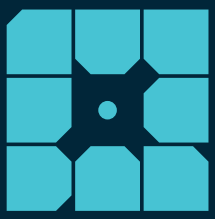


WP engine®

MAXIMIZE MOBILE ENGAGEMENT WITH PROGRESSIVE WEB APPS & WORDPRESS

WHITE PAPER

Introduction

Earlier this year, Google announced a new classification for websites: Progressive Web Apps. Previously the terminology had been used, but standards for what makes a web application "Progressive" had not yet been defined.

So what is a Progressive Web App (PWA) exactly? Google has released a list of requirements, as well as a browser extension that checks whether sites adhere fully to them. Though the list may be daunting at first, don't stress just yet. In this article we'll cover the major pillars of a PWA and how WordPress factors into the equation.

Why use a PWA?

Before we dive into the nitty gritty details of what qualifies a website as a Progressive Web App (PWA), it's important to call out the reasons why making a PWA is valuable to users. Firstly, studies showed that attaining and maintaining users from a native mobile application was extremely difficult, with steep user dropoffs. And yet mobile websites weren't able to engage users in the same way that native mobile apps were - they weren't able to get users to come back to the website, and experiences were heavily dependent on network speed. Often times the website, being built for desktop clients, was clunky and unusable on mobile devices.

The void created by these needs has been filled by the Progressive Web App, which offers users the experience of a native mobile app right in their web browser. Test results have shown huge success rates in both new user growth and engagement. In creating their PWA Forbes yielded a 43% increase in sessions per user, and their average session duration doued. And BookMyShow saw an 80% increase in conversion rates thanks to their new Progressive Web App. There are countless examples of success in user engagement to peruse if you need more convincing, all featured in Google's Showcase.

Core tenets of the PWA

Google has defined the Progressive Web App as "a new way to deliver amazing user experiences on the web," and must be **reliable, fast, and engaging**. While these are all positive, happy words, Google doesn't really get specific about **how** developers should make a PWA. With that in mind it may take some digging to get to what the PWA literally is, so we'll break it down.

In short, a PWA is a website that has mobile app-like functionality such as:

- Super fast loading speeds
- User interface which mimics a native mobile app
- Unique designs tailored to mobile and tablet devices
- Additional security with SSL enabled on all pages
- The ability to add the website to the home screen for the mobile device
- The ability to use the website even when offline
- The ability to send push notifications

Some functionality required to be a PWA is already something a **responsive** website includes, such as the mobile and tablet layouts. WordPress has made many strides in the past years towards enabling better mobile experiences such as the use of srcset, allowing for scaled images to be displayed on the proper devices. And with free Let's Encrypt SSL certificates from WP Engine, the security requirements are an easy win. But there are certain aspects to a PWA that WordPress websites wouldn't necessarily meet out of the box. This is where the "app" classification comes from. Because a PWA has functionality similar to that of a native mobile app, it makes the website more accessible to users both in terms of ease of use, and in accessibility.

So how do users get a true PWA experience from a WordPress site? The answer comes in the form of a buzzword around the WordPress community of late: **Headless WordPress**. Users can create a JavaScript App front-end which hooks into the WordPress REST API to grab site content and data.

Google doesn't get specific about how the PWA should be built, but they do specify that it should be constructed with JavaScript. That means your front-end can be built on tools like React, Angular, Vue, or many others. There are even PWA-specific tools like Gatsby which exist specifically for developing Progressive Web Apps on top of WordPress and other CMS providers. For specifics about how to build a Headless WordPress site, check out our Headless WordPress on WP Engine whitepaper.

