REALIZATION OF NATIVE APPS USING PROGRESSIVE WEB APPS

[1]Pratik M Thacker, [2]Dr.Andhe Dharani

[1]Postgraduate Student, [2]Professor

[1]Department of MCA, [2]Department of MCA

[1]RV College of Engineering, Affiliated to Visvesvaraya Technological University, Bangalore, Karnataka, India,
[2]RV College of Engineering, Affiliated to Visvesvaraya Technological University, Bangalore, Karnataka, India.

Abstract: Recent mobile platform developments allow functionality which were traditionally believed only in natively built applications. The unique approach, adopted Progressive Web Apps, can be enforced on any website by reasonable standards of a set of concepts and technologies that fulfill certain requirements**].**An overview to the state-of-the-art in research and practice is being presented. Progressive Web Applications are enhanced normal web applications with native behavior, such as offline support, installability, and push notifications.On this basis, various underlying concepts and technologies are discussed. The paper covers all aspects on the use of progressive web apps as a modern application development cycle.It examines building blocks of Progressive Web Application in cross-platform development and evaluate the status quo of current possibilities comparing against cross-platform app development approaches on both technical and overarching aspects ,thereby providing conclusion of findings .

Keywords: Progressive Web Applications (PWAs), Web Apps.

I. INTRODUCTION

Progressive Web Apps brings together a set of technologies,concepts and Web APIs that work in conjunction to deliver app-like functionality .Recent case study suggests AliExpress raises the conversion rate for new customers by 104% for Progressive Web App.[1] In other terms, when developing PWAs, it is not important to build an application that incorporates all of its capabilities,additional functionality will be introduced if the developer thinks like they might benefit from them and have enriched content.Therefore,PWAs allow companies to develop quicker, more reliable and more engaging websites that can be viewed by hundreds of millions of people around the world and establish trust with consumers and, where necessary, implement new technologies.

With the development of Twitter lite,there was a 65% rise in pages for every user session and 75% jump in the submitted Tweets[2]**.** Google has placed together a guide to insure that the PWA standard is produced in the best possible manner., that can be any further enhanced in the PWA model by offering additional, insightful off-line experience, achieving immersive speed in addition to putting many more key factors into account.The criteria to serve as a PWA benchmark are as follows:

- The website should implement Hypertext Transfer Protocol Secure (HTTPS) which often prohibits attackers to be able to remotely listen to interactions between sites and its customers.
- Mobile friendly designs for various screens that help consumers with seamless and improved User experience
- In the event of a network outage,the client will be able to access minimum offline material but never a NOT FOUND(404) error
- The Add to Home screen functionality comes from the web App manifest file declared in the source or root directory
- Cross-browser compatibility of the web app such that the website looks the same on any browser.
- Transitions will feel natural when moving around the application, including on a sluggish network.So if there is a pause in the response from the network, the device will include the loading sign.
- Individual pages will have URLs that guarantee strong linkability.This allows sharing through social network sites and each webpage to be accessed and viewed directly from a new browser window.

II. RELATED WORK

Majchrzak et al. described the Progressive Web App more as a website that provides offline and online usage and a modern graphical user interface.[3].As the description is based on the functionality of the website and relies on the implementation of the functionality, we include a clear functional description of the PWA.

Over the duration of 2014, the number of global consumers browsing the site on handheld devices exceeded those viewing it on a laptop.Organisations also feel the need to build native apps or hybrid apps in order to address the issues concerning websites.In certain instances, the site, iOS and Android apps need to be developed.The native application is typically written in a device-specific programming language and an integrated development environment (IDE).

Although the native Android applications are normally developed with the intellij IDE Android Studio with Java as programming language.Such apps are typically distributed by Playstores and have rich access to device hardware by platform-specific APIs.

These software may be updated from the respective operating system application store and operate in a native environment. When taken out of the native environment , these apps are unable to provide functionality due to web limitations.Progressive Web Applications (PWA) should solve the problem.

