

# CINETECH PRACTICAL ASSIGNMENT 5

23535246 : Lebogang Masenya

22747363 : Aundrea Ncube

21824241 : Mpho Siminya

22526171 : Nobuhle Mtshali

23539764 : Amantle Temu

23618583 : Malaika Kamangu

## **TASK 1**

### **Research**

What makes a good streaming service:

- Content Library:

A comprehensive content library is paramount to the success of a streaming service. It is imperative to seek out a platform that offers a diverse selection of movies, TV shows, documentaries, and original programming to cater to a wide range of viewer preferences and interests.

- Original Content:

Exclusive original shows and movies play a pivotal role in distinguishing a streaming service from its competitors. A prime example of this is

Netflix, renowned for its extensive catalogue of original programming, encompassing both acclaimed series and blockbuster films.

- Pricing:

When evaluating streaming services, it is crucial to consider the subscription cost. While some platforms offer competitive pricing, others may feature premium tiers with additional perks. Striking a balance between affordability and content quality is key to ensuring a satisfying user experience.

- User Interface:

A user-friendly interface greatly enhances the navigation and content discovery process for viewers. Services like Hulu and Disney+ are lauded for their intuitive interfaces, facilitating seamless interaction with the platform.

- Streaming Quality:

The capability to support high-definition (HD) and 4K streaming is imperative for delivering an immersive viewing experience. It is imperative to select a service that ensures smooth playback even under varying internet connection speeds.

- Device Compatibility:

Compatibility across a diverse array of devices, including smartphones, tablets, smart TVs, and gaming consoles, is essential to accommodate the diverse viewing habits of users. Most popular streaming services are readily accessible across multiple platforms.

- Ad-Free Options:

For users seeking an uninterrupted viewing experience, the availability of ad-free options is crucial. Platforms such as Amazon Prime Video offer competitive ad-free pricing, catering to this preference.

- Live TV and Sports:

For enthusiasts of live TV channels or sports content, platforms like YouTube TV or Fubo are noteworthy considerations, offering an extensive array of live programming options.

- Family-Friendly Content:

Streaming services that curate family-friendly content are highly valued by households. Disney+, for instance, stands out as an exemplary choice due to its extensive library of family-friendly entertainment options.

- Content Discovery and Personalization:

Effective content discovery features, such as personalised recommendations and curated playlists, help users discover new titles tailored to their interests. Implementing algorithms that analyse user viewing history, preferences, and ratings can enhance the accuracy of content recommendations over time.

- Offline Viewing:

Offline viewing functionality allows users to download content for later viewing without requiring an internet connection. This feature is particularly valuable for users with limited or unreliable internet access.

- Accessibility and Inclusivity:

Providing accessibility features such as closed captions, audio descriptions, and customizable subtitles ensures that the streaming service is inclusive and caters to users with diverse needs.

- Community Engagement:

Foster a sense of community among users through features such as discussion forums, user-generated content, and social media integration. Encourage user interaction and feedback to create a more engaging and immersive experience.

- Brand Identity and Differentiation:

Cultivate a distinct brand identity and positioning that sets the streaming service apart from competitors such as exclusive content partnerships,

niche genre offerings, or innovative features, to attract and retain subscribers

### **Movie/Film trends in genres**

- 1.Hardcore Sci-Fi
- 2.Female Superheroes
- 3.Film to TV adaptations.
- 4.Video On Demand
- 5.Classic Horror
- 6.Embrace of Niceness

### **Streaming sites trend**

- 1.Week-to-week releases
- 2.Platform exclusives
- 3.Docuseries
- 4.Traditional TV channels joining streaming services.
- 5.On-Demand convenience
- 6.Social media platforms hosting live-streamed video.
- 7.Spending on indie content

### **Streaming sites strengths:**

- Netflix: Best for Original shows
- Hulu: Best for Live and On-Demand content
- Peacock: Affordable
- Amazon Prime Video: Best for On-Demand shows and Movies.
- HBO Max: Best for HBO shows and Movies.
- Disney+: Best for Family friendly content
- Crunchyroll: Best for Anime

### **References**

The Best Video Streaming Services for 2024 | PCMag

<https://www.reviews.org/tv-service/best-streaming-service>

<https://mybroadband.co.za/news/broadcasting/525088-best-rated-streaming-apps-in-south-africa-with-prices.html>

11 Huge Streaming Trends 2024-2027 (explodingtopics.com)

10 Movie And TV Trends We Love (slashfilm.com)

An Industry Transformed: Four Emerging Trends in Film & TV  
(visualcapitalist.com)

