

Cometplex

Online Movie Rental Store

INSTRUCTOR – DR KUTSAL DOGAN



GROUP 18:

Priyanshi Khatri (pxk173330@utdallas.edu)

Sonali Gupta (sxg172531@utdallas.edu)

Raghav Seth (rxs174630@utdallas.edu)

TABLE OF CONTENTS

1. Company Background.....	3
2. Business Process.....	3
3. Process Simplified.....	5
4. Database Scope and Table Descriptions.....	5
5. Entity Relationship Diagram.....	7
6. Relational Database Schema.....	8
a) Viewer Table.....	8
b) Movie Table.....	8
c) Actor Table.....	9
d) MovieActorDetail Table.....	9
e) Rating Table.....	9
f) Promotion Table.....	10
g) Transaction Table.....	10
h) Cancellation Table.....	11
i) OrderDetail Table.....	11
j) WatchList Table.....	11
k) IMDB Table.....	12
l) GiftCard Table.....	12
7. Cometplex Main Menu.....	13
8. Data Input and Manipulation Screen Forms.....	13
9. Sample Reports.....	19
10. Contributions.....	23

1. Company Background:

Cometplex was a successful movie theatre of the of 90s in Dallas, Texas. With the development of technology and online market, it saw a decline in its business. The theatre owner Mr. KD realised the importance of the online space and understood that an online business will be cost effective, easy and convenient for both him and his customers. He is planning on closing his current business and moving it online, using his connections in the movie distribution industry. For the exclusivity of his project, he aims on providing users with the latest blockbusters which have been released recently.

He asked for help to develop a database which efficiently manages his business. To help him, we offered to develop a system called Cometplex, which is an online rental movie store offering movies of all languages and genres to the movie enthusiasts. Here are a few of the genres:

- Action
- Animation
- Biography
- Comedy
- Crime
- Drama
- Horror
- Mystery
- Romance
- Sci-Fi
- Sport
- Thriller

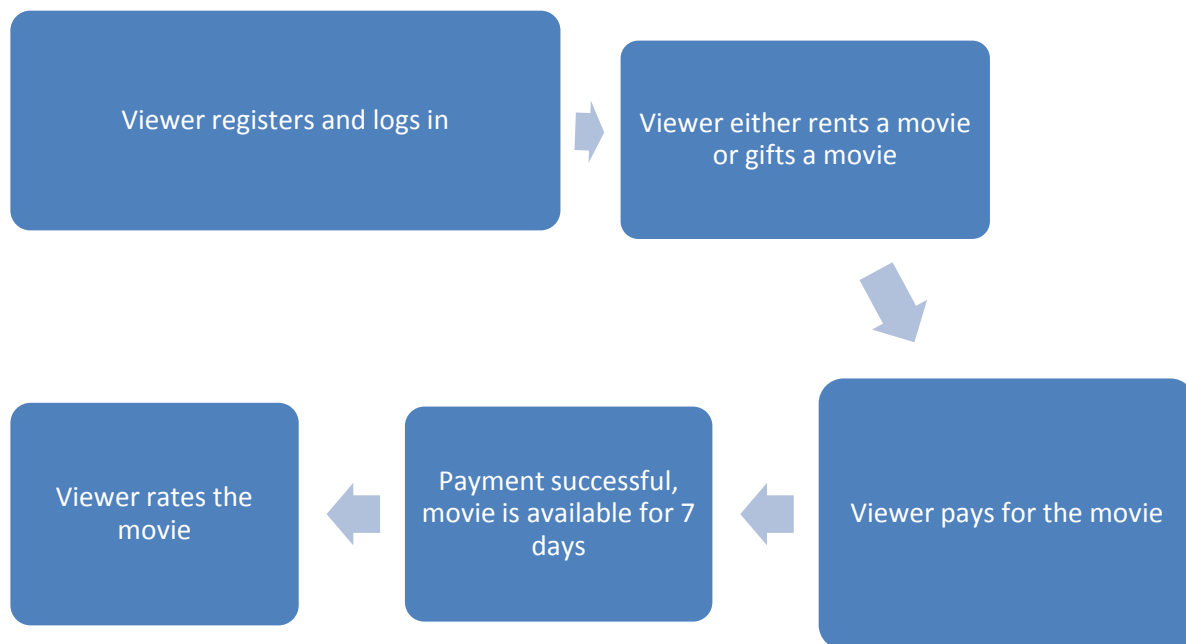
2. Business Process:

Business Process is as follows:

- Cometplex has two types of viewers: New and Existing.
- When a viewer is new, they should register on cometplex.com with their details to proceed. This, saves the information of he viewer.

