YaTV.app

OWEN LOVELUCK, RYAN MILLIGAN, ERIC QIN

CS3200 FALL 2020, 12/09/2020

Abstract

In the present day there exist all sorts of video-streaming applications, each with their own sorts of videos and shows. It can be real confusing trying to keep track of the mess that comes with a huge variety of apps, which is why we've created YaTV.app (Yet Another TV App).

YaTV combines the vast libraries of all your favorite video-streaming applications into one place, acting as a hub for your video needs. Users can easily register with their name, an email, a password and their country (to follow appropriate video content guidelines). Once registered, users can watch videos and shows, make playlists of their favorites, and add content they wish to watch to a "to-watch" list. YaTV also keeps track of all the content a user watches in case they want to watch something again!

Some videos and shows require a subscription to the app they are hosted on. Luckily, YaTV helps manage all a user's subscriptions by keeping track of the cost and expiration date of each subscription.

For users new to the video-streaming world, YaTV also keeps track of the apps themselves. Each application has set of platforms it supports. YaTV can tell the user each of those platforms, as well as if it supports mobile use. Additionally, YaTV includes the version number for each app on a platform on top of a rating so users can make the best choice for their app.

Description

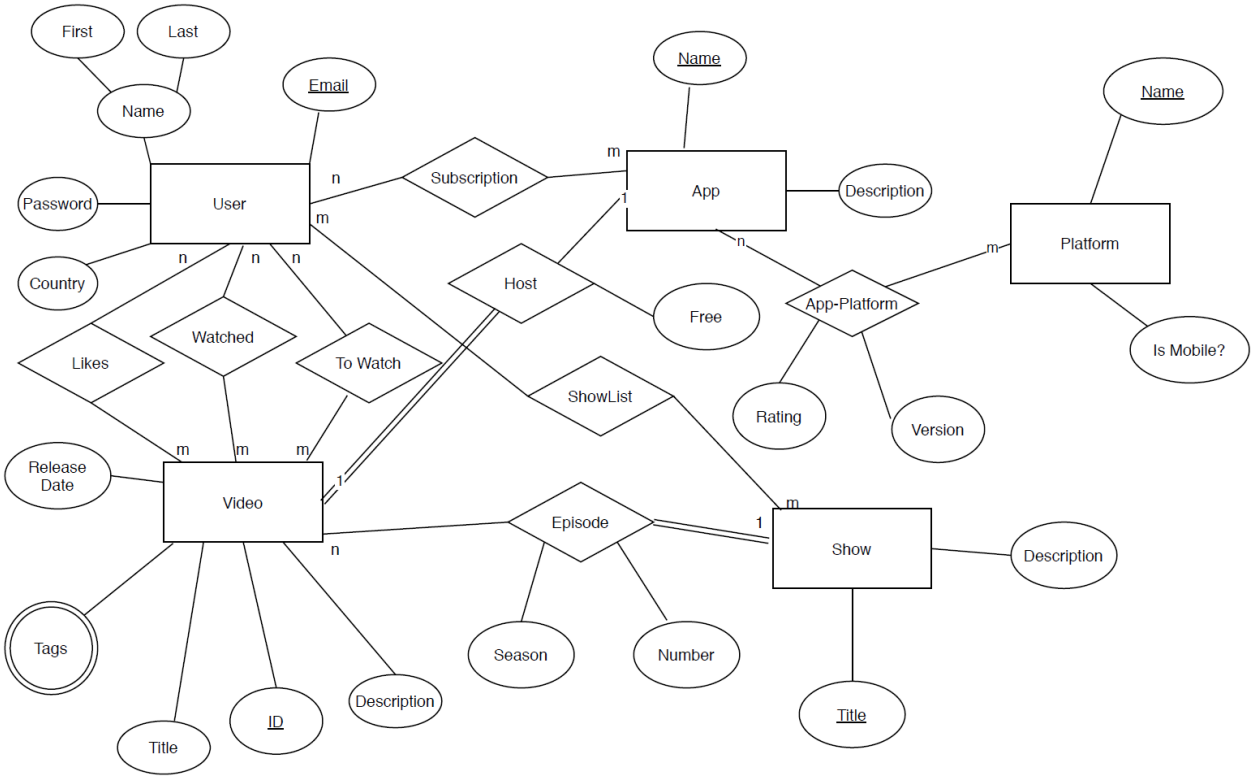
When designing the database schema for out final project, the YATV.app, we decided to create the following entities: User, App, Platform, Video, and Show. With these we created the following relationships: Likes (1 user likes 0 or many videos), List (1 user has a list with 1 or many videos they wish to watch), Host (1 video must have a single host), Subscription (1 app has 0 or many users), App-Platform (Many apps, can be on many different platforms), and Season (many videos can be on many groups of shows.