Genre Trend Analysis of 2023 Film and Television | Parrot Analytics

EER Diagram

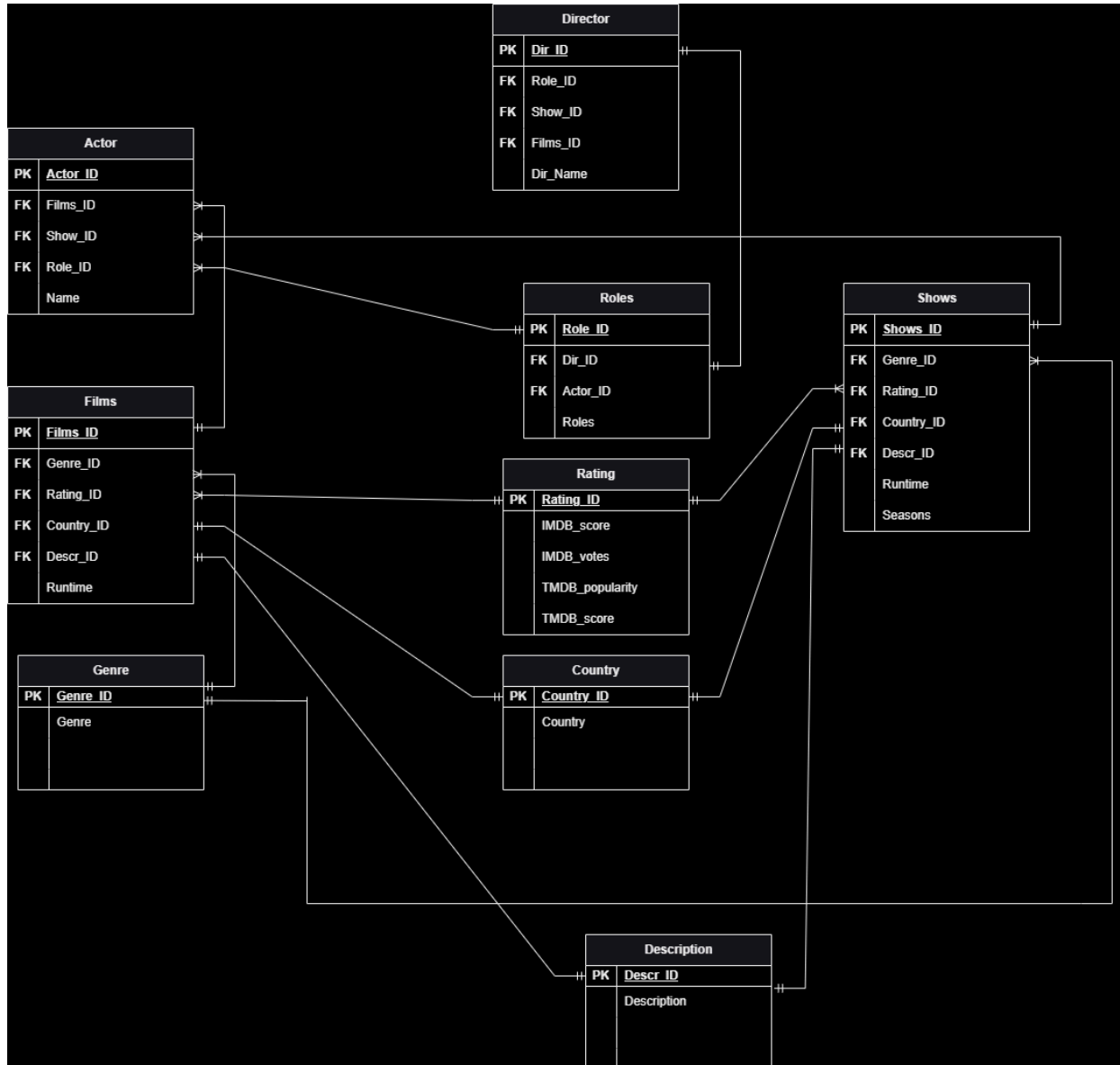
EER Diagram to Relational Mapping

Relational Schema

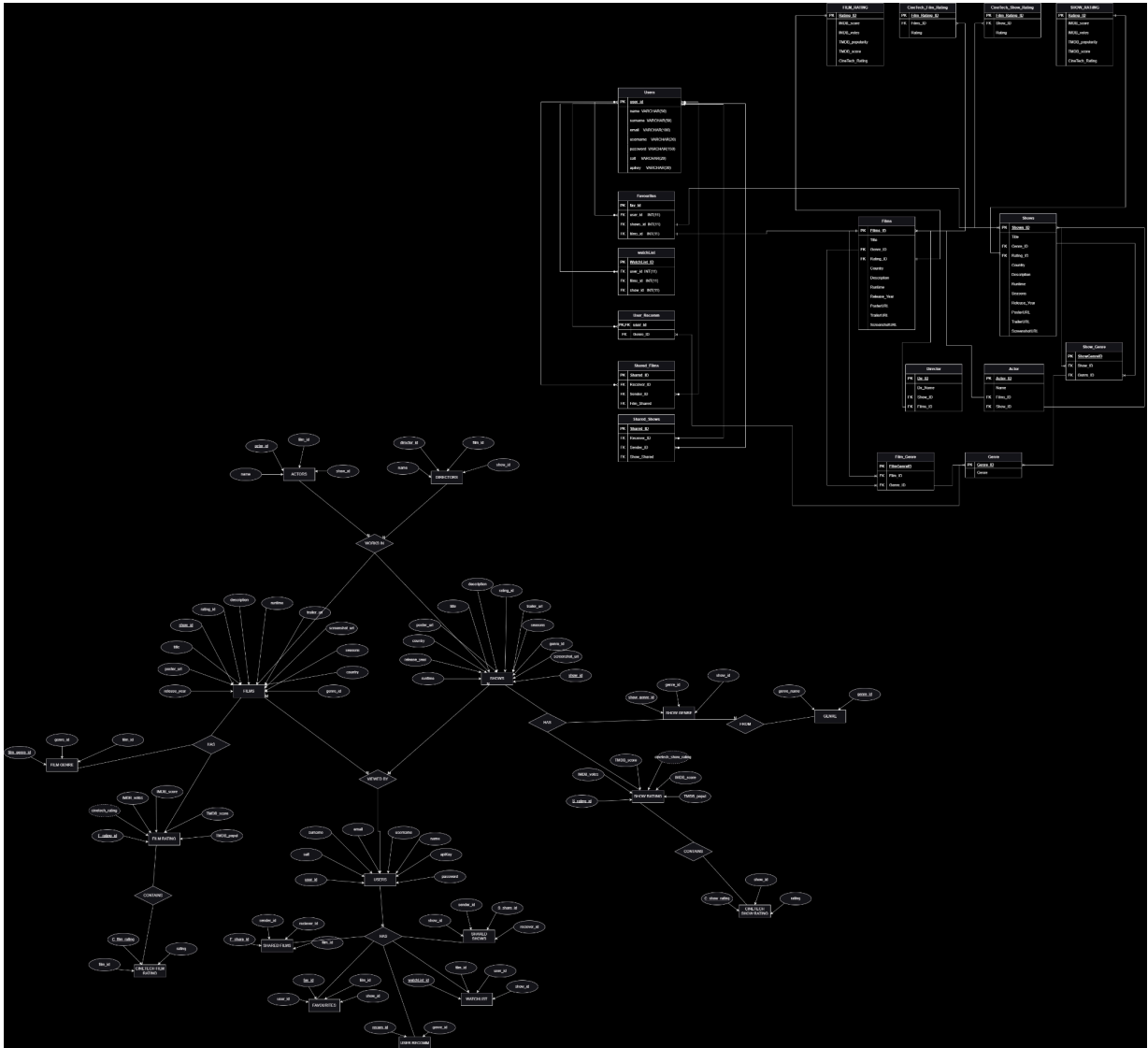
Dataset: Netflix

## TASK 2 : (E)ER-DIAGRAM

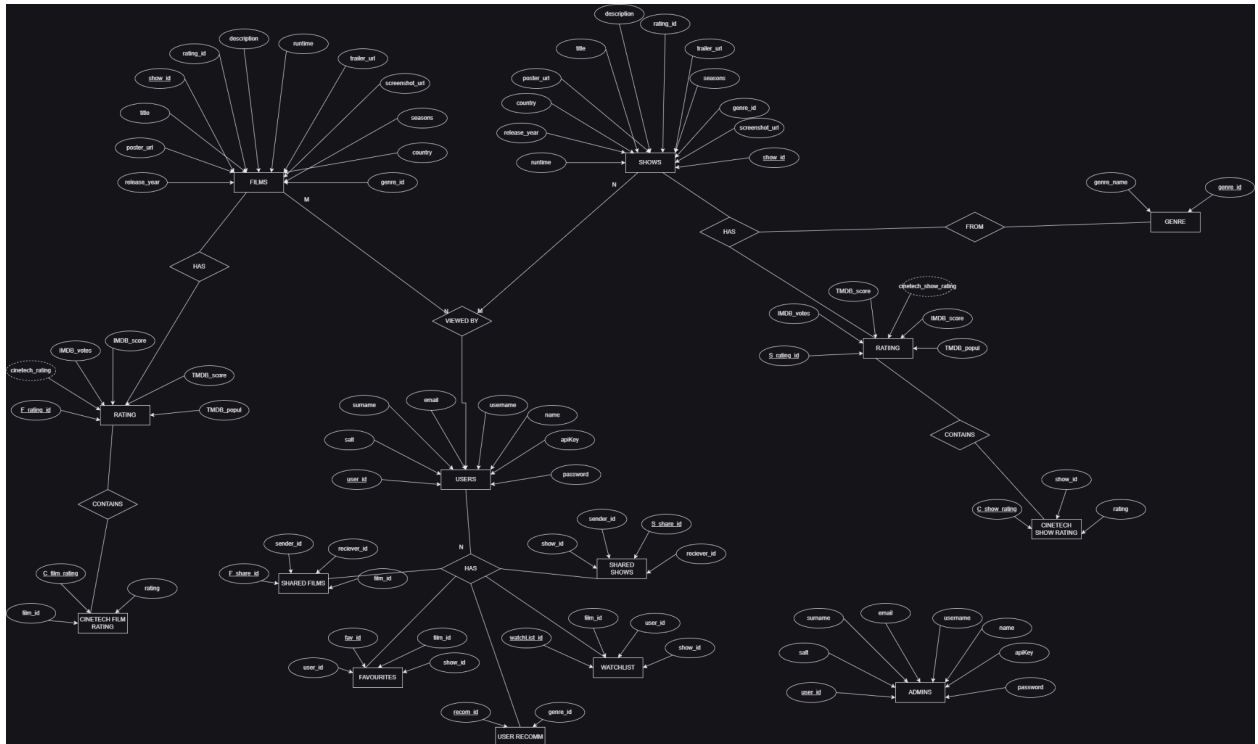
- First Draft:



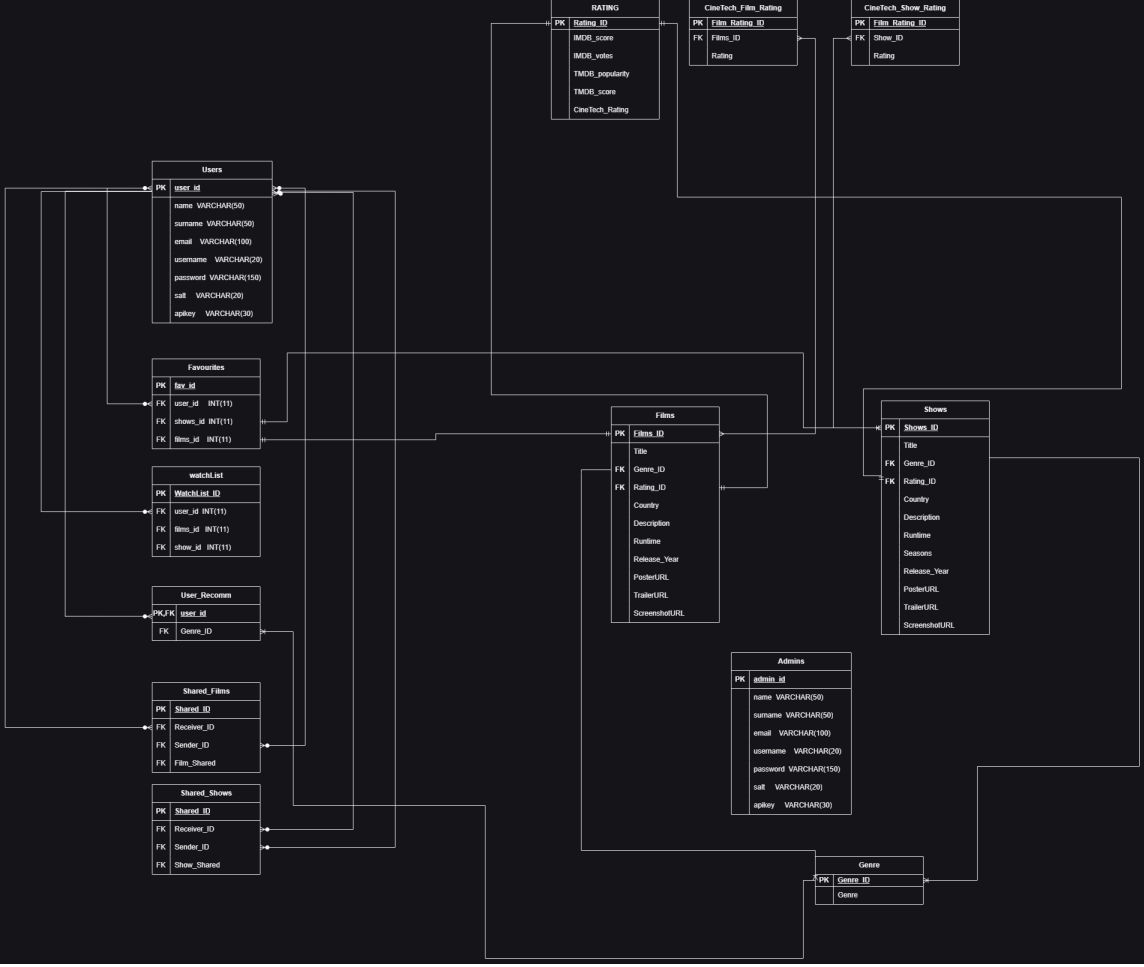
- Second Draft:



- Final Draft:

