Full Stack Developer Case Study

by Giancarlo Boas

Overview

myFlixApp is a responsive web application that allows users to explore a selection of movies, view details about each film, and manage their own list of favorites.

The app was built using React and connects to a custom-built REST API. It includes user registration, login, and profile management features.

This project was developed as part of my full-stack web development course.

Check the site

Context

myFlixApp was created as part of a project to practice and demostrate the general knowledge of Full Stack JavaScript development

Objetive

The objective was to build a single-page application using Fron-End and Back-End fresh knowledge to include in a professional portfolio
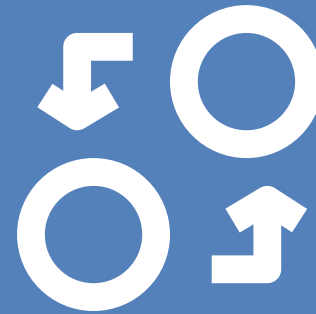
Hard Work, Great enviorement and the right tools

Duration

Developing both Client and Server Side were done in about a month. The process was longer on the client side as many functions had to be implemented

Methodologies

- Axios
- React
- JWT
- Node.js
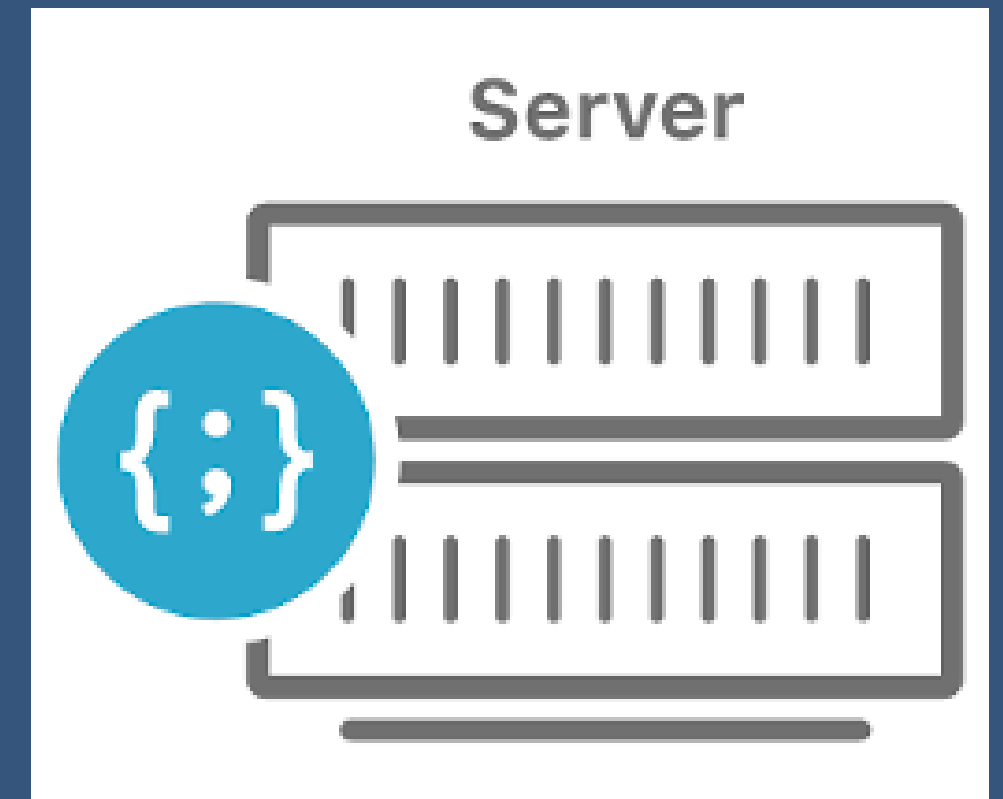- Mongo.DB
- Git
- Github

Credits

Lead Dev: Giancarlo Boas
Tutor: Marwa Jawad
Mentor: Renish Bhaskaran

The Server-Side

Creating the Restful API was the first part of the project using Node.JS and Express, adding all the data using also MONGODB. The API can be accessed via commonly used HTTP methods. In this API you will get the basic movie information

