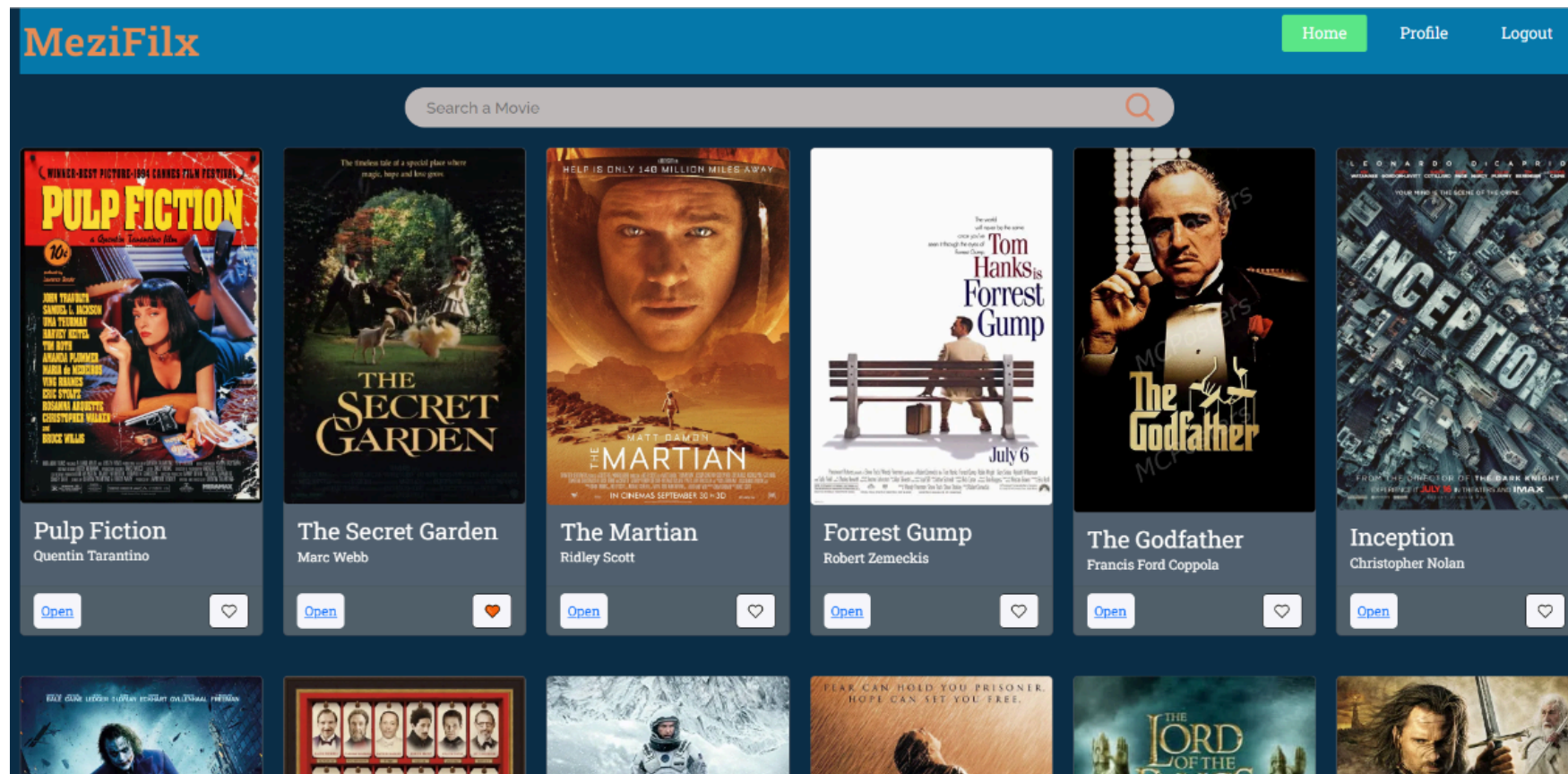


Case Study for myFlix full-stack project



[MeziFlix Demo](#)

Duration

- Serverside: 4 weeks
- Clientside: 3 weeks

Credits to

- Tutor: Othman Yamak
- Mentor: Stephen Barungi

Teck-stack - MERN

clientside

- React
- Bootstrap
- Javascript
- Netlify

serverside

- Nodejs
- Mangos
- Express
- Insamonia
- Render

Overview

MeziFlix is a movie web app built with the MERN stack. It is inspired by IMDB, but, well, not quite as big. It provides users with information about movies, directors, and genres. Users can create an account, update their personal information, create a list of favorite movies, and search for movies.

