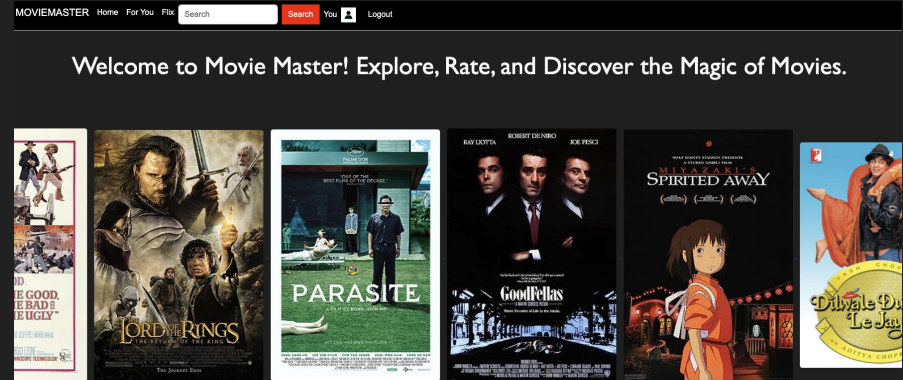# MovieMaster

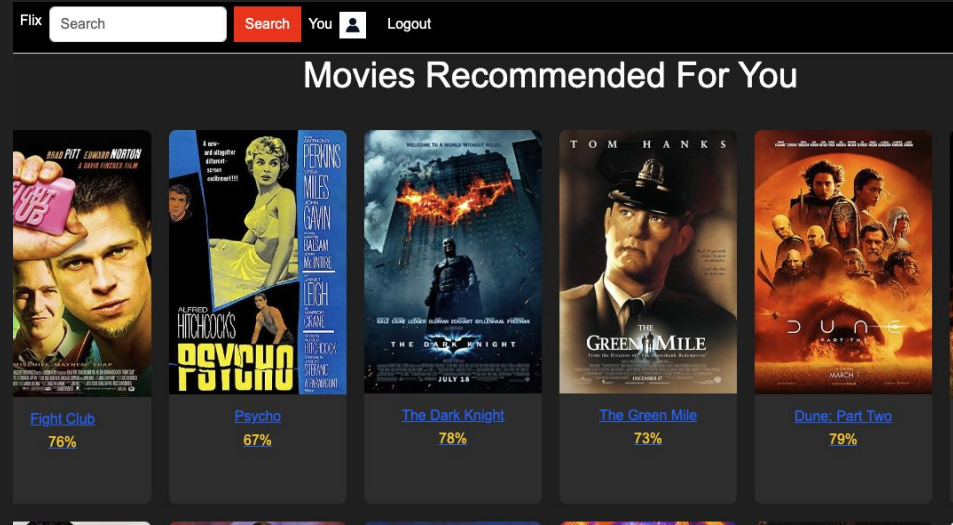William Hardee, Kayhaan Rashiq, Aiden Devine, Sopan Nair, Sameer Kulkarni, and Nicholas Kopiwoda

Description of App

- MovieMaster is a centralized platform for movie lovers who want a one-stop-shop for finding their next Movie
- Pulls ratings from IMDb, TMDB, and MetaCritic
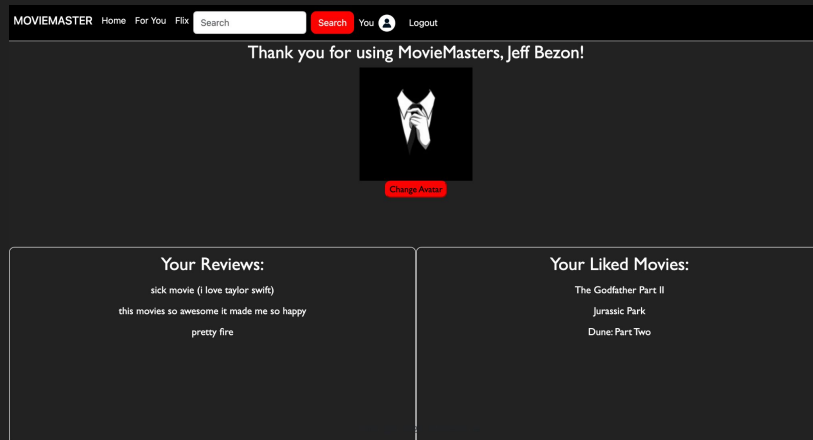- Allows users to add ratings, reviews, and "Like" movies
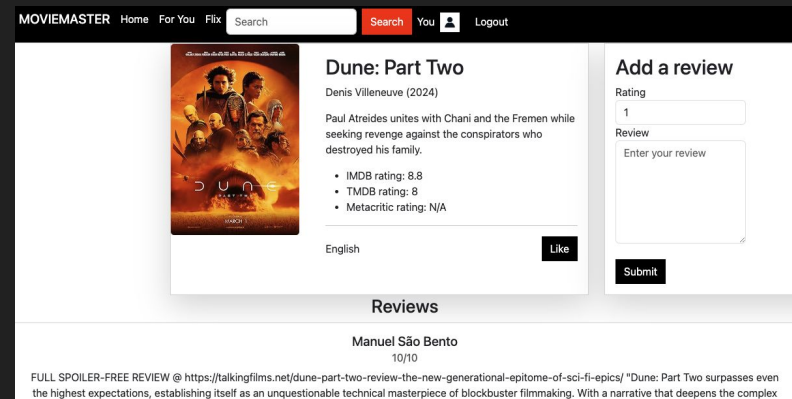
Description of App Continued

