

Canada Foxes - Emma Buller, Lucas Lee, Naomi Naranjo, Deven Maheshwari

SoftDev

P01 -- Design Document

2021-12-09

time spent: 1 hour

Proposal

A website where users can enter and store searches on a movie title, showing information about the movie's production/cast, reviews about the movie (NYT critics, rotten tomatoes scores), and recommendations for songs based on the soundtrack album/artist (most likely about 5 songs suggested).

Design Components

- Flask - Will serve as the python backend, routing the various python files and html front end code. Will be used to make API requests and manipulate the data into databases as well as host the website.
- HTML and CSS/JS (Foundation) - Use to display the front end and make the website aesthetically pleasing. Handle form inputs
- REST APIs - Spotify, MovieReview, Imdb. Request made to Imdb API to get information for actors, directors, etc. Then used in MovieReview API to see what critics said about the movie and matched with the Spotify API to recommend other soundtracks and movies for the user.
- Sqlite3 - Save user information and recommendations from past searches for each user profile. Registration and login capabilities.

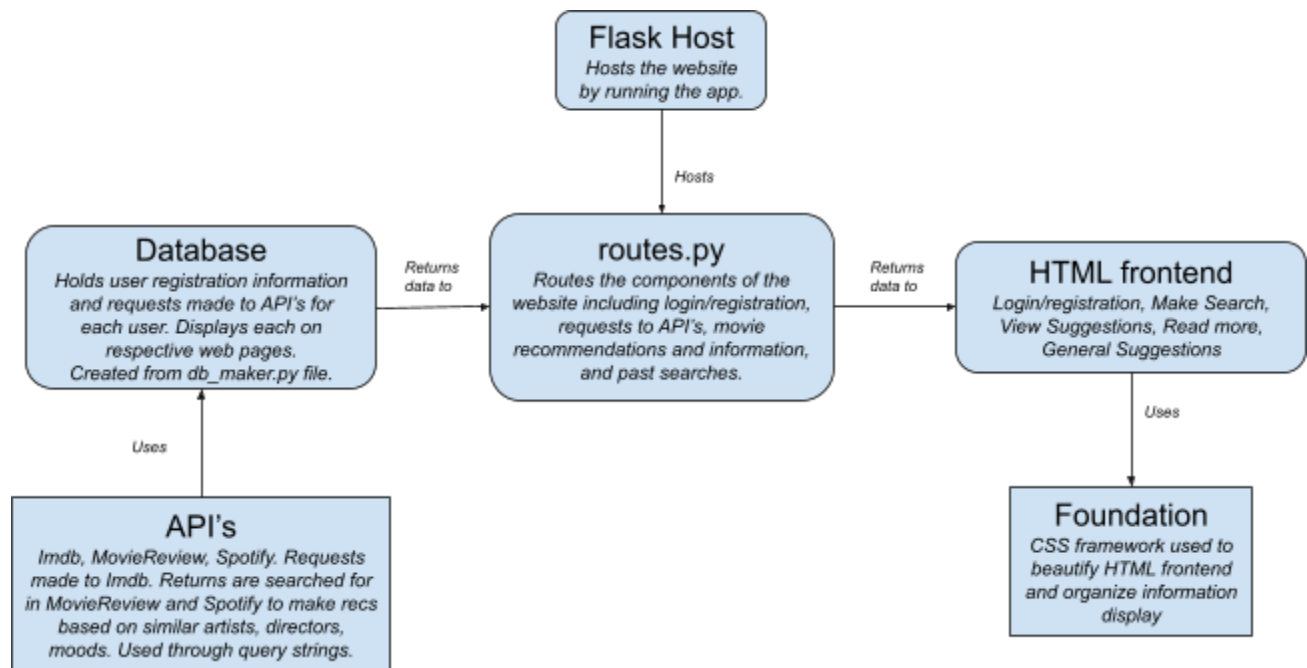
API's

1. Spotify API
2. MovieReviews API
3. Imdb API

Foundation

Foundation has all of the aspects of bootstrap and we find it easier to decipher and follow their docs. This framework is javascript heavy, but that allows us to experiment with animations and gather user attention. Its CSS features contain grid tables, helpful for displaying movie information and recommendations, as well as clear nav bar and abide functions to require user input.

Component Map



Database Organization

We will have three tables: user info that handles account information and credentials, and past searches that handles information from past user searches and is the users history page as well as a table that tracks movies and corresponding review and spotify rec.

User Info

user_id	username	password
PRIMARY_KEY INTEGER	TEXT	TEXT

Past Searches

movie query	timestamp	user_id	movie	reviews	spotify recs
PRIMARY_ KEY TEXT	TEXT	INTEGER	TEXT	TEXT	TEXT

Sitemap (Frontend)

- Login/Register Page
 - Allows user to sign up for an account
 - Redirect to main page once logged in
- Main Page
 - Link to login/register if not logged in
 - If logged in, form to enter a movie title query and display info on enter
 - Enter a single move name to be sent to IMDB to display top matching movie title, cast, release date, etc. (basic information)
 - SearchMovie gives list of movies that use the keyword
 - Choose movie you want from that list
 - From choice, we get the movie id, which is used to acquire the basic information
 - Pull full review about the movie from the NYT Movie Review API
 - Title of movie in query input of api call
 - Can also pull various shorter reviews from the IMDB API
 - Uses web scrapping to gather information from NYT as they only return a link to their review article when making an API request
 - Search Spotify for the movie soundtrack
 - Pull info about the movie soundtrack album and the artist(s) (if one exists)
 - Find album using Spotify API search function
 - Input values for album (film name) and year (release date year)

- Will display main artist of score/soundtrack if there is one
 - Ex: Black Panther the Album by Kendrick Lamar
 - Else, display multiple artists (if main artist of soundtrack is “Various Artists”)
- Display other albums/songs by the same artist
- Display recommended artists based on that artist
 - Based on spotify API “Related Artists” info
 - If various artists, it will probably be a mix of randomly selected related artists
 - One from each artist
 - Will check for repeats of artists
- History Page
 - Lists the past searches and recommendations that have been made to a user.
 - Chronological order
 - Users can search the contents of their history

Task Breakdown

Emma - IMDB and Movie Review API Backend, Frontend

Lucas - Spotify API Backend, CSS, Web Scraping

Naomi - Database management

Deven - Backend routing using python and making API requests, Frontend

Ship Date

2022-01-04