- After successful registration, they can browse through all movies from different genres as per their preference.
- After selecting the desired movie, the viewer will be redirected to Payments page where they must enter their card details and billing information to rent a movie.
- After the Payment is successful, the movie is available to the viewer for 7 days.
- The order cannot be cancelled once the viewer starts watching the movie.
- The viewer's movie viewing process is updated in a Watchlist table which stores the details about when a movie was started and stopped.
- These are some of the promotions that are offered on relevant genres throughout the year for specific days, where viewers get a 50% discount on the rent of the movie:
 - Valentine's Day: Romantic, Rom-com
 - Independence Day: Patriotic, Action
 - Halloween: Horror, Thriller, Mystery
 - Thanksgiving: Drama, Fantasy
 - Christmas: Fantasy, Animation
- Movies can be rated (star rating out of 5) by the viewers after completion of a movie. Also, IMDB ratings will be available to the viewers.
- A viewer can also avail a gift card option wherein one movie can be purchased and gifted to another registered user which will be available for viewing for 7 days.

3.Process Simplified:



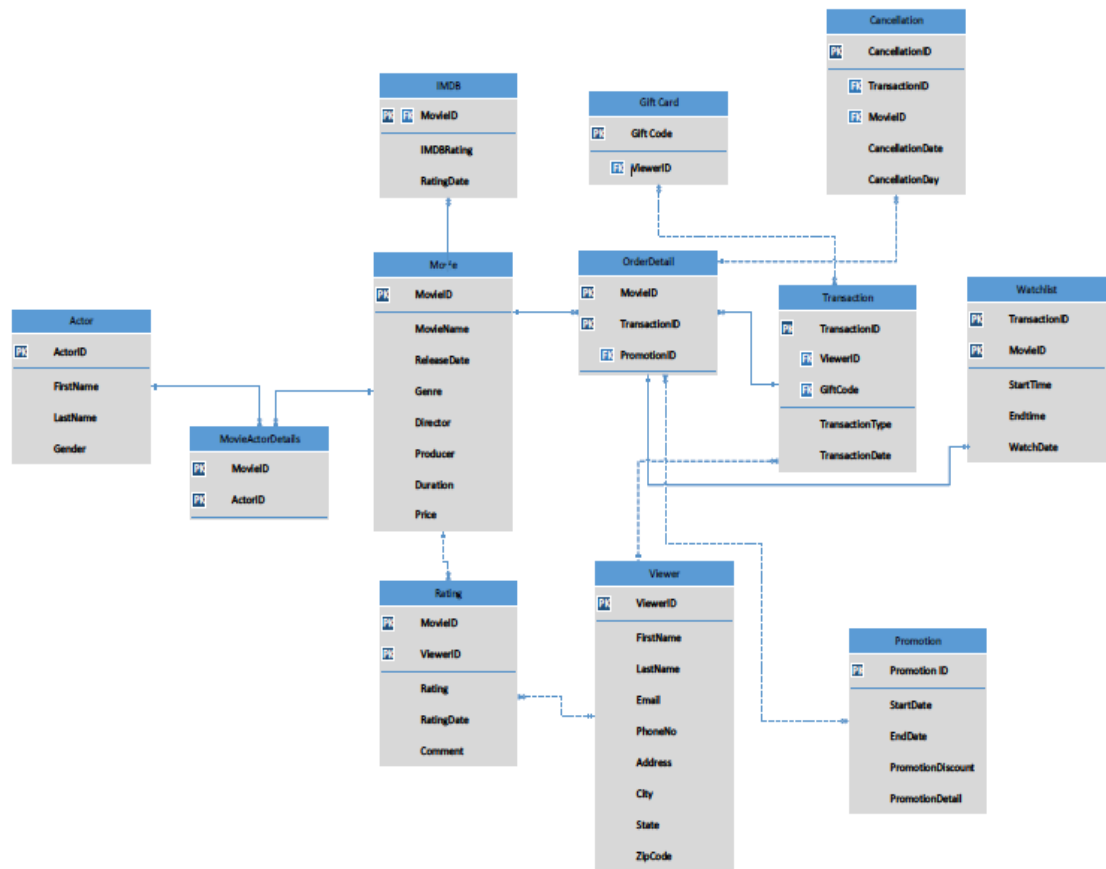
4.Database Scope and Table Descriptions:

Currently, Cometplex does not have database in place as it is new to the market, so following database tables will fulfill the business needs.