A hybrid application represents an application developed with web-based technology that can function and operate like a native application with the aid of hybrid software frameworks (e.g. Apaches Cordova)[4].Nonetheless, the progressive web applications (PWAs) launched by Google in 2015 may be viewed as a fourth option.[5]

III. CORE TECHNOLOGY

Progressive Mobile Apps cannot be considered a specific platform or idea, but instead a range of them used together to offer better user experience**.**In this portion, various facets of PWAs are incorporated. The Progressive Web App is indeed a web application, but it employs methods to have progressive user interface.[6]This is also not incorrect to talk about both mainstream web apps and traditional web apps when mentioning the same principles.Probably, PWA is also a progressive web app, although not all progressive web apps are PWAs in a limited context.This ambiguity can be overcome by describing the techniques and definitions in the following subsections:

3.1 Application Shell

According to Google's developer documentation: "An app shell is the minimal HTML, CSS, and JavaScript that is required to power the user interface of a progressive web app and is one of the components that ensures reliably good performance" (Google, 2019).

The App Shell Theory includes installing a negligible user interface before the PWA is released, accompanied by caching material, such that it is accessible offline for successive visits.It means that the UI is enabled from the cache automatically as the user opens the web page every time from the same platform and demands fresh information from the server afterwards.

The application shell design distinguishes the core program and the UI from the data.To assess the design of the app shell architecture, the following points need to be considered:

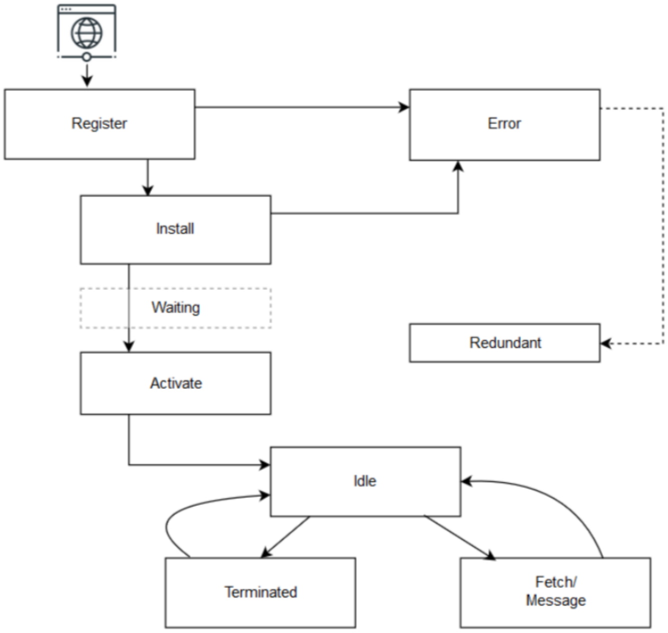
- What has to be achieved on the first load screen?
- Which other UI elements are the keys to the app?
- What supported assets are required for the application shell?

App Shell consists of static assets and device elements, which ensures that it does not rely on any external data.Examples of these static properties involve toolbars, menu bars, splash screens, and the like built in HTML, CSS, and JavaScript[7]**.**

3.2 Service Worker

Service Workers act as foundations of PWA, enabling features that make PWA unique.It allows apps to have functionalities like offline connectivity, regular background sync, and web-based notifications that usually involve a native application.A service worker is a JavaScript code which the browser executes on a separate thread from the main JavaScript code of the webpage in the background and thus have little access to the DOM[8]. Having almost negligible access to DOM, a service worker runs in the background on a different thread from the main code of JavaScript of the webpage.

Therefore, it will not obstruct the activity of the key JavaScript on the website when interacting across different settings.



3.2.1 Life cycle of Service worker

Figure 1 Service worker lifecycle modified from Hajian (2019) [11]

The lifecycle of the Service worker begins when the client opens the webpage.The execution of service worker file begins when the client application finds the file which it further downloads to parse.In the event of failure of the process,the registration declines thereby terminating the procedure**.**While, on the other side, if the registration is successful and the service worker is resolved, the state of the service worker returns to the installed state as seen in Figure 1.After successful setup and installation of service worker ,caching mechanism of static assets is applied. The set-up only takes place once and upon the effective launch, the service worker is allowed and has complete oversight over the website .

