## Chris Johnson

CPRG-303-B

Feb 4.2024

# React Native Environment Setup Guide

## System Requirements

CPU: Minimum duel-core 2.0 GHZ or equivalent

Ram: Minimum 8 GM, recommended 16 GB

Windows Version: Windows 10 or higher (64-bit)

Installation Instructions

1. Verify the version you are using
   1. Open command prompt enter ‘java -version’

Ensure its version is 17jdk or higher.

If version is out of date download the current version from [Java Downloads | Oracle](https://www.oracle.com/java/technologies/downloads/)

1. Node.js
   1. Download and install the latest version of Node.js from the Official Node.js website
   2. Verify version in command prompt ‘node -v’
2. Download and Install Android Studio from [Download Android Studio & App Tools - Android Developers](https://developer.android.com/studio)
   1. Install the Android SDK.
      1. To do that open Android Studio, click on “more actions’ button and select “SDK Manager”
      2. Select the “SDK platform" tab from within the SDK manager.
      3. Look for and expand the Android 12(tiramisu) entry check the following items.
         1. Android SDK Platform 33
         2. Intel x86 Atom\_64 System Image or Google APIs intel x86 Atom system Image
         3. Next, select the ‘SDK tools” tab and check Android SDK build- Tools entry and make sure its that 33.0.0 is selected
         4. Click Apply

#### 3. Configure the ANDROID\_HOME environment variable

The React Native tools require some environment variables to be set up to build apps with native code.

1. Open the **Windows Control Panel.**
2. Click on **User Accounts,** then click **User Accounts** again.
3. Click on **Change my environment variables.**
4. Click on **New...** to create a new ANDROID\_HOME user variable that points to the path to your Android SDK: \*\* this location can be found in android studio “setting” dialog, under Languages & frameworks 🡪 Android SDK
5. Open PowerShell
   1. Copy and paste **Get-ChildItem -Path Env:\ into power shell**
   2. **Verify ANDROID\_Home has been added.**

#### 4. Add platform-tools to Path

1. Open the **Windows Control Panel.**
2. Click on **User Accounts,** then click **User Accounts** again
3. Click on **Change my environment variables**
4. Select the **Path** variable.
5. Click **Edit.**
6. Click **New** and add the path to platform-tools to the list.

Default location is

%LOCALAPPDATA%\Android\Sdk\platform-tools

## Creating a new application:

* 1. navigate to your working path
  2. Run npm uninstall -g react-native-cli @react-native-community/cli

To uninstall previously installed package.

* 1. Run - npm install -g react-native-cli
  2. Run - npx react-native@latest init JohnsonToDoListApp
  3. Navigate to the project Directory:
     1. cd YourProjectName
  4. Run the Project
     1. Npm start
     2. Select run on android

That’s it if it’s installed correctly

## TroubleShooting

Debugging: Use the debugging tools provide by your IDE(Visual Studio code) for effective debugging.

Error messages: refer to the official react native troubleshooting guide.

## Resources:

React native official documentation.

React native GitHub repository.

Stack overflow- react native questions.