- a) **Viewer** – This table will store basic information for both existing and new viewers such as “ViewerID”, “FirstName”, “LastName”, “Email”, “PhoneNo”, etc.
- b) **Movie** – This table will hold all the information about the movies such as “MovieID”, “MovieName”, “ReleaseDate”, “Genre”, “Price”, etc.
- c) **Transaction** – This table will be used to store records for every transaction such as “TransactionID”, “TransactionType”, “TransactionDate”, etc.
- d) **Cancellation** – This table will store the cancellation details for the transactions done such as “CancellationID”, “CancellationDate”, “CancellationDay”, “MovieID” etc.
- e) **Promotion** – This table will store the ongoing deals, discounts on all the movies for different genres and will contain attributes such as “PromotionID”, “StartDate”, “EndDate”, “Promotiondiscount” etc.
- f) **Rating** – This table will store the rating provided by viewers to the movies and attributes such as “MovieID”, “Rating”, “ViewerID”, “Comment” etc.

- g) **IMDB** – This will be a live table which will fetch current IMDB ratings and will be updated biweekly such as “MovieID”, “MovieName”, “Genre”, “IMDBRating” etc.
- h) **GiftCard** – This table will hold the information about a gift card which can be purchased by one registered user for another and attributes like “GiftCode”, “MovieID”, “TransactionID” etc.
- i) **WatchList** – This table will store information about the user’s action once a movie is started such as “StartTime”, “EndTime”, “WatchDate”, “MovieID”, “TransactionID” etc.
- j) **OrderDetail** – This table stores the details of every transaction such as “MovieID”, “TransactionID”, “PromotionID” etc.
- k) **Actor** – This table stores the details of the cast of the movie such as “ActorID”, “FirstName”, “LastName”, “Gender” etc.
- l) **MovieActorDetails** – This table stores the movies and actors corresponding to those movies.

5. Entity Relationship Diagram



6. Relational Database Schema

VIEWER TABLE

CREATE TABLE VIEWER

(VIEWERID	INTEGER	NOT NULL,
FIRSTNAME	VARCHAR(50)	NOT NULL,
LASTNAME	VARCHAR(50)	NOT NULL,
EMAIL	VARCHAR(50)	NOT NULL,
PHONENO	VARCHAR(20),	
ADDRESS	VARCHAR(50),	
CITY	VARCHAR(50),	
STATE	VARCHAR(50),	
ZIPCODE	VARCHAR(20),	
CONSTRAINT PKVIEWERID PRIMARY KEY(VIEWERID))		

MOVIE TABLE

CREATE TABLE MOVIE

(MOVIEID	INTEGER	NOT NULL,
MOVIEName	VARCHAR(50)	NOT NULL,
RELEASEDATE	DATE	NOT NULL,
GENRE	VARCHAR(50)	NOT NULL,
DIRECTOR	VARCHAR(50)	NOT NULL,
PRODUCER	VARCHAR(50)	NOT NULL,
DURATION	INTEGER	NOT NULL,

YEAR INTEGER NOT NULL,
CONSTRAINT PKMOVIEID PRIMARY KEY(MOVIEID))

ACTOR TABLE

CREATE TABLE ACTOR

(ACTORID INTEGER NOT NULL,
FIRSTNAME VARCHAR(50) NOT NULL,
LASTNAME VARCHAR(50) NOT NULL,
GENDER VARCHAR(50) NOT NULL,
CONSTRAINT PKACTORID PRIMARY KEY(ACTORID))

MOVIE ACTOR DETAIL TABLE

CREATE TABLE MOVIEACTORDETAIL

(MOVIEID INTEGER NOT NULL,
ACTORID INTEGER NOT NULL,
CONSTRAINT FKMOVIEID FOREIGN KEY (MOVIEID) REFERENCES MOVIE
ON UPDATE CASCADE,
CONSTRAINT FKACTORID FOREIGN KEY (ACTORID) REFERENCES ACTOR
ON UPDATE CASCADE)

RATING TABLE

CREATE TABLE RATING

(MOVIEID INTEGER NOT NULL,
VIEWERID INTEGER NOT NULL,
RATING DOUBLE(10,2) NOT NULL,
RATINGDATE DATE NOT NULL,

COMMENT VARCHAR(50),
CONSTRAINT FKVIEWERID1 FOREIGN KEY (MOVIEID) REFERENCES
MOVIE ON UPDATE CASCADE,
CONSTRAINT FKMOVIEID1 FOREIGN KEY (MOVIEID) REFERENCES MOVIE
ON UPDATE CASCADE)

PROMOTION TABLE

CREATE TABLE PROMOTION

(PROMOTIONID VARCHAR(25) NOT NULL,
STARTDATE DATE NOT NULL,
ENDDATE DATE NOT NULL,
PROMOTIONDISCOUNT INTEGER NOT NULL,
PROMOTIONDETAIL VARCHAR(50) NOT NULL,
CONSTRAINT PKPROMOTIONID PRIMARY KEY(PROMOTIONID))

