



Remote Debugging Javascript

There are times when the mobile version of an app doesn't work but the actual version does, or we might want to run the app in an actual device to see how it works.

We need a way to run the code on a real mobile device so we can debug code issues and see how the app works under real mobile device conditions.

People sometimes suggest I should calm down about web performance because the web feels fine on *their* phone. [@DasSurma](#) and I pulled a little data to try to visualise why that might be both true and not particularly relevant: pic.twitter.com/9EfoaEJLgf

— Alex Russell (@slightlylate) [June 15, 2019](#)

Ideally you'd test with a device that reflects your users' and, sadly, that's not trending up like it should. The average device is not the iPhone or high-end Android device in most developer's pocket. It's more likely to be a lower powered multi-core device.

I won't go into the details of why mobile performance in mid and lower range of devices is not good. Alex Russell has a great [blog post](#) and a [presentation from 2016](#) that covers this topic.

The post will look at how to use the browser's DevTools to debug issues.

Safari in iOS on macOS

iOS has something that makes it hard to debug browsers, other than Safari. There are none. All browsers in iOS other than Safari are skins on top of WebKit WebView, similar to Safari but less powerful. So we'll start by looking at how to debug Safari in iOS using Safari's DevTools.

To get your iOS mobile device to work with Safari in macOS, follow these steps:

1. Connect your device to your computer via USB
2. Unlock the device
3. In Safari, go to the Develop menu. You should see the device name listed
 1. If the device is locked you will see a grayed out Unlock Device with Passcode message
4. In your mobile device navigate to your app
 1. Once the device is unlocked you should see the app and the associated service worker if one is running for the site/app
5. Click on the site name. That will open the developer tools on your desktop browser
 1. You can use the elements panel to inspect and interact with the page
 2. You can use the console to modify the content of the page

Safari on macOS with Chrome

Working with iOS and Chrome used to be possible using the [ios-webkit-debug-proxy](#) tool to proxy DevTools (Chrome Remote Debugging Protocol) for iOS devices (Safari Remote Web Inspector).

I could not get ios-webkit-debug-proxy to work on the current version of macOS with the current version of Chrome. Unfortunately, the tool is no longer actively maintained and the author (or one of the authors) has moved to a paid solution (<https://inspect.dev/>) with a 30-day trial.

So there's no free way to debug from an iOS device to a Desktop DevTools setup.

Debugging Chrome for Android in Chrome for desktop

Since I don't have an Android Device to test with, I'll defer to Kayce Basques [Remote debug Android devices](#) in the Chrome Developers site.

I'm hoping to get an Android device to test this with soon.