truefoundry

Logs and Metrics (React Project)

Problem Statement

You have to create a React application that will allow users to fetch & view system logs and metrics. The application will have 2 screens (Logs and Metrics) that can be toggled between using the Navbar. The SPA navigations are expected to be integrated with the React router.

You need to create both the screens based on the following Figma file:
https://www.figma.com/file/4Rpl8dg76PV53iN4NMcNvc/Assignment?type=design&node-id=3%3A2654&mode=design&t=LwiUcoErqVIDtx9K-1
Password: tothemoon

Icons/Assets Link:
https://drive.google.com/file/d/1uVB4GRYo34IWl9n3SmBhOkL77NP5XDZK/view?usp=sharing

If there are any icons/assets missing in the zip file then you can simply export from Figma.

For APIs, a file is being shared that has the classes that mimic the behavior of corresponding APIs for Logs & Metrics.

All the components that you create need to be added to a component library which you document using Storybook.

The libraries mentioned below are the ones that you should be using. You're free (even encouraged) to use any extra libraries apart from the ones mentioned.

Library Versions:

- react: 17+
- react-chartjs-2: 4+
- react-router-dom: 6+
- tailwindcss: 3+
- @storybook/react: 7+

Features:

Logs:

1. A terminal-like view that renders logs as shown in the Figma file.

2. A list of quick selections for timerange (Last {n} minutes/hours). While any of these is selected, live logs will be enabled.
   A custom time range selection that opens a date and time range picker to show logs from and to given timestamps.
3. Subscribe to live logs (if applicable) that keep populating at the bottom.
4. Autoscroll if the user position is at the latest log line otherwise let the new ones go below the fold.
5. Infinite scroll and paginate previous logs when scrolling up (scroll position needs to be maintained).
6. Show count of live logs that have come up below the fold and scroll to the latest logs on its click.
7. All the states of logs (filters) should reflect in the url and thus should be shareable.

Metrics

1. 4 charts (mix of line and area charts) to be rendered using chartjs with react.
2. Time range based filtering of system logs
3. Select and drag over any section within a chart (Peak/valley) and you should get an option to check the logs for the corresponding time range.
4. All the states of metrics (filters) should reflect in the url and thus should be shareable.

Storybook

1. Hierarchical list of all the components used
2. Preview of all components
3. Props usage of all the components

URL scheme of the app should be:

1. Logs: /logs
2. Metrics: /metrics
3. Storybook: /storybook

How to submit your solution?

******PLEASE SUBMIT ONLY IF YOU HAVE COMPLETED THE SOLUTION FOR ENTIRE PROBLEM STATEMENT******

Please follow these steps to make a submission:

4. You have to ***submit the solution and hosted webapp within 8 days*** of receiving the project.

5. Upload the code on a ***public GitHub repo*** and name it “Frontend-Assignment”. We will review your code and you will hear back from us if we are ready to move you to the next stage. Please write a proper README in your GitHub repo so that it's easy to understand how to run the code.
6. Please ***deploy/host your application on Firebase, Netlify, Vercel, or any other platform*** of your choice.

If you have any further questions or queries then feel free to contact me at qaifi@truefoundry.com.