Anime Recommendation System Database

by Anna Ryplewski and August Le

Step #1 Feedback:

Ronald Etie

- [X] remove unnecessary descriptions for obvious attributes (like title)
- [X] fix naming consistencies w/ AnimeSeries and AnimeShows
- [X] fix capitalization of variables in ER diagram
- [X] fix ER diagram structure (eliminate overlapping relationships)

Conrad Schreiner

- [X] update overview with more relevant figures/data
- [X] clarify the relationship between entities (user)
- [X] fix ER diagram capitalization

Leo Zimmer

- [X] condense animeShows and animeMovies
- [X] add attributes to ER diagram
- [X] rename episodes to numEpisodes

Joshua Knowles

- [X] combine animeShows and animeMovies
- [X] clarify which variables can be NULL/not NULL
- [X] fix optional relationship between animeStudio and animeMovie
- [X] fix ER diagram capitalization

Catherine Delachica

- [X] combine animeDisliked and animeLiked into an animePreference variable

  Note: Having two tables, animeDisliked and animeLiked could be clunky. Instead of having these two junction tables we redesigned our approach and instead created a single table animePreferences where the preference attribute was an integer. Negative

integer values mean with dislike where positive values mean the user liked said anime, with the magnitude of integer values conveying more like or dislike.

Devin Daniels

[X] remove crossed relationship lines in the ER diagram

Additional Changes:

- Changed **animeTitles** table name to **anime**
- Removed *popularity* attribute from **anime** table to reduce complexity
- Removed **animeStudio** and **studioWorks** to reduce complexity
- Change *userGender* in **user** to varChar to allow for string input
- Added **user_streamingServices** to act as a junction table between **streamingService** and **user**
- Combine **animeLikes** and **animeDislikes** into **animePreferences** as a junction table
- Remove *serviceID* from **user** table as part of 3NF normalization
- Make *userID* + *animeID* the primary key of **animePreferences** to ensure unique combinations
- Make *serviceID* + *userID* the primary key of **user_streamingServices** to ensure unique combinations
- Change *startDate* and *endDate* from varchar to Date
- Added constraint to *userAge* in **user** to be greater than 0

Updated Project Overview:

The number of shows and movies created by Japanese animation studios has been increasing ever since the medium gained popularity in the 1960s. Currently, as many as 300 animated television shows are aired or distributed on entertainment platforms in a given year. In 2023, the revenue generated by anime on streaming services was $5.5 billion globally. Hundreds of animation studios fuel a thriving industry of animated entertainment, which is eagerly consumed by a global audience (Hamilton).