An activated service worker has direct oversight over the sites.It can perform activities such as fetch, push and background sync. When no events are recorded, it can reach an idle state and, after a moment, reach terminated state.Terminated will not imply uninstalled or unregistered, and can become idle again as long as events continue to be triggered.

A service worker can often become obsolete if an event fails or if the current service worker replaces it with a new one.

3.3 Web App Manifest

Manifest is a JSON file that web developers may use to customize their PWAs.Configurations such as app name, page routes, splash screen graphic, background colors and display styles (e.g. fullscreen, with or without tab artifacts) may be defined in the file [9].The appearance and presentation of the app is based on the properties defined in the manifest file where every property serves specific objective The properties and their purpose used in the manifest file are as follows:

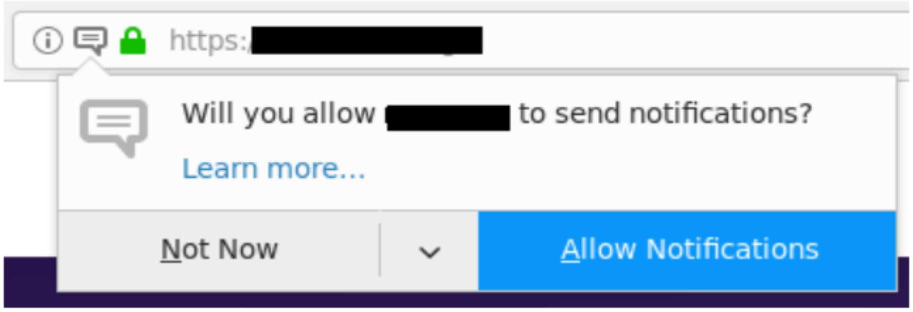
- short_name and name : The name which is visible for Add to home screen feature
- Start_url : URL relative to the manifest file that specifies where the application will be initialized. This usually points to the index.js file
- theme_color : Default theme color of app
- orientation : Orientation helps the developers impose it on applications that are supposed to function in only one orientation, such as landscape mode in games.
- scope : It determines the collection of urls that are deemed to be part of the application.

3.4 Web Push

Web Push consists of two primary platforms , Notifications API and Push API. Notifications API helps pop-up displays for different types of browsers. Push API enables asynchronous delivery of notifications

3.4.1 Notifications API

The Notifications API helps us display any pop-ups to the consumer or to view some other details when using the application.To view such alerts, the user's permission must be given which includes adding two lines of Javascript code in the service worker.Figure 2 clearly depicts the way in which it is displayed to the user



https:[redacted]

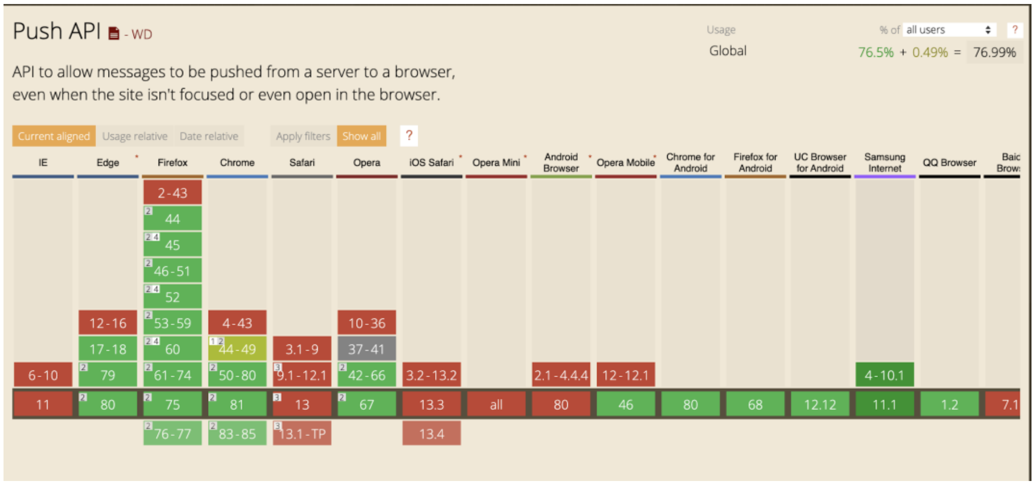
Will you allow [redacted] to send notifications?