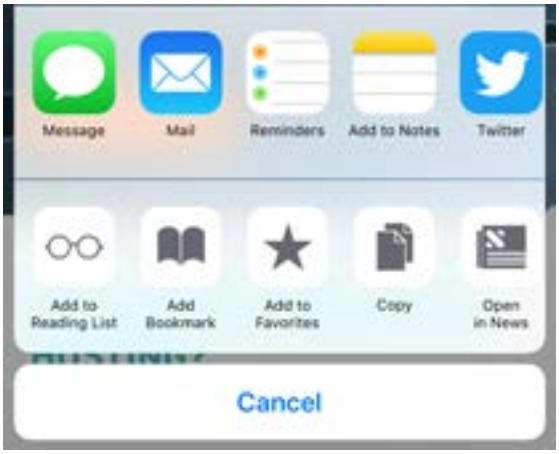
Attaining app functionality

Some of the attributes of a PWA are app-specific, and might be foreign if you're used to working within a fully native WordPress website. Let's dive into some of these unknown attributes and how they can be accomplished.

All URLs load offline

Loading content offline is something an app downloaded from the app store can certainly do, but most of the time it's unheard of for a website in a browser, mobile or otherwise to have this functionality. Part of the goal of the PWA is to create more meaningful mobile experiences, and that means loading content even when you have a slow connection or no connection at all.

So how can you accomplish this goal of the PWA? By using a service worker. The service worker is a browser-side JavaScript worker which controls how network requests are made. Most importantly, it allows websites to pre-cache important site resources, which reduces or eliminates your end-users' dependency on their network. To avoid abusing web server resources the service worker is not continuously running, but instead terminates when idle. The service worker aspect is the biggest reason why HTTPS is a hard requirement - without HTTPS encryption, service workers allow for connections to be hijacked. There are also a handful of need-to-know caveats which might affect your worker and/or workflow, so be sure to check out the full documentation from Google.



Metadata provided for Add to Home screen

This requirement isn't as tough as the last. It simply means that your PWA needs to have a Web App Manifest. The Web App Manifest is the JSON file containing metadata surrounding the display of your icon on the end-user's home screen when it is bookmarked. While really any website can have a Web App Manifest file, it is only required for Progressive Web Apps.

First load fast even on 3G

Since websites are at the mercy of the network speed they are accessed from, a fast load on 3G can be tough ask from developers. Limiting the number of network requests and ensuring asynchronous JavaScript downloads are great first steps. But this is where the JavaScript front-end really comes in handy - the very nature of a JavaScript web app means lightning fast speeds. Add in the capabilities of a service worker to pre-cache valuable site resources, and your dependency on fast network speeds is suddenly a remnant of the past. You can use tools like WebPageTest to test performance on specific mobile networks.

Page transitions don't feel like they block on the network

Another way to look at this requirement is that when users navigate to a different screen or page, it should feel quick shouldn't lag. Google hasn't made it a secret that they prefer Progressive Web Apps not be a single-page application, but rather a site with navigable, individual pages. But they also require that these pages don't slow down or inhibit the user experience. Headless WordPress implementations accomplish this inherently by only making an single WordPress REST API call when new pages are requested. And since only JSON is requested and returned in this process, the response is lightning fast.

Why is the PWA Important?

The Progressive Web App is a specific solution to help meet an ever-widening base of needs from mobile users. These apps bring together the best of both worlds in providing mobile users an easier way to access content, and providing a higher level of interaction to keep them coming back. The results speak for themselves: a true Progressive Web App will help improve user engagement and many times conversion rates as well. In a world where mobile web traffic now surpasses desktop, it's increasingly important to ensure your website is rich and engaging for all devices. With many JavaScript programming languages at your disposal, the question of whether to create a Progressive Web App becomes less "Why?" and more "Why not?"

About WP Engine

WP Engine powers amazing digital experiences for websites and applications built on WordPress. The company's premium managed hosting platform provides the performance, reliability and security required by the biggest brands in the world, while remaining affordable and intuitive enough for smaller businesses and individuals. Companies of all sizes rely on WP Engine's award-winning customer service team to quickly solve technical problems and create a world-class customer experience. Founded in 2010, WP Engine is headquartered in Austin, Texas and has offices in Limerick, Ireland, San Francisco, California, San Antonio, Texas, and London, England. www.wpengine.com

WP Engine 504 Lavaca Street, Suite 1000, Austin, Texas 78701 info@wpengine.com wpengine.com