

# TrashTrack

Your go-to web app for browsing the latest hot trash straight from Japan

Timothy Bui, Simon Louie, Jeff Deng

---

## Project Outline

We made a database showing currently airing and previously aired shows to allow for users who are interested in past and currently airing anime to find what they want and find information about it. This app is currently hosted on Heroku and uses Heroku PostgreSQL to host our database. The main use of the application will be to allow users to find what season their show is airing, find what studio animated the show, and characters in the show. To collect this information, we used Anilist's API to scrape for data to populate our database. There are many shows that air each season. With each season being approximately 12 weeks, this gives us a lot of data to work with. The complexity of this project makes it a good candidate for a database project.

## Database Overview

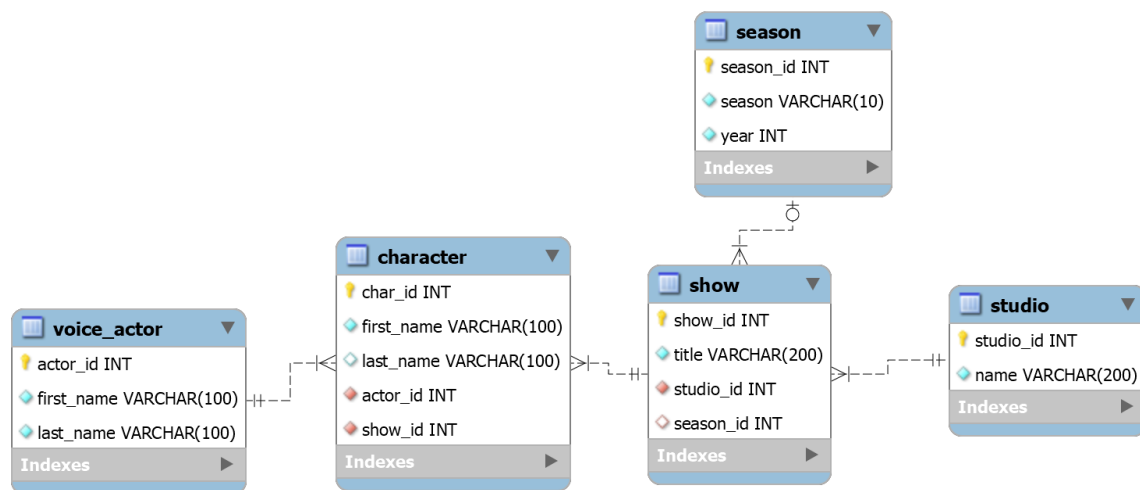
- **Voice Actors** — This table will store the names of the (Japanese) voice actors for the characters in the show. It contains the following attributes:
  - **actor\_id (PK)**: ID of the voice actor. Auto-incrementing counter. Used to uniquely identify them. Cannot be NULL and there is no default value. Int from 0 to 2,147,483,647.

