

Project Deliverable - 3

Movie Recommendation Application using Graph DB

Team Information:

- Aneesh Garg 180050007
- Ayush Garg 180050016
- Mohammad Taufeeque 180050062
- Samarth Singh 180050090

1) Models

Following is a textual description of the database tables along with their constraints and triggers.

1. OTT Table

id	PRIMARY KEY constraint
name	

2. GENRE Table

id	PRIMARY KEY constraint
name	

3. LANGUAGE Table

id	PRIMARY KEY constraint
name	

4. COUNTRY Table

id	PRIMARY KEY constraint
name	

5. SHOW Table

id	PRIMARY KEY constraint
title	
duration	
release_date	
avg_rating	Triggered after add rating for a show by the user
imdb_rating	

rotten_tomato
budget constraint- budget ≥ 0
overview
overall_rating
Is_movie
Indexes: id

6. OriginCountry Table

show_id PRIMARY KEY constraint, FOREIGN KEY to show
country_id FOREIGN KEY to country

7. OriginalLanguage Table

show_id FOREIGN KEY to show
language_id FOREIGN KEY to language
(show_id, language_id) PRIMARY KEY constraint

8. Genre_of Table

show_id FOREIGN KEY to show
genre_id FOREIGN KEY to genre
(show_id, genre_id) PRIMARY KEY constraint

9. Available Table

show_id FOREIGN KEY to show
ott_id FOREIGN KEY to ott
(show_id, ott_id) PRIMARY KEY constraint

10. Rating Table

id PRIMARY KEY constraint
numeric
review
upvotes
Downvotes
Indexes: id

11. Person Table

id PRIMARY KEY constraint
name
dob
gender
Indexes: id

12. User Table

person_id FOREIGN KEY to Person

username
email
password
nationality FOREIGN KEY to Country
Many-to-one relationship from user to country led to Nationality being added to User Table itself instead of a separate entity.
Specialisation of Person entity

13. LangPreference Table

person_id FOREIGN KEY to Person
language_id FOREIGN KEY to Language
(person_id, language_id) PRIMARY KEY constraint

14. WatchList Table

person_id FOREIGN KEY to Person
Show_id FOREIGN KEY to Show
(person_id, show_id) PRIMARY KEY constraint

15. Mru Table

Show_id FOREIGN KEY to show
username FOREIGN KEY to user
rating_id FOREIGN KEY to rating
(show_id, username) PRIMARY KEY constraint
Many-to-one relationship from (show, user) to rating

16. Actor Table

Show_id FOREIGN KEY to show
person_id FOREIGN KEY to person
(person_id, show_id) PRIMARY KEY constraint
Full participation from show to person
Specialisation of Person entity

17. Director Table

Show_id FOREIGN KEY to show
person_id FOREIGN KEY to person
(person_id, show_id) PRIMARY KEY constraint
Full participation from show to person
Specialisation of Person entity

2) Dataset

We have generated data using two datasets picked from kaggle.

<https://www.kaggle.com/ruchi798/movies-on-netflix-prime-video-hulu-and-disney>

<https://www.kaggle.com/ruchi798/tv-shows-on-netflix-prime-video-hulu-and-disney>

Corresponding files are kaggle_datasets/**movies.csv** and
kaggle_datasets/**tv_shows.csv**

Further we create the tables according to our schema using sql/**create.ddl**

Then we insert the data from above csv files into our schema accordingly using
sql/**data_gen_movies.py** and sql/**data_gen_tvshows.py**

In these scripts, we only add some selected columns which are required in our schema

3) Screen Designs and Controller Information

We are planning to build using django and internalized controllers.

User Role -

HTML files for given screen designs are provided in the template folder

PNG files for given screen designs are provide in the screen_design folder
(names of corresponding files are mentioned above each image)

For controller info given below redirection to a page will call get function corresponding to that page and the action bound to buttons will call corresponding post functions

1. Signup

signup.png, signup.html

Use cases: signup

Controller Info:

- Cancel button redirects to home(guest session)
- Sign up button will add user to db and redirect to home after login

The screenshot shows a 'Sign Up' form with a green header and a white body. The header contains the title 'Sign Up' and a sub-instruction 'Please fill in this form to create an account.' Below the header is a horizontal line. The form fields are labeled 'Email', 'User ID', 'Password', and 'Repeat Password'. Each field has a placeholder text ('Enter Email', 'Enter User ID', 'Enter Password', 'Repeat Password') and a corresponding input field. Below these fields is a checkbox labeled 'Remember me' with a checked status. At the bottom of the form are two buttons: a red 'Cancel' button on the left and a green 'Sign Up' button on the right. The entire form is set against a dark green background.

2. Login

login.png, login.html

Use cases: login

Controller Info:

- Cancel button will redirect to home(guest session)
- Login button will do user authentication and redirect to home and begin user session, or redirect again to login depending whether user is authenticated or not
- Ticking the remember me will save the user-id/password for future sessions
- Change password will send out a reset password mail and redirect again to login page (not sure whether we will implement this)
- Sign up button for new users to redirect to sign up page

The screenshot shows a login interface with a green header and a white content area. The header contains the word "Login". Below it is a instruction: "Please fill in this form to login to your account." The content area has two input fields: "Email Id/ User Id" and "Password", both with placeholder text "Enter Email Id/ User Id" and "Enter Password". There is a checked checkbox labeled "Remember me". Below the inputs are links for "Forgot Password?", "Change Password", "New User? Sign Up", "Cancel", and "Login". The "Cancel" and "Login" buttons are highlighted in red.