TRANSACTION TABLE

CREATE TABLE TRANSACTION

(TRANSACTIONID VARCHAR(255) NOT NULL,
TRANSACTIONDATE DATE NOT NULL,
TRANSACTIONTYPE VARCHAR(6) NOT NULL
PRICE INTEGER NOT NULL,
VIEWERID INTEGER NOT NULL,
GIFTCODE VARCHAR(50),
CONSTRAINT PKTRANSACTIONID PRIMARY KEY(TRANSACTIONID),
CONSTRAINT FKVIEWERID2 FOREIGN KEY(VIEWERID) REFERENCES
VIEWER ON UPDATE CASCADE,

CONSTRAINT FKGIFTCODE FOREIGN KEY(GIFTCODE) REFERENCES
GIFTCARD ON UPDATE CASCADE)

CANCELLATION TABLE

CREATE TABLE CANCELLATION

(CANCELLATIONID VARCHAR(50) NOT NULL,

CANCELLATIONDATE DATE NOT NULL,

QUANTITY INTEGER NOT NULL,

TRANSACTIONID VARCHAR(255) NOT NULL,

CANCELLATIONDAY VARCHAR(20) NOT NULL,

CONSTRAINT PKCANCELLATIONID PRIMARY KEY(CANCELLATIONID),

CONSTRAINT FKTRANSACTIONID FOREIGN KEY(TRANSACTIONID)
REFERENCES TRANSACTION ON UPDATE CASCADE)

ORDER DETAIL TABLE

CREATE TABLE ORDER_DETAIL

(MOVIE_ID INTEGER NOT NULL,

TRANSACTION_ID VARCHAR(255) NOT NULL,

PROMOTIONID VARCHAR(25),

PROMOTIONDISCOUNT INTEGER,

CONSTRAINT PKMOVIEID1 PRIMARY KEY(MOVIEID),

CONSTRAINT PKTRANSACTIONID1 PRIMARY KEY(TRANSACTIONID),

CONSTRAINT FKTRANSACTIONID1 FOREIGN KEY(TRANSACTIONID)
REFERENCES TRANSACTION ON UPDATE CASCADE)

WATCHLIST TABLE

CREATE TABLE WATCHLIST

```

        (TRANSACTIONID VARCHAR(255)    NOT NULL,
        MOVIEID          INTEGER        NOT NULL,
        STARTTIME        DATE           NOT NULL,
        ENDTIME          DATE           NOT NULL,
        WATCHDATE       DATE           NOT NULL,
        CONSTRAINT PKTRANSACTIONID2 PRIMARY KEY(TRANSACTIONID),
        CONSTRAINT FKMOVIEID2 FOREIGN KEY(MOVIEID) REFERENCES MOVIE
        ON UPDATE CASCADE)

```

IMDB TABLE

```

CREATE TABLE IMDB

```

```

        (MOVIEID          INTEGER        NOT NULL,
        IMDB_RATING       DOUBLE(10,2)   NOT NULL,
        RATING_DATE       DATE           NOT NULL,
        CONSTRAINT FKMOVIEID3 FOREIGN KEY(MOVIEID) REFERENCES MOVIE
        ON UPDATE CASCADE)

