

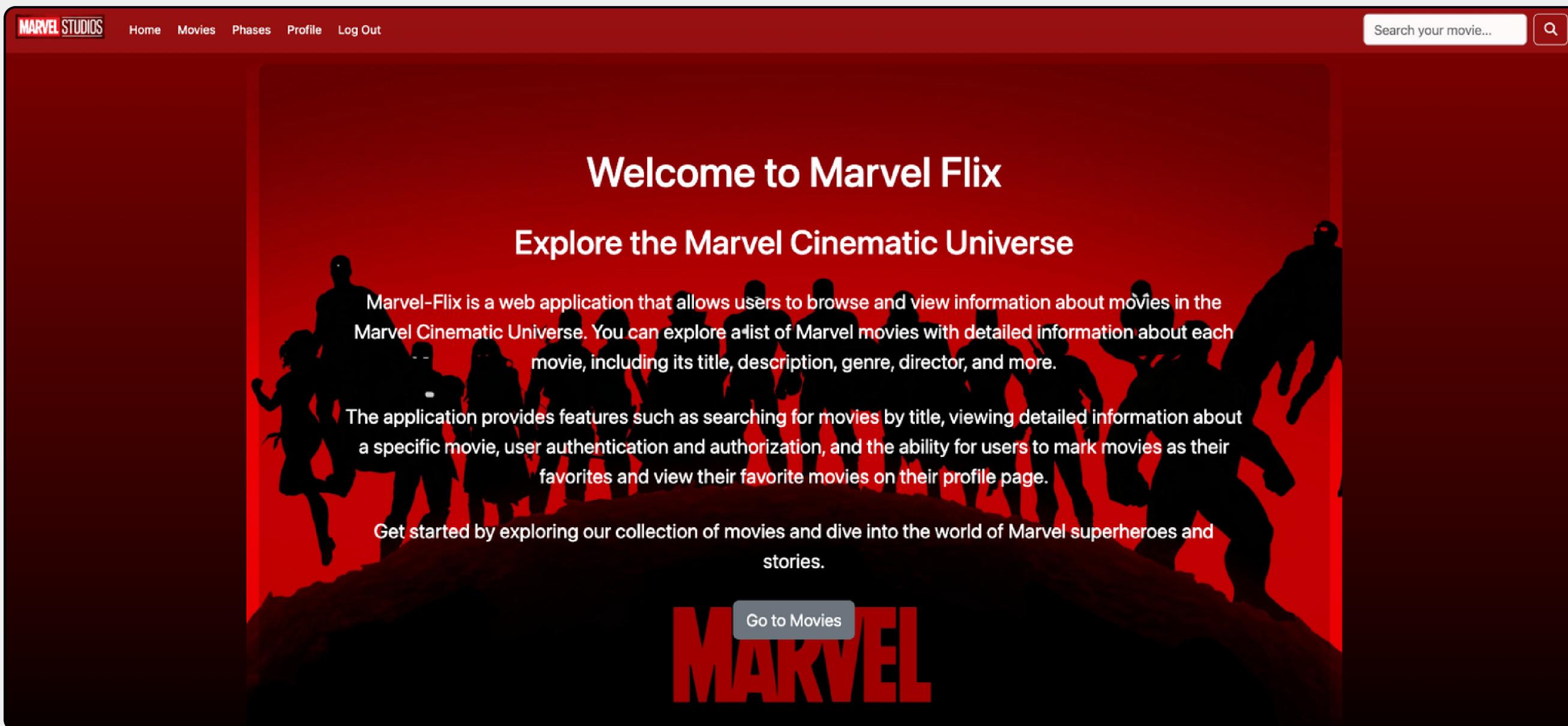
Marvel Flix Case Study

Presented by Manuel Fernando

Overview

The Marvel Flix Project is a web application developed using the MERN stack (MongoDB, Express.js, React, and Node.js). It offers users comprehensive information about superhero movies, including details about directors, Marvel's movie phases, and movie descriptions.

Users can create an account, update their personal information, and save a list of their favorite movies.

