movie_title(MOVIE_ID, MOVIE_NAME, DIRECTOR_ID, ACTOR_ID, GENRE_ID) **VALUES** ('M01','The Conjuring','D001','A001','G001'); **INSERT INTO** Movie_title(MOVIE_ID, MOVIE_NAME, DIRECTOR_ID, ACTOR_ID, GENRE_ID) **VALUES** ('M002', 'Jurassic Park', 'D002', 'A002', 'G002'); **INSERT INTO** Movie_title(MOVIE_ID, MOVIE_NAME, DIRECTOR_ID, ACTOR_ID, GENRE_ID) **VALUES** ('M003', 'Blade Runner', 'D003', 'A003', 'G003'); **INSERT INTO** Movie_title(MOVIE_ID, MOVIE_NAME, DIRECTOR_ID, ACTOR_ID, GENRE_ID) **VALUES** ('M004', 'An Inconvenient Truth', 'D004', 'A004', 'G004'); **INSERT INTO** Movie_title(MOVIE_ID, MOVIE_NAME, DIRECTOR_ID, ACTOR_ID, GENRE_ID) **VALUES** ('M005', 'A Quiet Place', 'D005', 'A005', 'G001'); **INSERT INTO** Movie_title(MOVIE_ID, MOVIE_NAME, DIRECTOR_ID, ACTOR_ID, GENRE_ID) **VALUES** ('M006', 'Indiana Jones and the Raiders of the Lost Ark', 'D006', 'A003', 'G002'); **INSERT INTO** Movie_title(MOVIE_ID, MOVIE_NAME, DIRECTOR_ID, ACTOR_ID, GENRE_ID) **VALUES** ('M007', 'Eternal Sunshine of the Spotless Mind', 'D007', 'A006', 'G003'); **INSERT INTO** Movie_title(MOVIE_ID, MOVIE_NAME, DIRECTOR_ID, ACTOR_ID, GENRE_ID) **VALUES** ('M008', 'March of the Penguins', 'D008', 'A007', 'G004'); **INSERT INTO** Movie_title(MOVIE_ID, MOVIE_NAME, DIRECTOR_ID, ACTOR_ID, GENRE_ID) **VALUES** ('M009', 'The Exorcist', 'D009', 'A008', 'G001'); **INSERT INTO** Movie_title(MOVIE_ID, MOVIE_NAME, DIRECTOR_ID, ACTOR_ID, GENRE_ID) **VALUES** ('M010', 'Avatar', 'D010', 'A009', 'G002'); **INSERT INTO** Movie_title(MOVIE_ID, MOVIE_NAME, DIRECTOR_ID, ACTOR_ID, GENRE_ID) **VALUES** ('M011', 'The Matrix', 'D011', 'A010', 'G003'); **INSERT INTO** Movie_title(MOVIE_ID, MOVIE_NAME, DIRECTOR_ID, ACTOR_ID, GENRE_ID) **VALUES** ('M012', 'Oppenheimer', 'D011', 'A012', 'G004'); **INSERT INTO** Movie_title(MOVIE_ID, MOVIE_NAME, DIRECTOR_ID, ACTOR_ID, GENRE_ID) **VALUES** ('M013', 'Get Out', 'D013', 'A012', 'G001');	   D: > Degree Computer Security & Networking > Year 2_Sem 1 > DB > Movie_title list.csv > data MOVIE_ID,MOVIE_NAME,GENRE_ID,DIRECTOR_ID,ACTOR_ID M001,The Conjuring,G001,D001,A001 M002,Jurassic Park,G002,D002,A002 M003,Blade Runner,G003,D003,A003 M004,An Inconvenient Truth,G004,D004,A004 M005,A Quiet Place,G001,D005,A005 M006,Indiana Jones and the Raiders of the Lost Ark,G002,D006,A006 M007,Eternal Sunshine of the Spotless Mind,G003,D007,A007 M008,March of the Penguins,G004,D008,A008 M009,The Exorcist,G001,D009,A009 M010,Avatar,G002,D010,A010 M011,The Matrix,G003,D011,A011 M012,Oppenheimer,G004,D012,A012 M013,Get Out,G001,D013,A013 M014,The Revenant,G002,D014,A014 M015,Interstellar,G003,D012,A015 M016,Wont You Be My Neighbor?,G004,D015,A016 M017,The Babadook,G001,D016,A017 M018,The Lord of the Rings: The Fellowship of the Ring,G002,D017,A018 M019,The Fifth Element,G003,D018,A019 M020,Super Size Me,G004,D019,A020