```

GIFT CARD TABLE

```

CREATE TABLE GIFTCARD

```

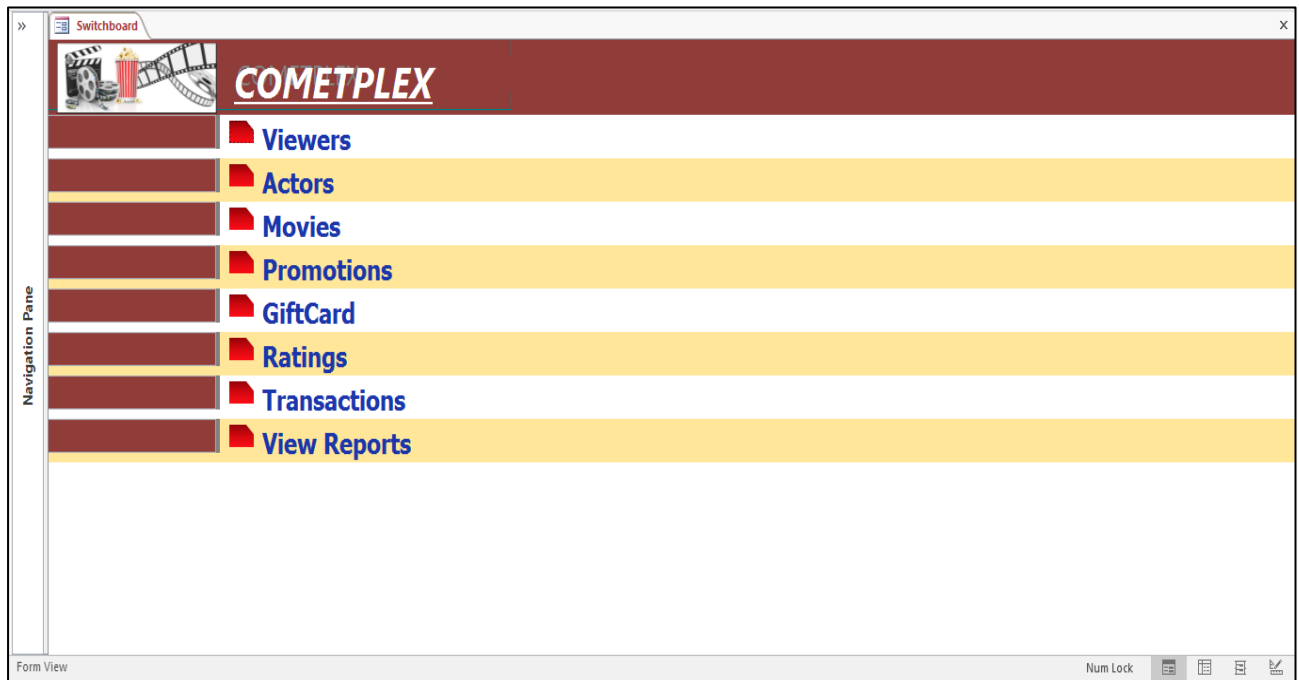
```

        (GIFTCODE  VARCHAR(255)    NOT NULL,
        VIEWERID   INTEGER          NOT NULL,
        CONSTRAINT FKVIEWERID3 FOREIGN KEY(VIEWERID) REFERENCES VIEWER)

```

7. Cometplex Main Menu

The main menu of Cometplex shown below helps to store all the information about Viewers, Movies, Transactions, Ratings and all other relevant information. There are sub menus for Transactions, Ratings and Reports tab to show more insights about the database.



8. Data Input and Manipulation Screen Forms

Viewer

This form can be used to view information about existing viewers as well as add a new viewer to the database.

Viewer

New Viewer?

ViewerID	FirstName	LastName	Email	PhoneNo	Address	City	State	Zipcode
1	Mike	Ross	mikeross@gmail.com	4692792184	12550 Whittington Dr. Apt 606	Houston	Texas	77077
2	Mary	Jacobs	mary.jacobs@gmail.com	9654566328	PO Box 7775, Lauder Lane	San Francisco	California	7775
3	Heather	Jones	heather.jones@gmail.com	4698775963	5400 N. Lakewood Avenue	Chicago	Illinois	60640
4	Rick	Brown	rick.brown@gmail.com	4685532187	113 Cherry St	Seattle	Washington	2205
5	Harvey	Specter	harvey.specter@gmail.com	4698752136	228 Park Ave S	New York	New York	54216
6	Racheal	Zane	racheal.zane@gmail.com	4697851236	20 Maple Avenue	San Pedro	California	90731
7	Sherlock	Holmes	thescienceofdeduction@gmail.com	7894526697	221B Baker St, Marylebone	London	London	56698
8	Bruce	Wayne	darkknight@gmail.com	4692356888	321 Batcave, Wildforest	Gotham City	Texas	75252
9	Clark	Kent	superman@gmail.com	4692356889	322 Random, Tower	Minneapolis	Minnesota	63589
10	James	Howlett	logan@gmail.com	4696184220	6634 Frankford Road	Dallas	Texas	75420
11	Louis	Litt	louis12@gmail.com	8568967845	7246 W. Windsor Dr.	Carmichael	California	95608
12	Tony	Stark	ironman@hotmail.com	7458962132	221 Stark Towers	Malibu	Florida	10001
*	0							

Record: 1 of 12 No Filter Search

Form View Num Lock

Actor

This form can be used to view all the actors and input the details of an actor or actress.

Actor

ActorID: 1 Add Actor

FirstName: Chris

LastName: Hemsworth

Gender: Male

Record: 1 of 28 No Filter Search

Movie

This form can be used to browse through all movies as well as add a new movie to the database.

MovieID	1003	<input type="button" value="Add Movie"/>
MovieName	Dunkirk	
ReleaseDate	21-07-2017	
Genre	Drama	
Director	Christopher Nolan	
Duration	106	
Producer	Christopher Nolan	
Year	2017	

Record: 14 of 17 No Filter Search

Cancellation

This form allows us to cancel an order provided he/she has not started watching the movie within 7 days of renting the movies as well as view all the cancellations.

CancellationID	1010	<input type="button" value="Add Cancellation"/>
TransactionID	T101	
CancellationDate	19-10-2017	
CancellationDay	Thursday	
Reason	Saw it elsewhere	

Record: 14 of 17 No Filter Search

GiftCard

This form helps us to apply a gift card for a particular viewer and also view all the gift cards that have been applied to a particular viewer.