3. Home

Logged-in User: home.png, home.html

Use cases: trending movies and tv-shows, recommended movies and tv-shows, top movies and tv-shows, goto selected movie/show goto movies, goto tv-shows, goto watchlist, goto recently added, goto profile, search

Controller Info:

- Home button will redirect to home
- Tv shows button will redirect to tv_shows page
- Movies button will redirect to movies page
- Recently added button will redirect to recently_added page
- MyList button will redirect to mylist page
- Search button will open a bar to type in query and on pressing enter or clicking the search button again will redirect to search_result page
- Account button will redirect to user profile page
- Clicking on any movie/tv-show image or title will redirect to item page

[Home](#)
[TV Shows](#)
[Movies](#)
[Recently Added](#)
[My List](#)

 Account

Trending Now

TV Shows

Movies

[Home](#)
[TV Shows](#)
[Movies](#)
[Recently Added](#)
[My List](#)

 Account

Movies

Top TV Shows

Top Movies

Guest User: home_guest{}.png, home_guest.html

Use Case: trending movie/shows, top movies and shows, goto movies, goto tv shows, goto recently added, search, login, goto to selected movie/show

Controller Info:

- Home button redirects to home_guest page
- Tv-shows button redirects to tv-shows_guest page
- Movies button redirects to movies_guest page
- Recently added button redirects to recently_added page
- Search button will open a bar to type in query and on pressing enter or clicking the search button again will redirect to search_result_guest page
- Login button redirects to login page

- Clicking on a movie/tv-show will redirect to item_guest page

The screenshot displays a user interface for a movie/tv-show recommendation system. At the top, there is a navigation bar with links for Home, TV Shows, Movies, and Recently Added. On the right side of the header are a search icon and a Log In link. Below the header, there are three main sections: "Trending Now", "Top TV Shows", and "Top Movies". Each section contains a grid of movie/tv-show thumbnails. The "Trending Now" section includes "HAWAII FIVE-O", "AGENTS OF S.H.I.E.L.D.", "STEP UP 2 THE STREETS", "marlon", "SHOOTER", and "THE NIGHT SHIFT". The "Top TV Shows" section includes "HAWAII FIVE-O", "marlon", "the FLASH", "BLACK LIGHTNING", "New Girl", "LEGENDS OF TOMORROW", "AGENTS OF S.H.I.E.L.D.", "marlon", "cooking ON HIGH", "QUEER EYE", "SHOOTER", and "JESSICA JONES". The "Top Movies" section includes "WANTED", "THE BOURNE ULTIMATUM", "GUARDIANS OF THE GALAXY VOL. 2", "NATIONAL TREASURE", "BAD BOYS", "STEP UP 2 THE STREETS", "AGENTS OF S.H.I.E.L.D.", "marlon", "cooking ON HIGH", "QUEER EYE", "SHOOTER", and "JESSICA JONES". Each thumbnail features the title of the show/movie and, in some cases, a "NEW EPISODES" or "NEW EPISODES" badge.

4. Movies/ TV-Shows page

(for representation purposes movies and tv-shows designs have been shown as one but while implementing they will be different)

Logged-in User: mtv{}.png, mtv.html

Use Cases: trending movies/tv-shows, recommended movies/tv-shows for different genres, top movies/tv-shows for different genres, goto home, goto watchlist, goto recently added, goto profile, search, goto selected movie/show
Controller Info:

- Navigation buttons work same as home page
- Clicking on a movie/tv-show image will redirect to item page

Home TV Shows Movies Recently Added My List

Account

Trending Now

Thriller

Comedy

Home TV Shows Movies Recently Added My List

Account

Romance

Action

Sci-Fi

Guest User: mtv_guest{}.png, mtv_guest.html

Use Cases: trending movies/tv-shows, top movies/tv-shows for different genres, goto home, goto recently added, search, login, goto selected movie/show

Controller Info:

- Navigation buttons work same as home_guest
- Clicking on a movie/tv-show image will redirect to item_guest page

