

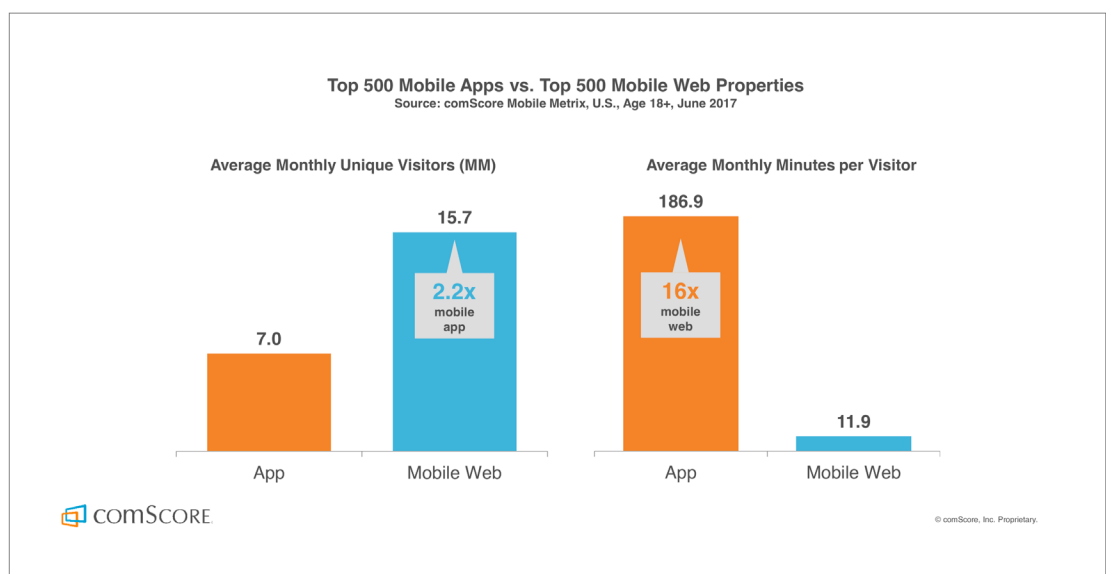
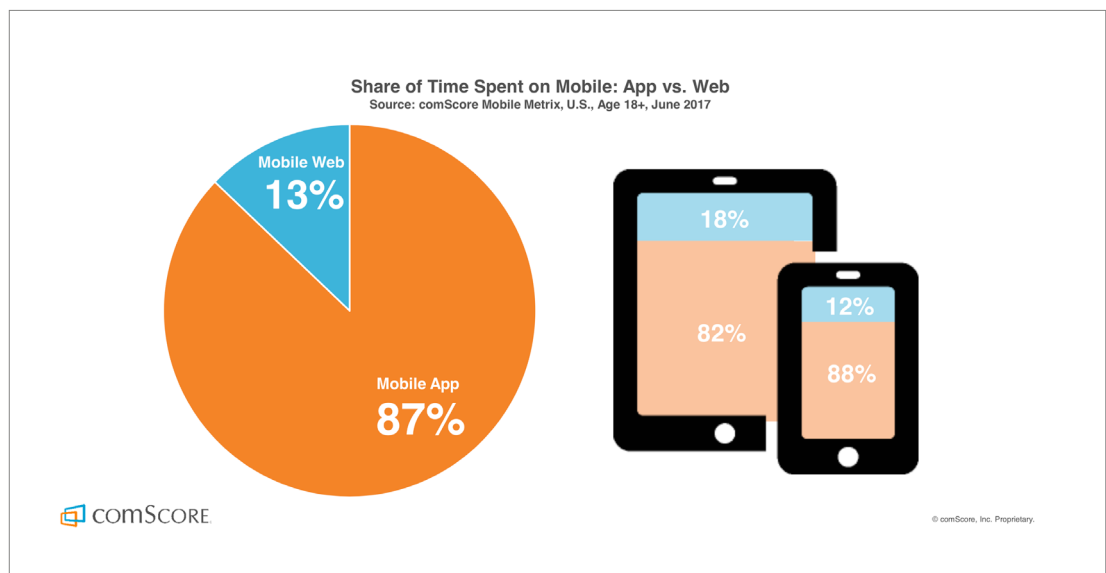
Web vs Native app

White paper

Introduction

Before we start elaborating on the different approaches, we should first have a look at the numbers. Those numbers have remained more or less the same since 2016. Users spend 87% of the time on their mobile device within apps, and only 13% in a browser. On phones it is actually 88% vs 12%, and on tablets it's 82% vs 18%.