- "For You" page uses "Liked movies" to compute movie suggestions using state-of-the-art Recommendation Model
- "Flix" Page allows you to watch uninterrupted stream of hot movie trailers

Purpose

- Provide movie buffs with one single platform to search, review, discover, and rate movies

- Aggregate rating data from different sources (IMDb, TMDB, MetaCritic)

- Suggest new movies with advanced Recommendation Model based on individual taste

- Doomscroll trailers with Friends/Family to find a suitable movie

Intended User Base

- Movie Buffs looking for new movie recommendations
- Users who want to leave a rating/review for a recently watched movie
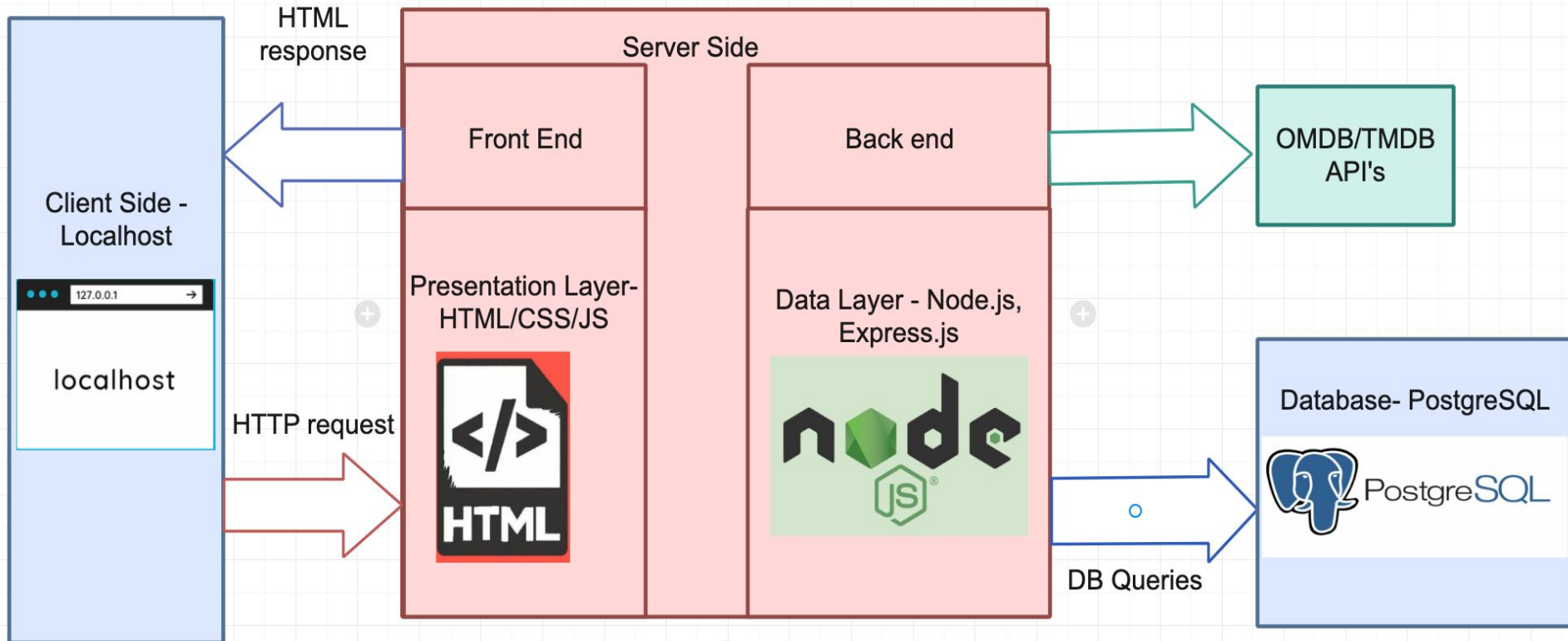- Friends/family deciding on a movie for the night

