

Scenario 1:

I would use the REACT framework for this first example. It appears that their biggest pain point is organization and maintaining up to date code. REACT would help with scalability as the user data grows. You could use the new framework as an overhaul for the new application you are being asked to create in order to create organization within the code that may scale up over time.

Scenario 2:

In the second example, I would use a library to build out the large data table. I would find a library that is compatible with a lot of different types of applications. A library would allow the different sub-apps to be updated as changes are made to different tools without having to completely overhaul all the work done in the sub-apps. It gives the application room to breathe as the application needs to grow and change.

Scenario 3:

This last example I would use the Vue.js framework. It specializes in the prototypes and would allow for a prototype to be created and then reviewed for further use. It would allow the developer to create a prototype without having to start from scratch for something that may or may not be used. Vue.js is also built out from Angular which is particularly good with animation so this may aid in the design part of the prototype.

Advantages of REACT for Achievement 3:

REACT allows different components to be combined to create an application. This project has many pieces such as a data layer, business logic layer, client layer and server. REACT will allow me to combine these pieces while easily updating each piece without recreating the whole application. It will keep the application be well organized as components increase.