



Accessible Tabs Code Challenge

The Accessible Tabs coding challenge is described in the following pages. Please review and complete an implementation that fulfils the requirements. Please take as much time as you need to complete the challenge. We would prefer you take the time to complete a solution you are happy with.

We use these submissions to help us understand your abilities and provide a discussion point should you be successful in progressing to the next stage. We are specifically interested in how you approach the problem in code. We like to see elegant solutions and clean code. We are not looking for elaborate patterns, architectures that go beyond what is necessary to solve this problem. Please minimise your use of 3rd party libraries or frameworks to those that offer convenience only. We are only interested in your skill in solving the problem, not your ability to use a framework.

Please complete the solution in a language relevant for the role being applied for. You will be advised when receiving the challenge by our hiring manager.

What to include in your submission

- Link to GitHub repository, or zip of source code
- Appropriate documentation such as a readme (it should be clear how to setup and run your app)
- Appropriate unit and/or integration tests included with your source code (these should be runnable)

Description and requirements:

The goal is to create a tab group fully compliant with the WAI ARIA specification

(<https://www.w3.org/TR/wai-aria-practices-1.1/#tabpanel>). The solution should be able to handle multiple tab groups rendered on a single page. Active tabs should retain their state in a similar manner to native HTML element including back/forward navigation and refresh. It should be possible to directly link to any given tab on a page.

The tab list should be fully operational in Chrome, Firefox, and Safari or Edge - ideally both.