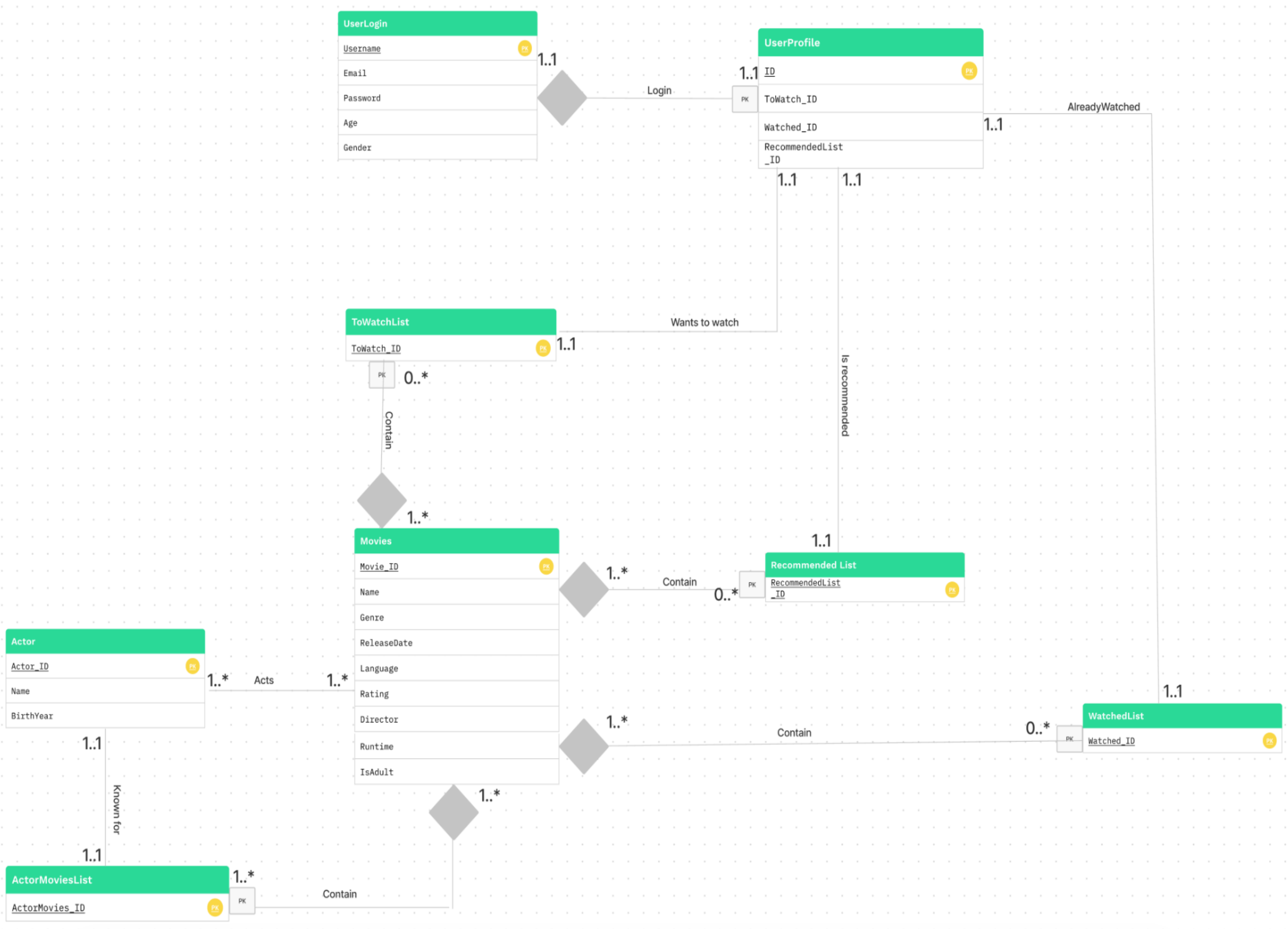
UserLogin
Username
Email
Password
Age
Gender
1..1
Login
1..1
PK
UserProfile
ID
ToWatch_ID
Watched_ID
RecommendedList _ID
AlreadyWatched
1..1
1..1
1..1
ToWatchList
ToWatch_ID
Wants to watch
1..1
PK
0..*
Contain
Is recommended
1..*
Movies
Movie_ID
Name
Genre
ReleaseDate
Language
Rating
Director
Runtime
IsAdult
1..*
Contain
0..*
PK
1..1
Recommended List
RecommendedList _ID
Actor
Actor_ID
Name
BirthYear
1..*
Acts
1..*
1..*
Contain
0..*
PK
1..1
WatchedList
Watched_ID
1..1
Known for
1..1
1..*
ActorMoviesList
ActorMovies_ID
1..*
PK
Contain

Relational Schema

UserLogin(Username : VARCHAR(50) [PK], Email : VARCHAR(50), Password : VARCHAR(20), Age : INT, Gender : VARCHAR(20))

UserProfile(ProfileID : INT [PK], ToWatch_ID : INT [FK to ToWatchList.ToWatch_ID], Watched_ID : INT [FK to WatchedList.Watched_ID], RecommendedList_ID : INT [FK to RecommendedList.RecommendedList_ID], Username : VARCHAR(50) [FK to UserLogin.Username])

ToWatchList(ToWatch_ID : INT [PK], MovieID : INT [FK to Movies.Movie_ID])

WatchedList(Watched_ID : INT [PK], MovieID : INT [FK to Movies.Movie_ID])

RecommendedList(RecommeadedList_ID : INT [PK], MovieID : INT [FK to Movies.Movie_ID])

Movies(Movie_ID : INT [PK], Name : VARCHAR(50), Genre : VARCHAR(20), Language : VARCHAR(20), ReleaseDate : VARCHAR(20), Rating : DOUBLE, Director : VARCHAR(20), Runtime : INT, IsAdult : BOOLEAN)

Actor(Actor_Id : INT [PK], Name : VARCHAR(50), BirthYear : INT)

ActorMovieList(ActorMovies_ID : INT [PK], MovieID : INT [FK to Movies.Movie_ID])

Assumption

1. Every user will have only one ToWatchList, one WatchedList, and one RecommendedList.
2. Each user will have one user profile, because of the condition in #1.
3. Each list (ToWatchList, WatchedList, RecommendedList) can contain from one to many movies.
4. One movie can feature many actors, and one actor can be featured in many movies/tv shows. This means a many-to-many relationship between movie and Actor.

5. All actors need to be featured in at least one movie.
6. All movies should have at least one actor in the movie.

Description of Relationship

1. Login: This relationship connects each user to the profile in our web application.
2. Wants to watch: This relationship describes for each user a list of movies they would like to see in the future.
3. Is recommended: This relationship describes for each user a list of movies filtered by the user preferences.
4. AlreadyWatched: This relationship describes for each user a list of movies that the user has already watched.
5. Contain (ToWatchList - Movies): This relationship represents all the movies in the ToWatchList.
6. Contain (RecommendedList - Movies): This relationship represents all the movies in the RecommendedList.
7. Contain (WatchedList - Movies): This relationship represents all the movies in the WatchedList.
8. Acts: This relationship shows what movies each actor was featured in and what actors each movie has.
9. Known for: This relationship represents the list of movies each actor is known for (has featured in). The list is unique for each actor.
10. Contain(ActorMovieList - Movies): This relationship represents the list of movies that each actor has featured in. The list will contain one to many movies, and each movie can be featured in one to many list.

IMDb