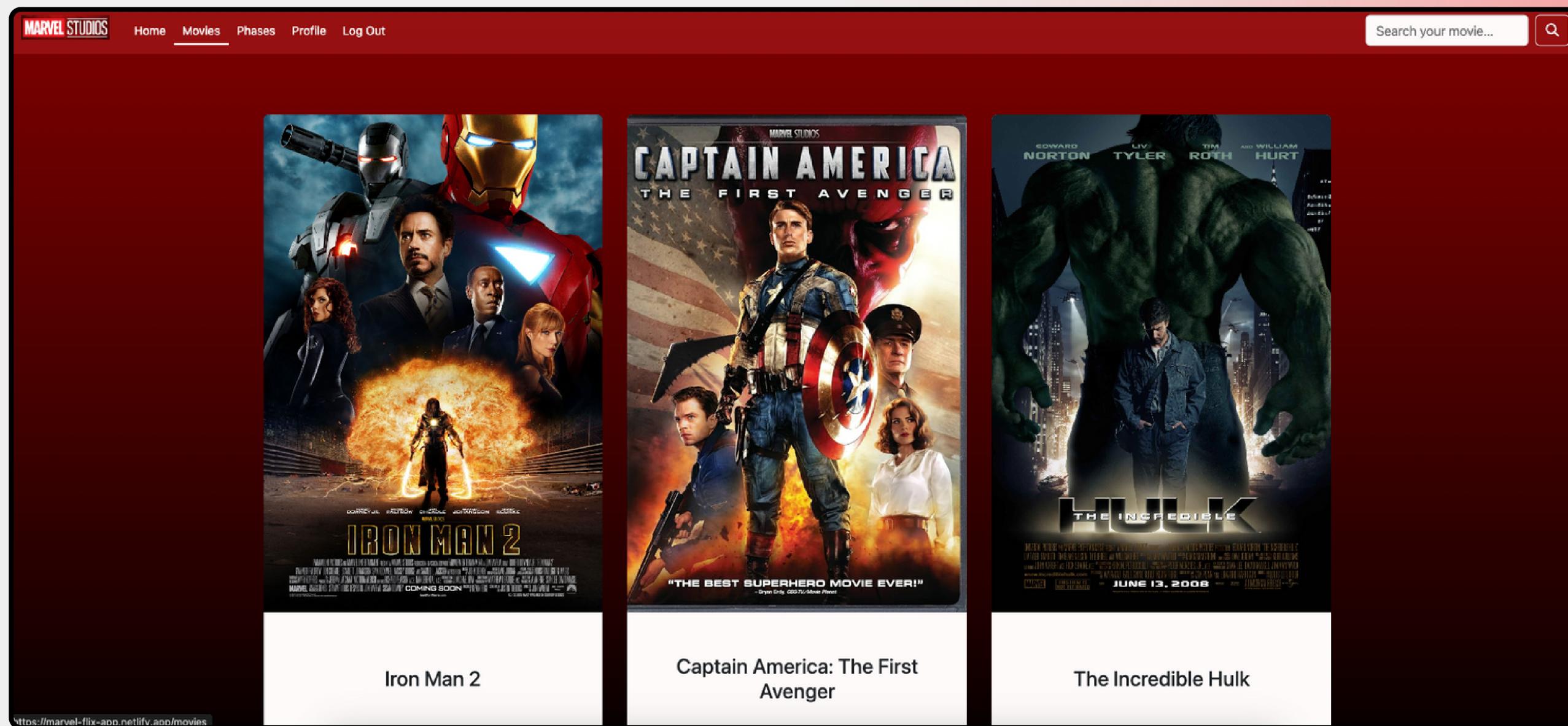


The screenshot shows the homepage of the Marvel Flix web application. At the top, there is a navigation bar with the "MARVEL STUDIOS" logo, followed by links for "Home", "Movies", "Phases", "Profile", and "Log Out". To the right of the navigation is a search bar with the placeholder "Search your movie..." and a magnifying glass icon. The main content area has a red background featuring silhouettes of various Marvel superheroes. The text "Welcome to Marvel Flix" is displayed prominently in white. Below it, the heading "Explore the Marvel Cinematic Universe" is shown. A descriptive paragraph explains the purpose of the application: "Marvel-Flix is a web application that allows users to browse and view information about movies in the Marvel Cinematic Universe. You can explore a list of Marvel movies with detailed information about each movie, including its title, description, genre, director, and more." Another paragraph describes the features: "The application provides features such as searching for movies by title, viewing detailed information about a specific movie, user authentication and authorization, and the ability for users to mark movies as their favorites and view their favorite movies on their profile page." At the bottom, a call-to-action button says "Go to Movies" and the word "MARVEL" is written in large, bold letters.

