



3 January 2023

Prepared by:

Ahmet Furkan Akdamar - 190316068

Tuğçe Durmaz - 190316051

Bahri Ün - 190316063

Contents

1.Introduction.....	2
2.E-R diagram	2
3. Table Details	3
3.1 Users.....	3
3.2 Movies.....	4
3.3 Series.....	5
3.5 Downloads.....	6
3.6 Watch History.....	7
3.7Watch List.....	8
3.8 New Releases.....	9
3.9 Popular on Netflix.....	10
4. Queries	11
4.1 Query 1.....	11
4.2 Query 2.....	11
4.3 Query 3.....	12
4.4 Query 4.....	12
4.5 Query 5.....	13
4.6 Query 6.....	13
4.7 Query 7.....	14
4.8 Query 8.....	14
4.9 Query 9.....	15
4.10 Query 10.....	15

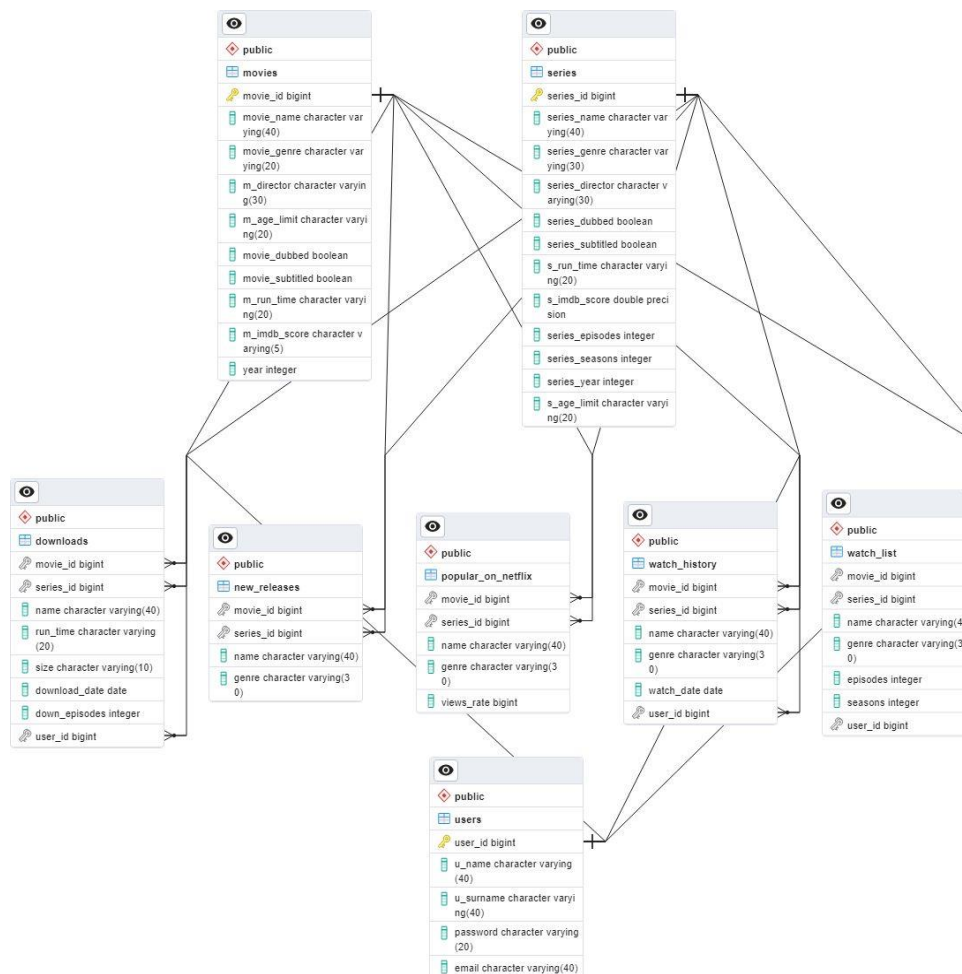
DATABASE MANANGMENT SYSTEM PROJECT

1- Introduction

In this project, we chose to design a NETFLIX database.

We created 8 tables while designing the database.

