**Lab Assignment 1: Spinning Up React Native App**

**﻿System Requirements**

React Native development requires the following system specifications:

* **Processor:** A modern-day multi-center processor, ideally i5 or better.
* Memory: Minimum of 8GB RAM, although 16GB is suggested for optimum performance.
* **Storage:** At least 10GB of free disk area to accommodate the installation and assignment files.
* **Operating System:** React Native helps each Windows and macOS X. [1] [2]

**Installation Instructions**

* **Install Node.Js:** Node.Js is needed to run JavaScript code. You can download it from the authentic Node.Js internet site.
* **Install React Native CLI:** The React Native Command Line Interface (CLI) is a tool that lets you create new React Native apps. Install the use of npm (which comes with Node.Js) by running npm set up -g react-local-cli to your terminal.

**Configuration Steps**

* **Set up Android improvement environment:** If you want to increase Android, you want to install your Android improvement surroundings. This entails installing Android Studio and the Android SDK. [3]
* **Set up iOS improvement environment:** If you’re on a Mac and need to develop for iOS, you want to install Xcode. [3]

**Project Creation**

* **Create a new React Native mission:** Now create a new React Native assignment with the aid of jogging npx react-native init YourProjectName (Amazing\_Todo\_List\_App) on your terminal. [3]
* **Navigate into the assignment directory:** Use the command cd YourProjectName (Amazing\_Todo\_List\_App).

**Running the Project**

* **Start the Metro Bundler:** In your project directory, run: npx react-local start. [3]
* **Run the app on an Android Device Simulator:** Make positive you have an Android Virtual Device installation in Android Studio. Then, through the project directory, run: npx react-local run-android. [3]

**Troubleshooting**

* **Port already in use:** The Metro bundler runs on port 8081. If every other procedure is using that port, you could terminate that manner or exchange the port that the bundler uses.
* **NPM locking error:** If you come across an blunders such as npm WARN locking Error: EACCES whilst the usage of the React Native CLI, strive jogging sudo chown -R $USER ~/.Npm and sudo chown -R $USER /usr/neighborhood/lib/node\_modules.
* **Missing libraries for React:** If you introduced React Native manually on your task, make certain you have got covered all the applicable dependencies which you are the usage of, like RCTText.Xcodeproj, RCTImage.Xcodeproj. [5]

**Citations**

[1] Intellipaat, “React native environment setup - the complete guide,” Intellipaat, https://intellipaat.com/blog/react-native-environment-setup/ (accessed Feb. 4, 2024).

[2] @ReactNativeMSFT, “System requirements · REACT native for windows + macos,” · React Native for Windows + macOS, https://microsoft.github.io/react-native-windows/docs/rnw-dependencies (accessed Feb. 4, 2024).

[3] Meta Platforms, “Setting up the development environment · REACT NATIVE,” React Native RSS, https://reactnative.dev/docs/environment-setup (accessed Feb. 4, 2024).

[4] A. Bouchefra, “Creating your first react native project,” Techiediaries, https://www.techiediaries.com/react-native-tutorial/create-project-using-react-native-cli/ (accessed Feb. 4, 2024).

[5] P. Aboyi, “Run a react native app on an Android device or emulator: Waldo Blog,” RSS, https://www.waldo.com/blog/run-a-react-native-app (accessed Feb. 4, 2024).