	INSERT INTO Movie_title(MOVIE_ID, MOVIE_NAME, DIRECTOR_ID, ACTOR_ID, GENRE_ID) **VALUES** ('M014', 'The Revenant', 'D014', 'A013', 'G002'); **INSERT INTO** Movie_title(MOVIE_ID, MOVIE_NAME, DIRECTOR_ID, ACTOR_ID, GENRE_ID) **VALUES** ('M015', 'Interstellar', 'D012', 'A014', 'G003'); **INSERT INTO** Movie_title(MOVIE_ID, MOVIE_NAME, DIRECTOR_ID, ACTOR_ID, GENRE_ID) **VALUES** ('M016', 'Wont You Be My Neighbor?', 'D015', 'A015', 'G004'); **INSERT INTO** Movie_title(MOVIE_ID, MOVIE_NAME, DIRECTOR_ID, ACTOR_ID, GENRE_ID) **VALUES** ('M017', 'The Babadook', 'D016', 'A016', 'G001'); **INSERT INTO** Movie_title(MOVIE_ID, MOVIE_NAME, DIRECTOR_ID, ACTOR_ID, GENRE_ID) **VALUES** ('M018', 'The Lord of the Rings: The Fellowship of the Ring', 'D017', 'A017', 'G002'); **INSERT INTO** Movie_title(MOVIE_ID, MOVIE_NAME, DIRECTOR_ID, ACTOR_ID, GENRE_ID) **VALUES** ('M019', 'The Fifth Element', 'D018', 'A018', 'G003'); **INSERT INTO** Movie_title(MOVIE_ID, MOVIE_NAME, DIRECTOR_ID, ACTOR_ID, GENRE_ID) **VALUES** ('M020', 'Super Size Me', 'D019', 'A019', 'G004');	
5.0	**SELECT** * **FROM** DIRECTOR; **SELECT** * **FROM** ACTOR; **SELECT** * **FROM** GENRE; **SELECT** * **FROM** MOVIE_TITLE;;	   

6.0	**SELECT** MOVIE_TITLE.MOVIE_ID,MOVIE_TITLE.MOVIE_NAME,MOVIE_TITLE.GENRE_ID,GENRE.GENRE_TYPE **FROM** MOVIE_TITLE **JOIN** GENRE **ON** MOVIE_TITLE.GENRE_ID = GENRE.GENRE_ID **ORDER BY** MOVIE_ID;	 
7.0	**INSERT INTO** USERNAME **VALUES ('U001','User001','user1234');** **INSERT INTO** USERNAME **VALUES ('U002','User002','user2222');**	 
8.0	**UPDATE** USERNAME **SET** USER_NAME = **'New_User002'** **WHERE** USER_ID = **'U002';**	 
9.0	**DELETE FROM** USERNAME **WHERE** USER_NAME = 'User001';	 

