

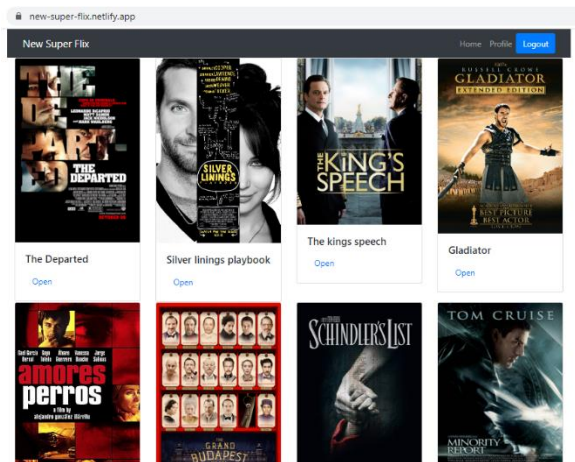
# Case of Study: Full-Stack project

## Overview

myFlix is a web app, developed using the MERN stack. It provides users with access to information to a magazine of movies. Each movie contents extra information about the director and the genre. Users are able to create and log in into an account, update their personal data and easily create a list of favorite movies.

## Purpose & Context

MyFlix is a personal project developed within the structure of my web development course at CareerFoundry to demonstrate my mastery of full-stack JavaScript development.



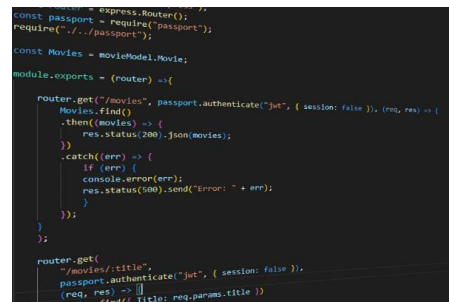
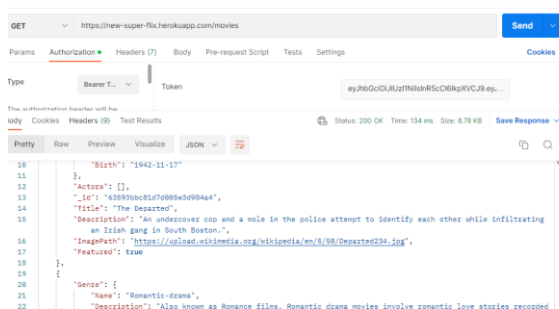
## Objective

The aim of the project was to have an ambitious full stack project that I can add to my professional portfolio, building a complete server-side and client-side application from scratch