2- E-R Diagram



The tables we created are as follows:

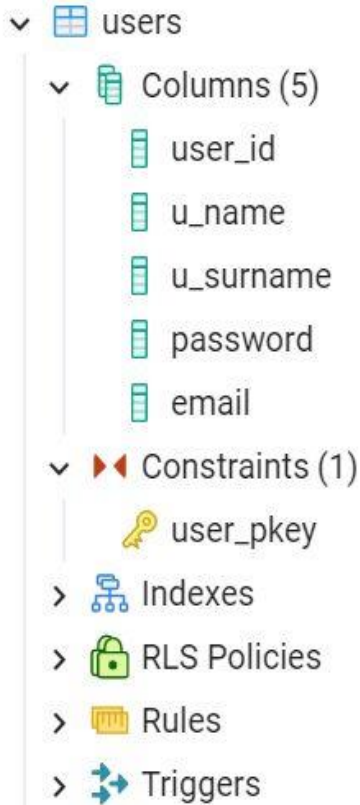
downloads , movies , new_releases , popular_on_netflix , series , users , watch_history , watch_list.














The data types are located next to the table name for each column in the E.R schema.

A more detailed explanation of the tables is below.

3- Table Details

3.1 Users



▼	 users
▼	 Columns (5)
	 user_id
	 u_name
	 u_surname
	 password
	 email
▼	 Constraints (1)
	 user_pkey
>	 Indexes
>	 RLS Policies
>	 Rules
>	 Triggers

We created the primary key of this table as user_id and it is used as a foreign key in many tables.

The reason for this is to keep the user's data without losing it.

user_id is a unique attribute as there is only one for each user.

Name, surname, password, e-mail information is kept in this table.

Also in the table

primary key : user_id

	user_id [PK] bigint	u_name character varying (40)	u_surname character varying (40)	password character varying (20)	email character varying (40)
1	1000001	Furkan	Akdamar	Fa123456	furkan123@gmail.com
2	1000002	Tuğçe	Durmaz	Td123456	tuğçe123@gmail.com
3	1000003	Bahri	Ün	Bü123456	bahri123@gmail.com
4	1000004	Sümeyye	Gömeç	Sg123456	sümeyye123@gmail.c...
5	1000005	Mehmet	Yılmaz	My123456	mehmet123@gmail.c...
6	1000006	Tony	Stark	Ts123456	tony123@gmail.com
7	1000007	Haluk	Bilginer	Hb123456	haluk123@gmail.com
8	1000008	Hugh	Jackman	Hj123456	jackman123@gmail.c...
9	1000009	Yağmur	Giriş	Yg123456	yagmur123@gmail.co...
10	1000010	Emine	Eker	Ee123456	emine123@gmail.com

3.2 Movies

▼	movies
▼	Columns (10)
	movie_id
	movie_name
	movie_genre
	m_director
	m_age_limit
	movie_dubbed
	movie_subtitled
	m_run_time
	m_imdb_score
	year
▼	Constraints (1)
	movie_pkey
>	Indexes
>	RLS Policies
>	Rules
>	Triggers

We have saved movie information of various genres.

We have given a unique id value for each movie. We have classified the films so that the films can be found.

According to their wishes; They can choose according to genre, director, age limit, subtitles, dubbing, run time, imdb score and production year.

Also in the table

primary key : movie_id

	movie_id [PK] bigint	movie_name character varying (40)	movie_genre character varying (20)	m_director character varying (30)	m_age_limit character varying (20)	movie_dubbed boolean	movie_subtitled boolean	m_run_time character varying (20)	m_imdb_score character varying (5)	year integer
1	1001	Nope	Fantastic	Jordan Peele	13+	true	true	2h 10m	6.9	2022
2	1002	Harry Potter and th...	Adventure	Chris Columbus	7+	true	true	2h 32m	7.6	2001
3	1003	The Lord of the Rin...	Adventure	Peter Jackson	13+	true	true	2h 58m	8.8	2001
4	1004	The Hobbit	Adventure	Peter Jackson	13+	true	true	2h 49m	7.8	2012
5	1005	Pirates of the Cari...	Action	Gore Verbinski	13+	true	true	2h 23m	8.1	2003
6	1006	The Shawshank Re...	Drama	Frank Darabont	13+	true	true	2h 2m	9.3	1994
7	1007	The Godfather	Drama	Francis Ford Copp...	18+	true	true	2h 55m	9.2	1972
8	1008	Forrest Gump	Drama	Robert Zemeckis	7+	true	true	2h 22m	8.8	1994
9	1009	Inception	Sci-Fi	Christopher Nolan	13+	true	true	2h 28m	8.8	2010
10	1010	The Matrix 1	Sci-Fi	Lana Wachowski-...	18+	true	true	2h 16m	8.7	1999

3.3 Series

series
Columns (12)
series_id
series_name
series_genre
series_director
series_dubbed
series_subtitled
s_run_time
s_imdb_score
series_episodes
series_seasons
series_year
s_age_limit
Constraints (1)
series_pkey
Indexes
RLS Policies
Rules
Triggers

We recorded various types of sequence information.

