**Logical Design**

A. MAPPING EER DIAGRAM TO SCHEMA (ALONG WITH INTUITIVE NORMALIZATION)

STEP – I

Map Strong Entities

i) movie\_table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| movie\_id | Mname | release\_date | duration | language |

ii) artist\_table

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| artist\_id | Name | | job\_title | region | age |
| first\_name | last\_name |

-To 1NF: Removing ‘composite attributes’

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| artist\_id | art\_fname | art\_lname | job\_title | age | orgin |

-Now the final table is in 3NF

iii) genre\_table

|  |  |  |
| --- | --- | --- |
| genre\_name | description | suggested\_audience |

-Now the table is in 3NF

iv) award\_table

|  |  |  |  |
| --- | --- | --- | --- |
| award\_id | award\_name | description | certifying\_organisation |

-Now the table is in 3NF

v) user\_table

|  |  |
| --- | --- |
| username | Password |

-Now the table is in 3NF

STEP – II

Map Weak Entities

-No weak entities to map

STEP – III

Map 1:1 relationships: Merged towards Total Participation

1) Prequel/Sequel Relationship

sequence\_table

|  |  |
| --- | --- |
| movie\_id  (fk:movie-\_table>movie\_id) | Sequel |

-Now the table is in 3NF

STEP – IV

Map 1:N relationships: Merged Towards the N side

1) movie belongs to relationship: movie-N :: Genre-1

movie\_table

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| movie\_id | name | release\_date | duration | language | genre\_name  (fk:genre\_table->genre\_name) |

-Now the table is in 3NF

STEP – V

Map M:N relationships: Add new relation with foreign keys referencing primary keys of both participating tables along with connecting attributes

1) film\_artist contributes for relationship

movie\_artist\_table

|  |  |
| --- | --- |
| movie\_id  (fk:movie\_table->movie\_id) | artist\_id  (fk:film\_artist\_table->artist\_id) |

-Now the table is in 3NF

**Note: To intuitively bring the tables into 3NF Ratings and Reviews have been made separate tables to remove the transitive dependencies to transform from 2NF to 3NF.**

**This takes care of setting primary keys during such that removal of any key would not hold the functional dependencies anymore. Also it ensures that there are no transitive dependencies.**

2) client rate movie relationship

rating\_table

|  |  |  |
| --- | --- | --- |
| username  (fk: user\_table->username) | movie\_id  (fk: movie\_table->movie\_id) | stars |

-Now the table is in 3NF

3) client review movie relationship

review\_table

|  |  |  |
| --- | --- | --- |
| username  (fk: user\_table->username) | movie\_id  (fk: movie\_table->movie\_id) | review |

-Now the table is in 3NF

STEP – VI

Map multi-valued attributes: Make a seperate table with foreign keys referencing primary keys of member relation (NF 4)

**Note: To intuitively bring the watchlist into 1NF, multiple valued watchlist attribute has been put up separate table to map multiple values effectively initially to bring it to 1NF. Now the table comes to 3NF as there are no transitive dependencies (3NF) and removal of any attribute from primary key would not fold functional dependency anymore.**

1) watchlist of client user

watchlist

|  |  |
| --- | --- |
| username  (fk: user\_table->username) | movie\_id  (fk: movie\_table->movie\_id) |

-Now the table is in 3NF

STEP – VII

Map N-Ary relationship: Add new relation with foreign keys referencing primary keys of both participating tables along with connecting attributes.

1) Wins Relationship: Ternary Relationship

Movie:N :: Film Artist:M :: Award:P

movie\_artist\_award\_table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| award\_id  (fk: award\_table  ->award\_id) | artist\_id  (fk: film\_artist\_table  ->artist\_id) | movie\_id  (fk: movie\_table  ->movie\_id) | category | year |

-Now the table is in 3NF

STEP – VIII: Addiotional Step

Mapping Subclasses of EER with help of binary discriminating attribute

1) User can be a client user or admin, hence adding “disjoint discriminating” attribute:

Admin specific attribute: None

Client specific attribute: Watchlist

user\_table

|  |  |  |
| --- | --- | --- |
| username | password | d\_administrator\_client |