## Approach

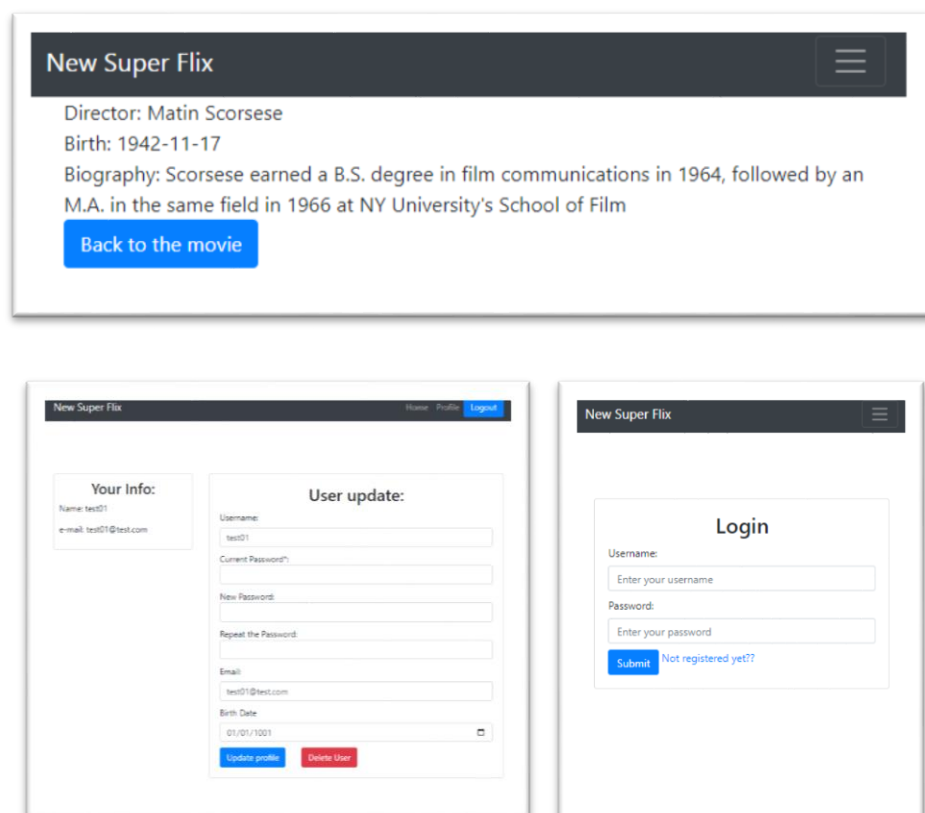
### Server-Side

I created a RESTful API using Node.js and Express. This API interacts with a non-relational database (allocated in MondoDB). The API can be accessed via commonly used HTTP methods like GET or Post. To retrieve data from and store data in the database, CRUD methods are used. The API returns the information in a JSON format.



## Client-Side

After completing the API and make sure that the endpoints are properly working, I started building the interface users would need to interact with the server-side. This application is a single-page, responsive application, developed with React. It provides several interface views, including, but not limited to, a main view (which shows a list of all movies available in the database), single movie view (returns data specific to a particular movie including a link to the director and genre information views, and also allows users to add the movie to their list of favorites), a login view, a registration view and a profile view (where users can update their data, check out their list of favorites or remove a movie from their favorites).



## Duration

The development of the client-side took me around three times as creating the server-side. On the one side I wanted to make sure I had a good grasp on how React works and achieve a desired result. On the other hand I faced some problems while uploading it to the platform that made the application perfectly work locally but not when uploaded online.

# Retrospective

## *What went well?*

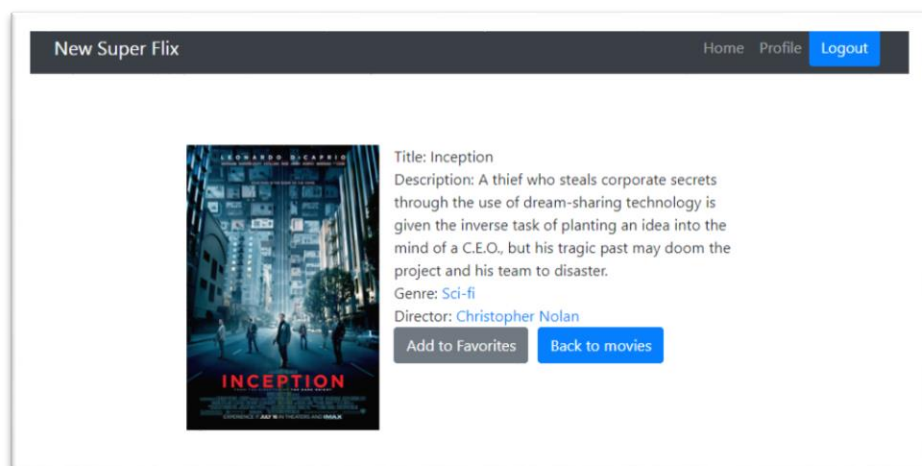
This was both my favorite and most challenging project. I have enjoyed building the API and working with the Database structures, as well as getting a full grasp of how a web works from what a user interacts with to where the information is permanently storage, going through every single step of the project while working in a topic (movies) which is one of my hobbies. Also, since this was the biggest project of the course, approximately 50% of the total time, is the project with which I got more engaged.

## *What didn't go well?*

The compatibility of the versions that I used in my local was anything but perfect, so I found multiple errors that took me a while to realize that weren't in the code itself but in them. With the help of my tutor and specially my mentor though three video call meetings (>1h) we could figure out what the problem was and eventually find a package of libraries that worked fine both locally and online.

## *Future Steps*

I would like to include some features such as not only a list of favorite movies but also a list of movies to watch, a list of movies watched and maybe feature series. Or even implement a logarithm that based on your favorites movies suggest you a tailored list instead of a list of all movies.



## *Final Thoughts*

Even though in my opinion, this is not a final product, since for that it should have a much bigger magazine of movies and the application being able to, at least save the movies that you have watched and the movies you want to watch, I am overall happy with the result of the project not only because I enjoyed working for it but also because I have got a deep understanding of the process of building a website while learning multiple different skills needed for being a web developer.

# Credits

Role: Lead Developer

Tutor: Blaise Bakundukize

Mentor: Renish Bhaskaran