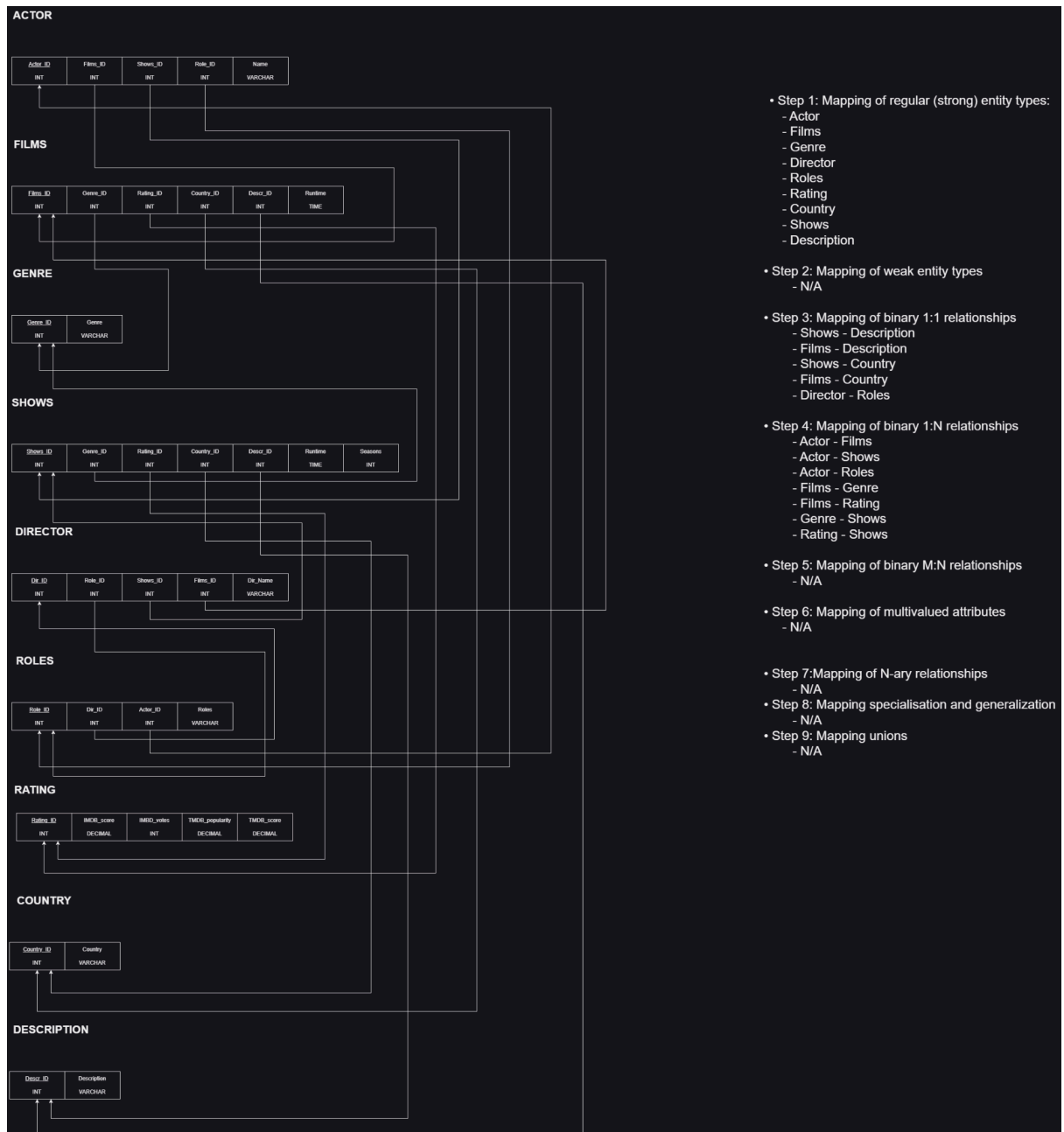






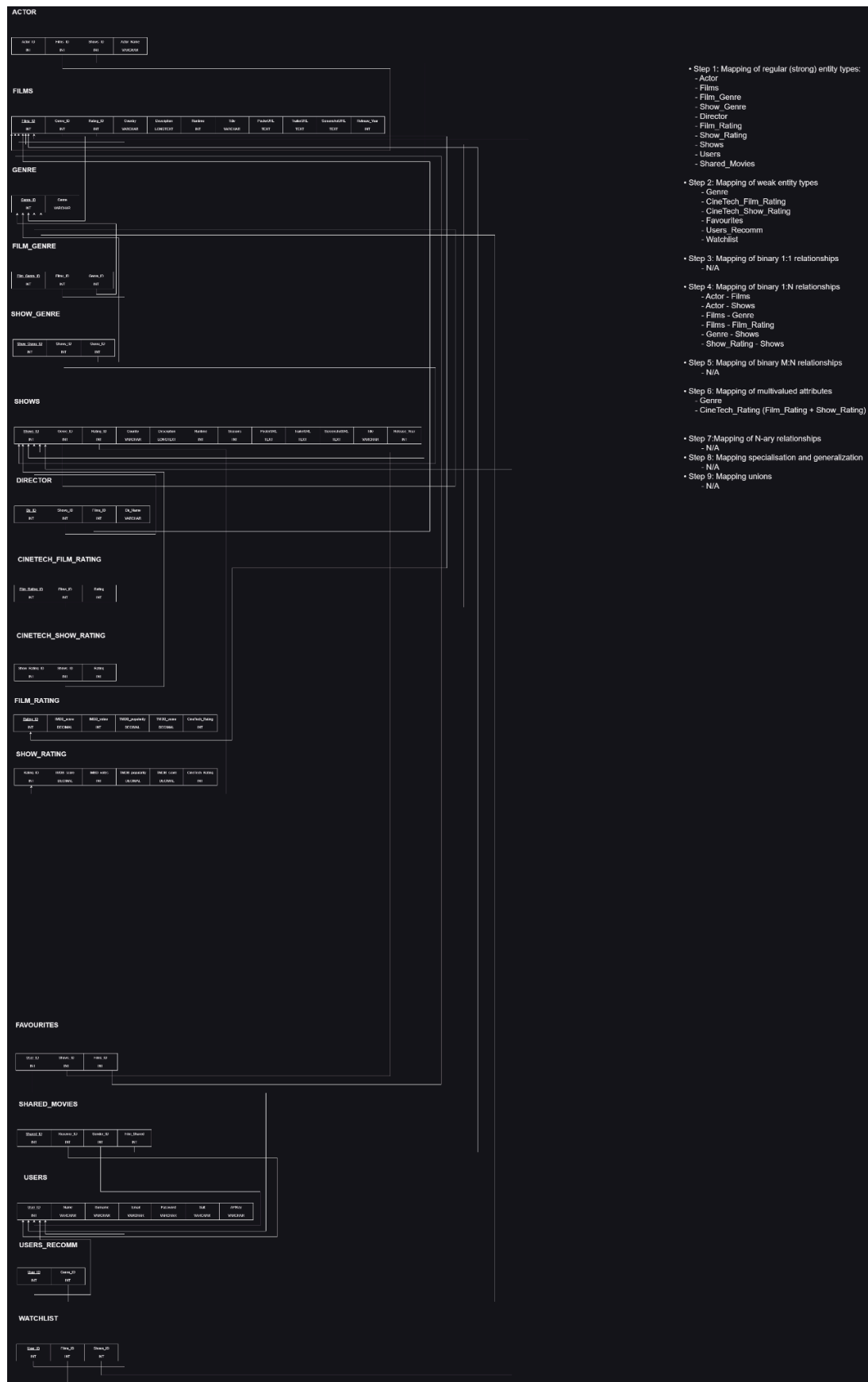
## TASK 3 : (E)ER-DIAGRAM TO RELATIONAL MAPPING

### • First Draft :

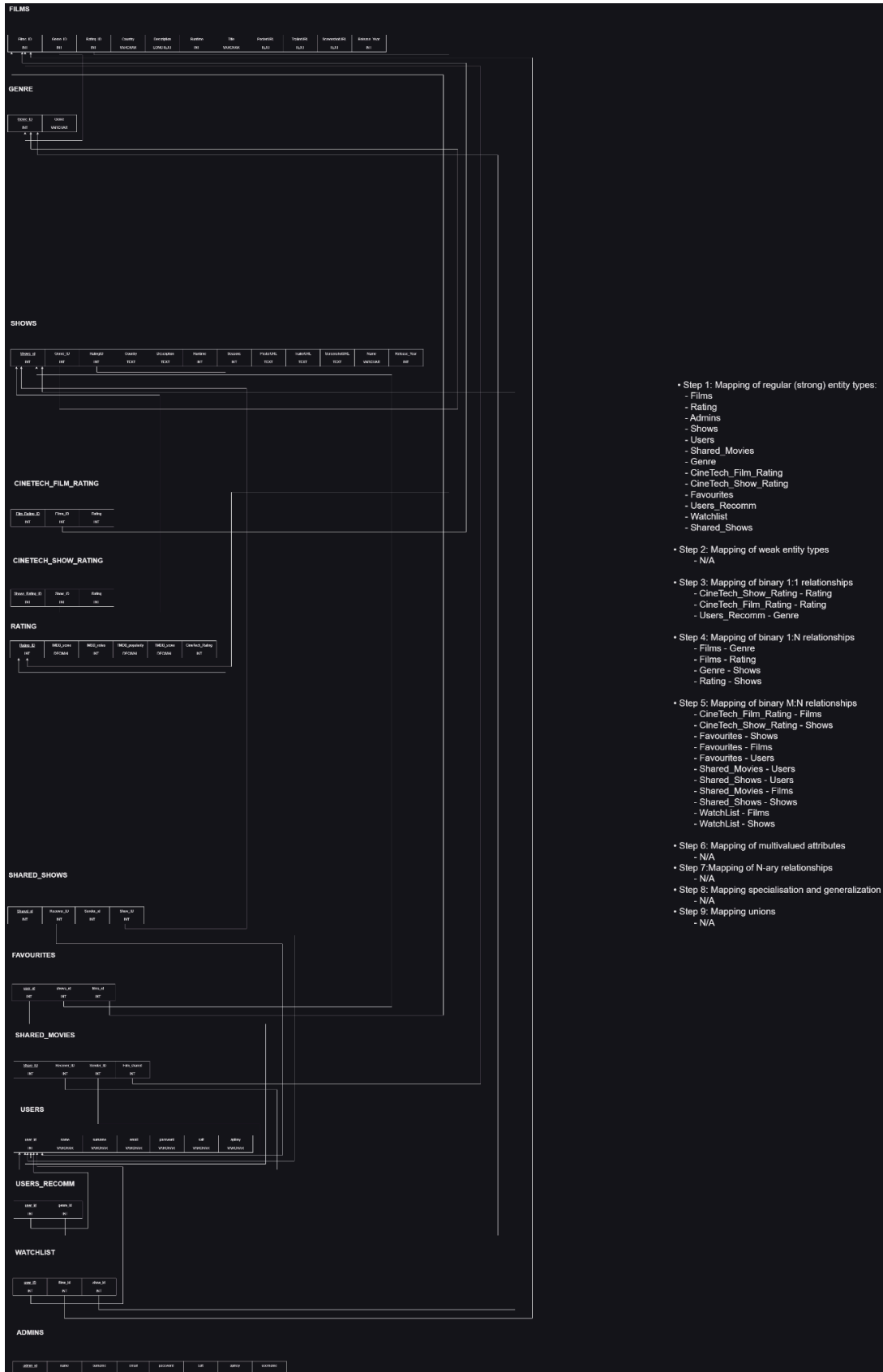




## ● Second Draft :



## ● Final Draft :



## TASK 4 : RELATIONAL SCHEMA

TABLE STRUCTURE										RELATION VIEW									
#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action										
<input type="checkbox"/>	1	Films_ID	int(11)		No	None		AUTO_INCREMENT	Change  Drop  More										
<input type="checkbox"/>	2	Title	varchar(255)	utf8mb4_general_ci	Yes	NULL			Change  Drop  More										
<input type="checkbox"/>	3	Genre_ID	int(11)		Yes	NULL			Change  Drop  More										
<input type="checkbox"/>	4	Rating_ID	int(11)		Yes	NULL			Change  Drop  More										
<input type="checkbox"/>	5	Country	varchar(255)	utf8mb4_general_ci	Yes	NULL			Change  Drop  More										
<input type="checkbox"/>	6	Description	longtext	utf8mb4_general_ci	Yes	NULL			Change  Drop  More										
<input type="checkbox"/>	7	PosterURL	text	utf8mb4_general_ci	Yes	NULL			Change  Drop  More										
<input type="checkbox"/>	8	TrailerURL	text	utf8mb4_general_ci	Yes	NULL			Change  Drop  More										
<input type="checkbox"/>	9	ScreenshotURL	text	utf8mb4_general_ci	Yes	NULL			Change  Drop  More										
<input type="checkbox"/>	10	Runtime	int(11)		Yes	NULL			Change  Drop  More										
<input type="checkbox"/>	11	Release_Year	int(11)		Yes	NULL			Change  Drop  More										

The above table is the films table with the following information:

- Primary & Foreign Keys : The Films\_ID is the primary key of the table to uniquely identify each film in the table. The foreign keys in this table are Genre\_ID which links to the genre table which consists of all existing genres, moreover the second foreign key is Rating\_ID which links to the Rating table.
- Constraints and Checks :
- Data Types and Length Constraints :

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	Film_genre_id	int(11)		No	None		AUTO_INCREMENT	Change  Drop  More
<input type="checkbox"/>	2	Film_id	int(11)		No	None			Change  Drop  More
<input type="checkbox"/>	3	Genre_id	int(11)		No	None			Change  Drop  More

The above table is the Film\_genre table with the following information:

- **Primary & Foreign Keys :** The Films\_genre\_id is the primary key of the table to uniquely identify each film linked to each genre in the table. The foreign keys in this table are Film\_ID which links to the Films table which consists of all existing genres, moreover the second foreign key is Genre\_ID which links to the Genre table. This is done so that we can facilitate a single Film with multiple genres
- **Constraints and Checks :** All the data types cannot be null
- **Data Types and Length Constraints :** Each foreign key has a data type of int with length 11 which is the same as the keys on their relevant table to make sure there is consistency across all tables.

