



Frontend coding challenge

The Task

Develop a dynamic online survey. The survey should consist of multiple pages.

A wireframe of a survey form. At the top, there is a horizontal progress bar with a grey segment on the left and a white segment on the right. Below the progress bar, the text 'Question: How old are you?' is centered. Underneath the text is a rectangular input field containing the number '23'. At the bottom of the form, there are two buttons: 'back' on the left and 'next' on the right. The 'next' button is highlighted with a grey background.

Each page includes...

- one question
- one of the following input types to answer the question
 - Text input field
 - Dropdown (several options, one selectable)
 - Radio buttons (multiple inputs, one selectable)
- One back button to go to the previous page of the journey (if not on first page)
- One next button to go to the next page of the journey (if not on summary / last page)
 - The button should be disabled if no input is given yet for the current page
- The last page shows a summary of all questions with answers (no wireframe included, just make something up)
- The survey can be browsed using the native browser back / next button
- **BONUS:** A progress indicator bar
- **BONUS:** When closing the browser window and reopening it, the progress with its data is restored

Implementation & hints

- Use data structures to create a dynamic user journey and make it easy to add / remove / edit pages via code
- Write plain, modern JavaScript (ES6 / ES7) or TypeScript
 - You are allowed to use libraries and frameworks to build web apps like React.js, Angular.js, Vue.js or Ember.js and libraries for state management.

- Do not use libraries explicitly build for the various features of the survey.
(progress bar, form elements, etc.)
- Structure CSS(SCSS, LESS, STYLUS...) with an eye on scalability & maintainability
- Think about UX & a11ys
- Please create a repository and deliver your code on Github
(**BONUS** if you make just regular commits as in your usual workflow instead of just pushing everything when you are finished)
- **have fun!**

Evaluation Criteria

Our goal is to find answers to the following questions and to get a sense for how you write code and solve problems.

- **UX / UI:** Does the survey adapt well to different devices (mobile / desktop) in terms of user experience and design.
- **Code Quality:** Is the code clean, consistent, well-structured and easy to understand? Do you demonstrate a good understanding of ES6 & semantic HTML?
- **Performance:** Is the app running smoothly? How does it work with a slow internet connection? Does the UI render quickly? Are the choice of libraries etc. appropriate for the given task?
- **BONUS:** Are code quality tools in place? Did you write tests?

Asking questions is good. We are happy to help you and will not penalize you for asking questions. In an actual work environment, we prefer that engineers spend 30 minutes talking a problem over with someone than spend an entire day building the wrong thing.

You do not need to accomplish everything, but obviously, the more the better.