Mobile websites reach 2,2 times more users than mobile apps but users spend 16 times more time in apps than they do on web sites. This means you need a web site to attract new users and you need an app to retain this user and to engage with him/her.



What is a web app?

A web app is basically a responsive web site displayed in a browser on a mobile device. It can be optimized to work well on touchscreens and it can be saved so that it has its own icon on your home screen.

It is not available in the app stores, and cannot access the hardware of the device, it does not work offline and the behavior is not the same as that of a native app.

In terms of development, it requires custom development and optimization for different browsers. When starting from a responsive web site, it will also require extra development to optimize for mobile.



What is a progressive web app?

A Progressive Web Application (PWA) is an app within your browser (Google Chrome only at the moment), displayed as a web site. You can't download and install it from Google Play or from Apple's App Store.

It's a link to a website that will install an icon on your home screen. Once you tap the icon it will open a browser in full-screen view without the navigation bar. You will only be able to navigate with the web site navigation.

Content for PWAs can be pre-downloaded in Chrome making most of the content immediately available.

For Google Chrome to recognize a web site as a PWA, specific HTML and JavaScript coding is required.

It can have limited hardware access, can send push notifications in the browser, and can work offline if the content had previously been downloaded.

It is a possible alternative for an Android app but it does not work on iOS.

In terms of development it requires once again custom development, with the limitation being that only Android devices are supported.

Now developers of PWA's are claiming a multitude of advantages in comparison to native apps. So let us take a look at those so called advantages and put our arguments against it.

PWA	Native app
<p>Low friction of distribution/ Universal access:</p> <p>If your PWA is online it is already accessible for Android. PWAs eliminate the need to go to the app stores, search for the app, click install, wait for the download and then open the app.</p> <p>A PWA is not subject to the sometimes arbitrary policies of the app stores or to geographic restrictions.</p>	<p>Yes, a PWA may be easy to install. But which source do you trust the most to install on your device? An app from the App Store or a just a web link?</p> <p>The App Store stands for quality, this is why it's not so easy to get through the approval process but in the end the user appreciates the quality.</p> <p>On Google Play the quality standards may not be the same, but at least there is some form of control as well.</p>
<p>Discoverability:</p> <p>Content in progressive web apps can easily be found by search engines but the content in a content-centric app won't show among app store search results.</p>	<p>Is this really an argument?</p> <p>If you are publishing content, you will normally have a web site where you publish the same type of content and people will find this content as easily as they do if it is published in a PWA.</p> <p>If you don't have it available on the web then you can use Twixl Publisher to create a browser client version and thus make it available.</p> <p>On the other hand search results for apps are always listed first so if someone is looking for your app via a search engine he will definitely find it, which is not the case with a PWA. Try it for yourself!</p> <p>Native apps can be published in the app stores and users tend to search in those app stores for their brands or magazines.</p> <p>You cannot publish a PWA in the app stores so users might search for you but may not find you at all.</p>

PWA	Native app
<p>Low development cost:</p> <p>The most important advantage of developing PWAs is that developers no longer need to develop multiple apps across multiple platforms. PWAs offer the promise of being able to develop a single app and have it work perfectly across all your platforms and devices.</p> <p>Instead of Desktop/mobile web + native Android + Native iOS it becomes just a PWA.</p>	<p>With Twixl Publisher you are using a solution to create your native app and you only need to concentrate on your content and how you are going to introduce it to your users/readers.</p> <p>Creating and managing your content is also something you'll need to do using a PWA.</p> <p>So with Twixl Publisher there is not really an extra cost involved to have a native app.</p> <p>Most of our customers have their own web site in place with a CMS and often a specific content editor. So in order to create a PWA they should throw all those systems out and start over. Sounds convincing!</p>
<p>Responsive:</p> <p>Because it's HTML all web pages can be made responsive, considerably lowering the creation cost.</p>	<p>This is absolutely correct, this is why we also support HTML content in native apps made with Twixl Publisher. Using an HTML editor you can create responsive layouts. You define your templates once and you just need to enter your content.</p> <p>But the reality is different, not all content our customers want to publish is HTML-based. They still have plenty of InDesign-based content and often also PDF content. How are they going to publish this content in a PWA? And if they manage to do that, how are they going to make it responsive?</p>
<p>Instant loading:</p> <p>The pages within a PWA are immediately available at least if they were already pre-downloaded.</p>	<p>In an article-based Twixl app, we offer exactly the same approach and we keep optimizing it.</p>