Non-technical users of the app should understand that a Video entity refers to a single video on a specific platform, this video can be part of a season as an episode. It is worth noting that a video or an episode on one platform may be exactly the same in terms of Name and description, between platforms, however, are described as different "videos" in the database. Users will have separate lists for shows or videos they have watched/like/want to watch and may subscribe to specific apps. Subscriptions are not specific to specific platforms.

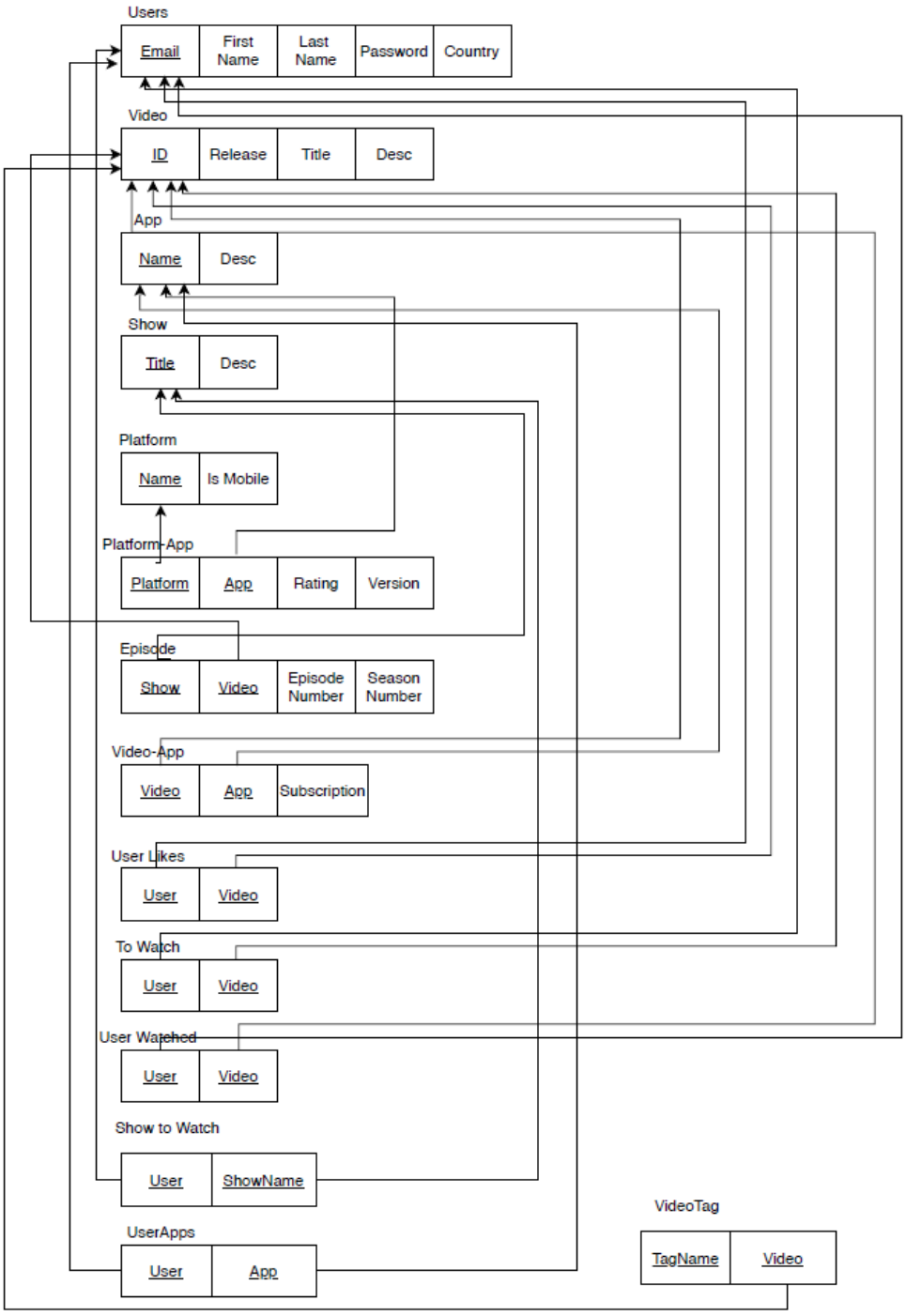
In terms of our interpretation of the narrative, we made assumptions about the uniqueness of the names of Apps, Platforms, Shows, and user email addresses. We added an ID to Video as the primary key, as we decided that it was unreasonable to have uniqueness across all apps for video names. We also assumed that the same video or show cannot exist on multiple Apps. This would have slightly changed the design of show and season to decrease the amount of duplicate data.

In terms of restrictions, the only major restrictions we have is that a video cannot be present in multiple seasons and that a video must have a host. This restriction was given in the spec; however, I think it would be perfectly reasonable for a video to exist without an App connected to it. Say for example a user had a video in their list and it was removed briefly from a platform before appearing on another app, the user would still want their personal data about that video. For example, if they had watched it, or if they wanted to in the future.