[Check it out](#)

Purpose

This project was created as part of my web development course at CareerFoundry to demonstrate proficiency in full stack JavaScript development.



Objective

The aim of this project was to build a comprehensive server-side and client-side web application from scratch to showcase in a professional portfolio.



■ Duration

Developing the client-side took approximately twice as long as the server-side. I focused on thoroughly understanding how React works to ensure the application met the desired outcome.

■ Methodologies

- MERN stack
- React
- Postman
- Bootstrap
- Heroku
- Axios

■ Credits

Lead Developer: Manuel Fernando Chavez

Tutor: Zaheer Abbas

Mentors: Ndi Selma Efkvei

MARVEL STUDIOS Home Movies Phases Profile Log Out the ave

The Avengers

Director

Name: Joss Whedon

Bio: Joss Whedon is an American producer, director, screenwriter, comic book writer, and composer.

Birth: January 1, 1964

Death: N/A

Genre

Name: Science Fiction

Description: A genre focused on imaginative and futuristic concepts often involving advanced science and technology.

 Add to Favorites

Process

Server-Side

[API code here](#)

I developed a RESTful API using Node.js and Express that interacts with a non-relational database (MongoDB).

The API can be accessed using common HTTP methods like GET and POST.

CRUD methods are implemented to retrieve and store data in the database.

The API delivers movie information in JSON format.

Basics

I first needed to determine whether to use a relational or non-relational database. After testing with PostgreSQL and MongoDB, I chose MongoDB for its flexibility as a non-relational database.

Deployment

After testing all endpoints with Postman, I deployed the app using Heroku and hosted the database on MongoDB Atlas.

Business Logic

I created models to maintain consistent data formatting and used Mongoose to interact with the database.

Security

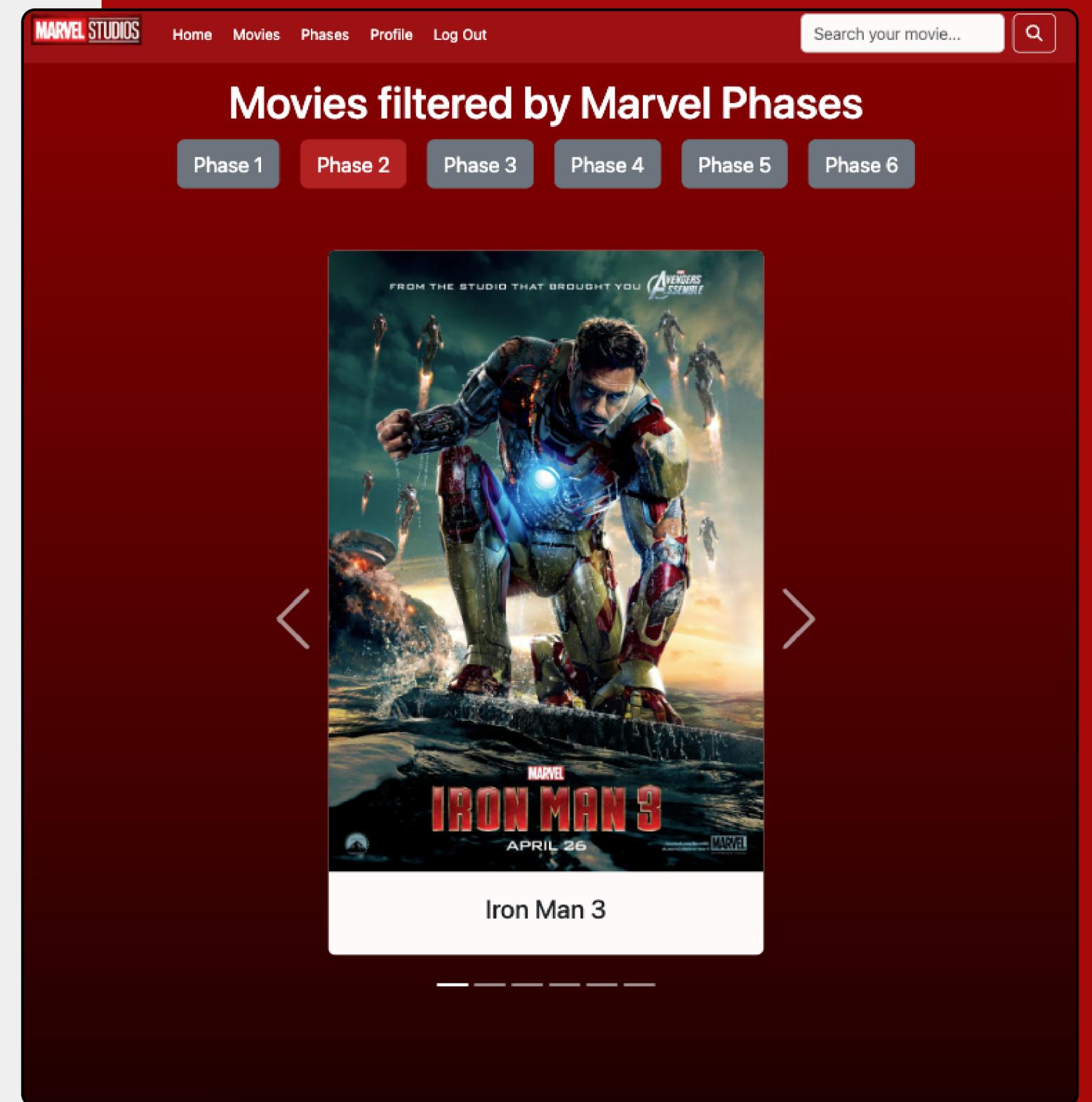
For security, I chose basic HTTP for initial login, paired with JWT token-based authorization. Additionally, I implemented CORS, password hashing, and data validation for enhanced security.