We have given a unique id value for each array. We have classified the sequences so that the sequences can be found.

According to their wishes; They can choose according to genre, director, age limit, subtitle, dubbing, broadcast time, imdb score, episode, seasons and production year.

Also in the table

primary key : series_id

	user_id [PK] bigint	u_name character varying (40)	u_surname character varying (40)	password character varying (20)	email character varying (40)
1	1000001	Furkan	Akdamar	Fa123456	furkan123@gmail.com
2	1000002	Tuğçe	Durmaz	Td123456	tuğçe123@gmail.com
3	1000003	Bahri	Ün	Bü123456	bahri123@gmail.com
4	1000004	Sümeyye	Gömeç	Sg123456	sümeyye123@gmail.c...
5	1000005	Mehmet	Yılmaz	My123456	mehmet123@gmail.c...
6	1000006	Tony	Stark	Ts123456	tony123@gmail.com
7	1000007	Haluk	Bilginer	Hb123456	haluk123@gmail.com
8	1000008	Hugh	Jackman	Hj123456	jackman123@gmail.c...
9	1000009	Yağmur	Giriş	Yg123456	yagmur123@gmail.co...
10	1000010	Emine	Eker	Ee123456	emine123@gmail.com

3.4 Downloads

▼	📁 downloads
▼	📄 Columns (8)
	📄 movie_id
	📄 series_id
	📄 name
	📄 run_time
	📄 size
	📄 download_date
	📄 down_episodes
	📄 user_id
▼	🔑 Constraints (3)
	🔑 movie_foreign
	🔑 series_foreign
	🔑 user_foreign
>	🔗 Indexes
>	🔒 RLS Policies
>	📋 Rules
>	⚡ Triggers

We have saved various movie and series information that the user has downloaded. Download here; We have added movie series id, film id , name genre, download date , size, run time information.

Also in the table

foreign key: movie_id , series_id, user_id

	movie_id bigint 🔒	series_id bigint 🔒	name character varying (40) 🔒	run_time character varying (20) 🔒	size character varying (10) 🔒	download_date date 🔒	down_episodes integer 🔒	user_id bigint 🔒
1	1001	[null]	Nope	2h 10m	1.5Gb	2022-03-21	[null]	1000001
2	1003	[null]	The Lord of The Rings...	2h 58m	2.9Gb	2022-04-28	[null]	1000001
3	1006	[null]	The Shawshank Rede...	2h 2m	1.7Gb	2022-05-15	[null]	1000001
4	1007	[null]	The Godfather	2h 55m	2.5Gb	2022-05-16	[null]	1000001
5	1010	[null]	The Matrix	2h 16m	2.1Gb	2022-05-26	[null]	1000001
6	[null]	10001	Breaking Bad	49m	2.5Gb	2022-05-26	5	1000001
7	[null]	10003	Peaky Blinder	1h	1.8Gb	2022-05-30	3	1000001
8	[null]	10004	Vikings	44m	3.5Gb	2022-06-03	7	1000001
9	[null]	10007	Lucifer	42m	0.7Gb	2022-06-05	1	1000001
10	[null]	10009	La Casa De Papel	1h 10m	2Gb	2022-06-07	4	1000001

3.5 Watch History

▼	watch_history
▼	Columns (6)
	movie_id
	series_id
	name
	genre
	watch_date
	user_id
▼	Constraints (3)
	movie_id
	series_id
	user_foreign
>	Indexes
>	RLS Policies
>	Rules
>	Triggers

We have recorded various movie and series information watched by the user. Watched here; We have added movie series id, film id, name genre, watch date information. And we provided the connection with the user with the user id data.

