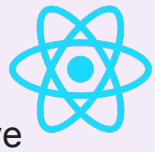
The React Native Development Process



tl;dr

- Your app must run in a device during the development process
- Expo is an emulated environment. You can run it ...

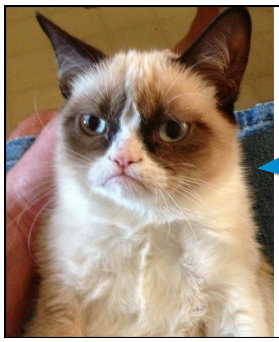
1. on a physical device (with Expo installed) by scanning a QR code
2. in an iOS emulator after installing xcode and the iOS emulator on your Mac
3. in an Android emulator after installing Genymotion and Oracle VirtualBox

- ... And there are actually more options

Where do I even start?

Three ways to start a RN project

1. Manually
2. react-native cli
3. expo



Manually

Really?!? We're considering this? Wow.

react-native-cli

- react-native init
- Creates an initial project
- You then code and deploy to a real device or run in an emulator

create-react-native-app

- A command-line tool that scaffolds a basic project.
- $ npx create-react-native-app cool-app
- $ cd cool-app
- $ expo start

What is react-native-cli?

- It's older.
- Allows you to init an app but it must be developed in Android Studio and in Xcode. JDK, Android SDK, Need Android 6+ (Marshmallow)
- Need npm version 4, not 5. Can't use the latest version of npm
- Windows to Android - Yes
- Windows to iOS - No
- MacOS to Android - Yes
- MacOS to iOS - Yes
- Linux to Android - Yes
- Linux to iOS - No

Which to use?

react-native init vs. expo	rni	expo
Can use native modules written in Java/Swift	☑	☒
Can code without Android Studio and XCode	☒	☑
Can develop for iOS on Windows	☒	☑
Can test without being tethered to a device via USB	☒	☑
JavaScript APIs provided/no extra installs needed	☒	☑
Setup and configuration time in minutes vs. hours	☒	☑
Easy to share the app during development	☒	☑
Officially recommended by the react-native team	☒	☑

```
$ npx create-react-native-app cool-app
npx: installed 28 in 2.652s
? Choose a template: blank
[14:27:05] Downloading project files...
[14:27:09] Extracting project files...
[14:27:16] Customizing project...

Your project is ready at /Users/you/coolApp
To get started, you can type:

  cd cool-app
  expo start
$
```

create-react-native-app uses Expo

Expo apps are React Native apps which contain the Expo SDK. The SDK is a native-and-JS library which provides access to the device's system functionality (things like the camera, contacts, local storage, and other hardware). That means you don't need to use Xcode or Android Studio, or write any native code, and it also makes your pure-JS project very portable because it can run in any native environment containing the Expo SDK

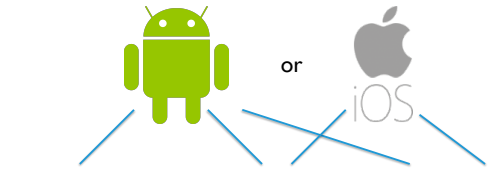
How do I run it?

`expo start`

- Builds and bundles your application into memory
- This starts a node server which serves a static json config at a given URL.
- If the expo client browses to that config file, your app will run

```
{
  "sdkVersion": "27.0.0",
  "name": "cool-app",
  "version": "0.1.0",
  "xde": true,
  "developer": {
    "tool": "crna",
    "projectRoot": "/coolApp"
  },
  "packagerOpts": {
    "hostType": "tunnel",
    "lanType": "ip",
    "dev": true,
    "minify": false,
    "urlRandomness": null
  },
  "env": {},
  "bundleUrl": "http://
172.16.2.232:19001/...",
  "debuggerHost": "172.16.2.232:19001",
  "mainModuleName": "./node_modules/...",
  "logUrl": "http://172.16.2.232:19000/logs",
  "id": "@anonymous/rn-..."
}
```

But Expo can't run in a browser. It needs a device like ...