Client Side

After completing the API, I built the user interface required to interact with the server-side. The client-side is a Single Page Application (SPA), developed with React, and is fully responsive.

It features multiple interface views including a main view, movies view, login view, signup view, phases view, and a profile view where users can update their personal data and manage their list of favorite movies.



Software Project Log

■ 1. The Build

A key goal of this project was to understand and implement MVC architecture as a design pattern. I chose React for its speed, maintainability, and comprehensive documentation. To manage the build operations, I used Parcel.

■ 2. Components

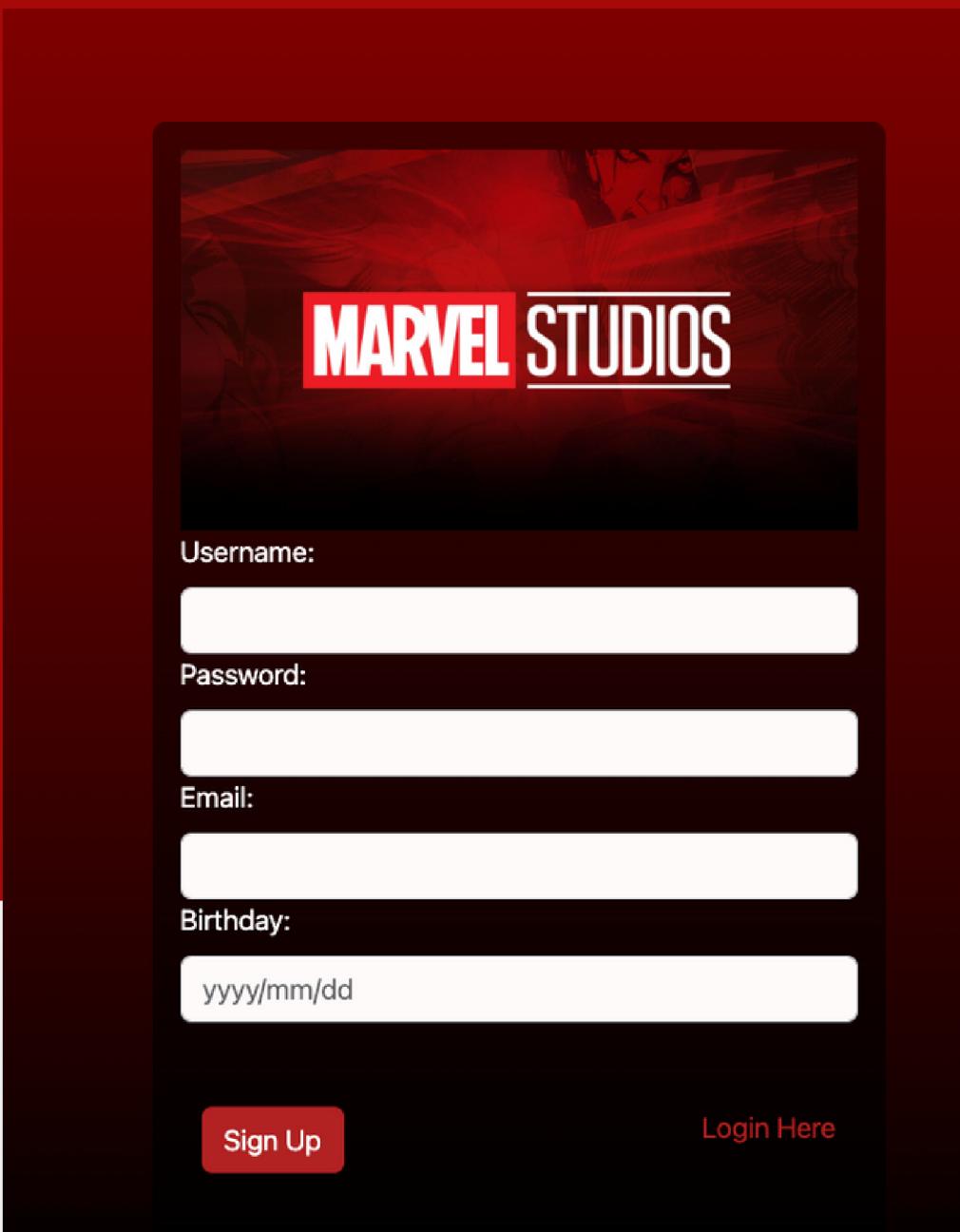
After completing the initial build, I created components for the various views and used hooks to manage the state. Axios was utilized to integrate with the API I had previously created.

■ 3. Design

I utilized React Bootstrap to design the layout of the pages and cards, ensuring consistent styling throughout the application. Additionally, I implemented client-side routing to manage authentication and control access to different views.

■ 4. Final Product

Ultimately, I implemented state management techniques to ensure the application remained scalable and efficient. Once the project was complete, I hosted it on Netlify.



The image shows the Marvel Studios login page. It features a red header with the "MARVEL STUDIOS" logo. Below the header is a form with fields for "Username", "Password", "Email", and "Birthday". At the bottom left is a "Sign Up" button, and at the bottom right is a "Login Here" link.

Username:

Password:

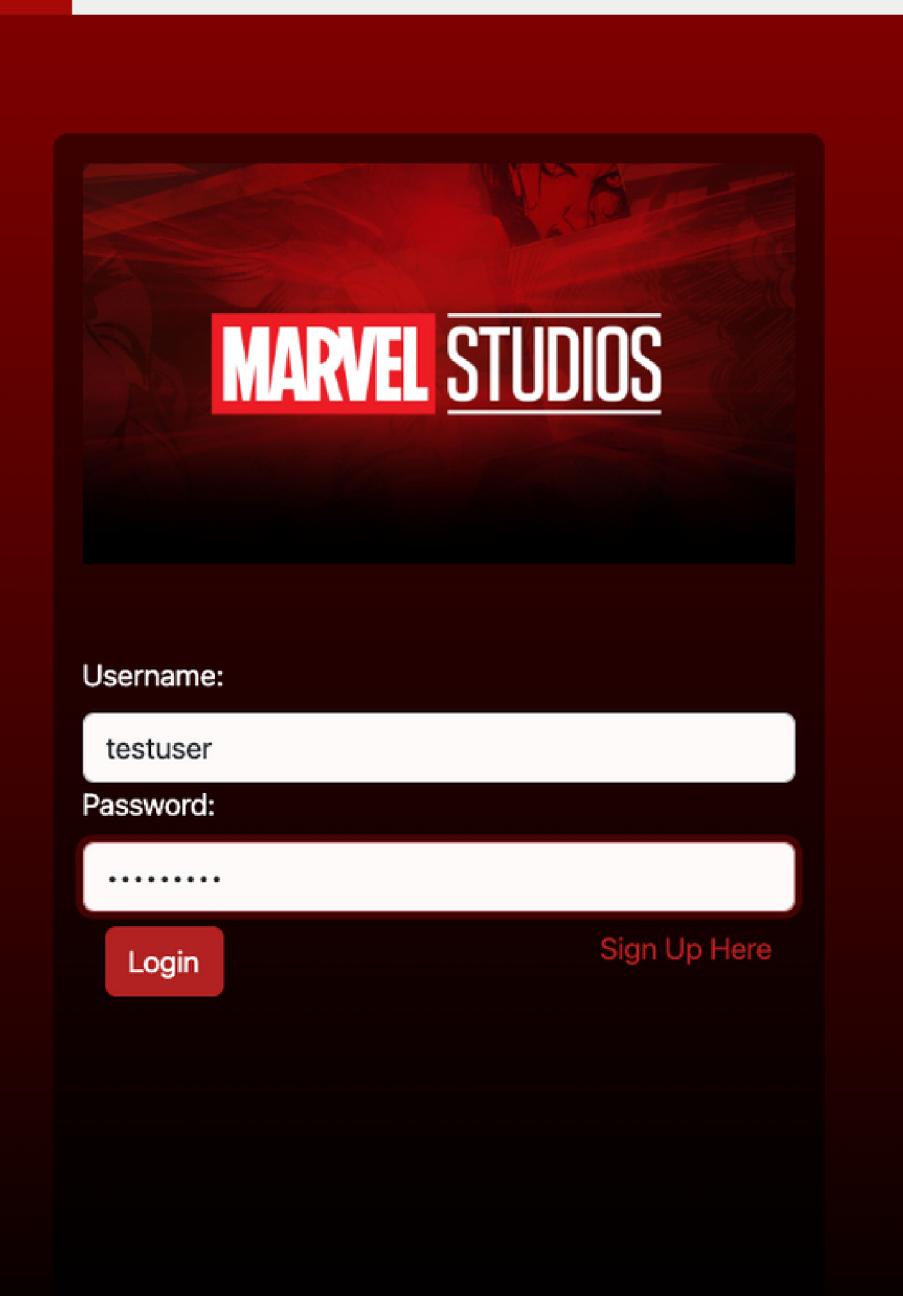
Email:

Birthday:

yyyy/mm/dd

Sign Up

Login Here



The image shows the same Marvel Studios login page as above, but with sample credentials filled in. The "Username" field contains "testuser", the "Password" field contains ".....", and the "Email" field is empty. The "Birthday" field is also empty. The "Login" button is visible at the bottom left, and the "Sign Up Here" link is at the bottom right.

Username:

testuser

Password:

.....