Server: localhost:3306 > Database: u23535246_CineTechDB > Table: Genre									
#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	Genre_ID	int(11)		No	None		AUTO_INCREMENT	Change  Drop  More
<input type="checkbox"/>	2	Genre	varchar(255) utf8mb4_general_ci		No	None			Change  Drop  More

The above table is the genre table with the following information:

- **Primary & Foreign Keys :** The Genre\_id is the primary key of the table to uniquely identify each genre in the table. There are no foreign keys in this table.
- **Constraints and Checks :** All the data types cannot be null and no default values
- **Data Types and Length Constraints :** Primary key Genre\_id has a data type of int with length 11 and Genre is varchar 255.

And the full database with the respective tables we used:

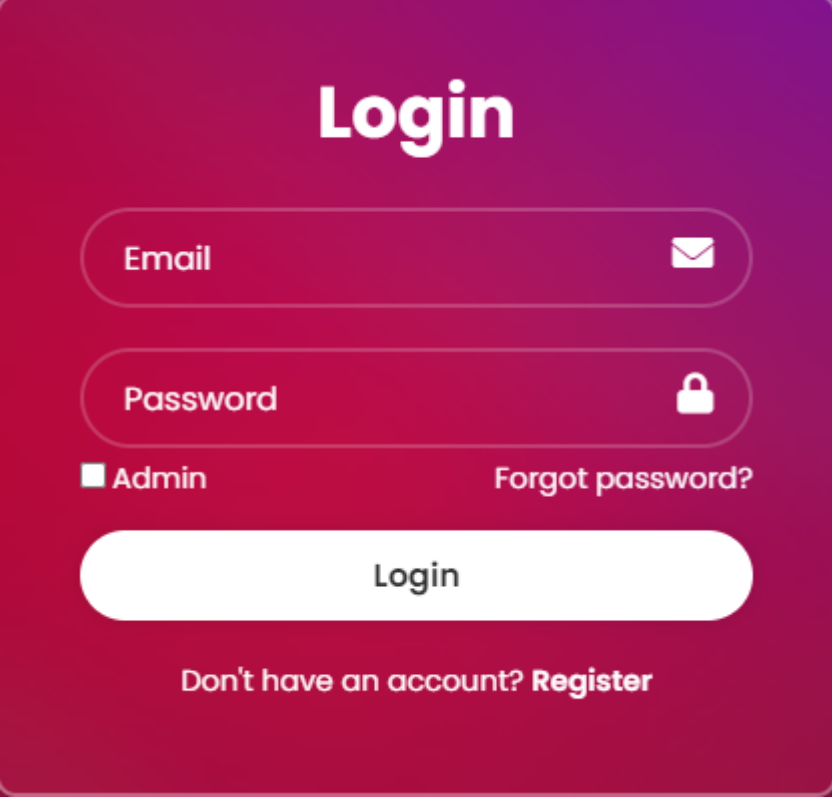
Table	Action	Rows	Type	Collation	Size	Overhead
Actor	Browse Structure Search Insert Empty Drop	500	InnoDB	utf8mb4_general_ci	88.0 K B	-
Admin	Browse Structure Search Insert Empty Drop	5	InnoDB	utf8mb4_general_ci	16.0 K B	-
CineTech_Film_Rating	Browse Structure Search Insert Empty Drop	2	InnoDB	utf8mb4_general_ci	16.0 K B	-
CineTech_Show_Rating	Browse Structure Search Insert Empty Drop	2	InnoDB	utf8mb4_general_ci	16.0 K B	-
Director	Browse Structure Search Insert Empty Drop	500	InnoDB	utf8mb4_general_ci	88.0 K B	-
Favouites	Browse Structure Search Insert Empty Drop	31	InnoDB	utf8mb4_general_ci	64.0 K B	-
Films	Browse Structure Search Insert Empty Drop	2,993	InnoDB	utf8mb4_general_ci	240.0 K B	-
Film_genre	Browse Structure Search Insert Empty Drop	101	InnoDB	utf8mb4_general_ci	16.0 K B	-
Genre	Browse Structure Search Insert Empty Drop	33	InnoDB	utf8mb4_general_ci	16.0 K B	-
Rating	Browse Structure Search Insert Empty Drop	507	InnoDB	utf8mb4_general_ci	48.0 K B	-
Shared_movies	Browse Structure Search Insert Empty Drop	6	InnoDB	utf8mb4_general_ci	64.0 K B	-
Shared_shows	Browse Structure Search Insert Empty Drop	2	InnoDB	utf8mb4_general_ci	16.0 K B	-
Shows	Browse Structure Search Insert Empty Drop	57	InnoDB	utf8mb4_general_ci	48.0 K B	-
show_genre	Browse Structure Search Insert Empty Drop	30	InnoDB	utf8mb4_general_ci	16.0 K B	-
users	Browse Structure Search Insert Empty Drop	10	InnoDB	utf8mb4_general_ci	16.0 K B	-
users_recomm	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	32.0 K B	-
watchList	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	64.0 K B	-
17 tables	Sum	4,779	InnoDB	utf8mb4_general_ci	848.0 K B	0 B




## **TASK 5 : WEB-BASED APPLICATION**


### **Login Page**

Added a feature to login as either a normal user or as an admin.

A login page with a dark blue gradient background. At the top, the word "Login" is written in large, bold, white sans-serif font. Below it are two rounded rectangular input fields. The first field is labeled "Email" in white text and has a white envelope icon on the right. The second field is labeled "Password" in white text and has a white padlock icon on the right. Below the password field, there is a checkbox labeled "Admin" in white text, followed by the text "Forgot password?" in white. Below these elements is a large, rounded white button with the word "Login" in dark blue text. At the bottom, there is a line of white text that reads "Don't have an account? Register".

**Login**

Email 

Password 

☐ Admin [Forgot password?](#)

**Login**

Don't have an account? [Register](#)

### **Add, edit, and delete Movies and TV series**

This gets directly done on the database itself.

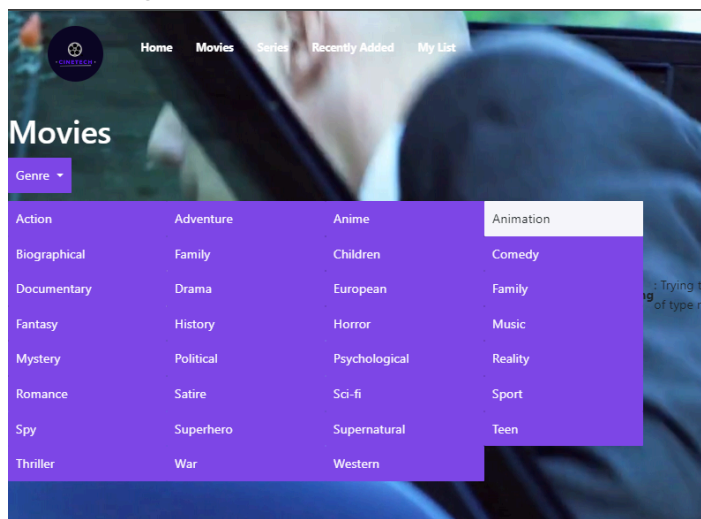
The screenshot shows the CineTech application dashboard with a dark blue header and a light blue sidebar. The main content area contains five functional boxes:

- Delete Movie or Series:** A box with a title input field and a "Delete" button. Description: "Remove Movies/Series no longer needed in the database."
- Add Movie:** A box with a title input field, Genre ID, Country, Rating, Description, Runtime, Year, Post URL, Video URL, and Screen URL input fields, and an "Add" button. Description: "This box will add a movie or series to the database."
- Add Series:** A box with a title input field, Genre ID, Country, Rating, Description, Runtime, Year, Seasons, Post URL, Video URL, and Screen URL input fields, and an "Add" button. Description: "This box will add a movie or series to the database."
- Edit Movie or Series:** A box with a title input field, a "Movie/Series" dropdown, an "Edit Here" dropdown, and an "Edit" button. Description: "This box will edit a movie or series from the database."
- Delete User from Database:** A box with an "Email" input field and a "Delete" button. Description: "This box will remove a user from the database."