Purpose & Context

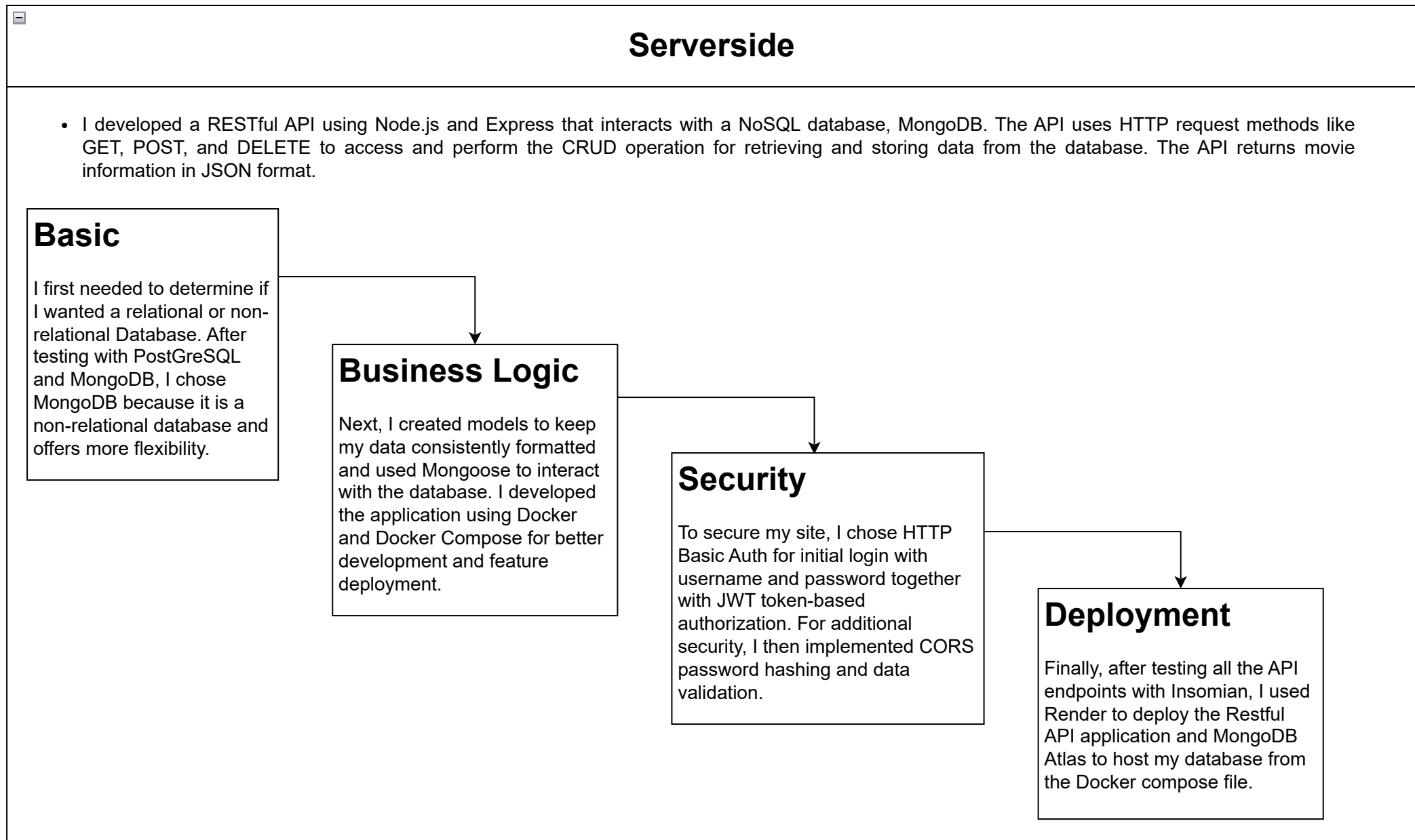
I developed Meziflix as the front-end client component of a full-stack web application as a part of the web development program at CareerFoundry. The application is built on top of the back end (database and RESTful API), which I previously created.

Objective

The primary objective was to acquire hands-on experience and create a fully functional, end-to-end web application in line with the principles of the MERN stack.

Approach

- This is the frontend client project of a backend API I developed in advance. Therefore I will expaint server side and client side parts below.



Client-side

- Once I completed the backend part, I used React.js as a frontend framework to create the UI needed for users to interact with the server side as components. It is a SPA (single page application) and responsive web application. It provides several interface views, including a movie view, a login view, a registration view, and a profile view (where users can update their user information and favorites list).

Frist Build

In this project, I learned and gained hands-on experience with MVC architecture. To accomplish this, I chose React because it is component-based, fast, easy to maintain, well-documented, and has a big community front-end framework. I then used the Vite build tool to create and complete the build operations.

Create Components

After the initial build was done, I created React components for each view and React hooks to control the state of the app. I utilized the Fetch-API to pull the API I had previously created.

Design

I used React Bootstrap to style the layout of the pages and cards and to ensure consistent styling. I also utilized client-side app-routing to add authentication to access views.

Deployment

Finally, after I had tested all of the functionality of the app, I hosted a finished version of the app on Netlify.

Retrospective

what I learned and enjoy?

- What did I learn and enjoy? I was able to build a fully functional, responsive, full-stack web application using the MERN stack, which includes MongoDB, Express, Node.js, and React. In addition, I was able to gain experience with HTTP user authentication and JWT authorization, and I tried out development with Docker and Docker Compose. Last but not least, I learned to deploy in render, Netflix.

What were the challenges?

- The initial challenge was to develop a Docker container with two services, Node.js and Express, and the MongoDB with Mongosh database. Besides that, managing the react state with hooks was a big challenge until I got my head around it. Thanks to many tutors and mentors for their support.

Next steps

- I will be refactoring the codebase to implement good proctics as I get more experience and update the UI. In addition, I would like to add an admin role to manage the content of the app, for example to add, update movies. I also want to include features like whats trending, whats popular, ability to watch trailers of the movies and movie ratings.

Final thoughts

- In conclusion, I've enjoyed the project and gained a lot of skills in developing a full-stack web application from scratch. This project taught me the ins and outs of full-stack web development. With it, I completed the MERN stack.

MeziFlix Login Signup

Username

Password

Email

Birthdate

Sign Up

MeziFlix Login Signup

Username:

Password

Login

MeziFlix Home Profile Logout

Search a Movie


300 × 300


User: UserFromReact5
Email: UserFromReact5@testemail.com
Birthdate: Not available

Update Data Delete Account

Favorite Movies

Normal





Links

- React Frontend link

[MeziFlix Demo](#)

- API Backend link

[API Code](#)