GiftCard

GiftCode: abcd123

ViewerID: 8

Add Gift Card

Records: 1 of 2

IMDB

This form helps us to view all the IMDB ratings, update rating for a particular movie and also insert an IMDB rating for a particular movie.

IMDB

MovieID: 1004

IMDBRating: 8.3

RatingDate: 04-11-1994

Add IMDB Record

Records: 3 of 10

Promotion

This form allows us to view details of promotions and also add a new promotion for particular dates.

Promotion

PromotionID: 8001

StartDate: 24-10-2017

EndDate: 31-10-2017

PromotionDiscount: 0.20

PromotionDetail: Halloween week

Record: 1 of 4

Transaction

This form helps us to add new transaction for a viewer and can also be used to view all the transactions for a viewer. The form has a subform to view the details of the viewer performing the transaction.

Transaction

TransactionID: 101

ViewerID: 5

GiftCode:

TransactionType: Credit

Price: 1

TransactionDate: 09-11-2017

Record: 1 of 26

Viewer

ViewerID	FirstName	LastName	Email	PhoneNo	Address	City	State	Zipcode
5	Harvey	Specter	harvey.specter@gmail.com	4698752136	228 Park Ave S	New York	New York	54216
5								

Viewer's Rating

This form allows us to view, insert and update rating for a movie. The form also has two subforms: Movie Details and Viewer Details. These subforms allows us to view movie and viewer details corresponding to a particular rating.

Viewer's Rating

Rating

MovieID: 1004 [Previous] [Next] [New Rating] [Delete Rating] [Refresh]

ViewerID: 3

Rating: 3.5

RatingDate: 03-10-2017

Comment: Climax was brilliant!

MovieDetails | ViewerDetails

MovieID	MovieName	ReleaseDate	Genre	Director	Duration	Producer	Year
1004	Andaz Apna Apna	04-11-1994	Comedy	Rajkumar Santosh	160	Vinay Kumar Sinha	1994

Record: 1 of 12 [Previous] [Next] [First] [Last] [No Filter] [Search]

Watchlist

This form helps us to view the watch details of a movie for a transaction such as when the movie was started, whether the viewer watched the full movie also the date on which it was watched.

WatchList

TransactionID: 1004 [Previous] [Next] [Refresh]

MovieID: 1004

StartTime: 13:30:00

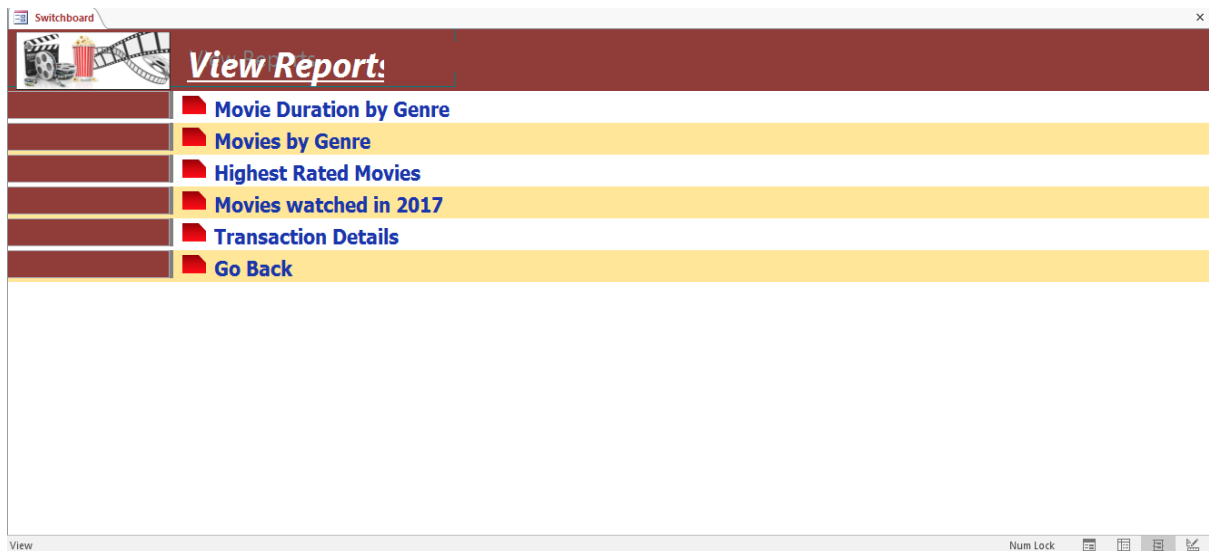
EndTime: 15:30:00

WatchDate: 18-10-2017

Record: 1 of 15 [Previous] [Next] [First] [Last] [No Filter] [Search]

9.Sample Reports

Below is the submenu to view all the useful reports for important insights for the overall performance and popularity of Cometplex.