Figure 2.Web push permissions user's point of view

3.4.1 Push API

The Push API enables updates to be viewed even if the window tab is closed.It offers us an opportunity to reach the client..The specification of the Push API seems to be more complex than the Notification API, however as an outcome, it gives the user the ability to connect with the app even without opening the tabs of the browser.Overall, rising the feeling of the native application.

Certain browsers do not have encouraging support for Push Api, particularly in Apple ecosystem of iOS devices, as can be seen from Figure 3 below.



Push API - WD

API to allow messages to be pushed from a server to a browser, even when the site isn't focused or even open in the browser.

Usage Global % of all users ?

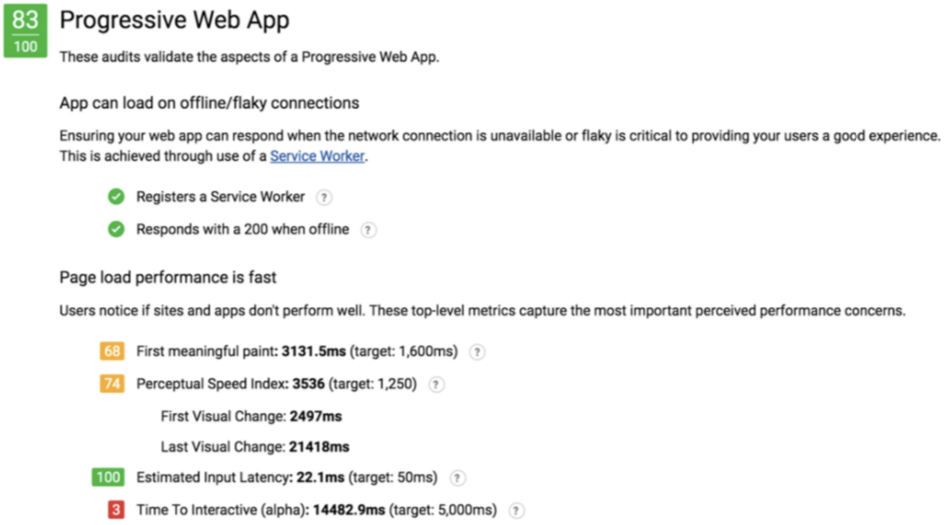
76.5% + 0.49% = 76.99%

IE	Edge	Firefox	Chrome	Safari	Opera	iOS Safari	Opera Mini	Android Browser	Opera Mobile	Chrome for Android	Firefox for Android	UC Browser for Android	Samsung Internet	QQ Browser	Baic Brow:
		2-43													
		[2] 44													
		[2 4] 45													
		[2] 46-51													
		[2 4] 52													
	12-16	[2] 53-59	4-43		10-36										
	17-18	[2 4] 60	[1 2] 44-49	3.1-9	37-41										
6-10	[2] 79	[2] 61-74	[2] 50-80	[3] 9.1-12.1	[2] 42-66	3.2-13.2		2.1-4.4.4	12-12.1				4-10.1		
11	[2] 80	[2] 75	[2] 81	[3] 13	[2] 67	13.3	all	80	46	80	68	12.12	11.1	1.2	7.1
		[2] 76-77	[2] 83-85	[3] 13.1-TP		13.4									

Figure 3.Push API browser support [10]

3.5 Lighthouse:PWA Testing Tool

The Lighthouse Chrome extension is an approved Google tool targeted at PWA programmers.This tool is available as a Google chrome extension, and will be helpful in performance analysis of websites against certain metrics and goals, which Google considers to be significant, particularly regarding PWAs.The extension is created to increase the overall compatibility and for ultimate mobile web experience[12].Lighthouse can produce a verbose report on the state of the website evaluated, provide guidance on how to improve the code and properties, and provide an overall score as shown in figure 4 below:



83 / 100 Progressive Web App

These audits validate the aspects of a Progressive Web App.

App can load on offline/flaky connections

Ensuring your web app can respond when the network connection is unavailable or flaky is critical to providing your users a good experience. This is achieved through use of a Service Worker.

- ✓ Registers a Service Worker ?
- ✓ Responds with a 200 when offline ?