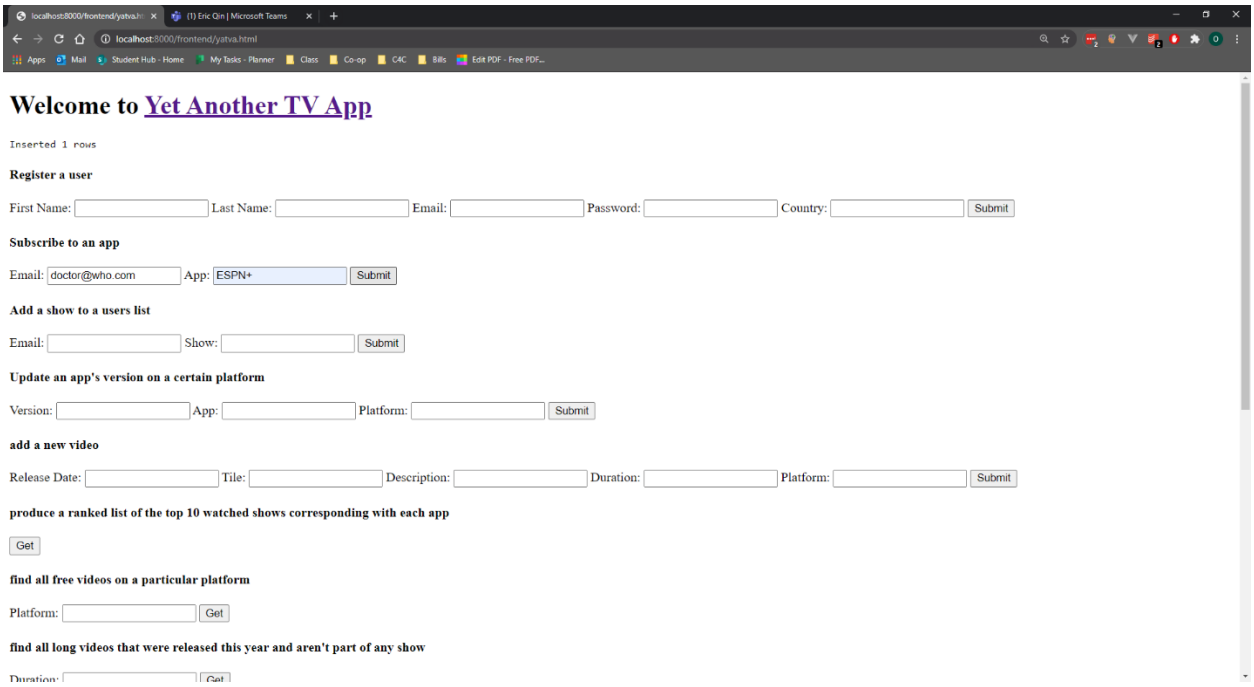
ERD


First
Last
Name
Email
Password
User
Country
Subscription
App
Name
Description
Name
Platform
Is Mobile?
Host
Free
App-Platform
Rating
Version
Likes
Watched
To Watch
ShowList
Release Date
Video
Episode
Show
Description
Tags
Title
ID
Description
Season
Number
Title
n
m
1
n
m
n
n
n
m
1
m
m
m
m
n
1
m

Relations


Users
Email
First Name
Last Name
Password
Country
Video
ID
Release
Title
Desc
App
Name
Desc
Show
Title
Desc
Platform
Name
Is Mobile
Platform-App
Platform
App
Rating
Version
Episode
Show
Video
Episode Number
Season Number
Video-App
Video
App
Subscription
User Likes
User
Video
To Watch
User
Video
User Watched
User
Video
Show to Watch
User
ShowName
UserApps
User
App
VideoTag
TagName
Video

Running the System



Retrospective

Overall this project was a great learning experience for us. We learned things like setting up a DBMS from scratch, parameterizing queries using the MySQL library in node.js, and how to translate a narrative into a DDL.

This project didn't come without its challenges. Setting up the database with DDL and DML were definitely the most difficult tasks for us. Also, actually compiling a presentation together when we were all working remotely from each other was also a challenge. Something we didn't really enjoy about this project were having to come up with our own data to populate the tables with.

All that being said, there were exciting parts to this project as well. Figuring out how to link up HTML to an actual interface was amazing. Also, while not quite as technical, learning how to stitch together a video with editing software was a refreshing change of pace from all the database work.

Conclusion

Our project was largely successful, as we finally produced a populated database that conforms to 3NF to the specification of the narrative. In addition to that, we also produced a successful html application as well as a JavaScript executable to host a server for our queries. Because of all of this work that we have done on the project, it is now possible to simply boot up a client and server on a computer with our code, and query our database using our interface and see the results of the query. For future work on the project, it would be interesting to expand upon the web interface to support more queries, and to further populate the database with data, perhaps even finding a way to convert data from real life databases into our own.