PWA	Native app
<p>Simple updates:</p> <p>With no app to be downloaded and updated every time you make changes to your services, a progressive web app can keep all your users up to date. It enables them to immediately enjoy the latest functionality, features, and security updates without the need to re-download the app. Using a PWA largely eliminates the problem of software “fragmentation” that may force a company needing to extend legacy support to old versions or risk cutting off sections of the user base until they initiate an update.</p>	<p>Strange argument coming from a company that is providing the Android OS.</p> <p>Anyway, in an article-based Twixl app, any content update is instantly available within the app without the need to submit whatever app update to the store.</p> <p>However we do advise our customers to update their apps regularly (software updates) because the environment in which they operate and with which they interact keeps changing.</p> <p>We think that an app update shows the involvement of the brand or publisher to his users and readers and also helps to remind them about the app. So essentially we see this as a positive thing.</p>
<p>Linkability:</p> <p>Any page/screen can have a direct link, which can be shared easily.</p>	<p>In the Twixl browser client, every article has its own URL as well. That way when a reader wants to share an article from the app he will automatically share the weblink. On the webpage he'll be able to read the article directly but he will also have a link to the app in the app store.</p>
<p>Large data savings:</p> <p>You only download what you read.</p>	<p>This is the same approach for article-based Twixl apps. Working publication-based with big size downloads is outdated.</p>

PWA	Native app
<p>Secure:</p> <p>HTTPS secures the connection between the publisher and the reader ensuring the users information is protected and isn't tampered with.</p>	<p>A native app is an island on the device of the user (this is called 'sandboxing'). It is a separate container that is not capable of infecting any other part of the system on your device.</p> <p>That is what we mean by secure!</p> <p>Every app that is published in the app stores is also checked by the review teams of those stores, adding an extra layer of control.</p> <p>User data is also transmitted via HTTPS from the device to our servers and to external servers.</p>
<p>PWAs can also send push notifications but only in a browser. They don't have access to the device ID.</p>	<p>Push notifications:</p> <p>With native apps you send push notifications to the device and not to the browser. You can also send targeted push notifications in relation to the behavior of the reader/user.</p>
<p>PWAs offer this possibility too but you need to download the content you want to be able to consult offline page by page.</p>	<p>Working Offline:</p> <p>Native apps can offer the possibility to download content in the app that can later be accessed offline. With Twixl Publisher we even offer different options to the user and the publisher to manage that aspect.</p>
<p>PWAs don't have access to the App stores and thus cannot use their store system. They can connect only with online stores from the publisher itself.</p>	<p>Subscriptions and in-app purchases:</p> <p>The app stores allow the publisher to sell subscriptions and to purchase individual issues. Native apps can use this, but they can also integrate with the web shop of the publisher, offering the best of both worlds.</p>

PWA	Native app
<p>This is correct but step by step Chrome keeps on adding new possibilities. In the future there may be less of a difference.</p>	<p>Hardware access: Native apps have better access to the hardware of the mobile devices than PWAs do.</p>
<p>Google hopes one day Apple will add PWA support on Safari...</p>	<p>What about iOS? PWAs don't work on iOS! So basically you only have a solution for half of the users if you replace your Android app by a PWA.</p>
<p>A PWA is just trying to reproduce that native behavior in an environment made for a click-behaviour. A replica is never as good as the original.</p>	<p>Native behavior: Native apps offer the behavior users have come to expect from their device, using the phone functionality, taking a picture, chatting, ... It's just the natural feeling.</p>



Why is it Google that is promoting PWA's?

The fact that users are spending more than 60% of their digital time on mobile and that 87% of that time is in apps is a huge threat for their business. Much of the content shown within those apps is not on the web!

Nearly all earnings for Google come from advertising. Ads within search results and ads on sites that are on the Google network. If the content is not on the web, they can't make money from it. It is the source of their existence that is in danger!

Now that mobile has become so important in such a short period of time it is also a threat for a lot of web developers. You could say that for each mobile developer you have about 20 web developers on the market. So let's use that force to be more vocal and say that web is better than native. If 20 people say web is better and only 1 says native is better then it must be true, right...?

Google added support for downloading content for PWA's in Chrome, providing the advantage that the content can be accessible immediately if it has been downloaded already. An approach Twixl is also using in article-based apps but there it is handled more server side. With this move Google is trying to bring more content to the web and more users to Chrome.