Home TV Shows Movies Recently Added

Log In

Trending Now

Thriller

Comedy

Home TV Shows Movies Recently Added

Log In

Romance

Action

Sci-Fi

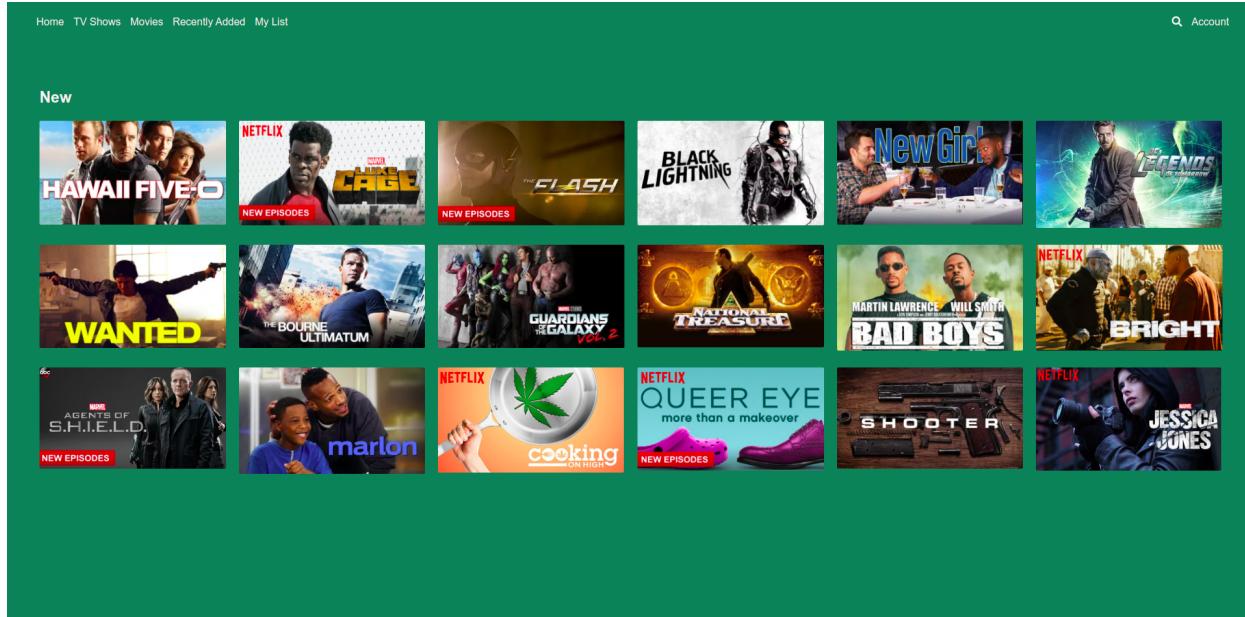
5. Recently Added

Logged-in User: recently_added.png, recently_added.html

Use Cases: movies/show recently added to db, goto selected movie/show goto movies, goto tv-shows, goto watchlist, goto home, goto profile, search

Controller Info:

- Navigation buttons work same as home
- Clicking on an image will redirect to item page

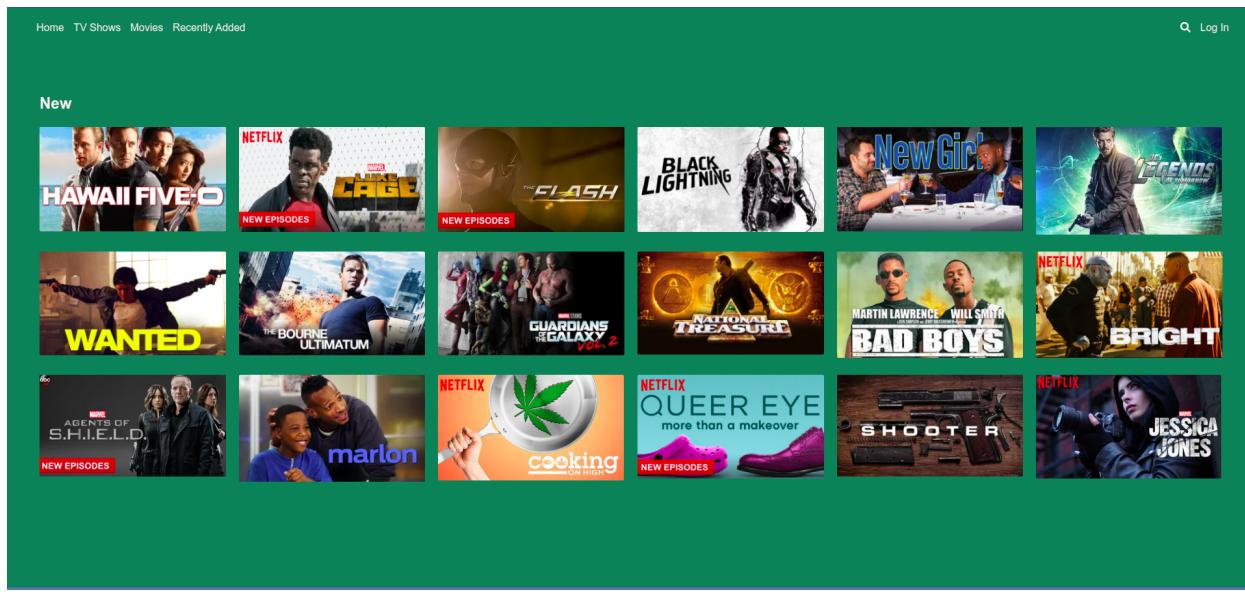


Guest User: recently_added_guest.png, recently_added_guest.html

Use Cases: movies/shows recently added to db, goto home, goto movies, goto TV shows, search, login, goto selected movie/show

Controller Infor:

- Navigation buttons work same as home_guest
- Clicking on an image will redirect to item_guest page