Login

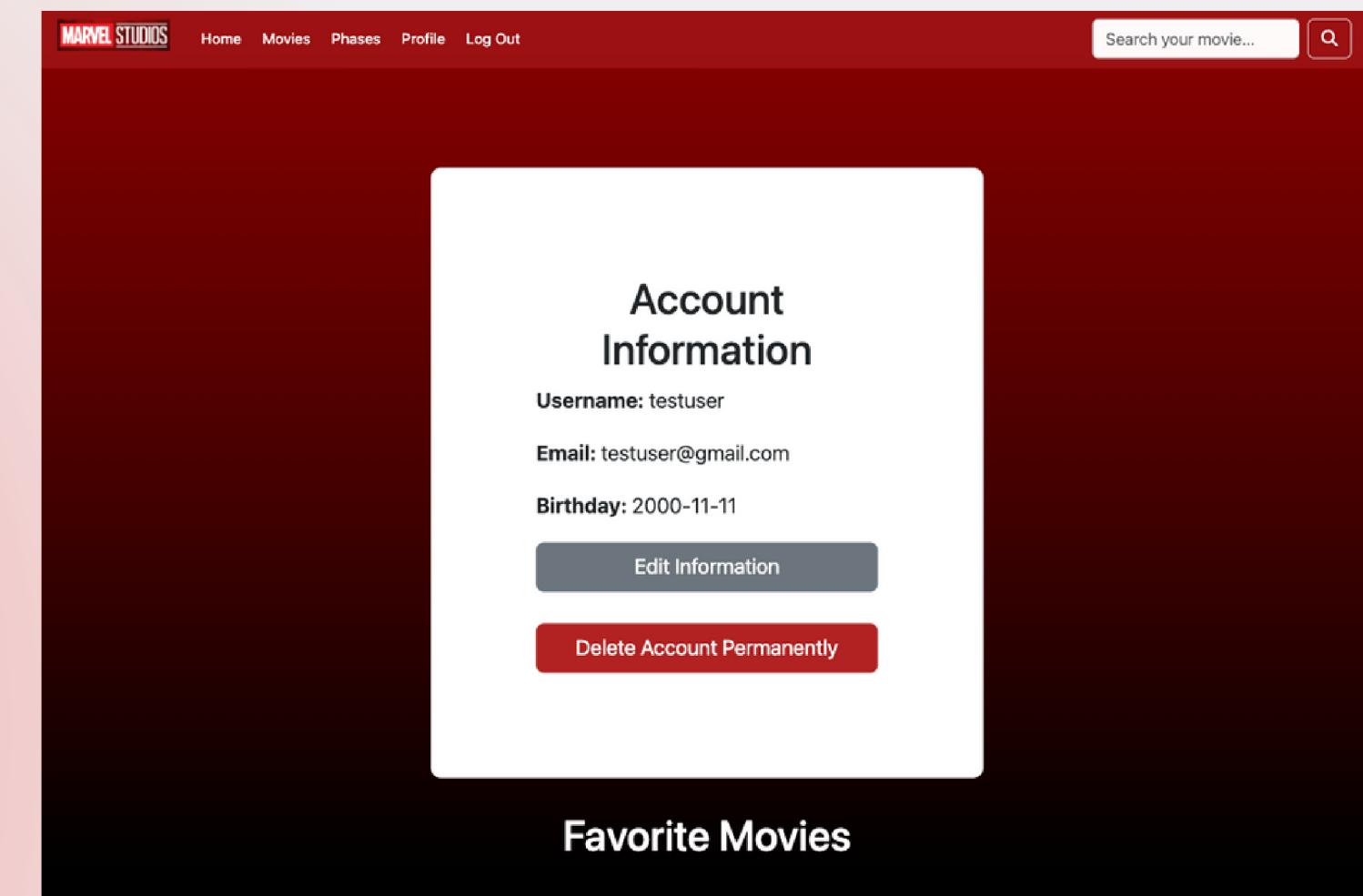
Sign Up Here

Page Views

try me
Username: testuser
Password: 123456789

Users can log in securely and access their profiles to update their information. On the Main View, they can see all available movies and search for specific titles.

By clicking on the desired movie users can view detailed descriptions of each film, including information about the superhero series it belongs to and details about the director.



The image shows the Marvel Studios account profile page. At the top, there's a navigation bar with links for "Home", "Movies", "Phases", "Profile", and "Log Out". A search bar is located at the top right. The main content area is titled "Account Information" and displays the user's details: Username: testuser, Email: testuser@gmail.com, and Birthday: 2000-11-11. There are buttons for "Edit Information" and "Delete Account Permanently". At the bottom, there's a section titled "Favorite Movies".

MARVEL STUDIOS Home Movies Phases Profile Log Out

Search your movie...

Account Information

Username: testuser

Email: testuser@gmail.com

Birthday: 2000-11-11

Edit Information

Delete Account Permanently

Favorite Movies

Retrospective

<https://marvel-flix-app.netlify.app>

What Went Well?

I thoroughly enjoyed building the database and setting up the backend. My analytical mindset appreciated the order and structure, allowing me to navigate the build process with ease.

What Didn't Go Well?

The biggest challenge I faced was time management. Balancing full-time work while learning to build this application was difficult. However, once I finished my shifts and could dedicate time to studying, everything fell into place, making the process much smoother. Taking my time also allowed me to gain a deeper understanding of React.

Future Steps

I plan to expand the database to include not only Marvel movies but also the associated series and potentially more content. Additionally, I would like to incorporate sorting features to organize movies chronologically by setting and release dates.

Final Thoughts

Overall, I am quite proud of the product I created. Initially, the connection between my database and frontend was challenging, but once I mastered it, I truly enjoyed the process and I have gained a strong understanding of building a SPA.