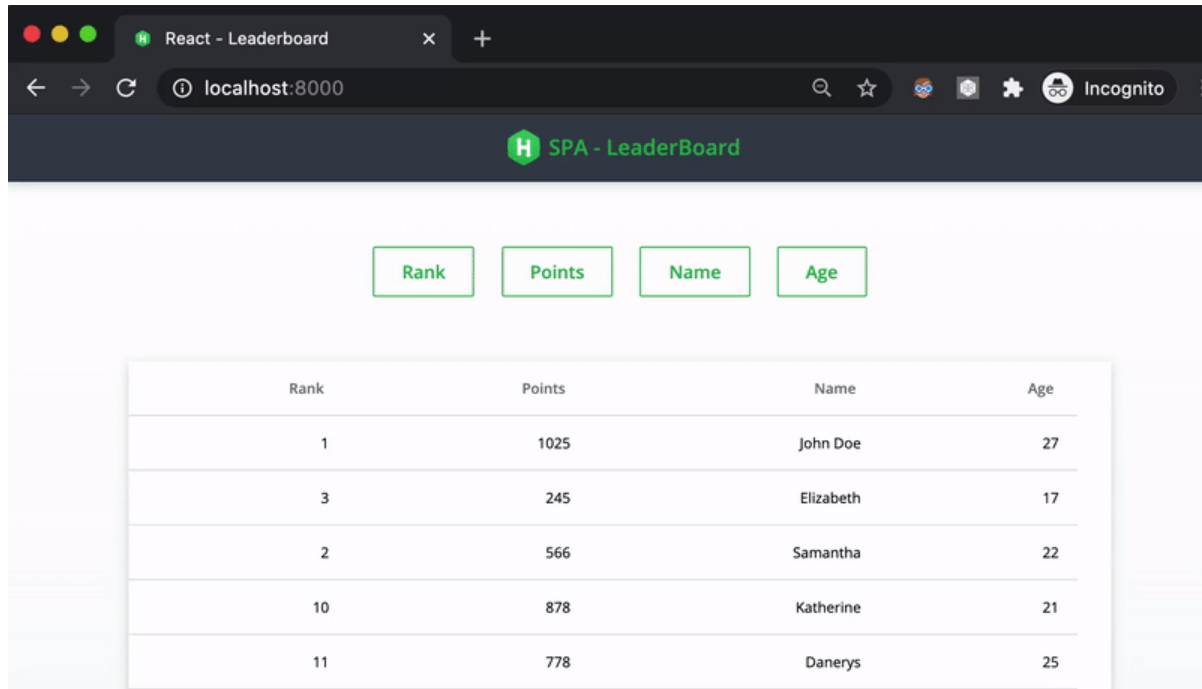


Create a basic sorting-based Single Page Application, as shown below. The HTML skeleton is already in place, but the application is not complete. Application requirements are given below, and the finished application must pass all of the unit tests.



The app has one component: the *LeaderBoard* view. The data to be filled into the table is stored in a variable called *response* inside the *response.js* file.

The response has the following properties:

- *name*: the name of the user [*String*]
- *age*: the age of the user [*String*]
- *points*: the points accumulated by the user [*String*]
- *rank*: the rank of the user [*String*]

The structure of the response is as follows:

```
{
  list: [
    {
      rank: "1"
```

```

      name: "John Doe"
      points: "100"
      age: "25"
    },
    { ... }
  ]
}

```

The app should implement the following functionalities:

- Initially, the table should be rendered with all 4 buttons, and the table contents should be filled in the order of how the data appears in the response when the route is "/".
- Each button click should navigate the user to the correct URL as given below:

○

<i>BUTTONS</i>	<i>Corresponding URL</i>
Rank	'/rank'
Points	'/points'
Name	'/name'
Age	'/age'

- It should sort the specified contents in the table in ascending order based on the type of button that is clicked: Rank, Points, Name, or Age.
- Directly landing on that URL should sort the table based on the URL path (e.g., "/rank" should sort by rank on load).

Please also note the following:

- The list of all user info to be filled into the table is present inside the *response.js* file.
- The application needs to be implemented like a Single Page Application (SPA), involving URL changes using React Router—not through the plain anchor tag that reloads the page.
- You can refer to the React Router API here: [React Router Reference](#)

The following data-testid attributes are required in the component for the tests to pass:

- The <table> should have 'app-table' and the <tbody> should have 'app-body' as data-testid .
- The Rank, Name, Points, and Age buttons must have 'route-rank', 'route-name', 'route-points', and 'route-age' as their data-testids respectively.
- Each <td> in the Rank column should have the data-testid in the following order: 'rank-0' for the first row, 'rank-1' for the second, and so on.
- Similarly, each <td> in the columns Name ('name-0', 'name-1', ...), Age ('age-0', 'age-1', ...) and Points ('points-0', 'points-1', ...) should also have its associated data-testids.

Please note that the component has the above data-testid attributes for test cases. It is advised not to change them.