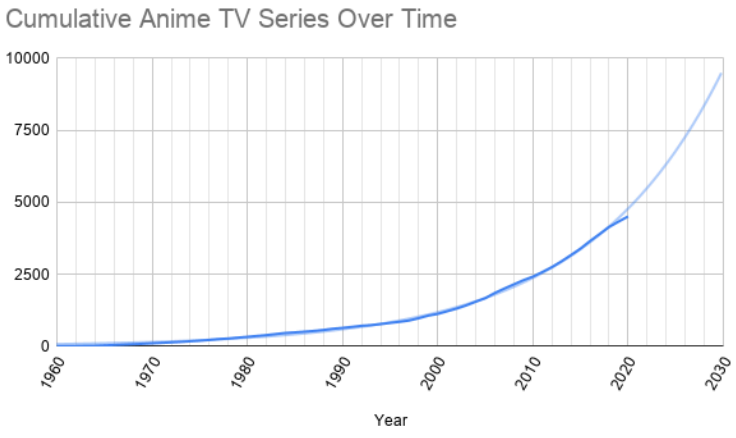
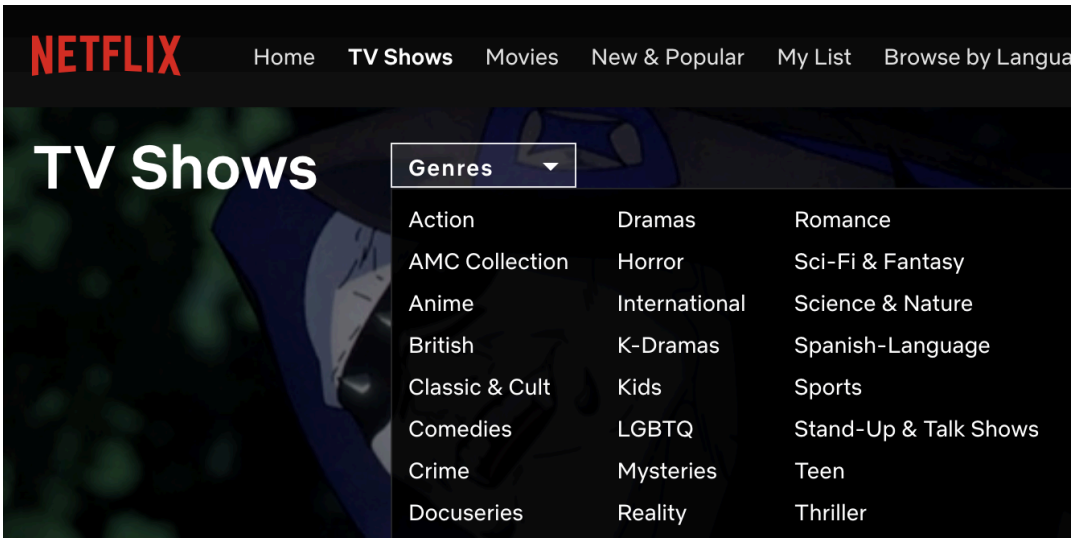
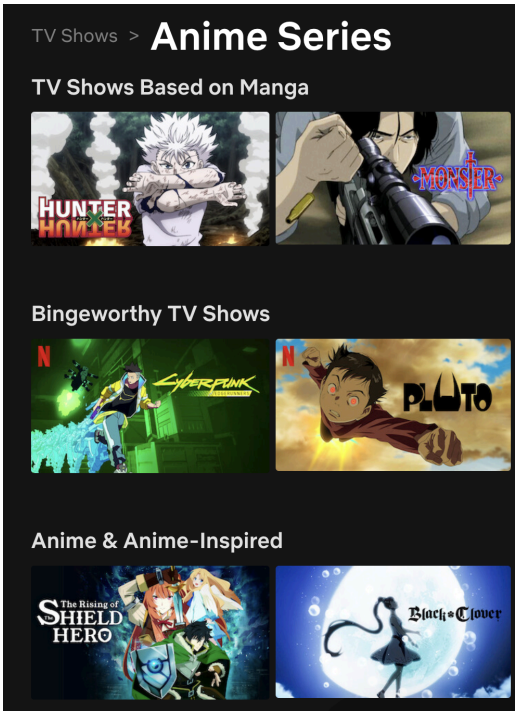


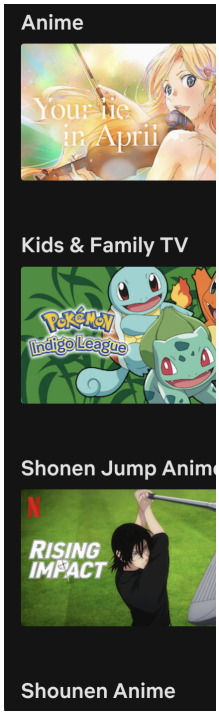
Chart of Number of Anime Per Year Over Time, TJSomething, Mar 1 2021, https://www.reddit.com/r/anime/comments/lvvexe/chart_of_number_of_anime_per_year_over_time/

And just as there's no shortage of viewership, there's no shortage of titles to peruse from. Thousands of anime TV shows and movies are available on multiple websites and apps such as Crunchyroll, Netflix and Hulu. However, western streaming platforms often fall short in their categorization of anime as a genre itself rather than a medium, as shown below.