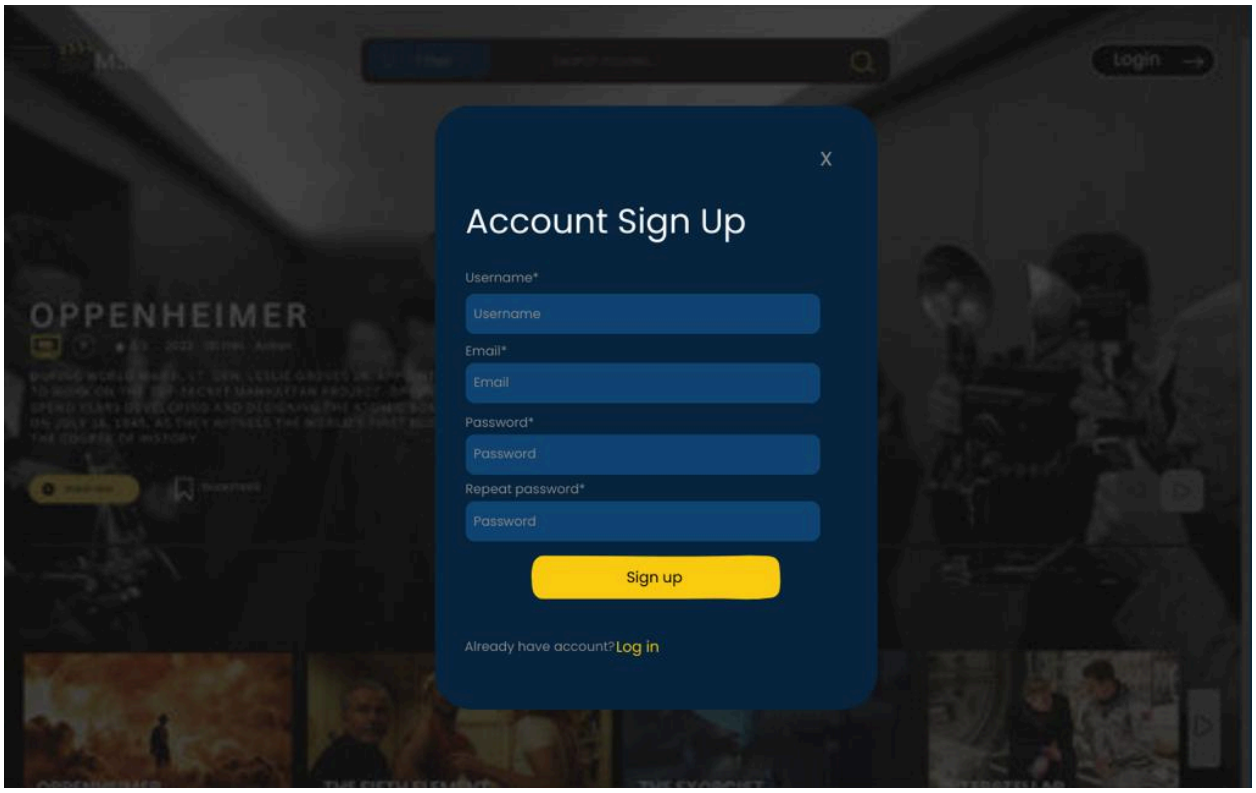
7.0 User Interface

7.1 Homepage



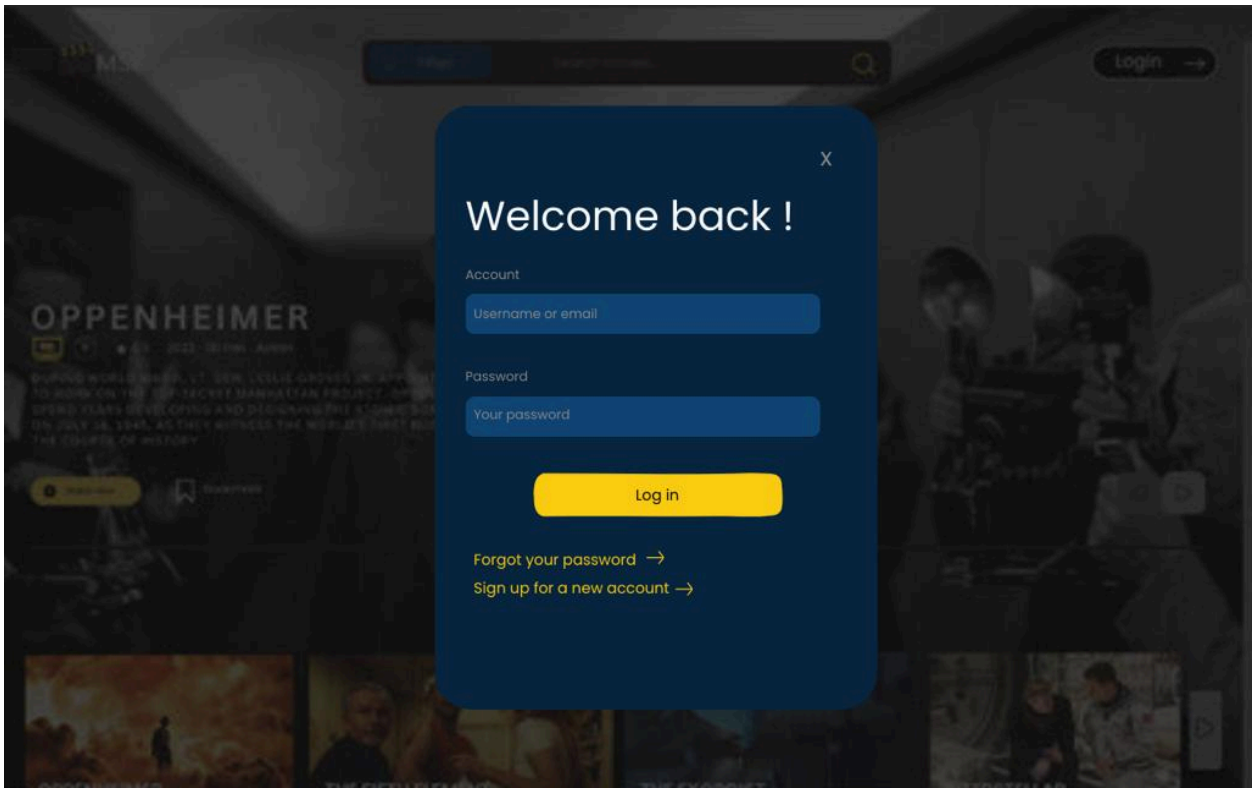
When a user goes to the website, it will display the homepage. The homepage has a search engine with filters on top of the page. There also has a login button for users to login their account.

