

Implement and Evaluate the progressive web application

Hout Tang

April 13, 2022

Abstract

This paper will mainly discuss about the theory and definition of progressive web application and way to implement and evaluate progressive web application using React and other user interface and discuss the possibility that progressive web application could be the future of mobile development.

Introduction

Progressive web apps (PWAs) are web applications that combine continuous improvement with service employees, interfaces, and other internet characteristics to provide users with an interface comparable to native apps. In other words, the users will no longer download the applications from the app store or Google Play. They can load and run in the web browser [Banb]. PWAs use a combination of site and native app techniques and principles to enjoy the benefits of both internet and native app capabilities. Through PWA, many websites are readily available on the mobile phone as a mobile application, and sometimes they work when we have no internet connection. The use of many websites has become so convenient because of PWA. Through PWA, many native programs are more tightly connected with the software and provide a more smooth user experience. Users prefer clicking their icons to access their favorite apps quickly rather than going to them through a browser since native apps operate offline. PWA developments utilize a vast community of Web-based tools and practices that Android or iOS devices developers can only hope of. They are simple to create and only support one version of the software. It works for all users regardless of browser preference. The application will run across a different browser and will gently decline when specific features are unavailable while providing a good customer experience [Mal+15].

Discuss the theory and definition of progressive web application

In conventional development, software reuse of code across native apps, the internet and handheld devices has been non-existent. It is due to native apps' non-interoperable software systems, leading in independent projects and production settings when compatibility for different major mobile operating systems is sought. Cross-platform programming has become a common choice for organizations lacking the skills or ambition to hire professional mobile developers for each desired system. The functionality of PWA depends on the "service worker" because it offers the theoretical base for a PWA. A service worker is a computer program that web browser runs in the background to allow web apps to function normally while it is in offline state. It acts as an intermediate between the computer and the internet, intercepting and handling network requests [Mer]. A PWA can store and then use the stored data thanks to service workers, resulting in a simple user interface. It allows web programs to load automatically as you travel across them, reducing data usage. It's pretty similar to what we're seeing in native apps. The web app artifact is a JSON file that specifies the PWA's menu system. It regulates how progressive web apps can be installed and displayed to users in the same manner as native apps can in the mobile app. The title, explanation, design, and listing of needed materials are all included in web app manifests. PWAS can use ordinary Web services and be put on a phone's main screen, like regular web services with only bookmarks [Banb]. Moreover, Web pack is a dynamic package campaign contributor. Components and templates are used in modern JavaScript systems. The development process is simplified thanks to these components and templates. However, it can be inconvenient for installation. When the developers use web pack, it scans developers' program and automatically generates a graph structure that contains all of the components developers will require, then packs them all into one or more packages and boosts the user's performance [Cse].

Implement and evaluate progressive web applications using React and another user interface

React is excellent tool for building flexible and fast online applications while minimizing skill levels. Performance concerns with DOM (Document Object Model) operations are common in JavaScript apps. The Document Object Model (DOM) is a conceptual depiction of the web page that scripting languages can deal with it. React leverages virtual DOM to perform. Varies broadly without reloading the entire page and risk displaying errors. Without using software initialization tools such as JQuery. Developers can implement alterations to the web page in the memory space with React. In addition, the quicker processing performance of React minimizes the time it takes for a page to load. React is not as difficult to understand as Ember JS technologies. If the developers are

contemplating on how to create a React dynamic web service, it will be difficult to operate [BBM96].”React supports JSX, a JavaScript syntax extension that allows developers to use HTML-like syntax in your code.” Furthermore, developers are not required to utilize JSX in React; it is simply a more convenient method. As a result, developer can create a robust React PWA using just JavaScript and, HTML CSS. Troubleshooting, validation, and design are all made easier using React’s web apps, boiler-plates, and frameworks. It provides several possibilities to select the one that best suits developers’ needs. ”React Developer Tools, Facebook’s Create React App. Semantic UI React, and Ant Design” are some of the more well-known examples. When dealing with big online Visualization System errors are a typical SEO issue. It means that portions of our web page will not appear in Search engine results. The information of the responsive websites will be searched appropriately due to the need for virtual DOM [Bana]. At the moment of typing, React was the seventh most commonly highlighted accessible project on GitHub, and the React developed world is robust and ethical. There are two excellent React-powered PWA which companies such as Starbucks and Tinder are using. These firms understand creating a robust and accessible React dynamic web service. As a result, these applications will provide a general consisting of React’s potential and what can be accomplished using it [Banb].

Discuss the possibility that progressive web applications could be the future of mobile development

PWA is widely regarded as the future of cross programming because of its application on multiple devices, faster speed, and ease of use that needs no maintenance or upgrades. PWA is a future tool because it is available for Android and iOS development. It is an essential part of mobile development future since many native apps might be replaced by PWAs in the future. In addition, the developers need to understand the fundamental of PWA technologies to master this technology. The business technology platform and the number of users will depend on PWA. Furthermore, User data is very confidential on PWA because every modern web service has SSL certificates. the PWA offers a quick and smooth experience by storing data based on the devices’ capabilities [dev]. For instance, AliExpress uses PWA. As a result, the company increase the industry’s exchange rate by 104 percent. On top of that, it also has positive influence on the time which consumers spend per session. It increases around 74 percents. Another example is that the developer team who used PWA at eXtra Gadgets, had generated 100 percents more sales and attract new users using web push notifications.

Conclusions

The implementation and evaluation process of progressive web application and its benefit in future development. PWA use a combination of sit and native app, techniques and principles to take advantage of both internet and native app capabilities. PWA enables developers to develop web apps in a very efficient way. PWA is an essential part of mobile development in the future since many native apps will be replaced by PWA.

References

- [BBM96] V.R. Basili, L.C. Briand, and W.L. Melo. “A validation of object-oriented design metrics as quality indicators”. In: *IEEE Transactions on Software Engineering* 22.10 (1996), pp. 751–761. DOI: 10.1109/32.544352.
- [Mal+15] Ivano Malavolta et al. “Hybrid Mobile Apps in the Google Play Store: An Exploratory Investigation”. In: *2015 2nd ACM International Conference on Mobile Software Engineering and Systems*. 2015, pp. 56–59. DOI: 10.1109/MobileSoft.2015.15.
- [Bana] Rajiv D. Banker. *Software complexity and maintenance costs*. URL: <https://dl.acm.org/doi/10.1145/163359.163375>. (accessed: 03.22.2022).
- [Banb] Robin Bansal. *Progressive Web App: Here’s Everything You Need To Know About PWA*. URL: <https://medium.com/gradeup/progressive-web-app-heres-everything-you-need-to-know-about-pwa-79181db5a969>. (accessed: 03.18.2022).
- [Cse] Gabor Cselle. *Every Step Costs You 20 percents of Users*. URL: <https://medium.com/gabor/every-step-costs-you-20-of-users-b613a804c329>. (accessed: 03.15.2022).
- [dev] devetry. *Are Progressive Web Apps The Future?* URL: <https://devetry.com/blog/are-progressive-web-apps-the-future>. (accessed: 03.19.2022).
- [Mer] I Mercado. *The impact of cross-platform development approaches for mobile applications from the user’s perspective*. URL: <https://dl.acm.org/doi/10.1145/2993259.2993268>. (accessed: 03.15.2022).