Tethered or Wireless or Emulated

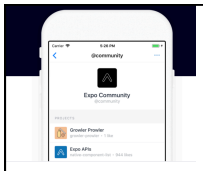
So we have these options

q	Show a QR code that you scan with your physical device's camera
a	Run in an Android emulator or tethered device
i	Open an iOS emulator and run it in there

```
$ expo start

> coolApp@0.1.0 start /Users/rap/Desktop/coolApp
> react-native-scripts start

16:14:20: Starting packager...
Packager started!

Your app is now running at URL: exp://192.168.1.16:19000

View your app with live reloading:

  Android device:
    -> Point the Expo app to the QR code above.
       (You'll find the QR scanner on the Projects tab of the app.)
  iOS device:
    -> Press s to email/text the app URL to your phone.
  Emulator:
    -> Press a (Android) or i (iOS) to start an emulator.

Your phone will need to be on the same local network as this computer.
For links to install the Expo app, please visit https://expo.io.

Logs from serving your app will appear here. Press Ctrl+C at any time to
stop.

› Press a to open Android device or emulator, or i to open iOS emulator.
› Press s to send the app URL to your phone number or email address
› Press q to display QR code.
› Press r to restart packager, or R to restart packager and clear cache.
› Press d to toggle development mode. (current mode: development)
```

1. Run it on your non-tethered device

The "q" option



Expo Client
for Android & iOS

Run your projects before you deploy. Open projects by scanning QR codes. If you need to, Download IPA 2.8.1 or Download APK 2.8.0.

Star

iOS App | Android App

Install the Expo client app on the device on which you want to test

- http://expo.io
- Apple App Store
- Google Play Store

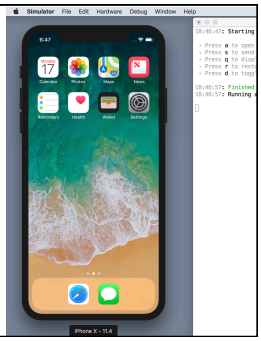
- Displays QR Code that you can scan on your connected device that has the expo app installed.
- Thus the device/emulator must be on the same network as the server (aka. your development machine)

2. Run it in an iOS Emulator

The "i" option

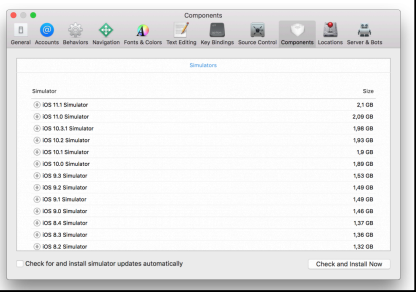
To use the iOS simulator you must ...

- Be on a Mac
- Have xcode installed and configured
- Have iOS simulator installed and configured



To install the iOS simulator ...

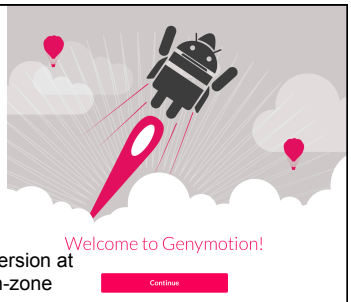
- Open Xcode
- Go preferences - components - simulators - Pick one.



3. Run it in an Android emulator

The "a" option

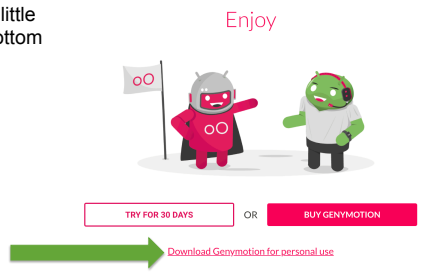
The official Android emulator is Genymotion



Install the free, personal version at http://Genymotion.com/fun-zone

Register with a real email address

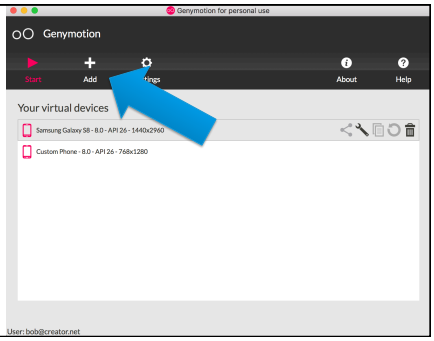
- Click on the little link at the bottom



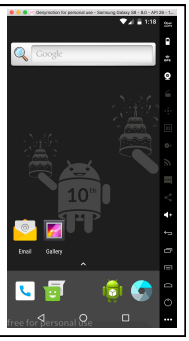
Genymotion needs VirtualBox

- Go to virtualBox.org and hit the comically large download button
- Your OS may block the install for security violations but when you put an exception on it, it installs okay.

