Wemolo - Frontend Challenge

Welcome to the Wemolo frontend coding challenge! We are excited that you have made it this far and we hope you enjoy working on our small challenge.
Feel free to reach out for any inquiries in case something is not clear enough.
Good luck .. !

Scenario:

When joining Wemolo you must get used to one thing: parking lots. They come in different shapes and sizes. Some of them are dark and dusty, others shine with architectural exceptionalism. True Wemoloians can evaluate the inner beauty of a lot from far distance. You seem to thrive to become one, so we suggest that you create a training tool for yourself. Tinder has turned out to be quite effective at rating beauty in a superficial manner, so you should create a UI where you can click "right" to label a parking lot as outstanding and "left" to add it to the list of avoidable places.
Finally, your UI should allow you to review your choices from the tinder session. Maybe there was one nice parking lot, you want to keep in mind for future visits…

Instructions:

- Implement a tinder-like UI that fetches the list of parking lots from our GraphQL API.
- Use ReactJs as the rendering framework -> client-side rendered.
- Provide type-safety using TypeScript
- Feel free to use a UI component library of your choice if you want to. Examples of such libraries include: Base Web, Ant Design, Material UI, etc.
- While we appreciate a good and responsive design, we urge you to keep it simple and focus on providing a clean, functional code.
- We would like to see how you ensure that your code works using a test framework of your choice. Add 1-2 effective tests that bring value

Features:

- Tinder View
  - The user can see metadata and the image of one parking lot at a time.
  - The user can label the beauty of a parking lot with "good" or "bad"
  - After any decision, the next parking lot will be loaded
  - The UI never fetches more than 5 cases in advance (you can use `limit` and `offset` query variables for pagination)
  - The user can stop tindering at any time and switch to the summary view
- Summary View
  - The user can see the good and bad lots in a compact fashion (e.g. list view) from this one UI session.
  - The results are grouped / sorted / filterable for better overview

Submission:

- Usually, the task takes 3-5 hours to complete. However, we took into consideration that you may have other commitments.
- We kindly ask that you push your code to a private git repository and send us a link as well as an access token to it.