Also in the table

foreign key: movie_id , series_id, user_id

	movie_id bigint	series_id bigint	name character varying (40)	genre character varying (30)	watch_date date	user_id bigint
1	1003	[null]	The Lord of The Rings...	Adventure	2022-03-01	1000001
2	1006	[null]	The Shawshank Rede...	Drama	2022-03-03	1000001
3	1007	[null]	The Godfather	Drama	2022-03-28	1000001
4	1008	[null]	Forrest Gump	Drama	2022-12-20	1000001
5	1010	[null]	The Matrix	Sci-Fi	2022-06-20	1000001
6	[null]	10001	Breaking Bad	Crime	2022-07-20	1000001
7	[null]	10003	Peaky Blinder	Crime	2022-08-25	1000001
8	[null]	10005	Black Mirror	Sci-Fi	2022-09-01	1000001
9	[null]	10007	Lucifer	Crime	2022-09-09	1000001
10	[null]	10010	The Quin s Gambit	Drama	2022-10-10	1000001

3.6 Watch List

watch_list
Columns (7)
movie_id
series_id
name
genre
episodes
seasons
user_id
Constraints (3)
movie_id
series_id
user_foreign
Indexes
RLS Policies
Rules
Triggers

We have recorded various movie and series information that the user has watched or wanted to watch and added to the watch list. Watched here; We have added movie serial id, name genre, watch date information. And we provided the connection with the user with the user id data.

Also in the table

foreign key: movie_id , series_id, user_id

	movie_id bigint	series_id bigint	name character varying (40)	genre character varying (30)	episodes integer	seasons integer	user_id bigint
1	1001	[null]	Nope	Fantastic	[null]	[null]	1000001
2	1004	[null]	The Hobbit	Adventure	[null]	[null]	1000001
3	1005	[null]	Pirates of the Caribbe...	Action	[null]	[null]	1000001
4	1007	[null]	The Gotfather	Drama	[null]	[null]	1000001
5	1009	[null]	Inception	Sci-Fi	[null]	[null]	1000001
6	[null]	10004	Vikings	Action	93	6	1000001
7	[null]	10006	Sense 8	Sci-Fi	22	5	1000001
8	[null]	10007	Lucifer	Crime	93	6	1000001
9	[null]	10009	La Casa De Papel	Crime	41	5	1000001
10	[null]	10010	The Quin s Gambit	Drama	7	1	1000001

3.8 New Releases

new_releases

- Columns (4)
 - movie_id
 - series_id
 - name
 - genre
- Constraints (2)
 - movie_id
 - series_id
- Indexes
- RLS Policies
- Rules
- Triggers

We have recorded the newly added movie and series information on the Netflix platform.

Here we used movie id , series id , name and genre information.

Also in the table

foreign key: movie_id , series_id

	movie_id bigint	series_id bigint	name character varying (40)	genre character varying (30)
1	1002	[null]	Harry Potter and Phil...	Adventure
2	1004	[null]	The Hobbit	Adventure
3	1005	[null]	Pirates of the Caribbe...	Action
4	1008	[null]	Forrest Gump	Drama
5	1009	[null]	Inception	Sci-Fi
6	[null]	10002	Wednesday	Horror
7	[null]	10004	Vikings	Action
8	[null]	10005	Black Mirror	Sci-Fi
9	[null]	10006	Sense 8	Sci-Fi
10	[null]	10008	Avatar: The Last Airbe...	Animation

3.9 Popular on Netflix

✓	popular_on_netflix
✓	Columns (5)
	movie_id
	series_id
	name
	genre
	views_rate
✓	Constraints (2)
	movie_id
	series_id
>	Indexes
>	RLS Policies
>	Rules
>	Triggers

We saved the most watched movie and series information on the Netflix platform.

Here we used movie id , series id , name and genre and views rate information.

Also in the table

foreign key: movie_id , series_id

	movie_id bigint	series_id bigint	name character varying (40)	genre character varying (30)	views_rate bigint
1	1002	[null]	Harry Potter and the ...	Adventure	100000
2	1004	[null]	The Hobbit	Adventure	80000
3	1008	[null]	Forrest Gump	Drama	90000
4	1009	[null]	Inception	Sci-Fi	60000
5	1010	[null]	The Matrix 1	Sci-Fi	75000
6	[null]	10002	Wednesday	Horror	50000
7	[null]	10005	Black Mirror	Sci-Fi	650000
8	[null]	10006	Sense 8	Sci-Fi	550000
9	[null]	10008	Avatar: The Last Airbe...	Animation	450000
10	[null]	10010	The Quin s Gambit	Drama	850000

4- Queries

4.1 Query 1

```
1  /*The last movie or series downloaded by the user whose user_id is 1000001.*/
2  select user_id, movie_id, series_id, name, download_date from downloads
3  where download_date = (select max(download_date) from downloads)
```

Data Output Messages Notifications

	user_id bigint	movie_id bigint	series_id bigint	name character varying (40)	download_date date
1	1000001	[null]	10009	La Casa De Papel	2022-06-07