7.2 Sign Up Page



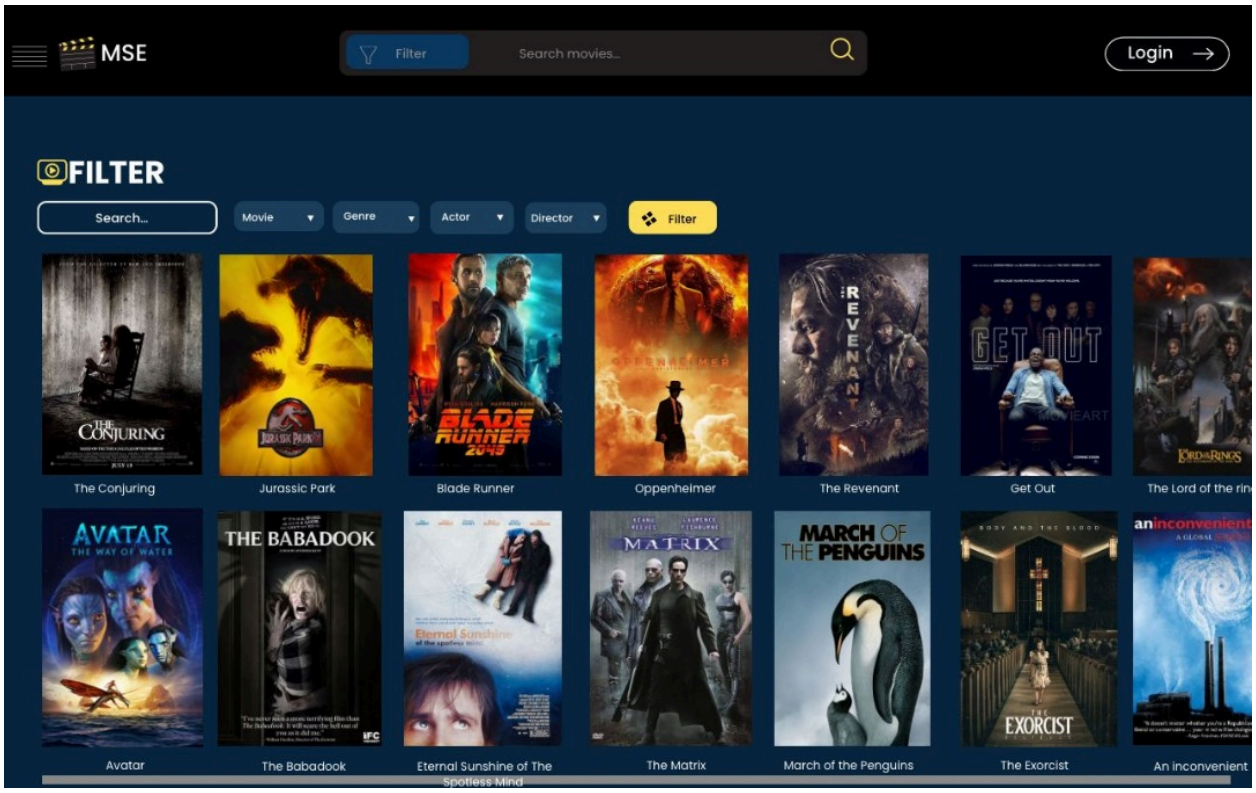
Before users login to their account, they must create their account first by sign up new account in the sign up page. They must enter the details which are username, email, password and repeat password.