-Now the table is in 3NF

B. FUNCTIONAL DEPENDENCY

1) movie\_table

movie\_id -> movie\_id

movie\_id -> mname

movie\_id -> release\_date

movie\_id -> duration

movie\_id -> language

movie\_id -> genre

movie\_id -> rating

movie\_id -> num\_of\_rating

movie\_id -> num\_of\_reviews

{movie\_id}\* ->{mname, release\_date, duration, language, genre, rating, num\_of\_rating, num\_of\_reviews}

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| movie\_id  (PK) | name | release\_date | duration | language | genre\_name  (fk:genre\_table->genre\_name) |

2) review\_table

user\_name -> user\_name

movie\_id -> movie\_id

{user\_name, movie\_id}\* -> {user\_name, movie\_id, review}

|  |  |  |
| --- | --- | --- |
| username  (fk: user\_table->username)  (PK) | movie\_id  (fk: movie\_table->movie\_id)  (PK) | review |

3) rating\_table

user\_name -> user\_name

movie\_id -> movie\_id

{user\_name, movie\_id}\* -> {user\_name, movie\_id, rating}

|  |  |  |
| --- | --- | --- |
| username  (fk: user\_table->username)  (PK) | movie\_id  (fk: movie\_table->movie\_id)  (PK) | rating |

4) watchlist

user\_name -> user\_name

movie\_id -> movie\_id

{user\_name, movie\_id}\* -> {user\_name, movie\_id}

|  |  |
| --- | --- |
| username  (fk: user\_table->username)  (PK) | movie\_id  (fk: movie\_table->movie\_id)  (PK) |

5) genre\_table

genre\_name -> genre\_name

genre\_name -> description

genre\_name -> suggested\_audience

{genre\_name}\* -> {genre\_name, description, suggested\_audience}

|  |  |  |
| --- | --- | --- |
| genre\_name  (PK) | description | suggested\_audience |

6) sequence\_table

movie\_id -> movie\_id

movie\_id -> sequel

{movie\_id}\* -> {movie\_id, sequel}

|  |  |
| --- | --- |
| movie\_id  (fk:movie-\_table>movie\_id)  (PK) | sequel |

7) user\_table

user\_name -> user\_name

user\_name -> password

user\_name -> d\_admin\_client

{user\_name}\* -> {user\_name, password, d\_admin\_client }

|  |  |  |
| --- | --- | --- |
| username  (PK) | password | d\_administrator\_client |

8) movie\_artist\_table

movie\_id -> movie\_id

artist\_id -> artist\_id

{movie\_id, artist\_id}\* -> {movie\_id, artist\_id}

|  |  |
| --- | --- |
| movie\_id  (fk:movie\_table->movie\_id)  (PK) | artist\_id  (fk:film\_artist\_table->artist\_id)  (PK) |

9) movie\_artist\_award\_table

award\_id -> award\_id

movie\_id -> movie\_id

artist\_id -> artist\_id

{award\_id, movie\_id, artist\_id}\* -> {award\_id, movie\_id, artist\_id, category, year}

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| award\_id  (fk: award\_table  ->award\_id)  (PK) | artist\_id  (fk: film\_artist\_table  ->artist\_id)  (PK) | movie\_id  (fk: movie\_table  ->movie\_id)  (PK) | category | year |

10) artist\_table

artist\_id -> artist\_id

artist\_id -> art\_fname

artist\_id -> art\_lname

artist\_id -> job\_title

artist\_id -> age

artist\_id -> origin

{artist\_id}\* -> {artist\_id, art\_fname, art\_lname, job\_title, age, orgin}

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| artist\_id  (PK) | art\_fname | art\_lname | job\_title | age | orgin |

11) award\_table

award\_id -> award\_id

award\_id -> award\_name

award\_id -> description

award\_id -> organisation

{award\_id}\* -> {award\_id, award\_name, description, organisation}

|  |  |  |  |
| --- | --- | --- | --- |
| award\_id  (PK) | award\_name | description | certifying\_organisation |