Install one or more devices to emulate

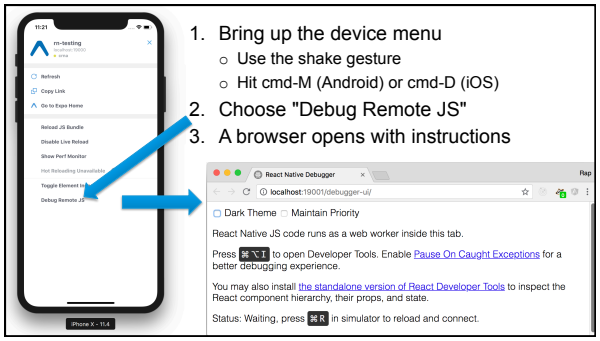


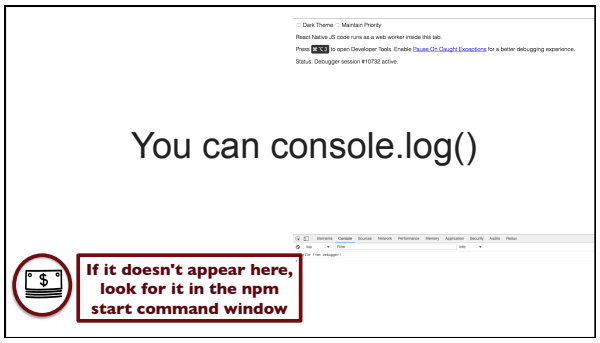
- Once installed, open the emulator and then hit "a"
- Your app opens in the emulator

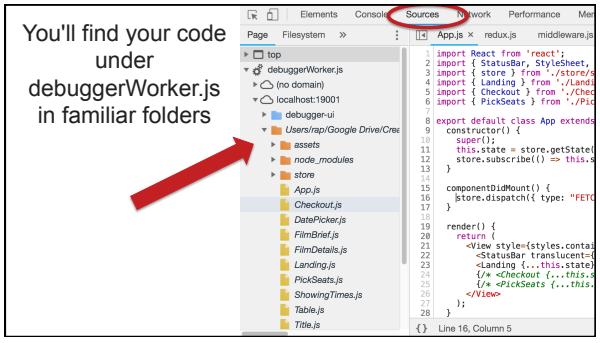


Debugging

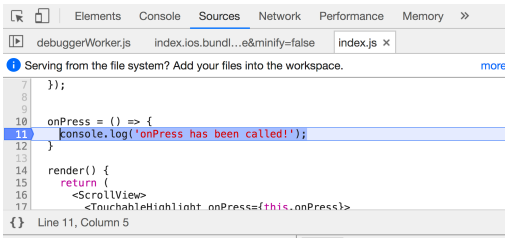
- Normally your app runs JavaScript in the device's engine called JavaScript core.
- But if you enable debugging through an emulator, Expo can connect the emulator to a browser, using the browser's JS engine. Therefore, you can use the typical browser's developer tools!
- console for console.log(), error(), etc
- JavaScript source debugging
- Network for examining Ajax requests and responses


1. Bring up the device menu
o Use the shake gesture
o Hit cmd-M (Android) or cmd-D (iOS)
2. Choose "Debug Remote JS"
3. A browser opens with instructions
React Native Debugger
localhost:19001/debugger-ui/
Dark Theme
Maintain Priority
React Native JS code runs as a web worker inside this tab.
Press ⌘⌥I to open Developer Tools. Enable Pause On Caught Exceptions for a better debugging experience.
You may also install the standalone version of React Developer Tools to inspect the React component hierarchy, their props, and state.
Status: Waiting, press ⌘R in simulator to reload and connect.


You can console.log()
If it doesn't appear here, look for it in the npm start command window

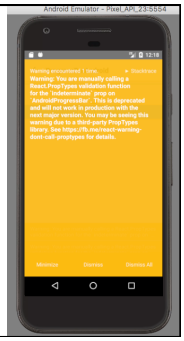

You'll find your code under debuggerWorker.js in familiar folders
Elements
Sources
Network
Performance
Page
Filesystem
App.js
redux.js
middleware.js
top
debuggerWorker.js
(no domain)
localhost:19001
debugger-ui
assets
node_modules
store
App.js
Checkout.js
DatePicker.js
FilmBrief.js
FilmDetails.js
Landing.js
PickSeats.js
ShowingTimes.js
Table.js
Title.js
Line 16, Column 5

You can set breakpoints



If there are warnings you see a "YellowBox"

Tap the alert to show details or to dismiss them.



If there are errors, you see a "RedBox"

- This can be dismissed.
- It can be started with console.error();

tl;dr

- Your app must run in a device during the development process
- Expo is an emulated environment. You can run it ...

1. on a physical device (with Expo installed) by scanning a QR code
2. in an iOS emulator after installing xcode and the iOS emulator on your Mac
3. in an Android emulator after installing Genymotion and Oracle VirtualBox

- ... And there are actually more options