7.3 Login Page



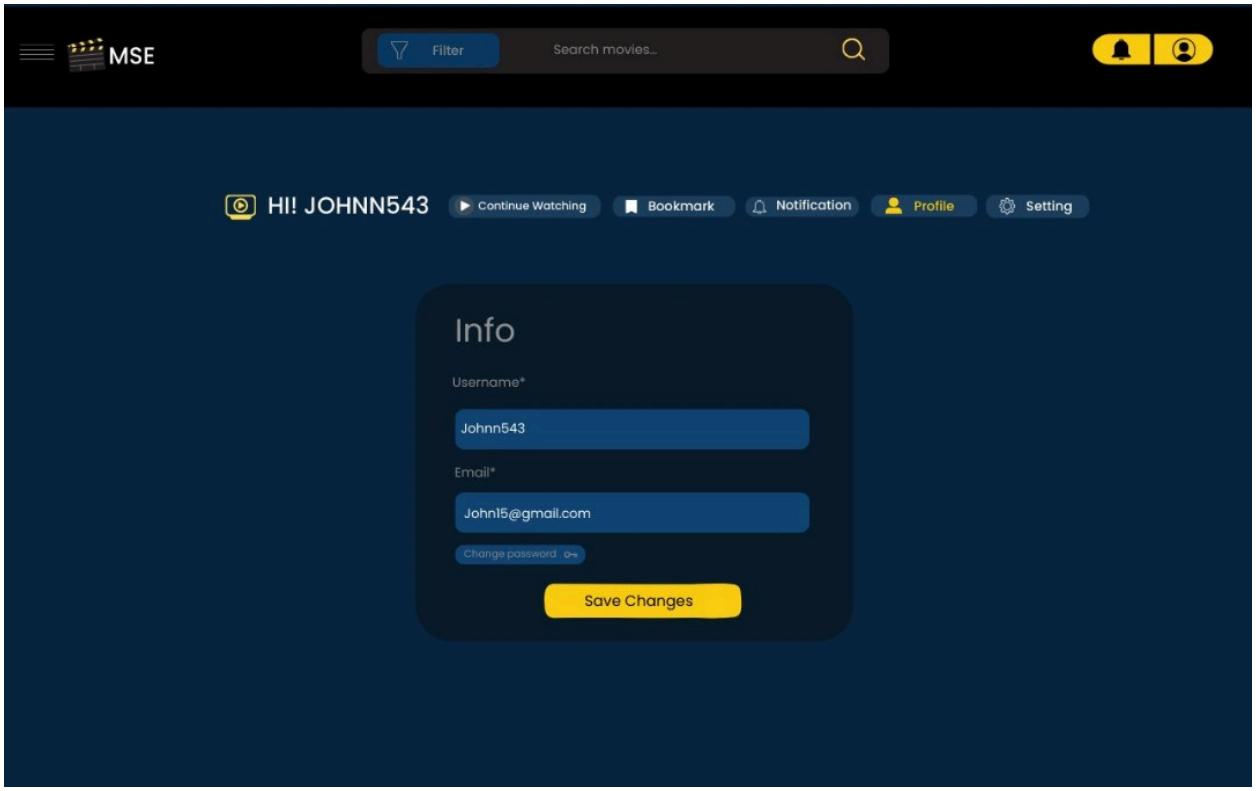
If a user wants to watch a movie, they must login their account first in the login page. They must enter their username or email and password in order to access their own account.