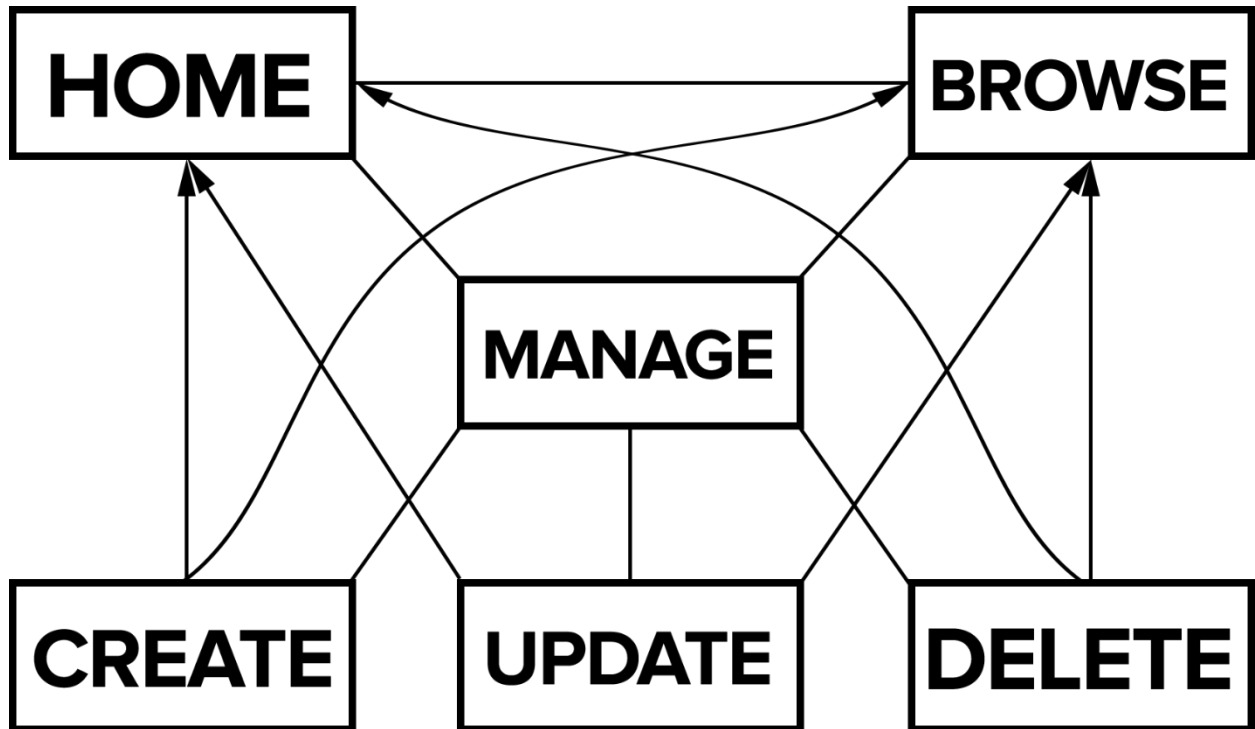
- first\_name: First name of the actor. Cannot be NULL and there is no default value. String with a max of 98 characters (VARCHAR(100)).
- last\_name: Last name of the actor. Cannot be NULL and there is no default value. String with a max of 98 characters (VARCHAR(100)).
- **Show** — This entity contains that data that identifies the show and essentially “makes” the show. It contains the following attributes:
  - show\_id (PK): Auto-incrementing counter. Used to uniquely identify them (Primary Key). Cannot be NULL and there is no default value. Int from 0 to 2,147,483,647.
  - title: This is the name of the show being identified. Cannot be NULL and there is no default value. String maximum of 198 characters (VARCHAR(200)). Cannot be blank.
  - studio\_id (Foreign Key): Lists the studio who animated the show. Cannot be NULL and there is no default value. Auto-incrementing INT with a max of 6 digits.
  - season\_id (Foreign Key): Lists what season the show is a part of. ID links to both season and year. Can be NULL and there is no default value. Auto-incrementing int value.
- **Characters** — These are the characters that appear in the show. It contains the following attributes:
  - char\_id (PK): ID of the character (two shows can have the same character). Cannot be NULL and there is no default value. Auto-incrementing int from 0 to 2,147,483,647.

