A Beginner's React Tutorial

Brady Hagen

`bhagen@oxy.edu`
Occidental College

1 Methods

1.1 Setup

I first began by setting up my own developer environment for React Native. For clarification purposes I'm running Manjaro Linux 5.15 on the KDE Plasma desktop environment. For this tutorial report I was following 'Made With Matt's' "Build Your First React App" tutorial [1]. The first step was to install Node which I did with sudo pacman -S npm.

With Node installed I could then use NPM to get the Expo CLI. sudo npm install -g expo-cli The console spit out a ton of errors, most of which were from deprecated packages which could be fixed with npm audit fix. sudo npm audit fix –force However in order to use npm audit I first need a package-lock file, which the CLI informs me I don't have. However, I can create with the command: sudo npm i –package-lock-only After trying it I found that this command doesn't work as it needs another .json file. It needs a (package.json to complete and make the package-lock.json file. Upon looking online, not having a package.json is a common error mainly due to NPM failing to install correctly because of currently open Node instances like VSCode or other Electron Apps (like Notion or Obsidian). I closed out VSCode, Notion, and Obsidian, reinstalled NPM with the same command, and this time was able to install Expo without it failing. I was now able to create and initialize my new React project with expo init TestProject. After installing various dependencies I finally had the basics to start development. To double check everything was running correctly I ran: npm run web. Sure enough, Node once again couldn't find my package.json even though it was there in the correct directory. After some troubleshooting I realized the name of the directory above my 'TestProject' was 'Tutorial App' and after a quick Google I found that Node hates it when there's a space in the path as it can cause it to totally break itself on build. I renamed some paths and ran it again and finally my Node server was up and running. What's really cool about the Expo CLI is that as you edit and change your code it can hot reload and update in real time. This means I'm able to constantly edit and adjust my code without having to compile after every change. Now that everything was setup properly I could start to work on the To-Do List.

1.2 App Development

React gives us some basic boiler-plate to start with, namely an App.js file where most of our code goes, and then a handful of .js and .json files that handle the building and deployment information. From here on I began to follow the tutorial presented in the video. We first went through and created the various view styles that were meant to represent the basics of our to-do list. At this point I realized that React Native is a terrifying combination of CSS, HTML, and JavaScript all jammed together. At the top of the JS file is a series of views, these views essentially dictate how the text is laid out on the page. From there each piece of text can be modified through various views or other components. At the bottom of the page is a mish-mash of CSS which defines the color of various components, the size of the font, and various areas of padding. The generall workflow was to layout the framework with basic HTML, then to follow up on the bottom in order to make it look nice with bits of styling. From there you add basic functionality with JavaScript and create interactable elements.

2 Evaluation

I picked this tutorial because I thought it would be the most applicable to my project, that of creating an interactive list to display information. While a To-Do List is significantly simpler to implement, a lot of concepts are transferable. Being able to modify views and create interactive elements in the form of views is super cool. It really opens up a world of possibility by making each and every element you use to be entirely customize-able. As awful as React Native is to look at it, it really does have some powerful capabilities.

What I was most impressed with however was definitely the Expo CLI. It was so nice making a change, hitting CTRL+S, and then bam being able to view my changes in real-time. Although I did find that while my IDE showcased no errors sometimes the page wouldn't display. I thought that it was because Expo was broken but nope, I found that that JavaScript due to its free-form nature is pretty lenient

with whatever it decides to flag as an error. While this is nice for people who are good at coding, for people who aren't very good this led to quite a bit of headache as I scanned each element for the capitalization I forgot.

3 Results

As I mentioned earlier this guy really goes super duper fast. It's less of a tutorial and more of him explaining what he needs to do, and then doing it. He really doesn't explain how things work, or why he decides to do things a certain way. Instead he has design goals that he had listed out before starting the video and then goes through and tries to create elements that accomplish them. It was cool to get hands on and create with the React Framework but I would still say I'm generally just as clueless as when I first started. I think I need something geared less towards already experienced developers and something instead for beginners. Moving forward from here I think I have to go pick up a JavaScript architecture textbook and get reading as I don't think YouTube tutorials will cut it.

References

[1] Matt, Made With. *Build your first React Native app - Todo List Tutorial*. URL: https://www.youtube.com/watch?v=0kL6nhutjQ8&t=364s.