Full Stack Developer Challenge

Scenario:

nib is revamping their Careers page to attract new employees and allow prospective employees to find the roles which they wish to apply for. The designer in your team has provided a design for the Careers page with variations at mobile, tablet and desktop resolutions.

As the team's developer you are required to implement the page. (Feel free to use any frameworks or tools you're comfortable with).

Challenge constraints:

- The jobs must be filtered and sorted on the server (imagine it's a large database of jobs, but in reality, it's a small JSON file. You can define the format of the JSON file).
- The jobs must be fetched via AJAX on the client.
- Location identifiers are provided in the json file and must be joined to data retrieved from the following locations API https://private-8dbaa-nibdevchallenge.apiary-mock.com/location. Documentation for the API can be found here: https://swaggerhub.com/apis/nib-dev-challenge/Locations/1.0.0.
- Location data must be retrieved **on the server**
- The JSON file must not be changed on disk.
- Display only the first 130 chars of the job description.
- A user will filter the results by selecting a location, e.g. Sydney.
- Then a user will select a job and be taken to a full job description page.
- When a user clicks back in their browser, they will return to the previous screen. The previous screen will have the same filter option selected eg. "Sydney".
- You will be required to commit to a GitHub git repository (This will be your personal GitHub repository) periodically throughout the challenge.
- Usage of boilerplate templates is **not** allowed.
- The server must be written in .net or .net core.
- Write at least one automated test either for the server or the client.

You will have three days to complete this challenge. You don't need to use the exact fonts, colours, sizes, images etc. We're looking for:

- Responsive layout.

- Experience with JavaScript on the client and server.
- Efficiency in API calls
- Error handling
- Maintaining of state across pages.