Server

Basic data

MongoDB was selected as the Database

Business Logic

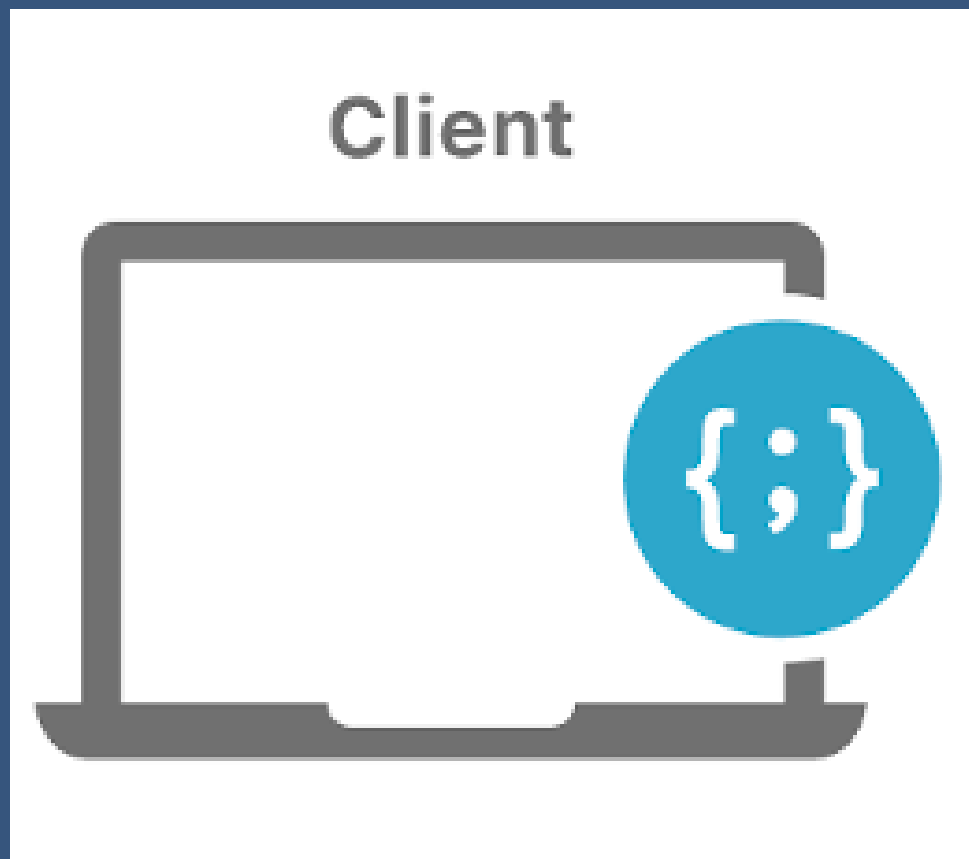Mongoose was selected to interact with the database