- show\_id (Foreign Key): ID of the show where the character is from. Cannot be NULL and there is no default value. INT with a maximum of 6 digits.
  - actor\_id (Foreign Key): ID of the voice actor playing the character. Cannot be NULL and there is no default value. INT with a maximum of 6 digits.
  - first\_name: First name of the character. Sometimes will represent the full name of the character if they do not possess a last name. Cannot be NULL and there is no default value. VARCHAR with a maximum of 98 characters (VARCHAR(100)).
  - last\_name: Last name of the character. Can be NULL and there is no default value. VARCHAR with a maximum of 98 characters (VARCHAR(100)).
- **Studios** – This table will represent the animation studios that produced the show. It contains the following attributes:
    - studio\_id (PK): ID of the studio used to uniquely identify them. Cannot be NULL and must be unique. Auto-incrementing Int from 0 to 2,147,483,647.
    - name: Name of the animation studio. VARCHAR with a max of 198 characters (VARCHAR(200)). Cannot be NULL.
  - **Seasons** — This table will hold the name of the season (Fall, Winter, Spring, Summer) with the year so that shows can be categorized by what season it aired. It contains the following attributes:
    - season\_id (PK): This is the ID of a season with its year. Auto-incrementing INT value from 0 to 2,147,483,647 that cannot be NULL and must be unique.

- season: This is the name of the season it aired without regard to the year. There can be duplicates (e.g. Fall [2018] and Fall [2019]). This cannot be NULL and there is no default value. VARCHAR with a max of 8 characters (VARCHAR(10)).
- year: This is the year in which the various shows aired. There can be duplicates (e.g. [Winter] 2019 and [Spring] 2019). This cannot be NULL and there is no default value. INT value with a max of 4 digits.

