

Q 1. You've been tasked with creating a new version of an application used by product Administrators to manage settings and display data about user activity and Information. The current version was written using jQuery but grew organically, and It's become hard to maintain and add new features to it. It's a medium-sized Application, with multiple views displaying a lot of data and forms.

Answer 1. I would like to use React to maintain and scale my app for some reasons:

- 1) Its declarative: Code written in React is easy to read and, therefore, easy to maintain.
- 2) Its component-based: To build with React, I create reusable components. This allows me to increase the scope of my web application with little to no performance issues (its super-fast!) or concerns about entropy.
- 3) It works well with data: The data I display to my users can change over time without having to reload the page.
- 4) it's compatible with other tools: For example, React Native, a framework for building native apps using React, and Redux, a library for managing application states.
- 5) It has a wide user base and plenty of online resources.
- 6) Its built, used, and supported by Facebook: React's documentation is well-maintained, and I won't have to wait several weeks for bug fixes!
- 7) Its relatively small.
- 8) the most important thing is that is a user friendly, easy to use and readable with an excellent documentation with many external developers.

Q 2: You've been asked to build a new piece of your company's web application. The Existing application consists of several sub-applications served independently under Different routes by an Express server, which allows for different technologies and Frameworks to cohabit. You'll be building what is essentially a huge table that will display hundreds of thousands of data records.

Answer 2. For this Project I would like to use React because:

React project is built using the so-called reusable components. That means each element of the interface that I've already built can be used anywhere in my Project by just calling from other components. I have created such a function once and can reuse it anywhere. What's more, I can group these elements into sections or pages and Insert them anywhere around the Project with the same functions.

Q 3: Your designers have tasked you with creating a prototype for a standalone application to be used internally by your product team. The application will showcase the product team's design system (colours, components, fonts, styling, etc.)

I will not use any framework. Last time I worked with Bootstrap, but I did not like it. Because of the colour combination, design sometimes gave me (eye) error. I used CSS myself. I think for any project, a simple HTML and CSS generated by a backend and a bit of Vanilla JavaScript might be enough to create a prototype. Maybe I will use the library instead since it's easier to replace once a library with another and framework once used will determine how an entire application will be built on the prototype stage. It might not be the best choice to make any decision earlier. Actually, the whole project work should depend on the situations and the user's interests.

Q 4: Discuss why React is the most suitable framework for the client-side Component of your myFlix application.

React is the most suitable framework for the myFlix application because:

- 1) It is a component-based framework that allows functionalities to be added structural way.
- 2) React is relatively small, which is suitable for the movie data because the movie database has the potential to grow to a significant size, and has a bulk framework which would just add weight to an already heavy application.
- 3) React is easy to read and maintain; that is essential because the myFlix will most likely need to be updated to add new movies that have come out.
- 4) React is the right choice because it works well with the data, especially in an iterative way where data viewed by the user can be changed without reloading.