## Sort and filter content based on various criteria

Added a sort by genre function on the movies and series pages. The movies and series pages were a way of filtering the data by type, either movie or series.

And a page that filters only the recently added movies by their release year.



## Update the database and/or extend, delete

Update to the database is done as an edit to either a movie or series, this is done, with no fuzzy search and section by section.



# TASK 6 : DATA

## Screenshot of the films (movies) table :

Film_ID	Title	Genre_ID	Rating_ID	Country	Description	PosterURL	TrailerURL	ScreenshotURL	Runtime	Release_Year
1	Rocky	28	1	US	When world heavyweight boxing champion, Apollo Cr...	https://image.tmdb.org/t/p/original/gg/vxxv8BtR5L...	u	u	119	1980
2	Grease	21	2	US	Australian good girl Sandy and greaser Danny fell ...	https://www.themoviedb.org/t/p/w1280/6GgZv00hwG...			110	1981
3	The Sting	2	3	US	A novice con man teams up with an acknowledged mac...	https://www.themoviedb.org/t/p/w1280/dm7hg37p8r...			129	1982
4	Rocky II	24	4	US	After Rocky goes the distance with champ Apollo Co...	https://www.themoviedb.org/t/p/w1280/M4uUcQTT7L...			119	1983
5	Monty Python and the Holy Grail	27	5	GB	King Arthur, accompanied by his squire, recruits h...	https://www.themoviedb.org/t/p/w1280/2DnG3u3j8L...			91	1984
6	Animal House	16	6	US	At a 1962 College, Dear Vernon Wormer is determi...	https://www.themoviedb.org/t/p/w1280/1Yf2tmdg9f9...			108	1985
7	Life of Brian	33	7	GB	Brian Cohen is an average young Jewish man, but th...	https://media.themoviedb.org/t/p/w94_and_7141_bes...			94	1986
8	White Christmas	3	8	US	Two talented song-and-dance men team up after the ...	https://media.themoviedb.org/t/p/w300_and_4450_bes...			120	1987
9	Heroses	26	9	US	A Vietnam veteran suffering from post traumatic st...	https://media.themoviedb.org/t/p/w300_and_4450_bes...			112	1988
10	Play Misty for Me	7	10	US	A brief fling between a male disc jockey and an ob...	https://media.themoviedb.org/t/p/w300_and_4450_bes...			102	1989
11	Caro Station	13	11	EG	Qinwei, a physically challenged peddler who makes ...	https://media.themoviedb.org/t/p/w300_and_4450_bes...			77	1990
12	Richard Pryor Live in Concert	29	12	US	Richard Pryor delivers monologues on race, sex, fa...	https://media.themoviedb.org/t/p/w300_and_4450_bes...			78	1991
13	Bande	3	13	IN	Mahang Bragham lives a wealthy lifestyle in Bihar...	https://media.themoviedb.org/t/p/w300_and_4450_bes...			141	1992
14	Prince	27	14	IN	To better himself, a spoiled prince temporarily st...	https://media.themoviedb.org/t/p/w300_and_4450_bes...			152	1993
15	FTA	31	15	US	A documentary about a political troupe headed by a...	https://media.themoviedb.org/t/p/w300_and_4450_bes...			97	1994
16	Robert's Career	26	16	DE	A keen chronicle of the unlikely rise to power of a...	https://media.themoviedb.org/t/p/w300_and_4450_bes...			150	1995
17	Amrapati	20	17	IN	After a failed conquest, Emperor Akashbhatu pret...	https://media.themoviedb.org/t/p/w300_and_4450_bes...			120	1996
18	Alibaba Aur 40 Chor	10	18	IN	The movie is based on the story of Ali Baba and th...	https://media.themoviedb.org/t/p/w300_and_4450_bes...			138	1997
19	Manoranjan	14	19	IN	Harvard Ratan is a rookie at the local police sta...	https://media.themoviedb.org/t/p/w300_and_4450_bes...			162	1998
20	Professor	2	20	IN	Sita devi is a very strict aunt for a number of ye...	https://media.themoviedb.org/t/p/w300_and_4450_bes...			163	1999
21	Knohn Knohn	30	21	IN	As a crazed killer biases a trail of blood through...	https://media.themoviedb.org/t/p/w300_and_4450_bes...			131	2000
22	Sadain the Victorious	21	22	EG	The first Sultan of Egypt and Syria leads the Musl...	https://media.themoviedb.org/t/p/w300_and_4450_bes...			186	2001
23	The Other Side of the Mountain	21	23	US	One year before the Olympics, Jill Kinmont, an 18...	https://media.themoviedb.org/t/p/w300_and_4450_bes...			103	2002
24	Dark Waters	11	24	EG	Ragab, a poor sailor, returns home to Alexandria a...	https://media.themoviedb.org/t/p/w300_and_4450_bes...			120	2003
25	The Return of the Prodigal Son	20	25	EG	In this Andre Gide adaptation, an activist is rel...	https://media.themoviedb.org/t/p/w300_and_4450_bes...			124	2005

## Screenshot of the shows table :

Show_ID	Name	Seasons	Rating_ID	Country	Description	Runtime	Release_Year	Genre_ID	ScreenshotURL	TrailerURL	PosterURL
1	Five Came Back: The Reference Films	1	2	US	This collection includes 12 World War II-era propa...	120	2017	28			https://www.themoviedb.org/t/p/w1280/2XO0KGGK9q7...
2	Monty Python's Flying Circus	4	1	UK	A British sketch comedy series with the shows bet...	30	1969	7			https://www.themoviedb.org/t/p/w1280/h2MMLPq9FA...
3	Monty Python's Fliegender Zirkus	2	14	DE	Monty Python's Fliegender Zirkus consisted of two ...	25	1972	12			https://www.themoviedb.org/t/p/w600_and_900_bes...
4	Seinfeld	9	30	US	A stand-up comedian and his three offbeat friends ...	22	1989	28			https://www.themoviedb.org/t/p/w600_and_900_bes...
5	Thomas & Friends	25	11	UK	Thomas & Friends is a British children's televis...	11	1984	24			https://www.themoviedb.org/t/p/w600_and_900_bes...
6	Freeman Sam	11	24	UK	Follow the adventures of Freeman Sam and his colle...	10	1987	26			https://www.themoviedb.org/t/p/w600_and_900_bes...
7	Danger Mouse	11	10	UK	Danger Mouse, the world's greatest secret agent, a...	25	1981	26			https://www.themoviedb.org/t/p/w600_and_900_bes...
8	Survivor	40	17	US	A reality show contest where sixteen or more casta...	42	2000	18			https://www.themoviedb.org/t/p/w600_and_900_bes...
9	One Place	20	10	JP	Years ago, the fearsome Pirate King, Gel D. Roger ...	24	1999	26			https://www.themoviedb.org/t/p/w600_and_900_bes...
10	Poldark	24	30	JP	Jon Ash accompanied by his partner Phachu, as he...	22	1997	7			https://www.themoviedb.org/t/p/w600_and_900_bes...
11	Gimme Girls	7	33	US	Set in the charming town of Stars Hollow, Connect...	45	2000	14			https://media.themoviedb.org/t/p/w300_and_7450_bes...
12	Cowboy Bebop	1	31	JP	In 2071, roughly fifty years after an accident wit...	24	1998	29			https://media.themoviedb.org/t/p/w300_and_7450_bes...
13	Neon Genesis Evangelion	1	28	JP	At the turn of the century, the Angels returned to...	23	1995	9			https://media.themoviedb.org/t/p/w300_and_7450_bes...
14	The Real World	33	29	US	Each year, seven strangers in their twenties, from...	60	1992	3			https://media.themoviedb.org/t/p/w300_and_7450_bes...
15	Power Rangers	28	30	US	A team of teenagers with attitude are recruited to...	22	1993	5			https://www.themoviedb.org/t/p/w600_and_900_bes...
16	Big Brother	23	28	NL	American version of the reality game show which fo...	45	1998	26			https://media.themoviedb.org/t/p/w300_and_7450_bes...
17	The Magic School Bus	4	5	CA	An eccentric schoolteacher takes her class on world...	22	1994	27			https://media.themoviedb.org/t/p/w300_and_7450_bes...
18	Inuyasha	8	29	JP	A teenage girl periodically travels back in time t...	24	2000	12			https://www.themoviedb.org/t/p/w600_and_900_bes...
19	Yu-Gi-Oh!	5	2	JP	A timid young boy who loves all sorts of games, on...	23	1998	29			https://www.themoviedb.org/t/p/w600_and_900_bes...
20	Adventures of Sonic the Hedgehog	2	12	US	Follow the adventures of Sonic the Hedgehog, and h...	22	1993	1			https://media.themoviedb.org/t/p/w300_and_7450_bes...
21	Kenan & Kel	4	24	US	Set in Chicago, the show follows the kid-friendly ...	22	1996	21			https://media.themoviedb.org/t/p/w300_and_7450_bes...
22	Grimm	8	10	US	The series revolves around the friendship of four ...	22	2000	10			https://www.themoviedb.org/t/p/w600_and_900_bes...
23	H	6	7	JP	H is a French sitcom with seventy-one, 22-minute e...	24	1995	16			https://www.themoviedb.org/t/p/w600_and_900_bes...
24	Ali That	11	15	US	The new kids of comedy bring the funny as Ali That...	23	1994	29			https://media.themoviedb.org/t/p/w300_and_7450_bes...
25	Heartbreak High	7	30	AU	The ins and outs of the classroom lives of a group...	22	1994	3			https://media.themoviedb.org/t/p/w300_and_7450_bes...