The generalization can be overlooked, but headers inside the genre itself don't shed any more clarity on the shows' content. Upon selection of the anime genre, users are greeted with headers like "TV Shows Based on Manga", "Anime & Anime-Inspired", and "Anime".





…Isn't this already the anime genre? The lack of proper organization is confusing for users to navigate and makes it difficult to find a new show based on previous likes. It also fails to separate shows into sub-genres or even maturity ratings.

To fix this, we are developing a database that catalogs anime shows and movies alongside a robust system of relevant attributes such as genres, maturity ratings, and trigger warnings.

This would streamline the user's browsing experience and make it much easier to find relevant content based on the anime's content instead of the arbitrary and redundant terms used at present.

Updated Database Outline:

anime:

- ***ID:*** int, auto_increment, unique, not NULL, PK
- ***title****:* varchar, not NULL
- ***startDate****:* varchar, not NULL
  start season/month and year that a show aired
- ***endDate****:* varchar
  can be NULL if ongoing/movie
- ***serviceID****:* int, FK, references animeStudio.serviceID
  the streaming service(s) a title can be found on
- ***genre****:* varchar, not NULL
  contains 2-3 (non-mutually exclusive) categories
- ***maturityRating****:* varchar, not NULL
  (mutually exclusive) rating for the series
- ***triggerWarnings****:* varchar
  contains disclaimers for sensitive content
- ***numEpisodes****:* int
  can be NULL if it's a movie/OVA

streamingService:

- ***serviceID***: int, auto_increment, unique, not NULL, PK
- ***name***: varchar, not NULL
  the name of the streaming service

user:

- ***userID***: int, auto_increment, unique, not NULL, PK
- ***username:*** varchar, not NULL
- ***password:*** varchar, not NULL
- ***userAge***: int, not NULL
- ***userGender***: varchar
- ***userLocation***: varchar

animePreference:

- ***userID****:* int, PK, FK
- ***animeID****:* int, PK, FK
- ***preference****:* int
  numerical rating of viewed shows

user_streamingServices

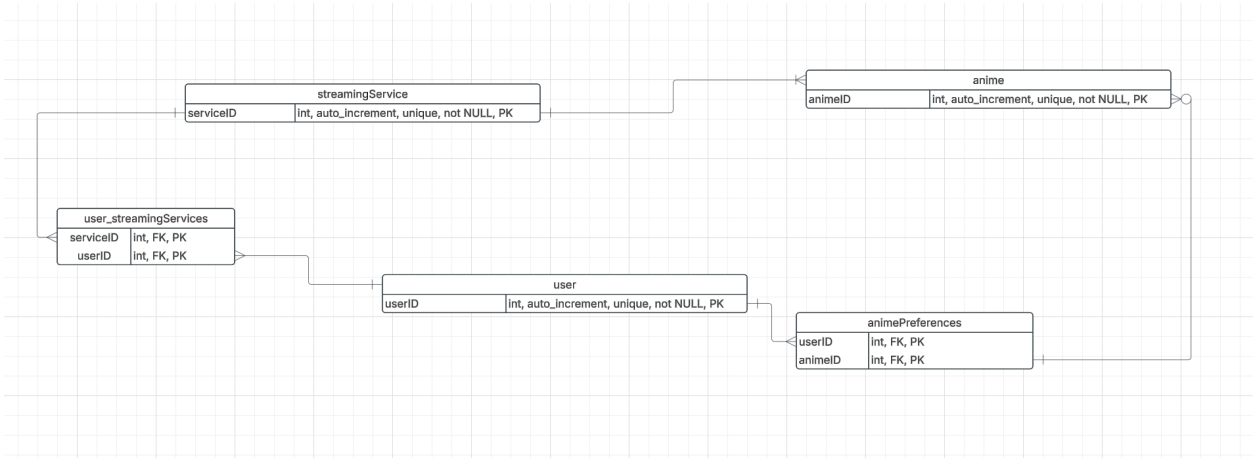
- ***serviceID****:* int, FK, PK
- ***userID****:* int, FK, PK