Page load performance is fast

Users notice if sites and apps don't perform well. These top-level metrics capture the most important perceived performance concerns.

- 68 First meaningful paint: **3131.5ms** (target: 1,600ms) ?
- 74 Perceptual Speed Index: **3536** (target: 1,250) ?
  - First Visual Change: **2497ms**
  - Last Visual Change: **21418ms**
- 100 Estimated Input Latency: **22.1ms** (target: 50ms) ?
- 3 Time To Interactive (alpha): **14482.9ms** (target: 5,000ms) ?

Figure 4.Example PWA report

4.0 APPLICATIONS USING PROGRESSIVE TECHNIQUES - SAMPLE CASE STUDIES

4.1 Pinterest

Pinterest is a visual search platform to find things like food, home and decor inspiration, and more.it runs a digital framework intended to archive and explore information on the World Wide Web by images and, on a smaller scale, Clips and animations.

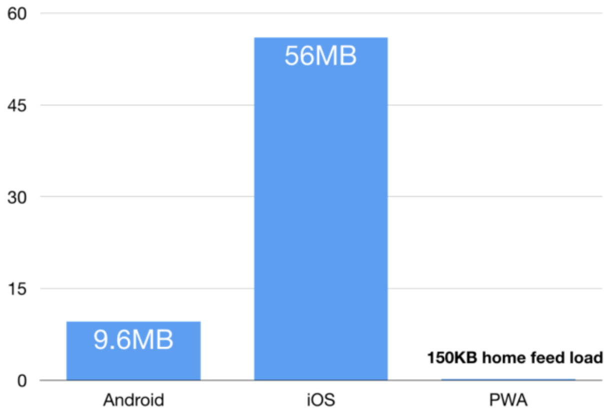


Figure 4.1(a).Comparing PWA to native apps of Pinterest

Implementation of PWA principles on Pinterest is as follows:

- Although Pinterest provides iOS & Android applications, they have been able to get the same current home feed functionality as such devices have on the web at a reasonable size— just ~150 KB minified & gzipped.[13]
- Pinterest has generated caching assets with Service workers, utilizing the Workbox libraries to deploy and produce service workers.
- Pinterest caches all Html or CSS packages that follow the cache-first approach and even caches their user experience (the Application shell).
- Pinterest has recently launched support forWeb Push notifications

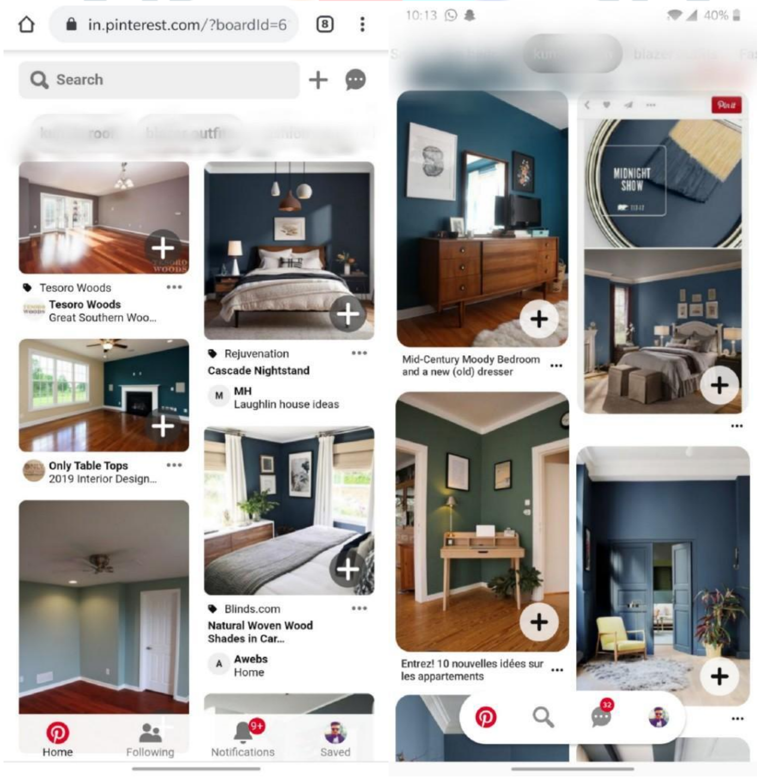


Figure 4.1(b) Pinterest Interface (PWA,Android)

