

TOPICS FOR REACT.JS TECHNICAL INTERVIEWS

Topics to cover during Tech Interview	Questions	Expected Answers per Seniority	Seniority
Version Control Systems Knowledge	What's the difference between a merge and a rebase?	Should have a good understanding of feature branching and how merge works. The candidate understands that rebase is just another way of integrating changes but is not able to explain the differences.	Junior
React Hooks	Which are the built-in React hooks? Briefly explain how each one is used and for what purpose.	Only mentions useEffect and useState . Doesn't fully understand the dependency array (or he/she leaves it empty, without fully understanding why and ignoring eslint rules)	Junior
Stage Management	What approach does he/she use for state management inside an app? Does he/she use global state? In which cases would they use one tool instead of the other?	They will mention the use of only one tool, probably Context , Redux , or Mobx . They won't be able to justify why they use it, a common answer might be <i>"I use this because that's what my company chooses."</i>	Junior
Testing	Do you have any experience with automated testing?	Little to non-experience in this subject. They might understand what it implies based on courses or articles they might have read, but they lack experience executing this type of task.	Junior
Communication Skills with client	A new feature has been added to the app. It's working correctly but the client expresses there are several performance issues. How would you proceed in order to solve the problem?	They will probably jump to quick conclusions and try to apply small but not focused fixes to the feature, without trying to deep dive with the client in order to have a better understanding of what is perceived as a "performance issue". They might suggest the implementation of certain techniques such as useCallback or useMemo in order to avoid any unnecessary renders. Nevertheless, without a proper analysis, this would only make matters worse, causing an increase in memory footprint.	Junior
Design Patterns	Wich design patterns are you familiar with?	They will probably lack any knowledge about design patterns. Maybe they've read about them but they can not elaborate on them and they don't understand their relevance.	Junior
Version Control Systems Knowledge	What's the difference between a merge and a rebase?	Should have a good understanding of git flows , and know that rebase rewrites commit history. The candidate should be able to provide examples of how code reviews and merge process works for promoting code to different environments.	Middle
React Hooks	Which are the built-in React hooks? Briefly explain how each one is used and for what purpose.	Mentions useCallback , useMemo , useRef and UseContext . They know in which cases they should apply them and understand useRef and useContext .	Middle
Stage Management	What approach does he/she use for state management inside an app? Does he/she use global state? In which cases would they use one tool instead of the other?	Will provide a good comparison between Redux and Context . - Redux allows you to manage your app's state in a single place and keep changes in your app more predictable and traceable. It also has dev tools that ease the development, along with middlewares that have utilities which allow extending its functionality. - Context is much more flexible than Redux. It allows exporting global functions that affect the behavior of a certain feature. It can work on an isolated feature/module level, and that provider can be reused as well.	Middle
Testing	Do you have any experience with automated testing?	Experience with unit testing . Probably will have experience with jest , enzyme , react-testing-library . They are able to test components, functions, and classes on their own.	Middle
Communication Skills with client	A new feature has been added to the app. It's working correctly but the client expresses there are several performance issues. How would you proceed in order to solve the problem?	They will probably suggest inquiring with the client about what are the perceived performance issues. Once this is checked, they will apply a solution that will involve cache and memorization.	Middle
Design Patterns	Wich design patterns are you familiar with?	Knows some design pattern and their definitions. Will probably have some theoretical knowledge but will lack practical experience applying them. They might use patterns without knowing them by name. They might also confuse design pattern concepts with architectures such as MCS , MVP , Clean Architecture , and some associated principles such as SOLID . In these cases, as interviewers, we should remark on the difference between these concepts and give the candidate a chance to answer or deep dive into architecture if he desires. It's acceptable as interviewers to mention some common Design Patterns such as Singleton , Factory Method , Decorator , or Strategy in order to help the candidate elaborate on their answer. CQRS knowledge is always valued.	Middle
Version Control Systems Knowledge	What's the difference between a merge and a rebase?	Rebase changes the point at which I branched to move to a new starting point by reapplying commits on top of another base branch, whereas Merge joins two or more development histories together. The candidate should be able to provide examples and a reasoned opinion on when to use either (<i>i.e., It depends on what is most important - a tidy history or a true representation of the sequence of development</i>). They should also be capable of explaining cherry-picking strategies, and squashing commits for building more meaningful commits/PRs.	Senior &+

TOPICS FOR REACT.JS TECHNICAL INTERVIEWS

Topics to cover during Tech Interview	Questions	Expected Answers per Seniority	Seniority
React Hooks	Which are the built-in React hooks? Briefly explain how each one is used and for what purpose.	Has a better understanding of when to use useCallback and useMemo . They know that they should not be used in any situation carelessly, we should evaluate the context, and do relevant benchmarking/profiling afterward. Another good indicator is knowing how to explain useImperativeHandle , forwardRef , Suspense API .	Senior &+
State Management	What approach does he/she use for state management inside an app? Does he/she use global state? In which cases would they use one tool instead of the other?	Will understand that these tools can be combined according to the necessity. In a large team or with microfrontends, redux can be a good convention in order to manage the global state of an app. They will understand that redux will not solve all of the problems, and many situations can be solved with libraries such as react-query , swr , or redux toolkit without introducing any accidental complexity. They can explain that in order to solve immutability issues, libraries such as immerjs or immutablejs can be used.	Senior &+
Testing	Do you have any experience with automated testing?	Has a more holistic comprehension of the different approaches to automated testing. They can distinguish between static , unit , integration , and e2e testing , and what each of them is used for. They know how to mock imports and globals, use spies, re-use tests and make them scalable. If they understand the importance of every type of automated testing, they will be able to explain the advantages of using unit testing (it allows to quickly verify the code functionality in the minimum unit), integration (it allows to verify the interactions between the various components of an application), and e2e (it allows to identify if a feature from an app is working when the end-user uses it.)	Senior &+
Communication Skills with client	A new feature has been added to the app. It's working correctly but the client expresses there are several performance issues. How would you proceed in order to solve the problem?	The candidate must propose working collaboratively with the client to identify the problem, replicate the issues, and define concrete expectations about a potential solution. They will be able to conduct an analysis of the execution time for the tasks using profiling measurements and tools such as React Devtools in order to understand if there are any unnecessary re-renders, and if so, why. After applying optimizations such as useMemo , useCallback , caching , avoiding any re-renders, and loading components using lazy-loading ; they will check if the processing time for scripts has improved, the memory footprint decreased, and if there is a perceived increase in the user experience or not. Before integrating these changes, it must be concluded if these updates increased the performance or if they have to be discarded. The purpose of this question is not only to assess the technical knowledge regarding benchmarking and optimization techniques but also what communication and collaboration with the stakeholders look like.	Senior &+
Design Patterns	Which design patterns are you familiar with?	Knows lots of Design Patterns and has relevant practical experience applying them. They can reason problems with help of patterns and explain them to other devs. They know the difference between patterns and architectures, and how they are related (e.g., MVC , Redux/Flux , and how they relate with Observer or Pub/Sub). They have to understand the advantages of each of the patterns and in which cases they should be used (Code reuse , resilience , control over data flows , etc).	Senior &+