Case Study for



Overview

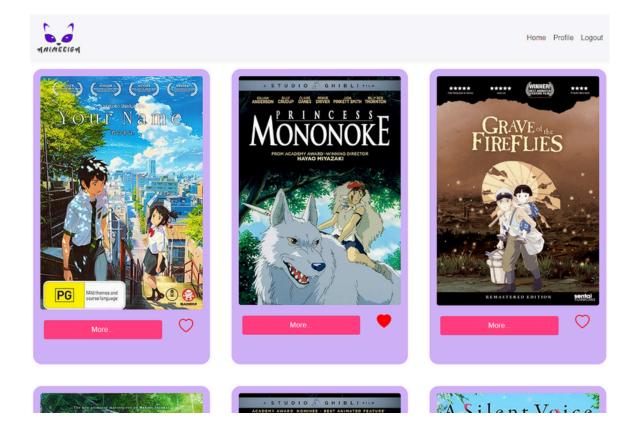
AnimeEiga is an innovative web application designed to cater to short-anime enthusiasts worldwide. Developed using the robust MERN stack (MongoDB, Express.js, React, and Node.js), this platform offers a seamless and interactive experience for users who wish to explore and keep track of their favorite animes.

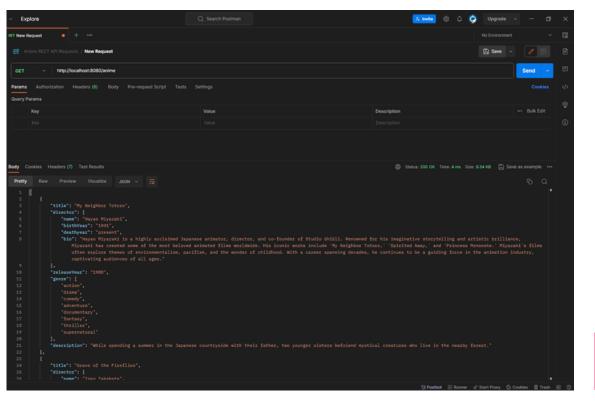
Purpose & Context

AnimeEiga was a personal project I built as part of my web development course at CareerFoundry to demonstrate my mastery of full-stack JavaScript development.

Objective

The project aimed to be an ambitious full-stack project I could add to my professional portfolio. I wanted to solve the problem of building the complete server-side and client-side of the application from scratch.





Server-Side

I created a RESTful API using Node.js and Express, that interacts with a non-relational database (MongoDB). The API can be accessed via HTTP methods like GET or POST. To retrieve data from and store data in the database, CRUD methods are used. The API provides anime information in JSON format.



Step 1

Initially, I had to decide between a relational and a non-relational database. After experimenting with PostgreSQL and MongoDB, I opted for MongoDB due to its non-relational nature and greater flexibility.

Step 2

To maintain data consistency and accommodate future scalability, I developed models for both the anime and user data. After that, I utilized Mongoose to communicate with the database.

Step 3

I opted for basic HTTP for the initial login, combined with JWT Token-based authentication to enhance the security of my site. Additionally, I integrated CORS, implemented password hashing, and conducted data validation to provide further protection.

Step 4

After thoroughly testing all endpoints with Postman, I successfully deployed my app on Heroku and utilized MongoDB Atlas to host my database.

Client-Side Development

Once the API was completed, I began constructing the interface that users would interact with for making requests and receiving responses from the server. This is a single-page, responsive application built using React and React-Redux. It features multiple interface views, including, but not limited to, a main view that displays a list of all movies, a single movie view that provides details about an individual movie and allows users to add it to their favorites, as well as login, registration, and profile views where users can update their information and favorite lists.

| ANIMEEIOA | \equiv | - АЛАКСИСА На ПАКСКО М | Home Profile Logout Search | Home | e Profile Logout |
|--|----------|-------------------------------------|--|---|-------------------|
| Log in! Meow ~ Username: Enter your username Password: Enter your password Submit | | <image/> | <complex-block></complex-block> | Profile Username: testing14 Name: Jane doe Birthday: 2000-08-08700:00:00:00002 Email: test@gmail.com Effet Delete Account | |
| ANIMEEIGA | | More More More | Моге. + 5 T U D I O ∕ G H I B L I //×× Тектору провологости то току с окладира | | |
| Join us! Meow~ Username: | | MARACECCOM | Home Profile Logout Search | Home | ie Profile Logout |
| Name: Password: | | | forms between a high mysterious woman in | Profile Name Jane doe Birthday mm/dd/yyyy | |
| Email: Birthday: | | Makoto Shinkai Genre: Romance | | Email test@gmail.com Seve Cancet Favorite Movies | |
| mm/dd/yyyy Submit | ٥ | | | The Garden & Words | |

Challenges

This project has become my favorite to date, as I constructed everything myself from the ground up, including the API, design, and the final interactive web application. The most significant challenge I faced was the integration of all my desired features. I didn't aim for merely a functional website; I wanted it to reflect my exact vision. Consequently, I dedicated time to grasp how React and Redux operate.

Duration

Despite facing various challenges, I successfully completed it within a month. However, this may not be evident on my GitHub timeline, as I paused work on it for a few months due to personal commitments.

Credits

Lead Developer: Win Win Khaing Tutor: Muhammad Usman Ali Mentor: Martin Macharia

