

CAREER*FOUNDRY*

Achievement 6 Project: Angular App (myFlix)

Objective

Using Angular, build the client-side for an application called myFlix based on its existing server-side code (REST API and database), with supporting documentation.

Context

As the modern workplace continues to transform, remote work has become more important than ever. Whether you end up working in an office or from home, it's crucial that you know how to communicate and collaborate effectively.

In this Achievement, you'll learn how to work with others by writing and sharing documentation with your peers, providing feedback, and making a contribution to the tech community. The ability to create detailed and straightforward handoffs for other developers is an indispensable skill that you'll need when applying for a job.

Opportunities to engage with these skills will be provided by your Achievement 6 project. For your project, you'll return to the myFlix app's server-side from Achievement 2 and build the client-side interface for a new single-page, responsive app, one that includes several interface views that will handle data through the previously-defined REST API endpoints.

This time you'll use Angular, instead of React. Angular is an extremely popular framework that can be used for a wide range of mobile and desktop applications. It's particularly well-suited to large projects, as well as projects requiring a complex user interface because it comes with a variety of built-in modules and services.

As you work on this project, you'll use Agile project management techniques including user stories, story points, and a kanban board. These techniques are commonly used by companies in the tech industry. At the same time, you'll practice communicating the skills and knowledge that you're acquiring with technical interview questions, before adding the project to, and finalizing, your online portfolio.

The 5 W's

- **Who**—The users of the myFlix movie app and codebase, including other developers and designers.
- **What**—A single-page, responsive movie app built with Angular, with routing and several interface views. The client-side developed in this Achievement will support the existing server-side from Achievement 2 by facilitating user requests and rendering the response from the server-side via a number of different interface views. The app will be accompanied by relevant documentation and handoff deliverables, including a kanban board containing user stories and story points.

- When—Users will be able to use the app whenever they want to read information about different movies or update their user information.
- Where—The app will be hosted online. It is responsive and can therefore be used anywhere and on any device, giving all users an equal experience.
- Why—Movie enthusiasts like to be able to access information about different movies, directors, and genres whenever they want. The app will demonstrate your Angular skills and your ability to create straightforward documentation for other developers and employers.

Design Criteria

User Stories

- As a user, I want to be able to receive information on movies, directors, and genres so that I can learn more about movies I've watched or am interested in.
- As a user, I want to be able to create a profile so I can save data about my favorite movies.

Key Features

- Your app should display a welcome view where users will be able to either log in or register an account.
- Once authenticated, the user should now view all movies.
- Upon clicking on a particular movie, users will be taken to a single movie view, where additional movie details will be displayed. The single movie view will contain the following additional features:
 - A button that when clicked takes a user to the *director view*, where details about the director of that particular movie will be displayed.
 - A button that when clicked takes a user to the *genre view*, where details about that particular genre of the movie will be displayed.

Technical Requirements

- The application must be written in Angular (version 9 or later)
- The application requires the latest version of Node.js and npm package
- The application must contain user registration and login forms
- The application must be designed using Angular Material
- The application's codebase must contain comments using Typedoc
- The project must contain technical documentation using JSDoc
- The project must be hosted on GitHub Pages

Your Project Deliverables

You'll complete your project step-by-step as you work through each Exercise. For each task, you'll submit a deliverable that directly contributes to the final product—in this case, an Angular version of the myFlix app with all the necessary documentation. Furthermore, you'll update your personal portfolio website that you'll need to apply for a job.

Below is a breakdown of your course project deliverables by Exercise:

Exercise 1: Collaboration and Project Management

- Set up a kanban board that you'll use as you work on this Achievement's project
- Create a user flow for your project
- Practice answering non-technical interview questions

Throughout this Achievement, you'll be asked to keep your kanban board up to date. Make sure to work with your kanban board for each task, and take some time to reflect on your project planning and assumptions. Were your tasks well broken down and your story points accurate? If not, what could you improve? By doing this, you can practice your various Agile skills.

Exercise 2: Introduction to Angular

- Create an app using Angular CLI
- Practice answering technical interview questions

Exercise 3: Advanced Angular Part 1

- Implement user registration and login forms using Angular Material
- Practice answering technical interview questions

Exercise 4: Advanced Angular Part 2

- Implement a movie card component in your app
- Define the routing in your app
- Practice answering technical interview questions

Exercise 5: Documentation & Handoffs

- Comment on the app's codebase
- Prepare all the necessary documentation for the app
- Finalize kanban board

Exercise 6: Providing Constructive Feedback

- Review a given screen design and provide constructive feedback

- Finalize portfolio preparation
- Practice answering non-technical and technical interview questions

Exercise 7: Contributing to the Tech Community & Beyond

- Join online and offline developer communities
- Incorporate your projects into your portfolio site