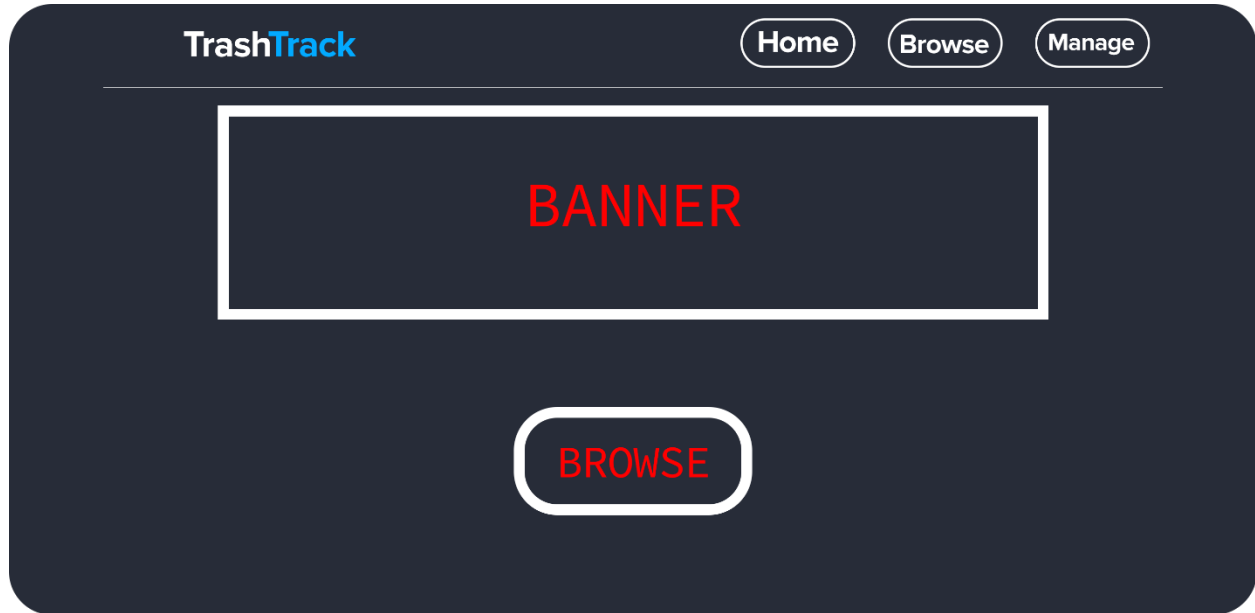
## Database Design



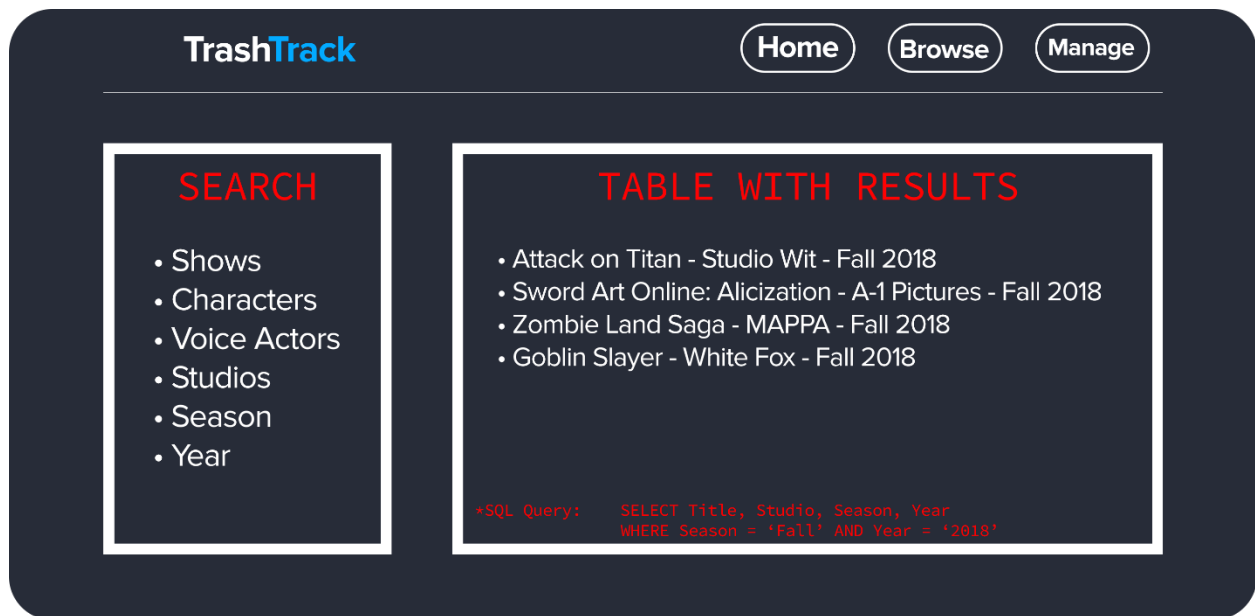


## Page-Specific Design

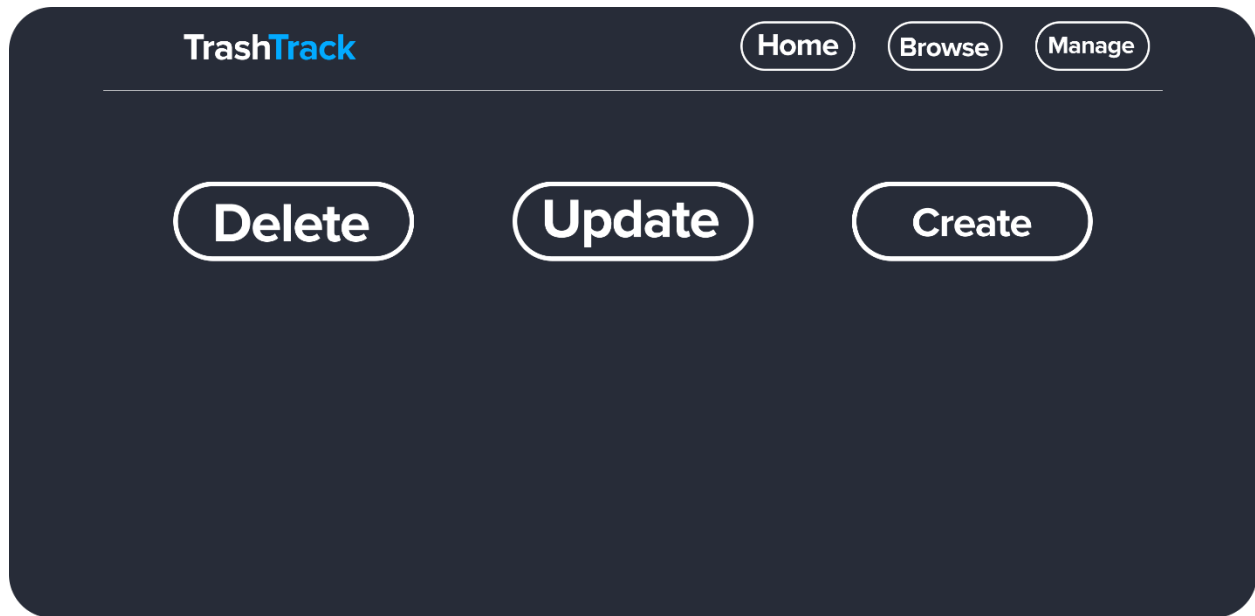
### Home Page



This is a static home page that the user will see upon arriving to our website. Here, there will be information regarding what the website will do and how to use it. There will be a button in the central area to redirect the user to our web app and a home button on the top to essentially reload the page. Since this page is static, there are no requests being made.



Here, the user will be able to interact with our web application to filter and search for various anime. The UI will contain search and filters on the left, navbar on the top, and query results on the center-right. The page will communicate with the server via HTTP GET with parameters involving season, studio, name, and year while changing the layout of the table displayed to suit the needs of the user and what they are attempting to view. The information received will be converted to a query to modify the database.



Here, administrators will be able to access pages to functions to update, delete, or create new entries in the database. This page shares the same navbar as all the other pages. Since this page is static, there are no requests being made. The information entered in the form will be converted to a query to modify the database.