We used to populate each table with the CSV file from [kaggle](#) and used mySQL workbench to populate the dataset into our database using a script. An additional database cleanup where by we added we split the initial table which has all the movies in one table into sub tables for optimising SQL queries.

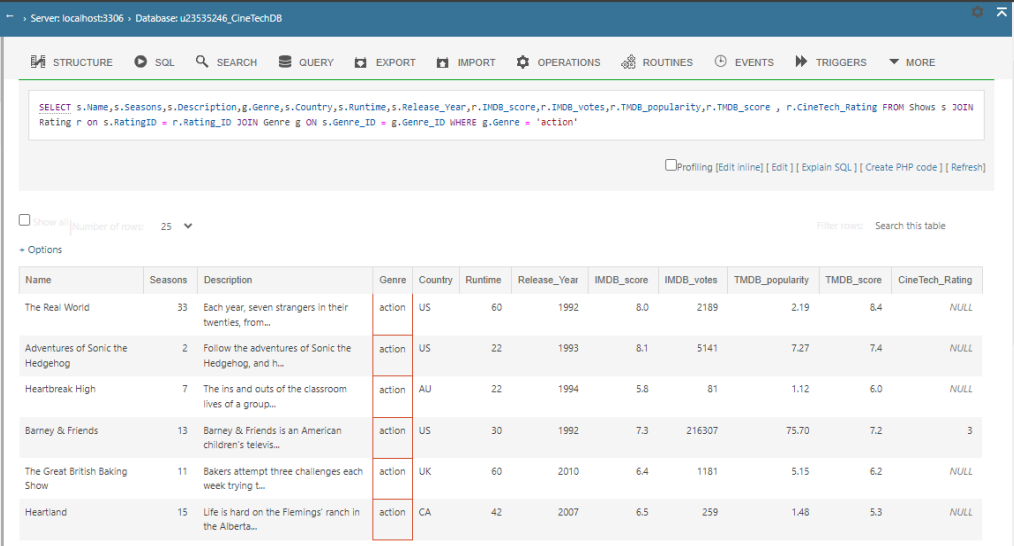
## TASK 7 : ANALYSE

We made use of phpMyAdmin as our chosen phpMyAdmin.

Execution Plan Output:

Chose query: SELECT

```
s.Name,s.Seasons,s.Description,g.Genre,s.Country,s.Runtime,s.Release_Year,r.IMDB_score,r.IMDB_votes,r.TMDB_popularity,r.TMDB_score , r.CineTech_Rating FROM Shows s JOIN Rating r on s.RatingID = r.Rating_ID JOIN Genre g ON s.Genre_ID = g.Genre_ID WHERE g.Genre = 'action'
```



The screenshot shows the phpMyAdmin interface with the following SQL query entered in the query box:

```
SELECT s.Name,s.Seasons,s.Description,g.Genre,s.Country,s.Runtime,s.Release_Year,r.IMDB_score,r.IMDB_votes,r.TMDB_popularity,r.TMDB_score , r.CineTech_Rating FROM Shows s JOIN Rating r on s.RatingID = r.Rating_ID JOIN Genre g ON s.Genre_ID = g.Genre_ID WHERE g.Genre = 'action'
```

Below the query box, the results are displayed in a table with 12 columns: Name, Seasons, Description, Genre, Country, Runtime, Release\_Year, IMDB\_score, IMDB\_votes, TMDB\_popularity, TMDB\_score, and CineTech\_Rating. The table contains 6 rows of data, all of which are action movies.

Name	Seasons	Description	Genre	Country	Runtime	Release_Year	IMDB_score	IMDB_votes	TMDB_popularity	TMDB_score	CineTech_Rating
The Real World	33	Each year, seven strangers in their twenties, from...	action	US	60	1992	8.0	2189	2.19	8.4	NULL
Adventures of Sonic the Hedgehog	2	Follow the adventures of Sonic the Hedgehog, and h...	action	US	22	1993	8.1	5141	7.27	7.4	NULL
Heartbreak High	7	The ins and outs of the classroom lives of a group...	action	AU	22	1994	5.8	81	1.12	6.0	NULL
Barney & Friends	13	Barney & Friends is an American children's televis...	action	US	30	1992	7.3	216307	75.70	7.2	3
The Great British Baking Show	11	Bakers attempt three challenges each week trying t...	action	UK	60	2010	6.4	1181	5.15	6.2	NULL
Heartland	15	Life is hard on the Flemings' ranch in the Alberta...	action	CA	42	2007	6.5	259	1.48	5.3	NULL

Report on the Initial Performance:

### Key Observations

#### 1. Type: ALL for Table s (Shows):

- This indicates a full table scan, which is inefficient especially if the table is large.

#### 2. Type: eq\_ref for Tables g (Genre) and r (Rating):

- These are using the primary key to fetch the rows, which is efficient.

### Optimization Strategies

#### 1. Indexing:

- Add indexes on columns that are used in joins and the **WHERE** clause to avoid full table scans and improve performance.

■ Index on **s.Genre\_ID**.

- Index on **s.RatingID**.
- Index on **g.Genre**.

## Implementing Indexes

Here are the SQL commands to add the necessary indexes:

```
CREATE INDEX idx_shows_genre_id ON Shows(Genre_ID); CREATE INDEX idx_shows_rating_id ON Shows(RatingID); CREATE INDEX idx_genre_genre ON Genre(Genre);
```

## Re-run EXPLAIN Command

After adding indexes, re-run the **EXPLAIN** command to see the new execution plan:

```
EXPLAIN SELECT s.Name, s.Seasons, s.Description, g.Genre, s.Country, s.Runtime, s.Release_Year, r.IMDB_score, r.IMDB_votes, r.TMDB_popularity, r.TMDB_score, r.CineTech_Rating FROM Shows s JOIN Rating r ON s.RatingID = r.Rating_ID JOIN Genre g ON s.Genre_ID = g.Genre_ID WHERE g.Genre = 'action';
```

## Expected Results

With the new indexes, the execution plan should ideally show **ref** or **index** types instead of **ALL** for the **Shows** table, indicating that indexes are being used.

## Optimised Execution Plan:

- **Optimised Query:**

```
CREATE INDEX idx_shows_genre_id ON Shows(Genre_ID); CREATE INDEX idx_shows_rating_id ON Shows(RatingID); CREATE INDEX idx_genre_genre ON Genre(Genre); EXPLAIN SELECT s.Name, s.Seasons, s.Description, g.Genre, s.Country, s.Runtime, s.Release_Year, r.IMDB_score, r.IMDB_votes, r.TMDB_popularity, r.TMDB_score, r.CineTech_Rating FROM Shows s JOIN Rating r ON s.RatingID = r.Rating_ID JOIN Genre g ON s.Genre_ID = g.Genre_ID WHERE g.Genre = 'action';
```

- **Expected Execution Plan:**

## Explanation of Performance Gain:

- **Improved Index Usage:**
  - The addition of indexes allows MySQL to use **ref** types instead of **ALL**, significantly reducing the number of rows scanned and improving query performance.

