Exploring the role of progressive web applications in modern web development

V S Magomadov
Faculty of information technology, Chechen State University, 32 Sheripov Street, Grozny, 364024, Russia

E-mail: vmagomadov@gmail.com

Abstract. This paper explores such a Web development technology as Progressive Web Applications. It is explained what this technology is and the purpose of it. The paper also compares progressive web applications to other Web technologies that are closely related to them, such as responsive websites and native applications. It highlights the most important differences and similarities between them. Furthermore, the paper discusses the advantages as well as the possible limitations of this technology. Finally, it provides some thoughts on whether or not this technology is the best solution for the online presence of a business.

1. Introduction
Today, a website is an integral component of any business, and in order to be able to compete, companies should consider incorporating the latest advancements in Web development into their online presence, making the user experience as seamless as possible. One such advancement is known as Progressive Web App which has gained some popularity in the last couple of years. These days, companies often request their Web developers to create a website that supports the PWA functionality, so being able to develop this kind of website is becoming an essential skill for Web developers.

At present, this technology is being widely discussed in the Web development community. Some say that it is not necessary to make PWAs because the existing Web technologies are already sufficient for making websites with the same capabilities. For instance, they argue that making a responsive website or a native application is enough to avoid using PWAs. While others argue that the aforementioned technologies lack the functionality available in PWAs. However, it is important to give some insight into what PWAs, responsive websites, and native apps are before delving into any of those arguments.

1.1. Progressive Web Applications
The purpose of a PWA is to give you the same user experience as a native mobile application. PWAs are not websites in the traditional sense but they are websites nonetheless. The technologies used for building PWAs are as follows:

- HTML5, CSS3, and JavaScript are used for building the user interface which is the static part of a PWA. This user interface, which is known as the app shell, is cached on the user’s mobile device so that it can be loaded without any delays which may take place because of the bad network connection.
• The client side uses web service calls for loading and rendering the content just like the way it is done in native apps. This helps avoid the classic scenario in which upon making each request, the user interface and data are put together on the server side and sent to the client all at once.
• A service worker is one of the most important technologies a PWA consists of. The service worker is JavaScript code that is run in the background by the browser. This code is not dependent on a web page which gives access to features that do not require a web page or interaction with a user.
• Such frameworks as Angular and React can be used for developing PWAs. These frameworks have some built-in service workers which make it unnecessary to rebuild them for every PWA.

Diagram 1 shows the architecture of a PWA.

1.2. Responsive websites
Responsive web design is all about creating websites that are able to fit into any screen size, whether it is a tablet or a mobile phone. These days there can hardly be found a professional website that does not meet this requirement, and it is considered to be a must. In fact, most of the users do not spend much of their time on a website that does not fit into the screen size of their device.

In order to make a website responsive, there is a set of tools that are commonly used, such as CSS media queries, flexible layouts and grids, and images. When the user switches from their laptop to a smartphone, the website automatically adjusts itself to the size of that device.

1.3. Native applications
Native apps are built for a specific device and can be launched using icons on the home screen of the device. They are usually downloaded from an application store and installed on the device. The most widely used application stores are Apple’s App Store and Google Play. One of the main characteristics of a native app is that it is able to use all the features available on the device. For example, it can use the device’s GPS, camera, sound, and many others. In addition, the native app can take advantage of the notification system of the device.

2. The difference between PWAs and responsive websites
The main difference between PWAs and responsive websites lies in the functionality, and there is no much of a difference in terms of the user interface. A PWA has webpages, which can be accessed through a URL, just like any other website (responsive or not). However, a PWA provides functionality that is more characteristic of a native app, and this is something a responsive website lacks.

Users are able to personalize PWAs according to their needs. For example, they can enable push notifications, access to the camera, and even add a PWA to the home screens of their devices the way they add native apps.
3. The difference between PWAs and native apps
One of the main differences between the two is that native apps are platform-specific, that is, they can only be run on an operating system that they were developed for. They heavily depend on the developer tools available in the operating system in order for users to be able to take advantage of their functionality.

PWAs mainly depend on the browser through which they can be accessed. They do not need to be developed for a specific platform since they can be run on any modern device which has a browser. They act like native apps in many ways but do not take control of the device’s hardware.

4. Advantages of PWAs
PWAs offer a number of advantages that businesses can greatly benefit from. Some of the main advantages will be described below.

- Cost-effective. Companies may be able to cut costs by developing a PWA for their online presence because there is no need to develop separate applications for Android, iOS, or any other operating system. In addition, PWAs also do not require to be constantly updated which can also save money.
- SEO-friendly. As it is the case with traditional websites, the content of a PWA appears in the search results which leads to a contribution to Search Engine Optimization. Through the use of the tools provided by Google, it is possible to check how a PWA ranks on the search results page.
- Not being dependent on application stores. In order to find a PWA, a search engine is used, or it could be offered through a social media page or any other channel. The point is that PWAs do not depend on application stores.
- Less amount of data is required. PWAs do not require to utilize as much data as regular websites which means they have a much better performance.

