Simas Simokaitis

Assignment 6

Problem Statement:

A physics laboratory needs an application for storing and displaying results from various experiments. Raw data may be input, saved, and recalled for viewing or editing. Data may be presented in numerical or graphical form. The system should also allow portions of this raw data to be selected and fed through any number of user defined functions and the output displayed and saved.

Summary:

This application will help a physics lab store and manage data from experiments. Users can enter raw data, save it, view or edit it later, and choose how it's displayed as numbers or graphs. The system will also let users select parts of the data and run it through custom functions, with results that can be viewed and saved.

Software Stack:

- **Frontend:** React.js – clean interface for users to interact with data and view graphs.
- **Backend:** Node.js with Express – handles data processing and connects everything.
- **Database:** MongoDB or PostgreSQL – stores raw and processed data.
- **Graphing Library:** Chart.js or D3.js – turns numbers into visuals.
- **Environment:** Development will be done on VS Code with Git for version control. Hosted either locally or on a platform like Heroku or AWS.

Team Size:

A team of **4–5 people** would work best:

- 1 frontend developer
- 1 backend developer
- 1 database person
- 1 tester (can also help with QA or documentation)
- 1 project lead (can also code)

Development Timeline:

Weeks	Task
1	Meet with customer, plan features

2–3	Design the interface and set up database
4–6	Build frontend and backend logic
7	Add graphs and data display features
8	Add support for user-defined functions
9	Test everything and fix bugs
10	Final polish and deploy the app

System Jobs:

- Let users input and save raw data
- View data as tables or graphs
- Edit previously saved data
- Choose data and run it through custom functions
- Save and view processed data

Questions:

1. What types of files should we support for importing or exporting data?
2. What kind of functions will users define?
3. Does the app need accounts and logins, or will it be used by one person at a time?