

Pro React Native

Master your tools. Build great apps.

[Articles](#)[Services](#)[About](#)

54

Shares



39

Getting Started with React Native on Ubuntu



5

LINUX



You're tempted to give React Native a shot.



If you're on Linux, you either prefer it for ideological reasons or for productivity reasons. And, between you and me, I'm happy that your money isn't going to



me. :)



In any case, if you're coming to React Native, I'm guessing that either you're frustrated with native Android development, or you're big on Javascript / React.js and want to apply your skills on mobile.



The good news is that the development experience on React Native is quite good!

The bad news is that getting it up and running on Linux can be a bit tricky, and the docs on the docs on the React Native site are a bit skimpy.

So I've put this walkthrough together to get you up and running in pure copy and paste fashion.

Give it a shot and please let me know in the comments if these directions didn't solve your problem.

Walkthrough

Prerequisites

Before following this guide, you should have installed the Android SDK and run a

successful Java-based "Hello World" app for Android.

Other than that, we should have you covered!

Let's dig in.

54 Shares **Installing everything**



Facebook NodeJS



First thing you need to do is to install NodeJS, a popular Javascript implementation.



Open a terminal and paste these commands in to install NodeJS from APT:



```
sudo apt-get install -y build-essential  
curl -sL https://deb.nodesource.com/setup\_4.x | sudo -E
```



bash -

```
sudo apt-get install -y nodejs
```



Install NPM



After installing NodeJS, you should now have NPM, too, the Node Package Manager.

You can check by typing `which npm` into the terminal.

If you have NPM, you'll see something like:

```
$ /usr/local/bin/npm
```


If you don't have NPM, you'll see:


```
$ which npm  
npm not found  
$
```


If that's the case, run the following command to install NPM:

```
curl http://npmjs.org/install.sh | sh
```


Install watchman

 **54** Shares **watchman** is a tool developed by Facebook for watching the filesystem. For best results, you'll want to install it.


 **39** Again, paste these into your terminal to get up and running:

 **5** git clone <https://github.com/facebook/watchman.git>

 cd watchman

 git checkout v4.1.0 # the latest stable release


 ./autogen.sh

 ./configure

 make

 sudo make install

 **|| flow**

 up: flow. This is a static type checker for Javascript.

Here's how you install it:

- `wget https://facebook.github.io/flow/downloads/flow-linux64-latest.zip`
- `unzip flow-linux64-latest.zip`
- `cd flow`
- `sudo mv flow /usr/local/bin/flow`

Note: for advanced users, you can put the flow executable anywhere on your PATH. It doesn't have to be in /usr/local/bin.

Install React Native

OK, with all of that out of the way, it's finally time to install React Native.

For most users of this guide, you'll need to use `sudo` to install React Native as root:

- `sudo npm install -g react-native-cli`

For users who installed NPM through other means, you'll likely want to install React Native as a normal user:

`npm install -g react-native-cli`

Awesome Project

Yay! React Native is installed.

Now it's time to do what we came here to do: make an app!

To create an app project, you'll need to sit down and input this command:

```
react-native init AwesomeProject
```

It may take several minutes, so don't panic if your terminal sits around for a while before spitting out any output.

Setting Up an Android Device

OK, now that we have a starter project waiting for us, we're almost ready to run the code.

But before we do, we'll need either an emulator or a real Android device to run our code on.

In this tutorial, we're going to use a real Android device.

First thing is to plug in your device and check the manufacturer code by using `lsusb`:

```
$ lsusb
Bus 002 Device 002: ID 8087:0024 Intel Corp. Integrated Rat
```

```

Bus 002 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root
Bus 001 Device 003: ID 22b8:2e76 Motorola PCS
Bus 001 Device 002: ID 8087:0024 Intel Corp. Integrated Rat
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root
~ ; 004 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root
54
Shares ; 003 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root

```



39



5

ns like a bunch of gibberish, eh? You want the line that represents your
e. If you're in doubt, try unplugging your phone and running the command
l:



.susb



```

; 002 Device 002: ID 8087:0024 Intel Corp. Integrated Rat
; 002 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root
; 001 Device 002: ID 8087:0024 Intel Corp. Integrated Rat
; 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root
; 004 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root
; 003 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root

```

You'll see that after removing the phone, the "Motorola PCS" disappeared from the list. This is the one that we care about!

```
Bus 001 Device 003: ID 22b8:2e76 Motorola PCS
```

From thabove line, you want to grab the first four digits from the device ID:

```
22b8:2e76
```

In this case, it's 22b8. That's the identifier for Motorola.