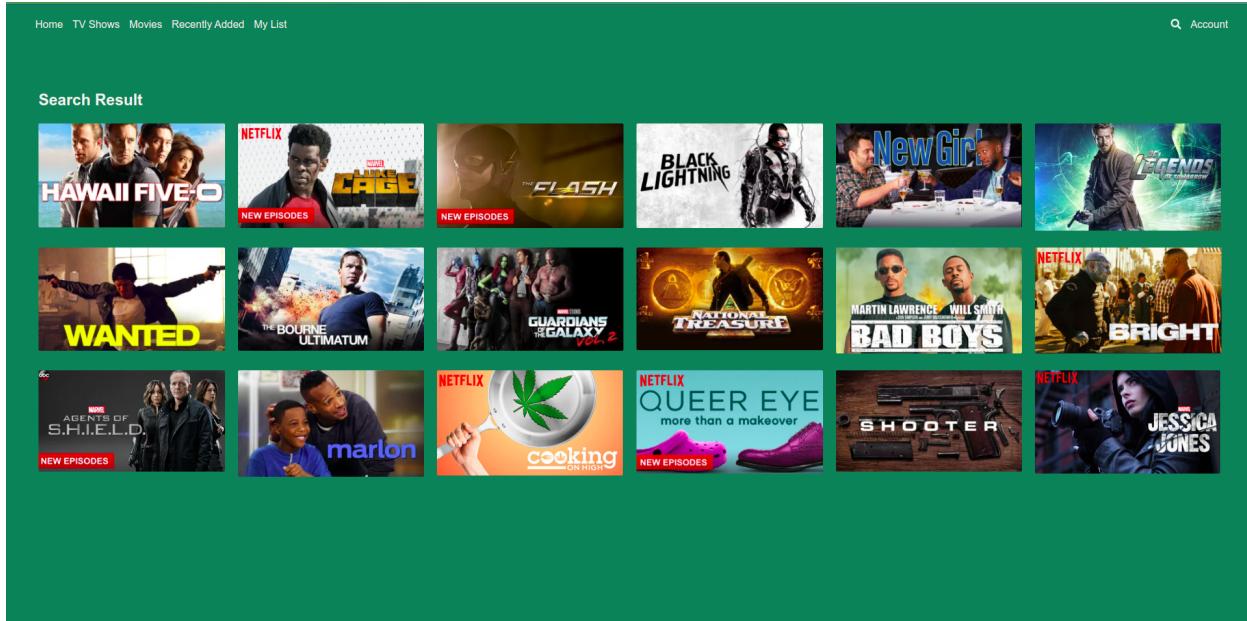
6. Search

Logged-in User: search.png, search.html

Use Cases: search result for given query, goto selected movie/show, goto movies, goto tv-shows, goto watchlist, goto recently added, goto home, goto profile, search

Controller Info:

- Navigation same as home
- Clicking on image redirect to item page



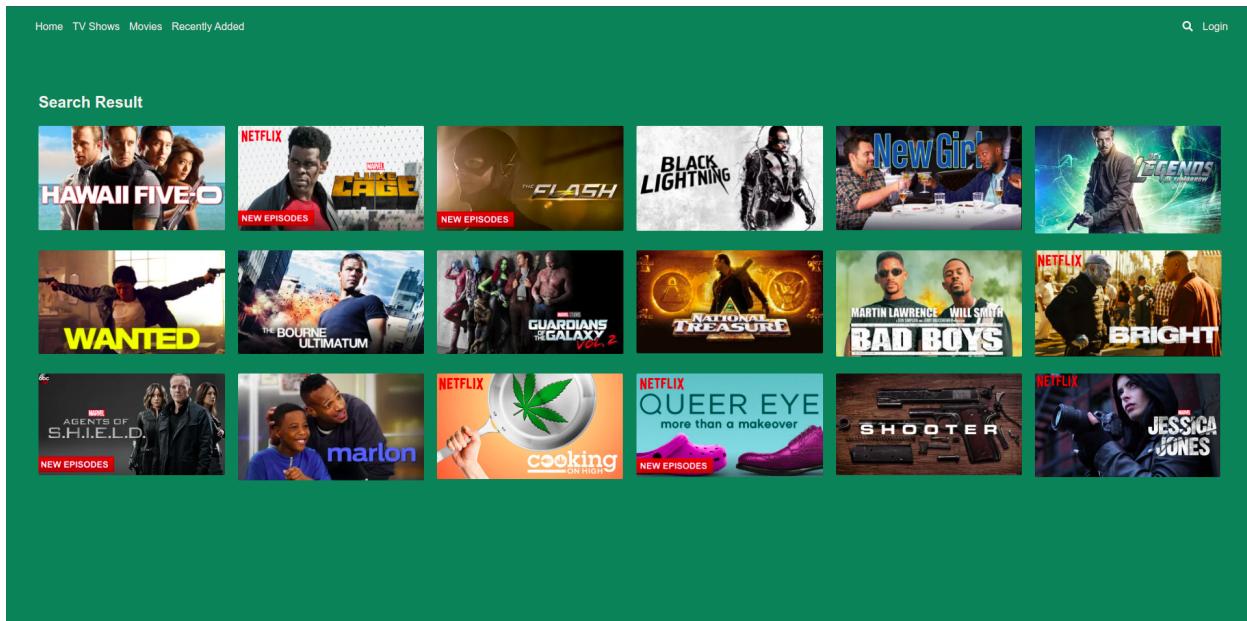
Guest User: search_guest.png, search_guest.html

Use Cases: search results for a given query, goto movies, goto TV shows,

goto home, goto recently added, search, login, goto selected movie/show

Controller Info:

- Navigation same as home_guest
- Clicking on an image redirects to item_guest page



7. Selected movie/ tv-show page

Logged-in User: item.png, item.html

Use Cases: selected movies/show details, goto movies, goto tv-shows, goto watchlist, goto home, goto recently added, goto profile, search, rate, top reviews, all reviews, add to watchlist, upvote and downvote reviews

Controller Info:

- Navigation buttons work same as home
- Clicking on Rate/Review redirects to rate page
- Clicking on add to mylist adds the item to watchlist
- Clicking on imdb link redirects to imdb page for this item in a separate tab
- Clicking on upvote or downvote (only one for a user) will change the upvotes/downvotes for this item
- Clicking on all reviews will redirect to review page



Bad Boys

Two hip detectives protect a witness to a murder while investigating a case of stolen heroin from the evidence storage room from their police precinct.