- **Reduced I/O Operations:**

- Indexes help in reducing the number of disk I/O operations, leading to faster query execution times.

Server: localhost:3306 • Database: u23535246\_CineTechDB

STRUCTURE SQL SEARCH QUERY EXPORT IMPORT OPERATIONS ROUTINES EVENTS TRIGGERS MORE

Your SQL query has been executed successfully.

```
EXPLAIN SELECT s.Name,s.Seasons,s.Description,g.Genre,s.Country,s.Runtime,s.Release_Year,r.IMDB_score,r.IMDB_votes,r.TMDB_popularity,r.TMDB_score , r.CineTech_Rating FROM Shows s JOIN Rating r on s.RatingID = r.Rating_ID JOIN Genre g ON s.Genre_ID = g.Genre_ID WHERE g.Genre = 'action'
```

[Edit inline] [Edit] [Skip Explain SQL] [Analyze Explain at mariadb.org] [Create PHP code]

Options

id	select_type	table	type	possible_keys	key	key_len	ref	rows	Extra
1	SIMPLE	s	ALL	NULL	NULL	NULL	NULL	57	Using where
1	SIMPLE	g	eq_ref	PRIMARY	PRIMARY	4	u23535246_CineTechDB.s.Genre_ID	1	Using where
1	SIMPLE	r	eq_ref	PRIMARY	PRIMARY	4	u23535246_CineTechDB.s.RatingID	1	

Query results operations

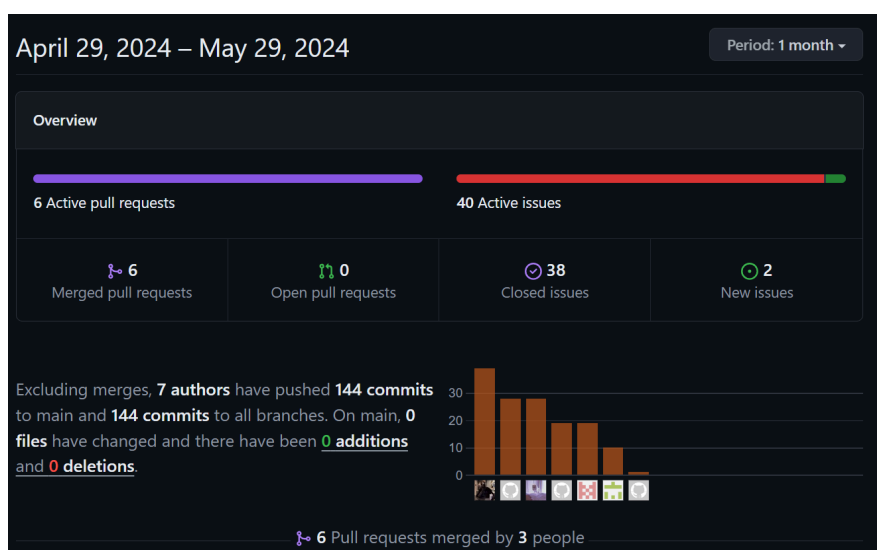
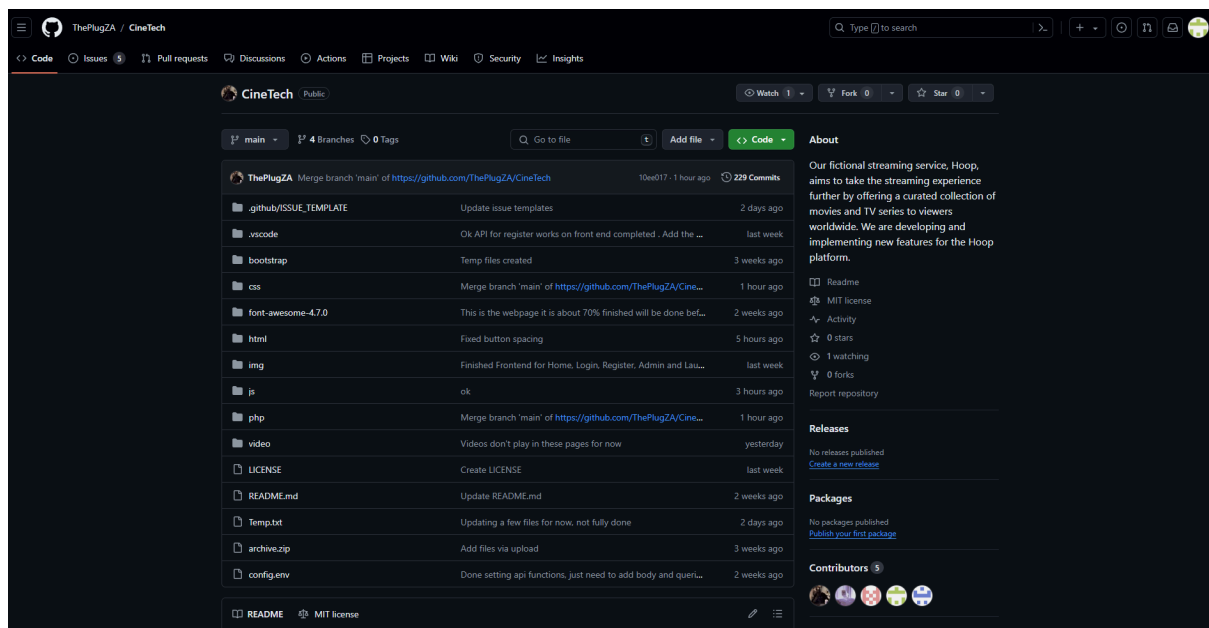
Print Copy to clipboard Create view

Bookmark this SQL query

# TASK 8 : DEVELOPMENT

## Usage of Git

In our project we used git (below) to effectively update the files and make sure each team member has recent developments of what each individual has contributed to the project. With every commit, each team member provided comments consisting of brief explanations to allow for the next person to understand and build upon the previous implementation. Repository: <https://github.com/ThePlugZA/CineTech>





## Data Validation Techniques

```
public function registerUser($name, $surname, $email, $password, $username, $admin) {
    if (empty($name) || empty($surname) || empty($email) || empty($password)) {
        return json_encode(array("message" => "All fields are required"));
    }

    // Validate email format
    if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
        return $this->errorResponse(time(), "Invalid email format");
    }

    // Validate password strength (e.g., minimum length, contain uppercase, lowercase, digit, symbol)
    if (strlen($password) < 8) {
        return json_encode(array('message' => 'Password must have at least 8 characters'));
    } else if (!preg_match('/[A-Z]/', $password)) {
        return json_encode(array('message' => 'Password should include at least one uppercase letter'));
    } else if (!preg_match('/[a-z]/', $password)) {
        return json_encode(array('message' => 'Password should include at least one lowercase letter'));
    } else if (!preg_match('/[0-9]/', $password)) {
        return json_encode(array('message' => 'Password should include at least one number'));
    }

    // Check if user already exists
    if ($admin === "true") {
        $stmt = $GLOBALS['connection']->prepare("SELECT admin_id FROM Admins WHERE email = ?");
    } else {
        $stmt = $GLOBALS['connection']->prepare("SELECT user_id FROM users WHERE email = ?");
    }

    $stmt->bind_param("s", $email);
    $stmt->execute();
    $result = $stmt->get_result();
    if ($result->num_rows > 0) {
        return $this->errorResponse(time(), "Person already exists");
    }
    // User is registered
}
```

The above snippet code illustrates the use of regular expressions such as the 'preg\_match' function to test the user input, if it is valid before inserting it into the database, we included error messages to alert the user in the event that their input does not meet the criteria and prompt them to enter the correct information.

## Package Manager

For our website we decided that it would be best if we do not use the package manager. These are the reasons we had for not using it:

- **Minimal Dependencies:** Since our project is a small project with minimal external dependencies, managing these dependencies manually was straightforward and did not warrant the overhead of a package manager.
- **Quick Prototyping:** For quick prototyping or proof-of-concept projects, manually adding a few libraries was faster than setting up a package manager in our case.
- **Security Concerns:** Directly managing dependencies provided tighter control over which libraries and versions we could use, potentially reducing the risk of introducing vulnerabilities from third-party packages.