Report 1: Movies by Duration

Reports the duration of all movies in ascending order of genre.

Related query:

```
SELECT Genre, MovieName, Duration
FROM Movie
ORDER BY Genre;
```

Genre_Movie_Duration		
Genre	MovieName	Duration
Action	Thor: Ragnarok	121
Adventure	Justice League	121
Animation	The Lion King	88
Biography	Schindler's List	195
Comedy	Andaz Apna Apna	160
Crime	The Godfather	175
Documentary	13th	100
Drama		

Report 2: Movie details by Genre

Reports the movie name, release year, rating, rating date and comments for the maximum rated movie by genre.

Related Query:

```
SELECT Sub2.Genre, MovieName, Year, Sub2.Rating AS HighRating, Comment
FROM (SELECT Genre, MAX(Rating) AS MaxRating FROM Genre_Rating GROUP
BY Genre) AS Sub1 INNER JOIN (SELECT * FROM Genre_Rating) AS Sub2 ON
(Sub1.Genre=Sub2.Genre) AND (Sub1.MaxRating=Sub2.Rating);
```

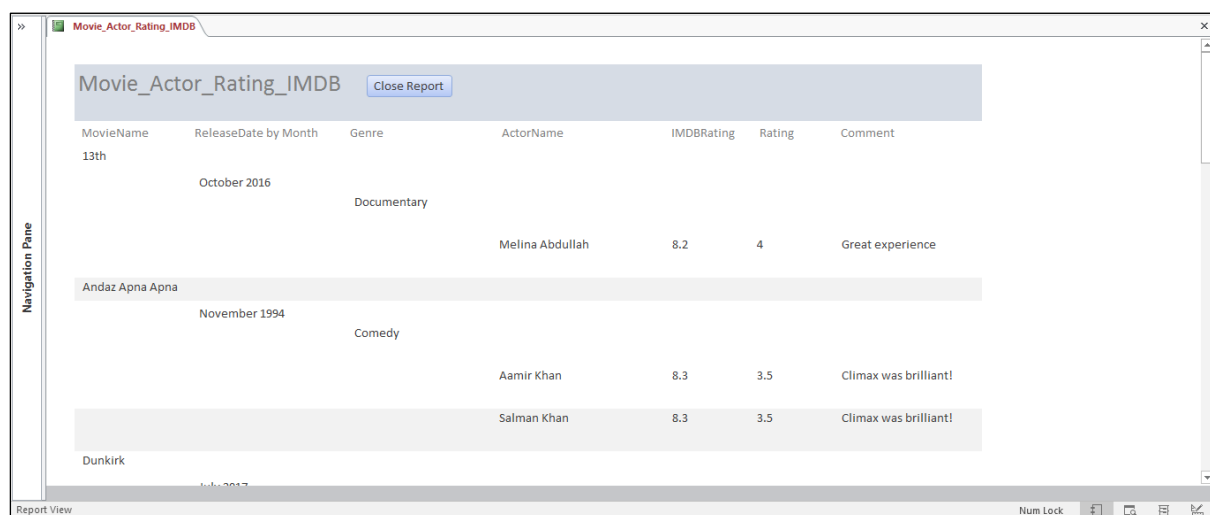
Movie by Genre					
Genre	MovieName	Year	Rating	RatingDate	Comment
Action	Thor: Ragnarok	2017	4	18-11-2017	Amazing!
Animation	The Lion King	1994	3.5	19-09-2016	Good.
Biography	Schindler's List	1994	3	13-10-2015	Didn't live up to the hype
Comedy	Andaz Apna Apna	1994	3.5	03-10-2017	Climax was brilliant!
Crime	The Godfather	1994	9	18-11-2017	khkh

Report 3: Movie Actor Rating IMDB

Reports the details of the movie, corresponding actor(s), ratings and comments corresponding to the movie.

Related Query:

```
SELECT Movie.MovieName, Movie.ReleaseDate, Movie.Genre, (FirstName+'
'+LastName) AS ActorName, IMDB.IMDBRating, Rating.Rating,
Rating.Comment
FROM ((Movie INNER JOIN IMDB ON Movie.MovieID = IMDB.MovieID) INNER
JOIN (Actor INNER JOIN MovieActorDetails ON Actor.ActorID =
MovieActorDetails.ActorID) ON Movie.MovieID = MovieActorDetails.MovieID)
INNER JOIN Rating ON Movie.MovieID = Rating.MovieID;
```



The screenshot shows a report window titled "Movie_Actor_Rating_IMDB" with a "Close Report" button. The report displays a table with the following columns: MovieName, ReleaseDate by Month, Genre, ActorName, IMDBRating, Rating, and Comment. The data is grouped by movie, with each movie having a header row and one or more actor rows.

