

From: Kevin Carroll [REDACTED]
Sent: 20 July 2022 05:44
To: BrowsersAndCloud
Subject: Comment on CMA investigation into Mobile browsers and cloud gaming

Dear CMA,

I am writing as a web developer and entrepreneur that works with web technologies on a daily basis. I would like to express my concerns and frustrations in dealing with Apple's ban on outside browsers on their IOS platform. Apple has been dragging its feet for years when it comes to including essential features into its mobile Safari browser. Some of which have already been implemented by other browser companies for many years now. As a result, Apple has been holding back web technologies and doing a disservice to consumers who would benefit from being able to use the simplest solution (like a web app), rather than having to install another app on their phones and tablets. I think the reasons why they are stifling the web are quite obvious, they want their app store and other ecosystems to be the only game in town when it comes to delivering a great customer experience on their devices. However, there are many instances where a web application (PWAs) would be much better suited instead of an native app. Not only that, but if a web application can deliver the same great experience as a native application, while simultaneously removing any friction associated with having to go the app store and install another app, then using web apps should be a no brainer.

Since Apple does not allow any other browser engines to run on their devices and because Apple's market share is so prevalent, web developers often have no other choice but to create an IOS app. This can waste a lot of time and resources that could be better spent working to improve their applications rather than having to spend so much time and energy, not innovating, but recreating their existing app so it will be able to run on Apple's devices. If Apple allowed other browser engines, like Chrome and Firefox, then web developers would have the option to include "enhanced features" on these browsers and inform their users when they are running in Safari to switch if they would like to experience any of these enhancements. This would at least give developers an avenue where they would still be able to deliver the great experiences the web has to offer without having to create a native application just so they can use features such as web push notifications, which has become so essential to applications.

I believe that Apple is doing the web a great disservice by acting in such a way as to slow the progression of some great technologies and possibilities so it can line its own pockets and solidify the great moat it has been building around its business. For example, Apple just recently announced that it would add support for the web push notification API on desktop Safari with the release of its newest operating system later this year, and IOS will also get this feature in early 2023. That is an amazing announcement, and it's unbelievably wonderful news given the fact that it has taken 8 years for them to support something that other web browser makers have been supporting since 2015. This is such a basic feature, it's really table stakes for applications in general. The ability to notify your users when a certain event occurs is essential for any app maker to be able to provide to their customers. Real-time communication has become such a basic need that you can't even run a business if you can't deliver this feature to your users. This is shutting out entire classes of applications from running on IOS inside of the browser. That is truly ridiculous, and it highlights the very sad state that web applications are currently in.

I am not confident that Apple will suddenly have a change of heart and start doing the right thing for web developers and their customers. That is why I am urging the CMA to use its power and take a very hard look at Apple's actions. They have been treating web applications like second class citizens for far too long and without some sort of regulation I fear they will continue on this path without consequence.