Perhaps there is also another reason why Google is promoting a web approach instead of a native app approach.

Android devices may be popular (approximately 2/3 of the mobile devices worldwide are on Android), but there is a complete fragmentation when looking at the versions of Android that are running on those devices. This makes it extremely hard to develop for, as features that are supported in the latest version may not have been supported in a previous release and so on.



Let's compare iOS with Android

iOS 11 was released in September 2017. In January 2018, 65% of users were on this release and 28% still on iOS 10. The 2 latest releases of iOS thus comprise 93% of the devices. Android 8 was released in August 2017. In January 2018 the adoption rate was 1,1%. For Android 7 it was 28,3 %. This is 29,4% for both releases together. Even worse, today still 12% of users are on Android 4.4, originally released in October 2013. Meaning that when you develop for Android today, you still have to develop for a platform that is more than 4 years old.

This is a real issue for Android, making the platform definitely less popular for developers so Google had to find another way to approach the mobile market. Or is it the developers that have to find another approach to publish on Android devices?

It is about the content and the cost to publish to different media

We are comparing technologies but it all starts with what kind of content the publisher or brand wants to publish. Where does the content come from? Is it already coming from the web?

Most of the time it isn't. It's available in print or as a PDF. Most of the time it is a mix of different types of content and this is why Twixl Publisher offers a flexible approach.

All our customers create specific print content and know how to get it printed. Most of our customers have a solution in place to go to web and create content specifically for the web. They have a separate content flow for web and print.

When publishing to mobile, you may decide to do this starting from the print content or starting from the web content. Then again, you start wondering if you shouldn't use both types of content. We think the content is central and from there you can start outputting to different media. Nevertheless, we have to be able to handle the setup and workflows that our customers are using today.

We offer a solution to publish to both mobile and the web with the same content, and even with content coming from print. Twixl Publisher offers the combination of native apps and the browser client.

If the content is fragmented, coming from different sources, we are flexible. We can publish PDF content, enriched InDesign content and using our API you can push content from a (web) CMS.



We definitely believe in the native approach, providing a native behavior, a wider range of possibilities and respecting the expectations of the user.

Does this bring an extra development cost? When using Twixl Publisher... not at all. On the contrary, the subscription cost for Twixl Publisher will always be far less than the cost of custom development and maintenance of any web alternative, that may only work on one medium and will not allow publishing of print content.

The challenge is to lower the production cost and to find one format to publish to print, to web and to mobile, adapting the layout per medium and being able to offer a captive reading experience. You need to adapt to the medium and to the user.

Respect the medium, respect the user!

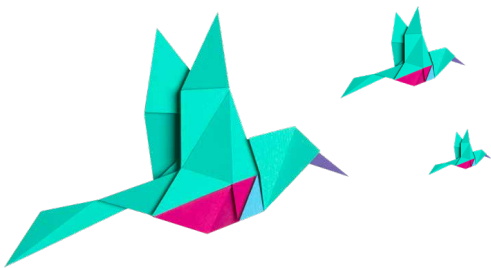
This is again why we believe in our flexible approach allowing to publish a combination of different types of content. Responsive web content with a lower creation cost, PDF content and enriched InDesign-based content with a separate layout for tablets and phones. Finding the ideal combination is the challenge.

What if the customer requires certain specific functionalities?

Then you may need some custom development.

Today Twixl Publisher apps can already integrate a lot of existing web functionalities. In the future we will open up our platform so you can more easily access certain hardware functionalities of the devices from within HTML-based content. While this may also be possible with custom development, we intend to offer easier built-in access in our solution to use those hardware functionalities.

When compared to creating and maintaining a custom app, the advantage of using Twixl Publisher in combination with an easy way to access specific hardware-related features cannot be underestimated! The cost savings will be huge.



Conclusion

- ✓ Whether it is a 'standard' web app or a progressive web app it requires custom development which you don't need with a solution like Twixl Publisher, that is creating the native app for you.
- ✓ A web app is just a link on the home screen and offers no real valuable alternative to a native app.
- ✓ A progressive web app is only an alternative on Android so that's replacing one solution by another.
- ✓ Publishing content to different media requires a flexible and open solution and not a solution only capable to publish web content.
- ✓ Even if you require certain specific features, Twixl Publisher will still be the best approach from an economic point of view and because it is a native and flexible solution, that lets you publish to both mobile and the web.