Sanket Mehta

TechDoc#6: ReactJS & MobX

What is ReactJS & why do we use it instead of plain JS?
ReactJS is an open source JavaScript library designed by Facebook for creating rich and engaging web apps fast and efficiently with minimal coding. The core objective of ReactJS is providing the best possible rendering performance. Its strength also comes from the focus on individual components - Instead of working on the entire web app, ReactJS allows a developer to break down the complex UI into simpler components. Also, React is very a simple and lightweight library that only deals with the view layer. It is not a beast like other MV* frameworks such as Angular or Ember. Any Javascript developer can understand the basics and start developing an awesome web application after only a couple of days reading tutorial.

Why do we need separate library for state management with ReactJS?
State is the piece of information that keep track of the elements present in your web application at any point. React is built with a way for components to internally manage their state & data without any need for an external library or tool. It does well for applications with few components but as the application grows bigger, managing states shared across components becomes a chore. In an app where data is shared among components, it might be confusing to actually know where a state should live. Ideally, the data in a component should live in just one component. So sharing data among sibling components becomes difficult. Given that the state management gets messy as the app gets complex, we need something for state management & we use MobX for this.

What is MobX & How does ReactJS and MobX make a good combination to build any WebApp?
MobX provides the mechanism to store and update the application state that React then uses. React and MobX together are a powerful combination. React renders the application state by providing mechanisms to translate it into a tree of renderable components. Both React and MobX provide optimal and unique solutions to common problems in application development. React provides mechanisms to optimally render the UI by using a virtual DOM that reduces the number of costly DOM mutations. MobX provides mechanisms to optimally synchronize application state with React components by using a reactive virtual dependency state graph that is only updated when strictly needed and is never stale.

Useful Links:

- Ten minute introduction to MobX and React - https://mobx.js.org/getting-started.html
- How to build your first app with Mobx and React - https://hackernoon.com/how-to-build-your-first-app-with-mobx-and-react-aea54fbb3265
- React - https://reactjs.org
- MobX - https://github.com/mobxjs/mobx