Challenges

- Finding free/affordable APIs to pull data from for movie ratings, descriptions, reviews etc.
- Embedding uninterrupted stream of YouTube trailers on "Flix"
- Developing algorithm for Recommendation Model on "For You"
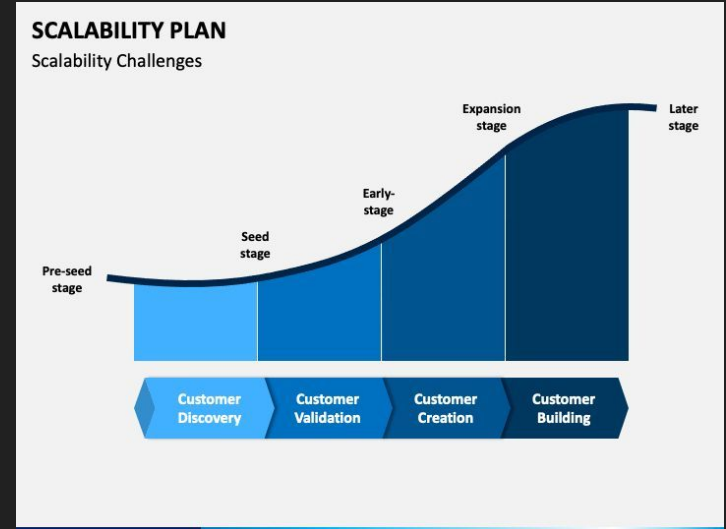- Dynamically displaying hot new movies on Home Page

Fixes

- TMDB, OMDB
- TMDB and filtering
- Euclidean Distance Algorithm
- Templating in Handlebars

Architecture Diagram
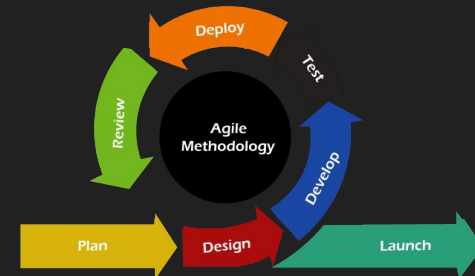
Future Scope

- Potential Growth opportunities:

  - Expand into TV shows to attract new users

  - Embed YouTube shorts of funny movie-clips

    to drive engagement of younger users

  - Improve Recommendation model to utilize

    user ratings and reviews in computation

    - Numerical, NLP

SCALABILITY PLAN
Scalability Challenges

Pre-seed stage / Seed stage / Early-stage / Expansion stage / Later stage

Customer Discovery | Customer Validation | Customer Creation | Customer Building
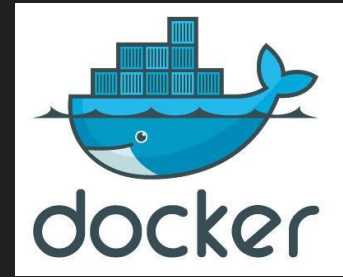
Technical Details

- Programming Languages:
  - JavaScript 4/5
    - ExpressJS was easy to use
  - HTML/CSS 4/5
  - SQL 4.5/5
- APIs
  - TMDB 5/5
    - Had a ton of movies and contained a route that would give a json result of a list of popular movies which was essential in inserting large amounts of movies into our DB
  - OMDB 2/5
    - Didn't have as many movies as TMDB and some of the json results of the API were poorly formatted. Only used to get ratings from external review sites.
- Project Management
  - Style: Agile Methodology
  - Collaborated with GitHub, Google Drive
  - Met twice weekly to discuss project requirements, progress, and hurdles
  - Communicated with Email, iMessage

Technical Details Cont

Mocha & Chai - 2/5:

- Caused some issues when starting docker container as asynchronously adding movies to our DB could cause the tests to time out.
- Had to docker compose down -v for every time we started the container.
- Made starting the container take a while.

Docker - 4.5/5:

- Easy to use and allowed us to not worry about installations.
- Standardization, if one thing worked on one teammates computer it will most likely work on all teammates computers.

Demo Time!

Q & A