**React Native Environment Setup**

**Development OS: MacOS**

**Target (Mobile) OS: Android**

1. Install Homebrew by opening a Terminal window and executing the following command:  
   /bin/bash -c “$(curl -fsSL <https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh>)”
2. Once Homebrew is installed, use it to install Node and Watchman by running the following commands in a Terminal window:  
   brew install node  
   brew install watchman
3. Install the Java Development Kit:  
   brew install --cask adoptopenjdk/openjdk/adoptopenjdk8
4. Download and install Android Studio from <https://developer.android.com/studio/index.html>
5. During Android Studio installation, make sure the following options are selected:
   1. Android SDK
   2. Android SDK Platform
   3. Android Virtual Device
6. Once Android Studio is installed, open it and click Configure, and then select SDK Manager.
7. Select the SDK Platforms tab and then check the box next to Show Package Details in the bottom right corner. Look for and expand Android 10 (Q) and make sure the following items are checked:
   1. Android SDK Platform 29
   2. Intel x86 Atom\_64 System Image OR Google APIs Intel x86 Atom System Image
8. Select the SDK Tools tab and check the box next to Show Package Details. Look for and expand the Android SDK Build-Tools entry and make sure that 29.0.2 is selected.
9. Click Apply to download and install the Android SDK and related build tools.
10. Open a Finder window and look for the .bash\_profile file. Open this file in a text editor and add the following lines:  
    export ANDROID\_HOME=$HOME/Library/Android/sdk  
    export PATH=$PATH:$ANDROID\_HOME/emulator  
    export PATH=$PATH:$ANDROID\_HOME/tools  
    export PATH=$PATH:$ANDROID\_HOME/tools/bin  
    export PATH=$PATH:$ANDROID\_HOME/platform-tools
11. If you are using zsh instead of bash as your terminal, you will need to add these lines to your .zprofile or .zshrc file instead of the .bash\_profile file.
12. Reopen Android Studio and click Configure and then select AVD Manager.
13. Click Create Virtual Device. Choose a device from the Phone category (I recommend one of the Pixel devices) and click Next. Click Download next to the Q release. Once downloaded, or if the Download option is not available, select the Q release and click Next. Complete the wizard.
14. Open up a Terminal window and execute the command npx react-native init TestProject
15. Node should begin downloading the React Native libraries and create a project called TestProject.
16. Once Node has finished downloading the necessary files and creating the project, navigate to the TestProject folder by executing the command cd TestProject
17. Execute the command npx react-native run-android
18. Your project should begin building and should then launch on the Android emulator.
19. Close the emulator and the Metro Bundler window. Delete your TestProject folder.

**React Native Environment Setup**

**Development OS: MacOS**

**Target (Mobile) OS: iOS**

1. Install Homebrew by opening a Terminal window and executing the following command:  
   /bin/bash -c “$(curl -fsSL <https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh>)”
2. Once Homebrew is installed, use it to install Node and Watchman by running the following commands in a Terminal window:  
   brew install node  
   brew install watchman
3. Open the Mac App Store and download and install Xcode.
4. Open Xcode and choose Preferences… Click Locations and install the tools by selecting the most recent version in the Command Line Tools dropdown.
5. Select the Components tab and choose a simulator with the corresponding version of iOS you wish to use.
6. Open a Terminal window and execute the command sudo gem install cocoapods
7. Once cocoapods is installed, execute the command npx react-native init TestProject
8. Node will download the necessary React Native libraries and create a project called TestProject.
9. Execute the cd TestProject command to navigate to the TestProject folder.
10. Execute the command npx react-native run-ios
11. Your project should begin building and then your app should launch on the iOS simulator. This may take several minutes.
12. Once you’ve confirmed the project has launched on the simulator, close the simulator and the Metro Bundler window and then delete your TestProject folder.