MovieName	ReleaseDate by Month	Genre	ActorName	IMDBRating	Rating	Comment
13th	October 2016	Documentary	Melina Abdullah	8.2	4	Great experience
Andaz Apna Apna	November 1994	Comedy	Aamir Khan	8.3	3.5	Climax was brilliant!
			Salman Khan	8.3	3.5	Climax was brilliant!
Dunkirk	July 2017					

Report 4: Viewer Movie 2017

Reports the number of movies purchased by the viewer(s) in 2017.

Related Query:

```
SELECT Sub1.ViewerID, (Viewer.FirstName+' '+Viewer.LastName) AS
ViewerName, Viewer.City, Count(Sub1.MovieID) AS NumberOfMovies
FROM (SELECT Transaction.ViewerID, Transaction.TransactionID,
Transaction.TransactionDate, OrderDetail.MovieID FROM [Transaction] INNER
JOIN OrderDetail ON Transaction.TransactionID=OrderDetail.TransactionID
```

*WHERE YEAR(TransactionDate)=2017) AS Sub1 INNER JOIN Viewer ON
Sub1.ViewerID = Viewer.ViewerID
GROUP BY Sub1.ViewerID, Viewer.FirstName, Viewer.LastName, Viewer.City;*

Viewer_Movie_2017			
ViewerID	ViewerName	City	NumberOfMovies
1	Mike Ross	Houston	4
2	Mary Jacobs	San Francisco	3
3	Heather Jones	Chicago	5
4	Rick Brown	Seattle	4
5	Harvey Specter	New York	7
6	Racheal Zane	San Pedro	1
7	Sherlock Holmes	London	4
8	Bruce Wayne	Gotham City	1
9	Clark Kent	Minneapolis	4
10	James Howlett	Dallas	1
11	Louis Litt	Carmichael	4
12	Tony Stark	Malibu	2

28 November 2017 Page 1 of 1

Report 5: Viewer Transaction Details

Reports all the transaction details along with the amount paid by each Viewer.

Related Query:

*SELECT Viewer.ViewerID, (FirstName+' '+LastName) AS Name,
Transaction.TransactionID, TransactionDate, MovieName, Year, (Price-
Nz(PromotionDiscount,0)) AS Amount
FROM Viewer INNER JOIN ([Transaction] INNER JOIN (Movie INNER JOIN
OrderDetail ON Movie.MovieID = OrderDetail.MovieID) ON
Transaction.TransactionID = OrderDetail.TransactionID) ON Viewer.ViewerID =
Transaction.ViewerID;*

Viewer_Transaction_Details Close Report					
Name	TransactionID	TransactionDate	MovieName	Year	Amount
Bruce Wayne	T120	29-06-2017	Rush	2013	1
Clark Kent	T103	14-10-2017	Dunkirk	2017	0.2
	T104	15-10-2017	Schindler's List	1994	2
	T122	25-07-2017	The Notebook	2004	1
	T104	15-10-2017	Andaz Apna Apna	1994	2
Harvey Specter	T111	28-10-2017	Wonder Woman	2015	2
	T107	17-10-2017	Schindler's List	1994	1
	T101	09-11-2017	Thor: Ragnarok	2017	0.2
	T111	28-10-2017	E.T. The Extra-Terrestrial	1982	2
	T114	03-11-2017	The Notebook	2004	1
	T117	04-11-2017	Andaz Apna Apna	1994	2
	T117	04-11-2017	Rocky	1976	2
Heather Jones					

10. Contributions:

- **Priyanshi Khatri**
 - Contributed to the scope and business process of the project
 - Completed the ERD of the project
 - Created all the tables in access database
 - Worked on data entry
 - Created the buttons and finalized the forms
 - Created the hierarchial forms
 - Created the drop-down menu in the tables
 - Created the switchboard in access and did final formatting
 - Completed the final project report with proper formatting
- **Sonali Gupta**
 - Contributed to the scope and business process of the project
 - Created the initial draft of ERD in Visio
 - Created the relational schema for the project.
 - Worked on data entry
 - Created the initial draft of all the forms
 - Created the first draft of entire report
 - Checked if relational schema and access db is consistent

- **Raghav Seth**

- Proposed the idea of the project
- Contributed to the scope and business process of the project
- Created the queries and reports based on that
- Finalized the ERD of the project
- Created the relational schema for the project.
- Worked on data entry
- Checked if relational schema and access db is consistent
- Created the switchboard in access
- Completed the formatting of reports