

**COMIC
FLICK**

The logo features the words 'COMIC' and 'FLICK' stacked vertically in a bold, orange, sans-serif font with thick black outlines. The text is set against a blue, jagged-edged shape that resembles a comic book explosion or a speech bubble. The background of the entire image is a vibrant yellow-orange gradient with a halftone dot pattern of small red dots. Several thin, parallel lines radiate from the bottom left towards the top right, creating a sense of motion or depth.

Objective

Comic Flick is a web app, developed using the MERN stack, that provides users with access to information about movies, directors, and genres. The basis for selecting the movies is that they are all adaptations of comic books. Users are able to create an account, update their personal data, and create a list of their favorite movies.



M.

MongoDB



R.

React



E.

Express



N.

Node.js

Backend

The Process

1st

RESTful API

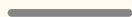
The RESTful API provides access to the database and was built using **Node.js** and **Express**



2nd

NoSQL Database

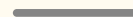
Create a **NoSQL** database using **MongoDB**



3rd

Business Logic Layer

Model the business logic layer using **Mongoose** to ensure the data in the database is consistent



4th

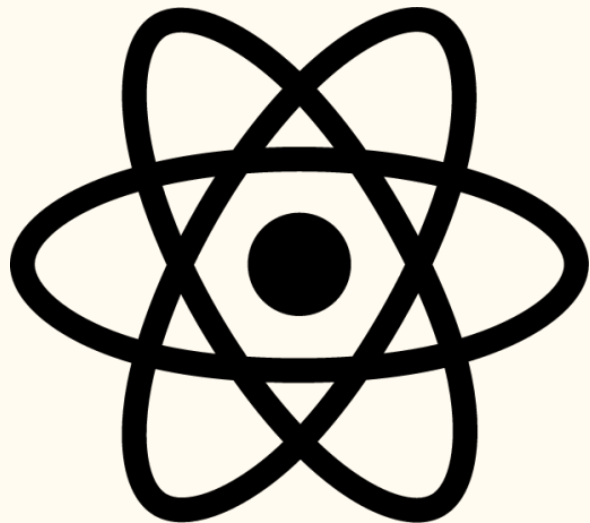
Authentication and Authorization

Implement authentication and authorization into the API using basic HTTP authentication and JWT (token-based) authentication

Frontend

React

React Bootstrap



The React Framework was used because it has a virtual DOM which allows for fast rendering views. It's also supported by a wide community of developers and engineers which will make it easier to fix any bugs I come across.

Creating the User Interface

User Stories

User stories were created to help me keep in mind what I want the user to be able to do and experience when using the application.

Build Process

The build process is operations that are done to the code, such as transpilation and minification, to make it more organized, readable and maintainable. **Parcel** was the build tool that was selected to handle the build process.

Client-Side App Routing

React Router was used to implement client-side routing which allows any change in state to be reflected in the URL.

Deploy

Netlify was chosen to deploy this application.

The Final Product

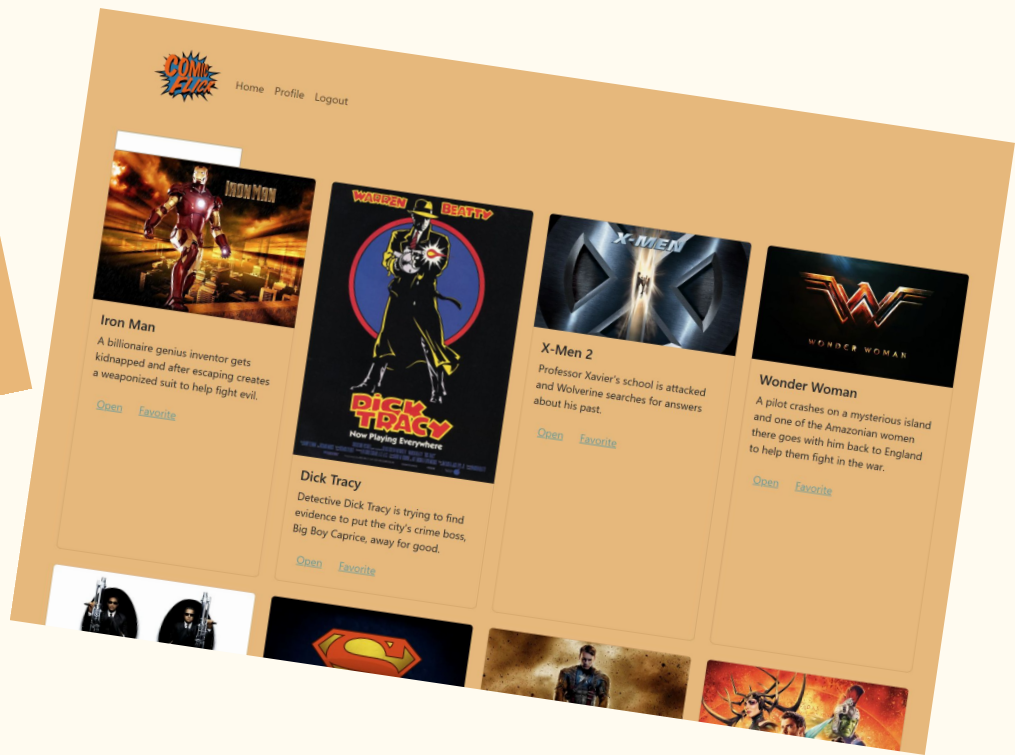


Site:

<https://comicflick.netlify.app/>

Github:

<https://github.com/Darnobles/myFlix-client>



Retrospective

The purpose of this project was to develop a full-stack application from scratch, enabling users to create profiles and search through movies in a database. I consider this project to be a success overall, but there are some aspects I would like to change, such as the design and color choices, in order to make the comic book theme stand out more.

Images provided by:
<https://Iconduck.com>
<https://www.vecteezy.com/>