You'll need to input this into your udev rules in order to get up and running.

So, fire up your favorite text editor. For me it's vim:

- `sudo vim /etc/udev/rules.d/51-android.rules`

And add the following line to the file:

```
SUBSYSTEM=="usb", ATTR{idVendor}=="22b8", MODE="0666", GROU
```

check that your device is properly connecting to ADB, the Android Debug
54 Shares |e, by using adb devices.



39

adb devices



5

list of devices attached

0300GLMK device



should see it as shown above.



do a little dance. We're almost there!



Finishing the Project



Now that we have a starter project waiting for us, we'll need to run the code.

In one terminal:

- `cd AwesomeProject`
- `react-native start`

In a second terminal:

- `cd AwesomeProject`
- `react-native run-android`

Did it fail?

OK, try this:

- `adb reverse tcp:8081 tcp:8081`

Still failing?

Leave a comment below or ping me on the [Reactiflux Discord server](#) and I'll help you out.

Did it succeed?

54

Shares

ET! Celebrate and then let's move on to the grand finale:



39

Open `index.android.js` in your text editor



5

Change "Welcome to React Native" to "Welcome to React Native on Android"

Press the menu button (F2 by default, or ⌘-M in Genymotion) and select

Reload JS to see your change!



Run `adb logcat *:S ReactNative:V ReactNativeJS:V` in a terminal

to see your app's logs



It's a doozy, but we did it!



Was this helpful? If so, don't miss my next post by joining the mailing list using the form below:



Get weekly tips to become a React Native Pro

* indicates required

Email Address

Subscribe

Written on February 3, 2016

David Kay writes about keeping you productive in React Native. Over the years, he's developed over a dozen successful native apps. If you found this article helpful, [join his weekly newsletter](#).

15 Comments Pro React Native

[1 Login](#) ▾[♥ Recommend](#) 2[🔗 Share](#)[Sort by Best](#) ▾54
Shares

Join the discussion...

LOG IN WITH

OR SIGN UP WITH DISQUS [?](#)

39



5

**AdrianLab** • 4 months ago

Just to point out, to install watchman in a fresh Ubuntu VM i also had to do
sudo apt-get install python-dev

2 ^ | ▾ • Reply • Share >

Kretawiweka Nuraga Sani • 10 months ago

I get error :

* What went wrong:

Execution failed for task ':app:mergeDebugResources'.

> Error: java.util.concurrent.ExecutionException: java.lang.RuntimeException:
AAPT process not ready to receive commands

please help me

1 ^ | ▾ • Reply • Share >

**pallav** • 3 months ago

Hi, Thank you so much for the tutorial. I cannot tell you how much I thanked you in my heart while following your steps. I am able to launch the app but I cannot see any text on the app. Is there a way to fix this? I am using Ubuntu 14.04 with node -v7.10.1 and npm 4.6.1 and Android 5 (Xiaomi). I am able to launch the app using Expo but not with native code. Where am I going wrong?

^ | ▾ • Reply • Share >

**Spartacus Gupta** • a year ago

Android Debug Bridge version 1.0.31

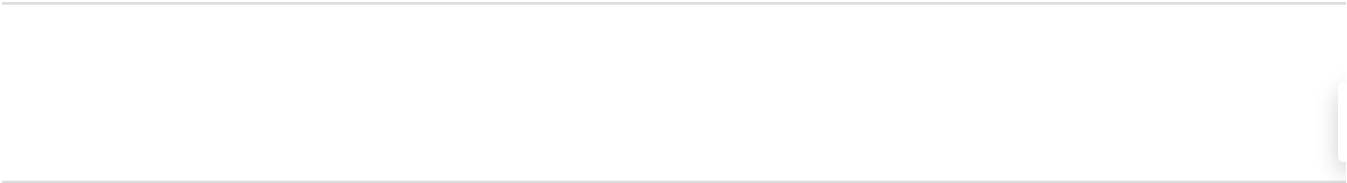
I am frustrated now, spent almost 3 days on this.

command adb reverse tcp:8081 tcp 8081

never work for me I want live reloading but using usb connection

please help me

^ | ▾ • Reply • Share >



54
Shares



39



5