Relationships:

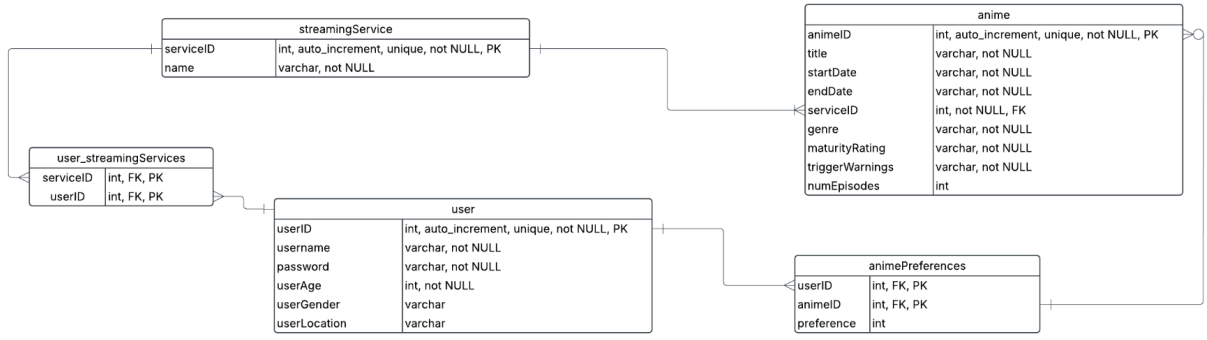
- 1:M relationship exists between **studio** and **anime**, where each anime is associated with a single studio

- Many **user**s can have many (or no) **streamingService**s, so their relationship is M:N as shown through the user_streamingServices junction table.
- Many **anime** are found across different **streamingServices** so their relationship is M:N.
- Additionally, a M:N relationship between **user** and **anime** is established by the ***animePreference*** junction table; each **user** can add a ***score*** to any ***animeID*** they want to base recommendations on.
- A 1:1 relationship exists between each **user**'s unique ***userID*** and the corresponding PK of their **animePreference** data.
- There is a 1:1 relationship between **studio.ID** and **worksList**, since there is only one **studio** per junction table.
- Finally, A 1:M relationship exists between **worksList** and **anime.ID** as there can be multiple titles associated with a single studio.

Updated Entity Relationship Diagram:



Schema:



Example Data:

streamingServices

serviceID	name
1	Crunchyroll
2	Netflix
3	Peacock
4	Hulu
5	Amazon Prime

user_ streamingServices

serviceID	userID
1	1
2	1
3	1
1	2
4	3

animePreferences

userID	animeID	preference
1	1	5
1	3	3
2	1	0
4	4	-3

user

userID	username	password	userAge	userGender	userLocation
1	fanik8d0	>-q7,a?+	20	F	USA
2	anime0q4	1atGvE	22	M	Canada
3	jerry12	0+U!\0u	31	N	Japan
4	userjh6gr	/1Pu?H4	19	F	Germany

anime

animeID	title	startDate	endDate	serviceID	genre	maturityRating	triggerWarnings	numEpisodes
1	Attack on Titan	2013-04-07	2022-11-13	1	Action	TV-MA	Violence	87
2	Naruto	2002-10-03	2017-03-23	2	Adventure	PG-13	Mild Violence	720
3	One Piece	1999-10-20	Ongoing	3	Fantasy	PG-13	Adventure	1000
4	Death Note	2006-10-04	2007-06-27	4	Thriller	TV-MA	Psychological	37
5	Demon Slayer	2019-04-06	Ongoing	5	Action	PG-13	Intense Action	44

Work Cited

Hamilton, Christofer. “The Anime Boom That’s Making Netflix Billions.” *Yahoo Finance*, 7 Jan. 2025,https://finance.yahoo.com/news/anime-boom-making-netflix-billions-183000698.html?guccounter=1.