Duration: 1h 59min
Release Year: 1995
Crew: Michael Bay(Director), George Gallo (story), Michael Barrie(screenplay)
Cast: Will Smith, Martin Lawrence, Lisa Boyle
Genre: Action, Comedy, Crime
Country: U.S.A
Budget: \$1000000
Language: English
Average Rating: 7.1
imdb: 6.5 ([Link to imdb](#))
Rotten Tomato: 72%
Available on: Netflix, Disney

Add to MyList
Rate/ Review

Top Reviews

- Good Movie [Upvote](#) [Downvote](#)
- Must watch [Upvote](#) [Downvote](#)

All Reviews

Guest User: item_guest.png, item_guest.html

Use Cases: selected movies/show details, goto movies, goto tv-shows, goto home, goto recently added, login, search, top reviews, all reviews

Controller Info:

- Navigation buttons work same as home_guest
- Clicking Imdb link redirects to imdb page for this item
- Clicking all reviews button redirects to review_guest page

Bad Boys



Two hip detectives protect a witness to a murder while investigating a case of stolen heroin from the evidence storage room from their police precinct.
Duration: 1h 59min
Release Year: 1995
Crew: Michael Bay(Director), George Gallo (story), Michael Barrie(screenplay)
Cast: Will Smith, Martin Lawrence, Lisa Boyle
Genre: Action, Comedy, Crime
Country: U.S.A
Budget: \$1000000
Language: English
Average Rating: 7.1
imdb: 6.9 ([Link to imdb](#))
Rotten Tomato: 72%
Available on: Netflix, Disney

Top Reviews

- Good Movie
- Must watch

[All Reviews](#)

8. Rate

rate.png, rate.html

Use Cases: rate movie/tv-show , write a review

Controller Info:

- Clicking confirm will add rating and review to db and redirect to item page
- Clicking cancel redirects to item page

Rate

Please fill in this form to rate the movie.

Rate

Enter rating (out of 10)

Review

Write Review (optional)

[Cancel](#) [Confirm](#)

9. All Reviews

review.png, reviews.html

Use cases: read all reviews and rating, upvote and downvote a review, goto movie/tv-show details page, goto home, goto tv shows, goto movies, goto recently added, goto mylist, search, goto profile

Controller Info:

- Navigation button same as home
- Upvote and downvote buttons will update up(down)votes in db
- Clicking go back to item page will redirect to item page



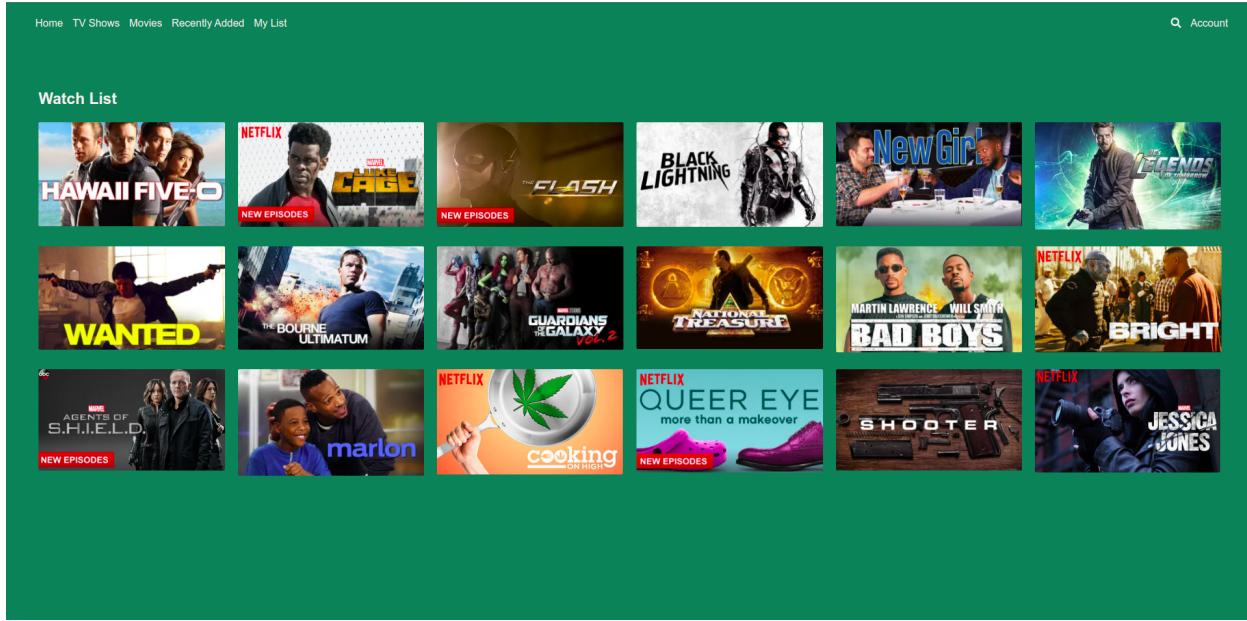
10. Watchlist

mylist.png, mylist.html

Use Cases: display movies/tv-shows in watchlist, goto home, goto tv-shows, goto recently added, goto profile, search

Controller Info:

- Navigation works same as home
- Clicking on image redirects to item page



11. Profile

profile.png, profile.html

Use Cases: display user details, goto edit profile, goto change password, goto home, goto tv-shows, goto movies, goto recently added, goto mylist, search
Controller Info:

- Navigation same as home
- Clicking Edit profile button redirects to edit_profile page
- Clicking Change Password button redirects to change_psw page



12. Edit Profile

edit_profile.png, edit_profile.html

Use Cases: edit profile

Controller Info:

- Clicking cancel will redirect to profile page
- Clicking confirm will update db and redirect to profile page

Edit Profile

Please fill in this form to edit your profile.

