Tutorial Writeup

Malcolm Holman

`holmanm@oxy.edu`
Occidental College

1 Introduction

The selected tutorial is about how to make a mobile app using react native, created by Vadim Savin who goes by notJust.dev on YouTube. I chose this tutorial because my comps proposal is a mobile app and I am a beginner app developer so a video breakdown of the fundamentals will be more informative than solely documentation. Using a video tutorial allows for somebody with more experience than me to expand on the concepts within the code and documentation. The app itself is a replica of the tesla homepage at the time of recording (December 2020).

2 Methods

2.1 Setup

Immediately I noticed Vadim made the tutorial easier for the viewer by having a set powerpoint with a general roadmap of what would be discussed in the tutorial. The first thing that I did was download the Expo CLI using instructions that can be found on the Expo website. This download allowed me to skip some of the tedious framework and library setup that would come with creating an application from scratch. This is particularly useful for me as a beginner because currently I care more about learning to code in React rather than the setup of an application. Additionally, Vadim stated that the React setup from scratch would take too long and could be a different tutorial. Expo allowed me to start a local server on which I could host my application and see live updates as I make them. It also allowed me to choose from an iOS emulator, Android emulator, and web view options for how I wanted to view the application. Expo is downloaded and run by using node package manager. One thing that Vadim left out of the tutorial was the need for an iOS sdk to locally emulate an iPhone on a mac. This issue can be bypassed by either downloading the SDK or using expo's convenient mobile app to render the app.

2.2 Text Editing

I followed along as Vadim broke down what the files that are automatically generated by Expo do. Many of the files did not change and did not pertain to the project at hand so they were skimmed over. The most important folders are the assets folder, which contains any images or files that need to be rendered, and the app.js file that actually renders the application. Vadim then went on to show what happened when text was modified in the app.js file and the live updates were then rendered on the iPhone emulation. While explaining how to modify, format, and position text, Vadim pointed out similarities to other languages which I was familiar with, which gave me further clarification.

2.3 Background Image

The next thing that I did was begin the process of rendering a background image. Vadim provided a free bundle that contained all of the necessary images for the project and linked the download in the description of the video. He then showed how to import it into the project by adding the images to the assets folder. Image background is not in javascript by default, instead I had to import it from react. I then adjusted the style of the image to make it more similar to the background image found on Tesla's actual website.

2.4 Car Component

Following the creation and styling of a background image, Vadim explained what a component is in react and how to use them. Reusable components allow for code to be more modular and reduces the amount of code that has to be written. There is not a default components folder, so I had to create one. To make the car component I took the style and text that was in the app.js file and added them to a new folder called CarItem in the components folder. Then, the displayed content and the style were split into an index.js file for content and a styles.js file containing the style information. This is not necessary as the styles can be appended to the bottom of the index.js file but it improves readability. In order to render the new car item on the application I had to import the component to app.js.

2.5 Button Component

The next feature that I added was two clickable, reusable buttons. Because the button needed to be reused with minor text or functionality changes, a component was created. The component setup process for the button was the same as that of the car item: create a button folder within the components folder, create an index.js for rendered items and logic, create a styles.js file to give the index file the necessary formatting. An issue arose when creating the button because the button needed to have different properties depending on whether or not it was a primary button. This meant that I had to utilize the properties, props for short, feature of react to pass in parameters upon button render.

2.6 Car component

Finally, in order to render all of the available models without rendering a car item component on the main app.js file for each car, a cars list component had to be created. Vadim provided a cars.js file which contained an array of all available Tesla models and the information that needed to be passed into car items for each different model. This dynamic rendering greatly reduced the amount of duplicate code that needed to be written.

3   Evaluation

The evaluation of this tutorial is rather straightforward because the goal was simply to create my first React Native application. The tutorial did not cover any app functionality or anything that needed to be loaded at a certain speed, it was solely visual. The app successfully runs on both my computer and my phone so I would say that it was a success.

4   Discussion

Vadim succeeded in providing a comprehensive step by step guide on how to create a mobile app using React Native with no react experience. He did not leave any key details out and all necessary files were linked and free to use. I learned the fundamentals of React and the general workflow necessary for mobile app development. Additionally, I picked up more broad coding tips such as "don't repeat yourself", abbreviated to DRY. The tutorial made me more confident in my ability to produce a mobile app for my comps project. I will likely have to watch several more tutorials in the future but I now have a stronger foundational understanding of React development.