CAC 1 - Django Full-Stack Web Development Project - Phase 1
Streamify Database Schema

Introduction:

Streamify is an OTT platform offering a vast collection of latest and classic movies and series in one place. It provides real-time viewing analysis, enabling users to gain insights into their watching behaviour and preferences, enhancing their overall entertainment experience with personalized recommendations and detailed analytics.

1. Database Design:

Streamify consist of 8 tables. Below are the name of the tables:

Table Name: User_auth
Table Description: This table is for user authentication at the time of login and signup.

Column Name	Data Type	Constraints	Description (if any)
User_id	Int	PRIMARY KEY	
username	Varchar	UNIQUE	
password	Varchar	UNIQUE	

Table Name: User_profile
Table Description: This table holds the information about the user. This is for user management.

Column Name	Data Type	Constraints	Description (if any)
user_id	Int	FOREIGN KEY	
first_name	Varchar		
last_name	Varchar		
dob	Date		
age	int		Calculated field from Dob
mobile	double	UNIQUE	
email	Varchar	UNIQUE	
door_no	int		
street	Varchar		
city	Varchar		
state	Varchar		

pincode	int		
genre preference	Varchar		
subscribed	boolean	ENUM(“Yes”,”NO”)	
gender	boolean	ENUM(“M”,”F”)	

Table Name: Movies

Table Description: This table holds the information about the movies along with other details this is for movie / series management.

Column Name	**Data Type**	**Constraints**	**Description (if any)**
movie_id	Int	PRIMARY KEY	
movie_name	Varchar		
duration	time		
ratings	int		
releasedate	date		
type(movie/series)	boolean	ENUM (“Movie”,”Series”)	

Table Name: Genre

Table Description: This table holds the information about different types of Genre available. This table is for Genre Management.

Column Name	**Data Type**	**Constraints**	**Description (if any)**
genre_id	Int	PRIMARY KEY	
genre_name	Varchar		

Table Name: Actor

Table Description: This table is used to manage the details of Actor / Actress who has worked in the movie / series.

Column Name	**Data Type**	**Constraints**	**Description (if any)**
actor_id	Int	PRIMARY KEY	
actor_name	Varchar		
dob	date		
age	int		
gender	boolean	ENUM(“M”,”F”)	
awards	int		

Table Name: Movie_Genre

Table Description: This table is used to record the details of movie and genre. This act as a junction table between movie and genre table

Column Name	**Data Type**	**Constraints**	**Description (if any)**
movie_id	Int	FOREIGN KEY	
genre_id	Int	FOREIGN KEY	

Table Name: Movie_Actor

Table Description: : This table is used to record the details of movie and actor. This act as a junction table between movie and actor table

Column Name	**Data Type**	**Constraints**	**Description (if any)**
movie_id	Int	FOREIGN KEY	
actor_id	Int	FOREIGN KEY	

Table Name: Watchlist

Table Description: This table is used to record the details of the movies watched by the users along with the genre and actor for analysis.

Column Name	**Data Type**	**Constraints**	**Description (if any)**
user_id	Int	FOREIGN KEY	
movie_id	Int	FOREIGN KEY	
genre_id	Int	FOREIGN KEY	
actor_id	Int	FOREIGN KEY	
start_time	time		
end_time	time		