Name	<input type="text" value="Enter Name"/>
DOB	<input type="text" value="Enter DOB"/>
Gender	<input type="text" value="Enter Gender"/>
Language Preference	<input type="text" value="Enter Language Preference"/>
Nationality	<input type="text" value="Enter Nationality"/>
<input type="button" value="Cancel"/> <input type="button" value="Confirm"/>	

13. Change Password

change_psw.png, change_psw.html

Use Cases: change password

Controller Info:

- Clicking cancel redirects to profile page
- Clicking confirm updates db and redirects to profile page

Change Password

Please fill in this form to change your password.

Current Password	<input type="text" value="Enter Current Password"/>
New Password	<input type="text" value="Enter New Password"/>
Repeat New Password	<input type="text" value="Repeat New Password"/>
<input type="button" value="Cancel"/> <input type="button" value="Confirm"/>	

Admin Role

The screenshot shows the Django Admin homepage at 127.0.0.1:8000/admin/. The top navigation bar includes links for Site administration, Groups, Users, and a plus sign icon. The main header says "Django administration" and "WELCOME, AVUSH285". The left sidebar lists "AUTHENTICATION AND AUTHORIZATION" (Groups, Users) and "MOVIE" (Actors, Availables, Countries, Directors, Genre_ofs, Genres, Languages, Origin countries, Original languages, OTs, Persons, Shows). The right sidebar displays "Recent actions" and "My actions" with entries like "India Country", "Italy Country", "United States Country", "Disney On", "Prime On", "Hulu On", "Netflix On", "Hindi Language", and "English Language".

Admin Use Cases:

1. Admin homepage
Admin_page.png

Controller use cases:

1. Clicking the Add button lets the admin add new entries to the tables
2. Clicking the edit button lets the admin edit attributes of the existing entries in the tables

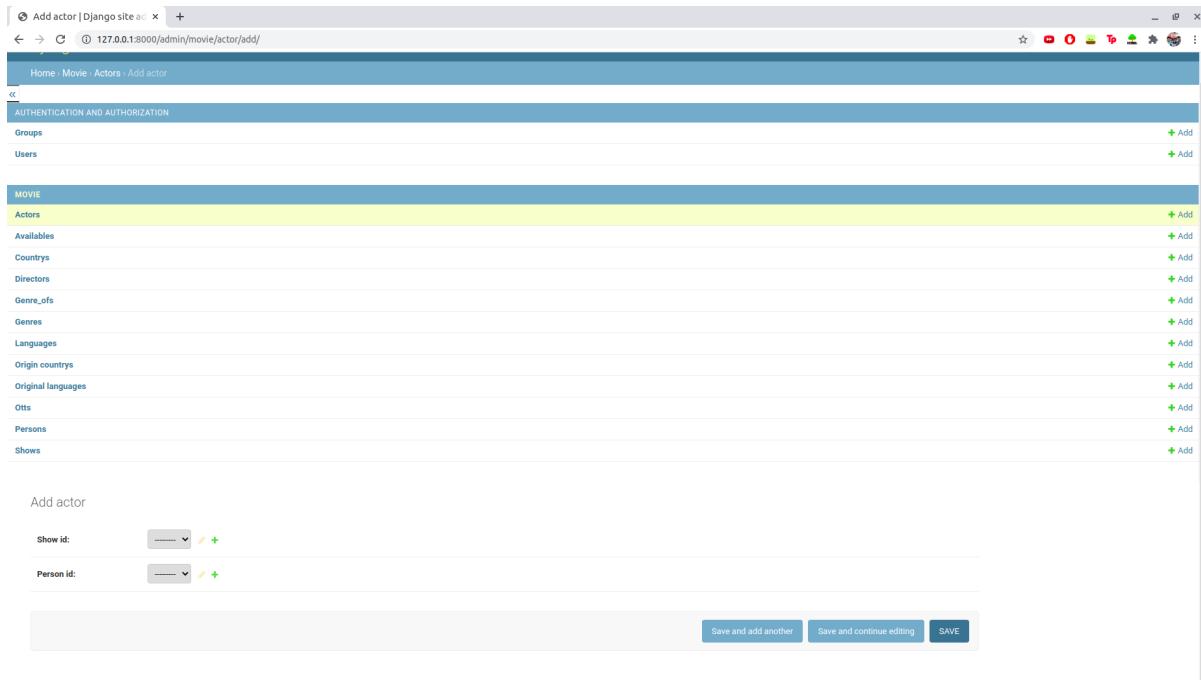
The screenshot shows the Django administration interface for the 'Country' model. At the top, there's a navigation bar with links like 'Select count', 'SamarthSing', 'Time zones', 'python - You', 'WebMail-8 II', 'CS 387-2020', 'Project.Deliv', 'Project.Deliv', 'database scl', 'What Is a Da', 'Movie-Reco', and a '+' button. Below the navigation is a header bar with 'Django administration' and a user welcome message 'WELCOME, AYUSH285 VIEW SITE / CHANGE PASSWORD / LOG OUT'. The main content area has a left sidebar with sections for 'MOVIE' (Actors, Availables, Countries, Directors, Genre_ofs, Genres, Languages, Origin countries, Original languages, OTTs, Persons, Shows) and 'COUNTRIES' (Italy, India, United Kingdom, United States). The 'COUNTRIES' section is highlighted in yellow. At the bottom of the sidebar, there's an 'ADD COUNTRY' button. The main body of the page shows a table with columns for 'COUNTRY' (with checkboxes for each row), 'Action' (with a dropdown menu and a 'Go' button), and '0 of 4 selected'. The table lists four countries: Italy, India, United Kingdom, and United States.

