React Developer
Recruitment Questionnaire

Soft questions for a start

1. How and why did you start your journey with front-end / JavaScript?
2. On which frameworks or libraries you worked already?
3. How do you keep up with the news in industry?
4. What are you planning to learn in the near future?
5. Which technologies excites you the most?
6. What kind of project you have done lately? What was your role?
7. How would you describe your level as a developer? Junior / Regular / Senior? *(this question can help you filter technical questions)*

Working methodology / approach to programming

1. What do you think about writing tests?
    a. Tests: did you wrote them? If yes, what types? Which libraries did you use?
2. Are you trying to use some design patterns in your work?
    a. If yes, which ones?
3. Did you ever do pair programming?
4. Did you work with code review? What good practices of code review you know?
5. Did you already worked with project where english language was mandatory?
    a. How do you fell in those projects?
    b. Did you ever work with clients from another time zone?

CSS / SCSS / HTML Questions

1. Did you ever converted web design into website without using third party libraries?
2. What is processor, which ones do you know?

3. What’s the difference between SASS/LESS and CSS? What new processors can add to CSS?
4. What do you do to check browser support for a specific CSS rule (CANIUSE)?
5. Did you use flex/grid?
6. SASS/LESS:
    a. What is mixin?
    b. Did you use include, exclude, what’s the difference?
7. Do you know any methods to name classes (BEM)?
8. Do you know what is mobile first approach?
9. Which CSS frameworks you know? What do you think of using them?
10. Did you ever have anything to do with SVG?
11. How can you pass different styles for different browsers?
12. How can you detect if a specific browser supports specific functions? (Modernizer)
13. Do you know ITCSS or other CSS code architecture? If no, how did you sort CSS/SASS/LESS files, how do you sort the folders to make yourself code keeping easier?

JS questions

1. Difference `let`, `const`, `var`
2. Strict Equality Comparison vs Abstract Eqality Comparision in JS
3. Do you know any methods to save things on the client side, so it can be still available after closing it?
    a. Difference between localstorage and cookies
4. What “this” points in js?
5. Difference: “arrow function”, and function
6. Which array methods you know? forEach, map, filter, reduce
7. Difference: undefined, and not defined
8. What’s hoisting

```
console.log(a);
var a = 1;

console.log(b);
let b = 2;

console.log(b);
const b = 2;
```

9. What’s closure?
10. What’s IIFE (Immediately Invoked Function Expression) and how can you use it?

```
for (var i = 0; i < 3; i++) {
  setTimeout(function() {
     console.log(i);
  }, i * 1000)
}
```

11. How to setup the default parameter in ES6, and how to implement default parameter in ES5?
12. Spread operator, rest parameters (If he doesn't know anything, it's a lost case)
13. Destructuring
14. What's Promise, do you know async await syntax?
15. Difference between `Call, Apply, Bind`
16. String template
17. What's immutable?
18. How to block the object modification in JS?
19. Functional programming
20. What's ServiceWorker?
21. What are the basic types in JS
22. Event loop, how does it work?

Tools / automation

1. What's npm, yarn, what do you use it for?
2. What package.json file contains? What is package-lock.json for?
    a. What's the difference between devDependencies and dependencies?
3. Did you ever use/configure webpack, gulp?
4. Do you know what linter is? code linting? did you ever use linter?

React Questions

1. Compare React with any other framework you know.
2. Differentiate between Real DOM and Virtual DOM.
    a. What do you understand by Virtual DOM? Explain its working.
3. What is JSX?
4. Lifecycle methods
5. In which lifecycle event do you make AJAX requests and why?
6. Component state vs component props?
7. How do you change state in React?
8. What do you use to manage the state?
    a. What are the differences between Redux and Mobx?

9. How do you start writing a new project in React? (create react app?)
10. Which libraries you import to your project when you start?
11. What are the differences between functional components and class components?
12. What are Higher Order Components (HOC)?
13. How setState in React Works, is it synch or async?
14. Have you've been using Pure Components?
15. What are refs in React and why are they important?
16. What are keys in React and why are they important?
17. What is the difference between a controlled component and an uncontrolled component?
18. How do you tell React to build in Production mode and what will that do?
19. Which methods of styling apps in React do you know?
20. What is React Router?
21. Why is switch keyword used in React Router v4?

Redux / State Management

1. Explain Flux.
2. What is Redux?
3. List down the components of Redux. (Action, reducer...)
4. How are Actions defined in Redux?
5. What differences you have between Redux & MobX?

GIT questions

1. How can you describe your GIT knowledge level?
2. Do you know what's git flow?
3. Did you practice code review? How? Did you download branch, or in PR, or your friend PC (😂)?
4. Difference between merge and rebase?
5. What's cherry pick?

TypeScript

1. What TypeScript is used for? What are the advantages of using it?
2. What are differences between TS and Flow?

Additional questions

1. How to block default <form /> saving behaviour?
2. Did you work with <canvas /> object? If yes, which libraries did you use?
3. List basic HTTP methods, difference between POST, PUT, PATCH
4. Which design patterns you know?
5. Do you know what's Serverless?
6. Did you work with NodeJS?
7. Did you use debugger?
8. Did you work in Scrum / Agile?
9. Do you have any experience in public presentations? Are you able to give a presentation in the future?
10. How can you keep the app safe on the frontend side?