2. Relationship Schema Diagram:

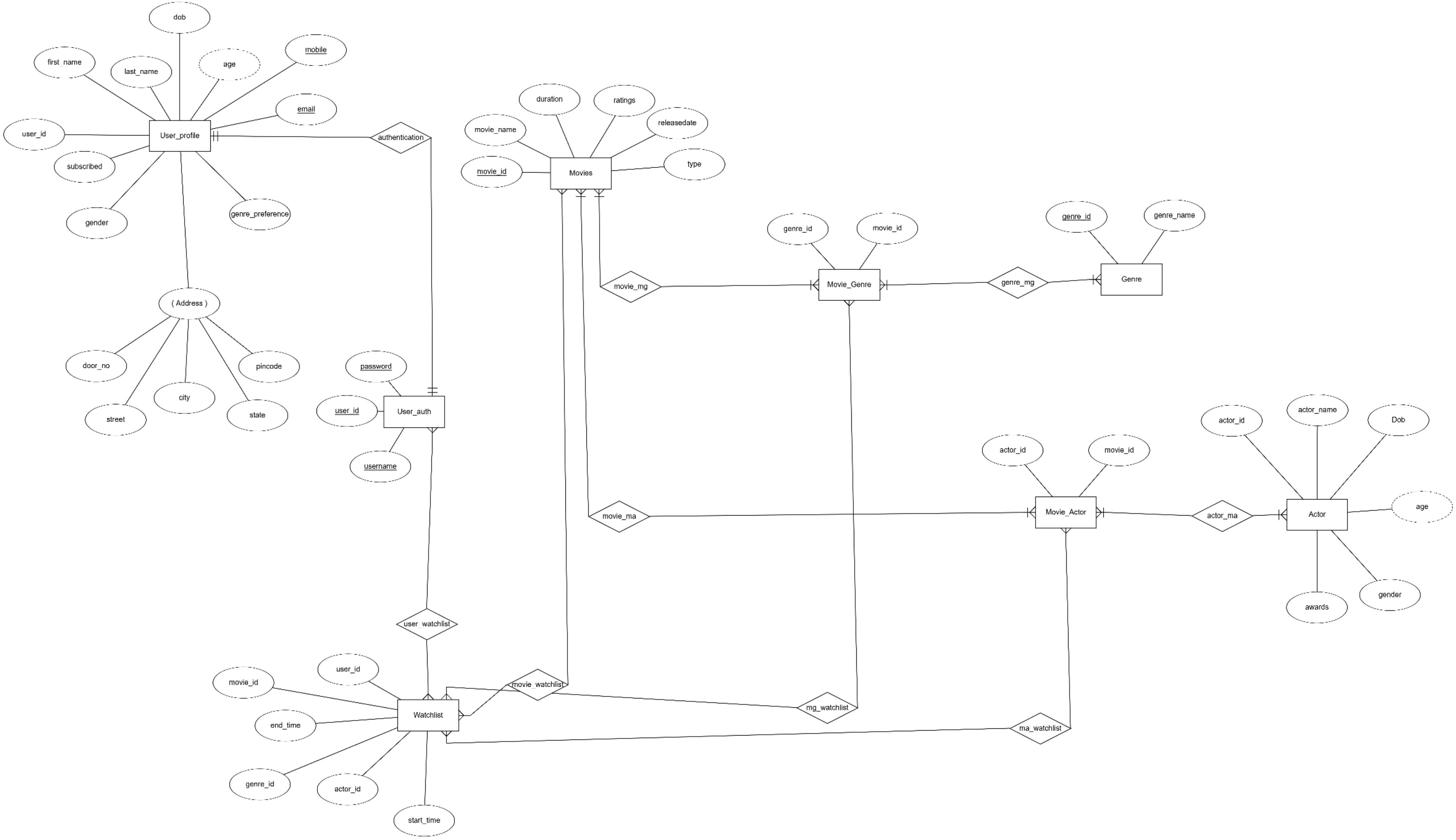
Below are the list of the tables we have in Streamify along with column name and constraint

1. User_auth (**user_id PRIMARY KEY**, **username UNIQUE**, **password UNIQUE**)
2. User_profile (**user_id FOREIGN KEY,** first_name, last_name, dob, age, **mobile UNIQUE, email UNIQUE,** door_no, street, city, state, pincode, genre_preference, **subscribed ENUM ("Yes","NO"), gender ENUM ("M","F")**)
3. Movies (**movie_id PRIMARY KEY,** movie_name, duration, ratings, releasedate, **type ENUM ("Movie", "Series")**)
4. Genre (**genre_id PRIMARY KEY,** genre_name)
5. Actor (**actor_id PRIMARY KEY,** actor_name, dob, age, **gender ENUM ("M","F"),** awards)
6. Movie_Genre (**movie_id FOREIGN KEY, genre_id FOREIGN KEY**)
7. Movie_Actor (**movie_id FOREIGN KEY, actor_id, FOREIGN KEY**)
8. Watchlist (**user_id FOREIGN KEY, movie_id FOREIGN KEY, genre_id FOREIGN KEY, actor_id FOREIGN KEY,** start_time, end_time)

Relationships:

- **User_auth → User_profile:** user_id in User_profile references user_id in User_auth (One – to One Relationship)
- **User_auth → Watchlist:** user_id in Watchlist references user_id in User_auth (One – to – Many Relationship)
- **Movie → Watchlist:** movie_id in watchlist references movie_id in Movie (One – to – Many Relationship)
- **Movies → Movie_Genre:** movie_id in Movie_Genre references movie_id in Movies (One – to – Many Relationship)
- **Genre → Movie_Genre:** genre_id in Movie_Genre references genre_id in Genre (One – to – Many Relationship)
- **Movies → Movie_Actor:** movie_id in Movie_Actor references movie_id in Movies (One – to – Many Relationship)
- **Actor → Movie_Actor:** actor_id in Movie_Actor references actor_id in Actor (One – to – Many Relationship)
- **Movie_Genre → Watchlist:** movie_id in Watchlist references movie_id in Movie_Genre, genre_id in Watchlist references genre_id in Movie_Genre (Many – to – Many Relationship)
- **Movie_Actor → Watchlist:** movie_id in Watchlist references movie_id in Movie_Actor, actor_id in Watchlist references actor_id in Movie_Actor (Many – to – Many Relationship)

3. E-R Diagram:


dob
first_name
last_name
age
mobile
email
user_id
User_profile
subscribed
gender
genre_preference
authentication
(Address)
door_no
pincode
city
street
state
password
user_id
User_auth
username
duration
ratings
movie_name
releasedate
movie_id
Movies
type
genre_id
movie_id
movie_mg
Movie_Genre
genre_mg
genre_id
genre_name
Genre
actor_id
actor_name
Dob
actor_id
movie_id
movie_ma
Movie_Actor
actor_ma
Actor
age
gender
awards
user_watchlist
user_id
movie_id
Watchlist
movie_watchlist
mg_watchlist
end_time
ma_watchlist
genre_id
actor_id
start_time