2. Edit Country Table

Admin_edit_country.png

Controller use cases:

1. Add country button lets the admin add new entries to the Country table
2. Clicking on any country name lets the admin change the information in the country table



3. Add Actor to Table

admin_add_actor.png

4) Transaction Queries

User use cases:

```
-- edit profile
-- nm,db,g,nation,pid
update person
set name = nm, dob = db, gender = g, nationality = nation
where person_id = pid

-- add lang pref
-- pid,l
insert into langpreference
select pid,id from language
where name = l

-- search movie
-- s
select id from show
where show.name = s
```

```

-- rate a movie
-- uid,sid,r
if exists (select * from mru where mru.user_id = uid and mru.show_id=sid)
begin
    update rating
        set numeric_rating = r
        where id = (select rating_id from mru where mru.user_id = uid and mru.show_id=sid)
end
else
begin
    insert into rating(id,numeric_rating) select count(*),r from rating
    insert into mru select uid, sid, count(*) from rating
end

-- add to watchlist
-- uid,sid
if not exists (select * from watchlist where user_id=uid and show_id = sid)
begin
    insert into watchlist values(uid,sid)
end

-- add a review
-- uid,sid,r
update rating
    set review = r
    where id = (select rating_id from mru where mru.user_id = uid and mru.show_id=sid)

-- upvote a review
-- uid,sid
update rating
    set upvote = upvote + 1
    where id = (select rating_id from mru where mru.user_id = uid and mru.show_id=sid)

-- downvote a review
-- uid,sid
update rating
    set downvote = downvote + 1
    where id = (select rating_id from mru where mru.user_id = uid and mru.show_id=sid)

-- all reviews of a movie
-- sid
select review from rating,mru
where rating.id = mru.rating_id and mru.show_id = sid

```

```
-- top movies  
-- t
```

```
select * from show  
order by imdb_rating  
limit t
```

Admin use cases:

```
-- add show  
-- list  
insert into show  
    select count(*)+1,list from show
```

```
-- update avg_rating  
-- r,sid  
update show  
    set avg_rating = r  
    where id = sid
```

```
-- add ott  
-- ott_name  
insert into ott  
    select count(*)+1,ott_name  
    from ott
```

```
-- add director  
-- dir_name,sid  
insert into person(id,name)  
    select count(*)+1,dir_name  
    from person  
insert into director  
    select sid,count(*)  
    from person
```

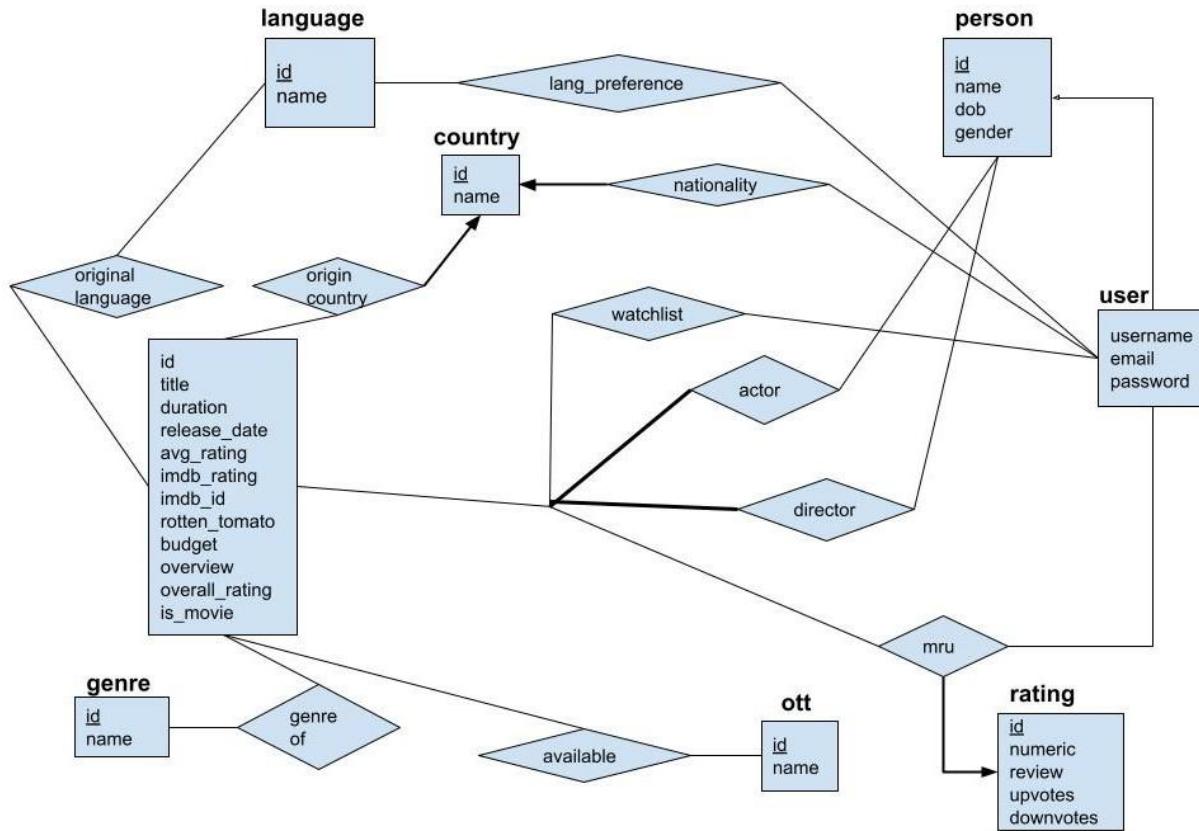
```
-- add actor  
-- ac_name,sid  
insert into person(id,name)  
    select count(*)+1,ac_name  
    from person  
insert into actor  
    select sid,count(*)  
    from person
```

```
-- add genre
-- g
insert into genre
    select count(*)+1,g
from genre
```

```
-- add language
-- l
insert into language
    select count(*)+1,l
from language
```

```
-- add country
-- c
insert into country
    select count(*)+1,c
from country
```