5. Limitations
PWA functionality requires more time and work for development. Developing a PWA means that it is necessary to develop a regular website and add PWA functionality on top of it. This means that Web developers have to invest more time and work developing those extra features. Furthermore, these features have not been standardized yet which means this may cause some issues when attempting to develop an MVP in a limited time-frame.

- Restricted access to the hardware of the device. When using an application through a browser, it is not possible to take full advantage of the hardware of the device. However, this is not an issue with the hardware but with JavaScript due to its single-threaded nature. This requires developers to rely on some workarounds for running native code that was written in C or any other language.
- Lack of support in older browsers. When it comes to PWAs, they mostly work well with modern browsers and older browsers do not have much support. Some companies still use legacy software and upgrading to new software is a difficult process that requires a lot of work, time, and money.
- Restricted support for iOS. While PWAs are supported by iOS (Safari), so far this support is not as great as what is available through the browsers on Android or any other major browser. Of course, this does not mean PWAs do not function well on iOS but the functionality is somewhat limited. For example, push notifications are not supported yet.
- Not being present on application stores. Not being dependent on application stores is mentioned above as one of the advantages but it may also be a limitation in a different context. The fact that PWAs are not present on these stores may result in their delegitimization. Application stores serve as a kind of social proof for applications that is achieved by means of reviews. Furthermore, they can catalogue apps and provide users with the most relevant apps according to their search. The majority of the users search for their desired apps right on these application stores, and because of this a company may miss out an opportunity to reach some of their target audience if they have a PWA instead of a native app. Thus, not appearing on application stores may be a disadvantage in some circumstances.
6. Discussion
This paper has shown that PWAs have a lot of advantages which makes them a good option for many businesses. They may be more cost-effective compared to other solutions, and it is quite possible that they are here to stay. However, the fact that they have not been around for too long compared to other well-established technologies, which were mentioned above, makes it harder to fully embrace them. Since native applications have existed for a relatively long time, software developers are more experienced in this area and may be able to deliver a product faster. There is no guarantee that a PWA is going to be a less expensive option until it has as much support as other Web technologies. Thus, due to these constraints, PWAs may not be the better solution for some companies yet but they do have the potential to be one shortly soon.

7. Conclusion
It is clear that PWAs have much potential to become an invaluable tool in Web development even though they have not yet been widely adopted by companies which may seem unusual to some people considering the advantages of this technology. It is difficult to say whether PWAs may replace native apps and responsive websites or not. For some kinds of companies, it may be best to have a native app. However, as users keep encountering PWAs while surfing the Web through their browsers, they may get accustomed to this native app-like user interface that is available to them without having to install anything on their devices. In the best-case scenario, users will go to the application stores for certain apps that should probably remain native. For example, it may be a better idea for apps related to games, messaging, social media, and entertainment to remain native. For other Web-related activities, users will take advantage of PWAs.

References
[1] Rojas C 2019 Building Progressive Web Applications with Vue.js: Reliable, Fast, and Engaging Apps with Vue.js (New York: Apress) p 31-2
[2] Love C 2018 Progressive Web Application Development by Example: Develop Fast, Reliable, and Engaging User Experiences for the Web (Birmingham: Pakt Publishing Ltd) p 43-4
[3] Callaghan M D 2020 Developing Progressive Web Applications with Angular (and Ionic): How to Build and Deploy Mobile Applications Without Paying Apple or Google for the Privilege (Seattle: Amazon Digital Services LLC – KDP Print US) p 56-7
[4] Blokdyk G 2020 Progressive Web Applications a Complete Guide (Brisbane: Emereo Pty Limited) p 47-8
[5] Kolce J, Buckler C, Seitz S, Brown M, Jelisejevs P, Mardan A, Green D M and Rometty J 2017 Modern JavaScript (Melbourne: Sitepoint) p 117-8
[6] Lewis S and Dunn M 2019 Native Mobile Development: A Cross-Reference for iOS and Android (Sebastopol: O’Reilly Media, Inc) p 87-8
[7] Novick V 2017 React Native – Building Mobile Apps with JavaScript (Birmingham: Pakt Publishing Ltd) p 26-7
[8] Frain B 2020 Responsive Web Design with HTML5 and CSS: Develop Future-Proof Responsive Websites Using the Latest HTML5 and CSS Techniques (Birmingham: Pakt Publishing Ltd) p 53-4
[9] Ranjan A, Sinha A and Battewad R 2020 JavaScript for Modern Web Development: Building a Web Application Using HTML, CSS, and JavaScript (New Delhi: BPB Publications) p 131-2
[10] Sheppard D 2017 Beginning Progressive Web App Development: Creating a Native App Experience on the Web (Tinley Park: Apress) p 21-2
[11] Chivukula S R, Iskandar A 2019 Web Development with Angular and Bootstrap: Embrace Responsive Web Design and Build Adaptive Angular Web Applications (Birmingham: Pakt Publishing Ltd) p 117-8