Santex Front-end Developer Hiring Test: E-commerce

Summary

To help us better assess your technical chops we ask that you complete the following exercise.

The requirements are listed below, but if there are any additional items you want to add, please feel free. We are always impressed by individuals who show initiative.

Description & requirements

Create a very simple version of an e-commerce.

We will provide a GraphQL API that you must use to read and write data in order to create this e-commerce.

We will also provide a set of wireframes that you can use as a guideline, but feel free to use your own designs if desired.

The user should be able to:

- Navigate through a list of products
- Search for an specific product
- Access to a product details page
- Add that product to the cart
- See a preview of the current cart through a header icon / button
- Access a checkout page (listing cart items and total cart price)
- Perform a fake purchase
- When the user clicks on PURCHASE, just show a JSON with the cart details through console.log / alert / add a textarea in the screen

Technical constraints

- Create a React app (could be CRA, Next.js, or any other option of your preference) with Typescript.
- Fetch data from our server using Apollo client GraphQL from https://demo.vendure.io/shop-api
- Handle store/state global using React Context and custom hooks for handling (do not use libraries like Redux, Mobx, Recoil)

- Use Styled components for styling. You can use any component libraries from the ecosystem https://styled-components.com/ecosystem
- Create at least one custom hook
- User experience has to be taken into account

Nice to have

- GraphQL Fragments
- Media queries

Bonus

- Add Unit tests
- Use a linter
- Coverage report

Deliverables

Create a public repo into your bitbucket/github account. Once complete, send an email with the URL to your repo.

Code quality is important to us. Your code must be well-structured and built in the spirit of maintainability and extensibility.

We will be evaluating your work and will expect you to demonstrate how you tested your app and what considerations should be included in a professional level application.

Don't forget to add a README and any other documentation you think is necessary (train of thought, troubleshooting).

Notes

Feel free to send any questions/concerns. We will help you out.

Have fun! :)