science

Coding Exercise: Lens Demo in React

The exercise is meant to be more freeform: think hackday instead of final exam. The objective here is to do something interesting or useful. You should spend about four hours or so on it. The goal is not to create a production system, but rather to show you can.

Beyond node.js and React, you are welcome to choose your tools. Just make sure we have a way to run everything. Submission is by tarball, git repo, or something else reasonable. Include a README.md with the steps needed to run your project.

Lens is our product for reading information from faxed medical documents and putting it into the patient's medical record. Your task is to prototype some of the user interface.

For context, assume there are 200 faxes per day, each 1-40 pages long, and that most fall into a number of common categories: lab results, notes from a specialist, hospital reports, prescription requests from pharmacies, and records requests from insurers. Nearly all documents correspond to a single patient. Some documents, especially from hospitals, are concatenations of several documents, for example from different departments at a hospital. Your users are medical assistants. They are reasonably sophisticated and will be familiar with the product from using it daily.

You can download some sample documents and parse results at:
https://jobs.science.clinic/exercises/faxes.tgz

An existing sample prototype of part of the Lens UI can be found at:
https://jobs.science.clinic/exercises/lens-demo/ (live demo)
https://jobs.science.clinic/exercises/lens-demo.tgz (downloadable archive)

There are a number of options you might consider.

1. **Reactify the prototype:** Take the sample prototype and treat it as a UX designer's spec to be react-ified. As animations and visual effects are an important part of our design language, be sure to include them. You may depart from the spec if you have a good reason, but be prepared to explain why your approach is better.
2. **Inbox:** Create an inbox for Lens. You are strongly encouraged to think creatively and expansively about what this might look like. Can we do better than just a simple table of links called "Incoming Fax" with a timestamp. Is there an alternative design that might make for an easier workflow? Is there more status that could be shown, perhaps along the lines of a dashboard? If you have an idea but it would need more (plausible) metadata, feel free to just add it to the sample parse results data.

3. **Visualization:** Lens keeps track of where in the document it pulls information from. Create your own visualization using this information. For example, you might assemble cropped clippings into a sort of visual summary. Alternatively, you could build a document search UI using those clippings. Feel free to convert the PDFs to PNGs if that will simplify your life. Preprocessed clippings on disk are a completely reasonable way to mock out a service that would do that for you.
4. **UI for rapid editing:** Improve the existing prototype for rapid editing. Without confusing new users, add features that experienced users can discover and use to work more quickly. For example, make the mouse redundant through extensive use of keyboard accelerators, shortcuts, and jumps in the same way you can with gmail. If you use them for navigation, make sure it is clear where "here" is. Make sure to handle navigation/movement for the right pane as well the center document window.
5. **Storyboard widget library:** Taking the design language you see in the prototype, build a library of widgets in Storyboard. Include UI components you don't see but could reasonably anticipate. For example, you might include widgets you need for displaying an inbox or a dashboard.
6. **Your own adventure:** if you have an idea that you think is fun or interesting, we are not here to get in your way. Check in with us about it to make sure we're all on the same page about it being reasonable and then have at it.

We will discuss your submission in the technical interview. In particular, we will talk about your approach, any interesting decisions you made or tools you used, and how you might build upon your submission for use in the real world.

The exercise is somewhat open ended to allow you a certain amount of creativity. It is not meant to consume too much of your time. If you feel you will exceed the recommended time, let us know and we will add stipulations to constrain the exercise to something that makes sense.