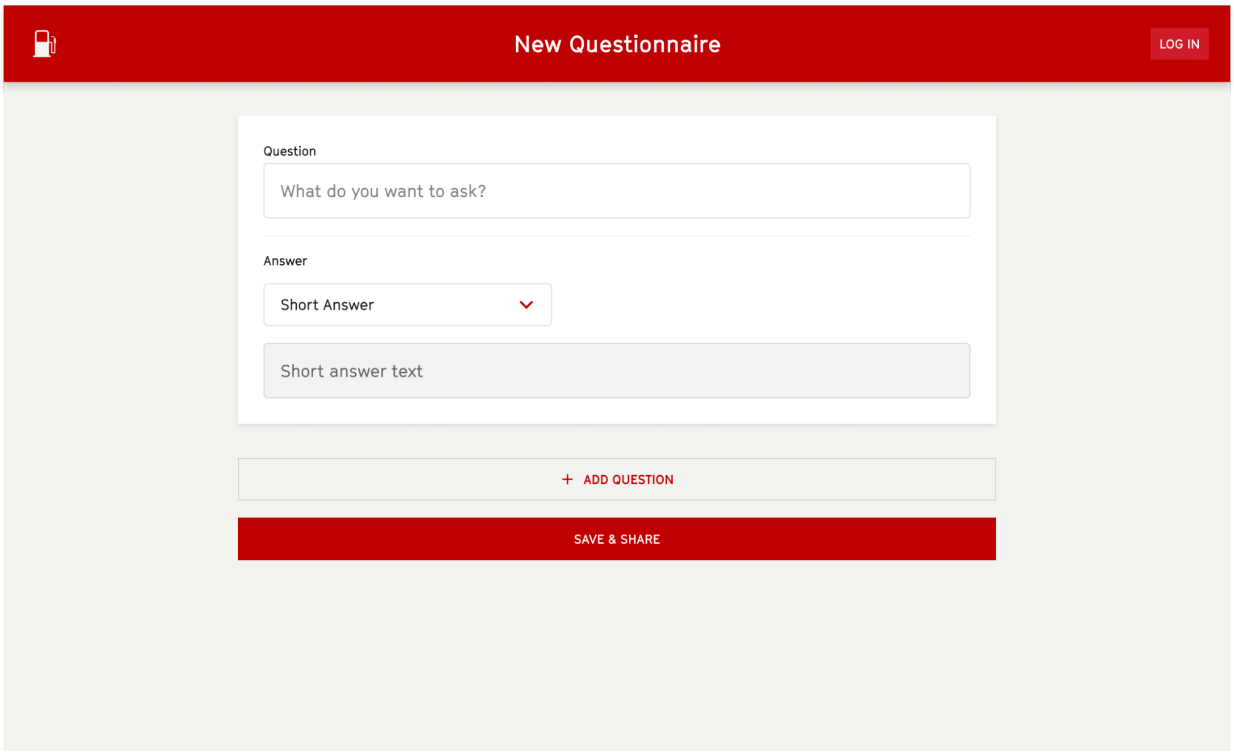
CODE EXERCISE

Position: Senior Backend Engineer

THE BRIEF

Your task is to take this design in Figma and build a GraphQL API using this boilerplate:



The API must provide the following functionality using queries and mutations:

- Add/update a questionnaire containing a title and any number of questions.
- You can delete or re-order any question in the questionnaire.
- Implement only one type of answer, i.e., short answer type.
- You should be able to publish it & get a shareable link.

WHAT WE'RE LOOKING FOR

- **A well-designed database & GraphQL schema.** You should understand the requirements from the wireframes and build an API schema and supporting data models in PostgreSQL.
- **Code organization and variable naming.** You should pay extra attention to the naming of variables and functions. You should organize your code based on the separation of concerns.
- **Database Optimisations.** You need to optimize your code for the most efficient storage and retrieval of data from PostgreSQL. This app can be designed in several ways; you should be able to explain your design choices around database schema.
- **You need to write tests.** Writing meaningful tests to ensure the correctness of APIs schema and behavior of different parts of your code.

WHAT WE'RE NOT LOOKING FOR

- **Don't worry about caching.** You don't need to write code to cache data to improve performance.
- **Don't worry about the handling of malicious queries.** You don't need to worry about query analysis like depth-limiting or cost-complexity analysis.
- **Extensive Documentation.** Your code and GraphQL schema should be self-documenting. Don't bother yourself with documenting everything very extensively.
- **Logging.** Don't go crazy about logging everything.

SUBMITTING YOUR SOLUTION

Along with your actual code, put together a short `README.md` file for your solution that talks about any interesting things you ran into, decisions you made, etc. Think of it as a little postmortem to talk through anything you found notable about the project. You can also use it as a place to explain any *"I know this might look weird at first glance but hear me out..."* sort of decisions you made.

When you're done, create a private repo on GitHub that just contains your project and invite me to it (I'm `theskumar`). Also, create an archive `“.zip”` of your project and send it to us.

Once we've had a chance to review your submission, we'll get back to you with the next steps.

Thank you!