# **Application Description**

This application is designed to effectively manage individuals' personal movie collections. Users can store, update, and review their movie collection, ensuring they have a convenient tool at their disposal to track and organize their favorite films. We will be using a Microsoft Access application as our front-end and an Oracle database back-end to store our application data.

Our application database will encompass five primary entities: Movies, Distributors, Directors, Actors, and Genre and two association tables: Movies\_has\_Contributors and Roles. These entities constitute the foundational elements of our database structure.

Our application database will encompass five primary entities: Movies, Distributors, Directors, Actors, and Genre, two association tables: Movies\_has\_Contributors and Roles, and six additional tables: Contributors\_First\_Name, First\_Name, Actors\_Weight, Weight, Role\_Screen\_Time and Screen\_Time, which store historical data. The original entities include:

· Movies: This entity represents individual movies. Store movies’ information, contains movieID(PK), title, releaseDate, duration and genreID(FK). A movie can have multiple contributors (many-to-many relationship).

· Contributors: This entity represents individuals who contribute to the creation of movies and stores contributorID(PK), lastname, nationality, dateofBirth, and dateofDeath. A contributor can contribute to many movies.

· Movies\_has\_Contributors: This entity serves as an association table between Movies and Contributors in a many-to-many relationship. It contains foreign keys referencing the primary keys of Movies and Contributors {MovieID, ContributorID} which combined, is a composite primary key. It also has the field Roletype(Director/Actor).

· Directors: Directors table contains directorID (PK), contributorID (FK).

· Actors: Actors table contains actorID(PK), height, contributorID (FK), and haircolor.

· Genre: Genre table includes genreID(PK), genreName.

· Roles: This entity serves as an association table between Actors and Movies in a many-to-many relationship. Each movie has one or more roles. Each role must be played by one actor, an actor can play many roles (but usually not in the same movie). Roles table contains { ActorID, MovieID } (PK), characterName.

The original table "Actors" with a multi-valued field "weight" is extended as follows: the original table with just the key and the single-valued fields; a new view "Actors\_View"; a new table for the multi-valued field named "Weight"; and an association table that links the new table to the multi-valued fields named "Actors\_Weight."

A white background with black text

Description automatically generated

The original table "Contributors" with a multi-valued field "First\_Name" is extended as follows: the original table with just the key and the single-valued fields; a new view "Contributors\_View"; a new table for the multi-valued field named “First\_Name"; and an association table that links the new table to the multi-valued field, named "Contributors\_First\_Name".

The original table "Role" with a multi-valued field "screen-time" is extended as follows: the original table with just the key and the single-valued fields; a new view "Roles\_View"; a new table for the multi-valued field named "Screen\_Time"; and an association table that links the new table to the multi-valued fields named "Role\_Screen\_Time".

By integrating these entities, we aim to create an application that enables users to efficiently manage, explore and enjoy their personal movie collections.

Business Rules:

A movie must belong to a genre.

A genre can contain many movies.

A movie has many roles.

An actor plays many roles (but not in the same movie)

Directors and Actors are both Contributors.

A movie has many contributors and a contributor can contribute to many movies.

A movie must have at least one contributor.

A movie must have a director

A director can direct one or more movies

A movie has multiple actors in it

An actor can act in one or more movies