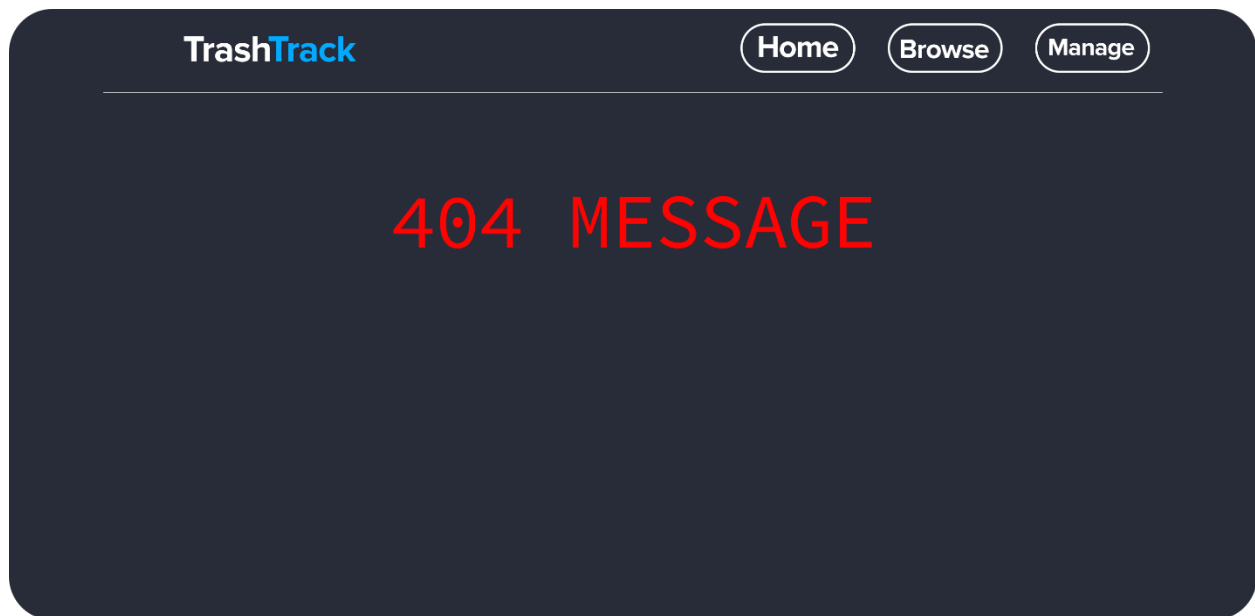
Manage Create/Update/Delete



The image shows a dark-themed web interface for a application named "TrashTrack". At the top left is the logo "TrashTrack" in white and blue. To the right of the logo are three navigation buttons: "Home", "Browse", and "Manage", each in a rounded rectangle. Below the navigation bar is a large white rectangular box with the text "INPUT FIELD" in red, indicating where a user would enter data. Below this box is a "Submit" button in a rounded rectangle.

This will be the general layout for the manage subpages where one can create, update, or delete entries. The input field is where the user will enter the info requested to modify/add/remove entries in the database. These pages will be sending requests via HTTP POST.

404




The 404 page will display for any address entered in our domain that does not exist. This is a static page and there will be no requests made.

## Database Tables

Show

	🔍 show_id 🔍	📄 title	🔍 studio_id 🔍	🔍 season_id 🔍
1	99255	Shokugeki no Souma: San no Sara	104	2
2	99420	Shoujo Shuumatsu Ryokou	314	2
3	97922	Inuyashiki	569	2
4	98436	Mahoutsukai no Yome	10	2
5	98707	Houseki no Kuni	6077	2
6	97886	Kekkai Sensen & BEYOND	4	2
7	100077	Hinamatsuri	91	3
8	99749	Fairy Tail (2018)	397	1
9	99629	Satsuriku no Tenshi	7	4
10	100049	Re:Zero kara Hajimeru Isekai Seikatsu OVA	314	1




## Studio

	 studio_id +	 name +
1	858	Wit Studio
2	10	Production I.G
3	6223	Studio Ouch
4	7	J.C. Staff
5	104	Lantis
6	561	A-1 Pictures
7	6101	Signal.MD
8	464	flying DOG
9	6186	comico
10	569	MAPPA




## Character

	 char_id +	 first_name +	 last_name +	 actor_id +	 show_id +
1	129386	Send	Help	129352	21127
2	124912	Pochi	<null>	128640	97907
3	124578	Chiaki	Oogaki	107013	98444
4	89028	Izuku	Midoriya	116971	100166
5	128037	Shizue	Izawa	116543	101280
6	123328	Isaac	Foster	95270	99629
7	76026	Megumi	Tadokoro	117003	99255
8	123181	Chisa	Kotegawa	106030	100922
9	126791	Shera	L Greenwood	116515	101004
10	122978	Haru	Okumura	95890	99693

## Voice Actors

	 actor_id	 first_name	 last_name
1	129352	Nani	Dafuq
2	107961	Megumi	Han
3	106184	Nao	Touyama
4	106297	Inori	Minase
5	119202	Yuuhei	Takagi
6	106661	Tomoyo	Kurosawa
7	102263	Youko	Hikasa
8	117003	Minami	Takahashi
9	119869	Shun	Horie
10	112209	Yurika	Kubo

## Season

	 season_id	 season	 year
1	1	Fall	2018
2	2	Fall	2017
3	3	Spring	2018
4	4	Summer	2018
5	5	Winter	2018
6	6	Winter	2017

## Group Responsibilities

**Timothy Bui:** Worked on written documents, updated web design mocks, update web relation diagram, CSS, HTML, set up Heroku, initialized GitHub repository, project lead, scraped data, imported data

**Jeff Deng:** CRUD elements, HTML, initial draft of web design mocks, initial draft of web relation diagram, CSS

**Simon Louie:** CRUD elements, HTML, CSS, relational schema (both initial draft and updated), set up Heroku PostgreSQL tables, maintained and initialized Node.JS server code, set up Handlebars