***** Updated ER Diagram**



DDL

```
DROP TABLE if exists mru;
DROP TABLE if exists rating;
DROP TABLE if exists watchlist;
DROP TABLE if exists langpreference;
DROP TABLE if exists userp;
DROP TABLE if exists director;
DROP TABLE if exists actor;
DROP TABLE if exists person;
DROP TABLE if exists origin_country;
DROP TABLE if exists country;
DROP TABLE if exists original_language;
DROP TABLE if exists language;
DROP TABLE if exists genre_of;
DROP TABLE if exists genre;
DROP TABLE if exists available;
DROP TABLE if exists ott;
DROP TABLE if exists show;
```

```
CREATE TABLE show (
    id int,
    title text,
    duration int,
    release_year int,
    avg_rating float,
    imdb_rating float,
    rotten_tomato text,
    is_movie int,
    PRIMARY KEY (id)
);
```

```
CREATE TABLE ott (
    id int,
    name text,
    PRIMARY KEY (id)
);
```

```
CREATE TABLE available (
    show_id int,
    ott_id int,
    PRIMARY KEY (show_id,ott_id),
    FOREIGN KEY(show_id) references show(id),
    FOREIGN KEY(ott_id) references ott(id)
);
```

```
CREATE TABLE genre (
    id int,
    name text,
    PRIMARY KEY (id)
);
```

```
CREATE TABLE genre_of (
    show_id int,
    genre_id int,
    PRIMARY KEY (show_id,genre_id),
    FOREIGN KEY(show_id) references show(id),
    FOREIGN KEY(genre_id) references genre(id)
);
```

```
CREATE TABLE language (
    id int,
    name text,
    PRIMARY KEY (id)
);
```

```
CREATE TABLE original_language (
    show_id int,
    language_id int,
    PRIMARY KEY (show_id,language_id),
    FOREIGN KEY(show_id) references show(id),
    FOREIGN KEY(language_id) references language(id)
);
```

```
CREATE TABLE country (
    id int,
    name text,
    PRIMARY KEY (id)
);
```

```
CREATE TABLE origin_country (
    show_id int,
    country_id int,
    PRIMARY KEY (show_id,country_id),
    FOREIGN KEY(show_id) references show(id),
    FOREIGN KEY(country_id) references country(id)
);
```

```
CREATE TABLE person (
```

```
    id int,
    name text,
    dob date,
    gender text,
    PRIMARY KEY (id)
);

CREATE TABLE actor (
    show_id int,
    person_id int,
    PRIMARY KEY (show_id,person_id),
    FOREIGN KEY(show_id) references show(id),
    FOREIGN KEY(person_id) references person(id)
);

CREATE TABLE director (
    show_id int,
    person_id int,
    PRIMARY KEY (show_id,person_id),
    FOREIGN KEY(show_id) references show(id),
    FOREIGN KEY(person_id) references person(id)
);

CREATE TABLE userp (
    person_id int,
    username text,
    email text,
    password text,
    nationality int,
    PRIMARY KEY (person_id),
    FOREIGN KEY(person_id) references person(id),
    FOREIGN KEY(nationality) references country(id)
);

CREATE TABLE langpreference (
    person_id int,
    language_id int,
    PRIMARY KEY (person_id,language_id),
    FOREIGN KEY(person_id) references person(id),
    FOREIGN KEY(language_id) references language(id)
);

CREATE TABLE watchlist (
    userp_id int,
```

```
show_id int,  
PRIMARY KEY (userp_id,show_id),  
FOREIGN KEY(userp_id) references userp(person_id),  
FOREIGN KEY(show_id) references show(id)  
);
```

```
CREATE TABLE rating (  
id int,  
numeric_rating float,  
review text,  
upvotes int,  
downvotes int,  
PRIMARY KEY (id)  
);
```

```
CREATE TABLE mru (  
show_id int,  
userp_id int,  
rating_id int,  
PRIMARY KEY (show_id,userp_id),  
FOREIGN KEY(show_id) references show(id),  
FOREIGN KEY(userp_id) references userp(person_id),  
FOREIGN KEY(rating_id) references rating(id)  
);
```

```
CREATE INDEX "show.id"  
ON show  
USING btree  
(id);
```

```
CREATE INDEX "person.id"  
ON person  
USING btree  
(id);
```

```
CREATE INDEX "rating.id"  
ON rating  
USING btree  
(id);
```