4.2 Query 2

```
1  /*Show adventure movies from pre-2005 movies and crime dramas from post-2005 series. */
2  select movie_name as name, year, movie_genre as genre from movies where movie_genre = 'Adventure' and year < 2005
3  union
4  select series_name, series_year, series_genre from series where series_genre = 'Crime' and series_year > 2005
```

Data Output Messages Notifications

	name character varying (40)	year integer	genre character varying
1	Breaking Bad	2008	Crime
2	Harry Potter and the ...	2001	Adventure
3	La Casa De Papel	2017	Crime
4	Lucifer	2016	Crime
5	Peaky Blinder	2013	Crime
6	The Lord of the Rings 1	2001	Adventure

4.3 Query 3

```
1  /*The oldest watched movie or series by the user whose user_id is 1000001.*/
2  select user_id, movie_id, series_id, name, watch_date from watch_history
3  where watch_date = (select min(watch_date) from watch_history)
4
5
6
```

Data Output Messages Notifications

	user_id bigint	movie_id bigint	series_id bigint	name character varying (40)	watch_date date
1	1000001	1003	[null]	The Lord of The Rings...	2022-03-01

4.4 Query 4

```
1  /* Top 5 series with the highest imdb rating.*/
2  select series_name, s_imdb_score from series order by s_imdb_score desc limit 5
3
```

Data Output Messages Notifications

	series_name character varying (40)	s_imdb_score double precision
1	Breaking Bad	9.5
2	Avatar: The Last Airbend...	9.3
3	Black Mirror	8.8
4	Peaky Blinder	8.8
5	The Quin s Gambit	8.6

4.4 Query 5

```
1  /*Top 5 movies on imdb */
2  select movie_name, m_imdb_score from movies order by m_imdb_score desc limit 5
3
```

Data Output

Messages

Notifications

	movie_name character varying (40)	m_imdb_score character varying (5)
1	The Shawshank Rede...	9.3
2	The Godfather	9.2
3	Inception	8.8
4	The Lord of the Rings 1	8.8
5	Forrest Gump	8.8

4.5 Query 6

1

/* Top 5 rankings of popular series and movies on Netflix.*/

2

select name, views_rate from popular_on_netflix order by views_rate desc limit 5

3

Data Output

Messages

Notifications

≡+

	name character varying (40)	views_rate bigint
1	The Quin s Gambit	850000
2	Black Mirror	650000
3	Sense 8	550000
4	Avatar: The Last Airbender	450000
5	Harry Potter and the Philosopher s Sto...	100000

4.6 Query 7

```
1  /* Show series that are dubbed and have less than 30 episodes in total.*/
2  select series_name, series_genre, series_dubbed, series_episodes from series
3  where series_dubbed = True and series_episodes < 30
4
```

Data Output Messages Notifications

	series_name character varying (40)	series_genre character varying (30)	series_dubbed boolean	series_episodes integer
1	Wednesday	Horror	true	8
2	The Quin s Gambit	Drama	true	7

4.7 Query 8

```
1  /*Find the average number of episodes by genre of series released after 2015.*/
2  select series_genre, avg(series_episodes) as avg_episodes
3  from series where series_year>2015 group by series_genre
4
```

Data Output Messages Notifications

	series_genre character varying (30)	avg_episodes numeric
1	Drama	7.0000000000
2	Horror	8.0000000000
3	Crime	67.0000000000

4.9 Query 9

```
1 /*Find the series and movie genres that are popular on netflix with an average view rate of more than 90000.*/
2 select genre, avg.views_rate) as avg_views_rate
3 from popular_on_netflix group by genre having (avg.views_rate)>90000)
```

Data Output Messages Notifications



	genre character varying (30)	avg_views_rate numeric
1	Animation	450000.000000000000
2	Drama	470000.000000000000
3	Sci-Fi	333750.000000000000

4.10 Query 10

```
1 /*Find series or movies watched after 2022-03-03 and
2 movies or series watched before 2022-09-09 from watch history.*/
3 select name, watch_date from watch_history where watch_date < '2022-09-09'
4 intersect
5 select name, watch_date from watch_history where watch_date > '2022-03-03'
6
```

Data Output Messages Notifications



	name character varying (40)	watch_date date
1	Peaky Blinder	2022-08-25
2	The Godfather	2022-03-28
3	Breaking Bad	2022-07-20
4	Black Mirror	2022-09-01
5	The Matrix	2022-06-20