7.4 Movie List Page



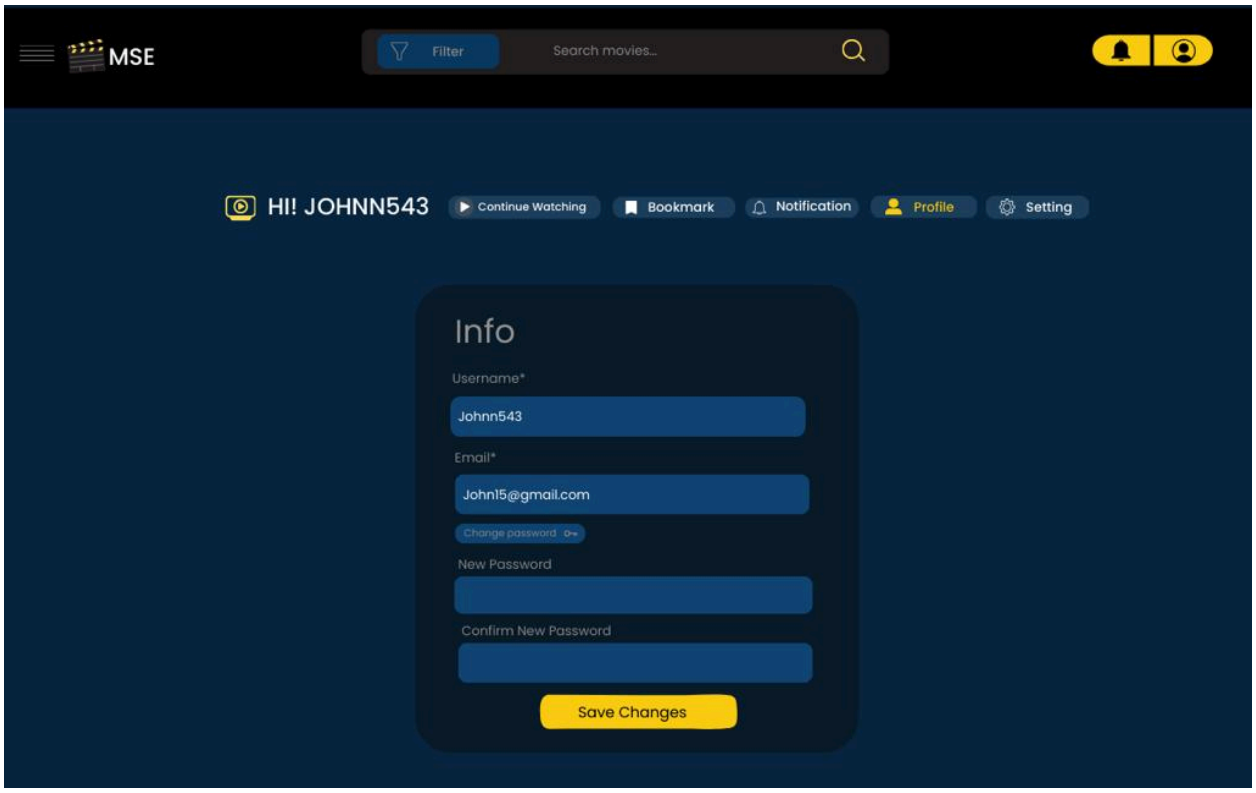
In the movie list page it will display all the movies that exist. Users can search what movie they want and also filter their movie selection by four categories which are searched by movie, genre, actor or director.

7.5 Update Username and Email



Users also can update their username and email in their profile page.

7.6 Update Password



Users can update their new password in their profile by entering their username, email, new password and confirming the new password.

8.0 Summary

In conclusion, we carried out in-depth data analysis taking into account both current business rules and existing data to guarantee a successful database conceptual design. This in-depth knowledge of the existing system made it possible to create a new, cutting-edge system that streamlines procedures. To ensure the new system runs well, we also revised the conceptual ERD and business rules. During this step, we also updated the data dictionary. We were also able to create normalization for our ERD by working on this project. The aforementioned report already has the SQL statements we used to put on it. In this project, we use the DDL and DML that we have studied in class. We also create a few user interfaces to demonstrate the SQL's functionality. Finally, we discovered that working on this project together has improved our knowledge of one another, since we will now turn to one another for support in times of need.