Security

JWT token was selected to make the site secure

Deployment

After Testing Endpoints with Postman, Heroku was selected to deploy the app

The Client- Side

Client

With the API ready I created the interface for user to be able to navigate around all the movies that the server offer in a way that they can interact with it using the responsive single page design on React-Redux.
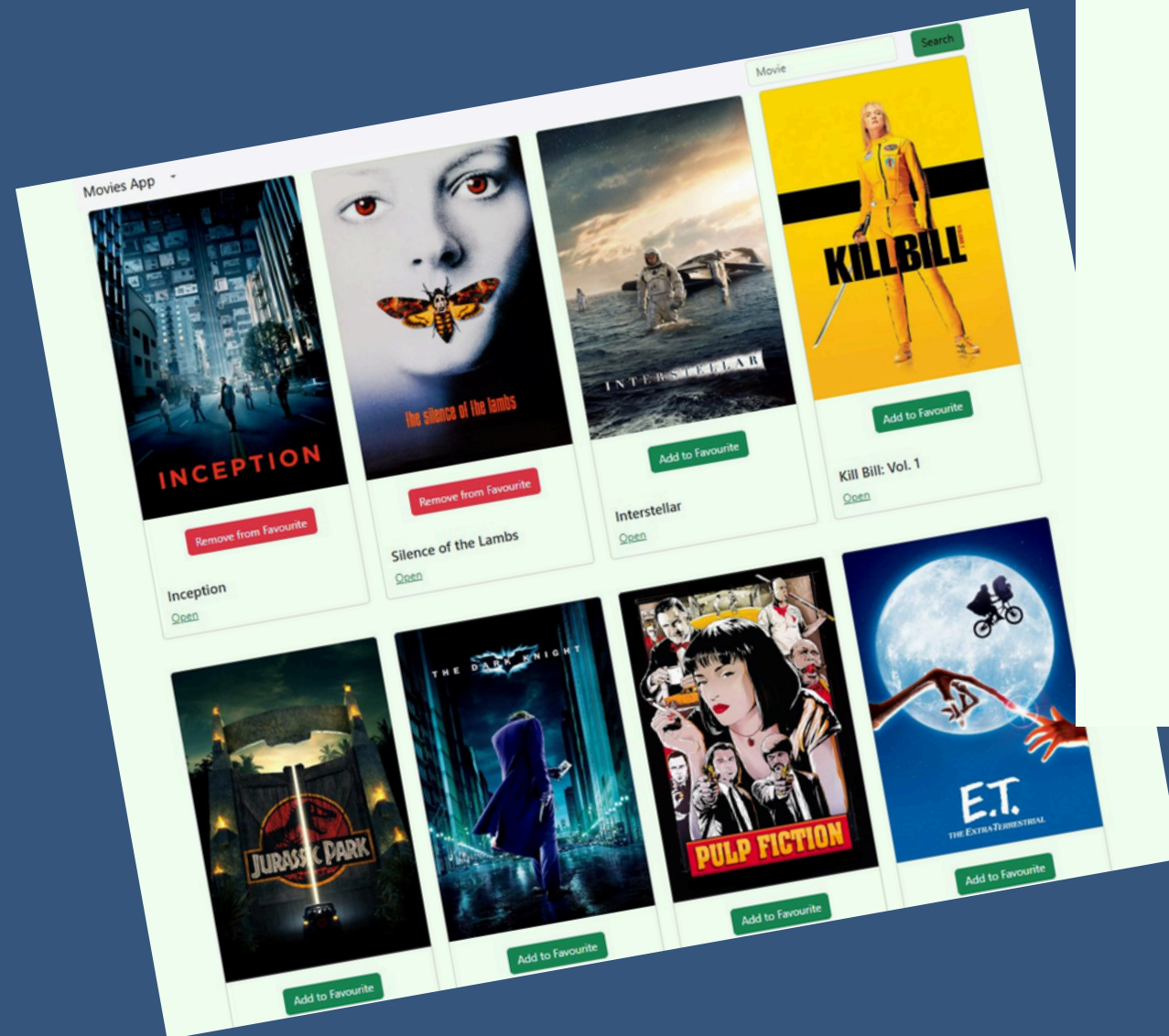
Build

React was selected because it is a powerful JavaScript library that makes it easy to build fast, scalable, and maintainable user interfaces using reusable components.

Design

I used React Bootstrap because allows you to build responsive, modern interfaces quickly by combining Bootstrap's sleek design with fully React-based components.

Final Product

To improve state management and support future scalability, I implemented Redux in the application. Once completed, I deployed the final version

Users can securely log in to the application and manage their profiles. In the main view, they can browse the full movie catalog or search for specific titles. By clicking on "More Info," they can access detailed information about each film.

Page Views

What went well?

The result and all the knowledge behind it is something that, even tricky, it was satisfying to see

What didn't go well?

Time was the biggest challenge here, its true that this was a 1 person work that didn't includes a team but the dates to deliver were sometimes really short

Retrospective

Future Steps

1. Increase the database to show more titles
2. Add search and filter options
3. Allow users to leave ratings or reviews
4. Add Reccomendations functions

Final Thoughts

It was hard to learn a new technology specially that fast but doing it was more satisfying that reading it. I would like to create more projects that can make me practice the React processes