1. Introduction

This Web Application has the function to receive datas from user input, convert those datas to JSON format and plot them in a multiple lines chart. This way, the user can analyze the chart and reach certain conclusions.

2. Methodology

To develop this application were used technologies such as react js framework, in addition we used libraries such as Chart.JS for the implementation of the graph and Dynamic-Form in an attempt to create a dynamic form for input of data by the users.

Chart.JS has shown itself to be very consistent and easy to learn, it has functions that help at the time of visualization, such as hiding a certain series of graphs. The most difficult part because of the little experience with the technology used was to make the graph update by clicking the button, because of this, the data was entered manually being provided from a JS file.

The editor used for development was Sublime Text 3.

3. Suggestions

My suggestion to improve this Web Application's operation is to develop another method of reading data's, once that if the user wants to analyze thousand data's, for example, he would have to fill the form thousand times. Having this in consideration, a method to improve this process is to read the data's from a file in JSON format or from a Database. Additionally, a way to upgrade this application is to plot the chart in a new window and in a different type of chart. This way, if the user wants to analyze and compare a thousand values from the chart, it would be better to visualize if the chart would occupy the whole

page and not only a part of this. Moreover, the multiple lines chart is not the suitable if the user wants to analyze the data's in this case the bars chart type is the appropriate.

3.1 Functionalities

One of the functionalities that can be applied to the application is the recording and handling of errors, since it is important to record all the errors that occur during the execution of the application so that they can be corrected in the future.

Making the application customizable to the user is another important functionality, once the user has what he needs in a practical way the software becomes even more valuable for it.

Depending on the application it is important to keep in mind a way to protect it, in this question we can think of authentication, making it necessary to authenticate all users.

3.2 Usability

- The graph should always be visible to the user.
- Color-related color blindness should be avoided.

3.3 Reliability

It is possible that when working with an external data source errors occur, be they connection or whatever the reasons, so it is important to try to keep as much data as possible locally stored in an optimized way, in case there is a failure the user can continue using the application.

3.4 Performance

It is important that the user can access data and graphics quickly, so the system is expected to respond in a matter of seconds or milliseconds.

4. Observations

Unfortunately I could not develop the application as it should. I understood what should have been done in the challenge, however, due to my lack of experience, since I had never mess with the react technology and within 12 days I reported it, I could not finish it completely. However, I consider the experience to be extremely valuable since I gained tremendous experience and enthusiasm to continue developing the application. As for the deadline, I'm sending the application today, even if incomplete since meeting the deadlines is an essential task.