In Figure 4.1(b) It is noticed that the User Interface of Progressive Web is the same as native Android Application

4.2 OYO lite

It is an electronic accommodation booking company and a global medium for delivering inexpensive stays around the globe, offering a variety of facilities from luxury rooms to backpackers hostel. Oyo is one of the world's largest and fastest expanding lodging companies of leased and franchised hotels, apartments and living spaces.

Implementation of PWA principles for Oyo lite is as follows:

- The relative size of Oyo lite is barely 1 MB
- Quick loading and response times
- Reliability whether the app has minimal to no network access
- A close look and feel to native apps (providing, for example, a splash screen and device color)

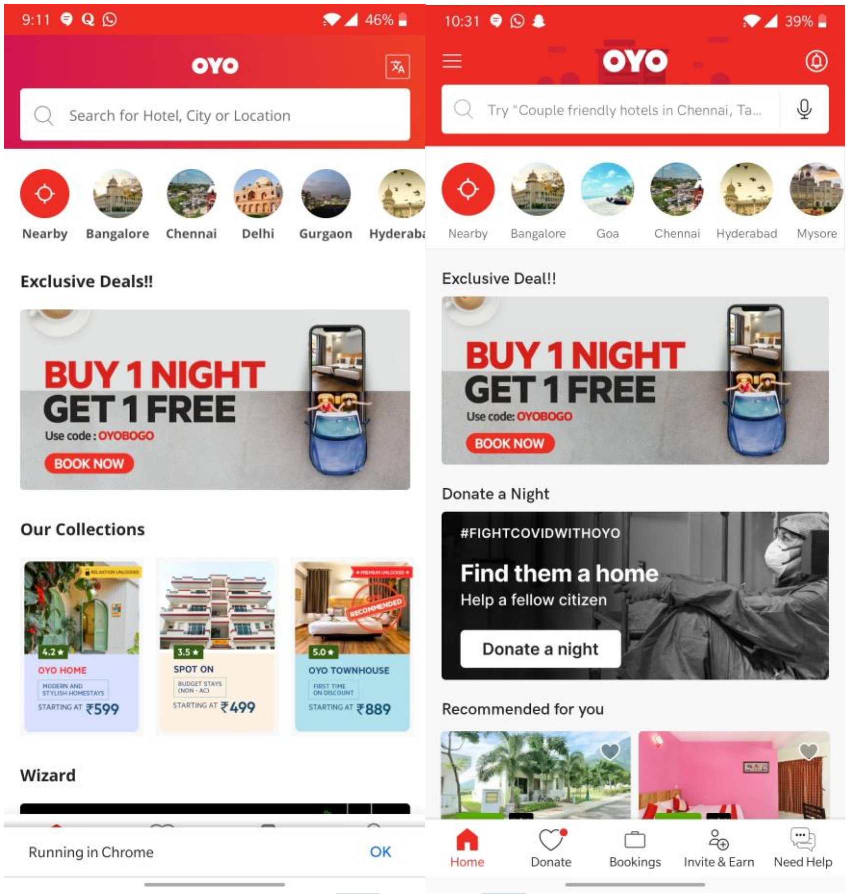


Figure 4.2(a) Oyo Interface (PWA,Android)

In Figure 4.2(a) It can be observed that the User Interface of Progressive Web(OYO lite) is the same as native Android Application and OYO lite is running on Chrome

4.3 BookMyShow

BookMyShow is India's biggest ticketing company, with more than 50 million users per month. They have now developed an enhanced version of their smartphone app utilizing the Progressive Web App (PWA), which offers an 80 + percent increase in revenue, which implies that more customers are purchasing seats.

- They were also pleased to utilize the "Add to home screen" feature to own a native-app-like experience.
- The PWA is only over 440KB—54 times smaller than the Android app and 180 times smaller than the iOS version.
- BookMyShow also optimally used Service Workers to provide consistent results on sluggish or unstable networks.

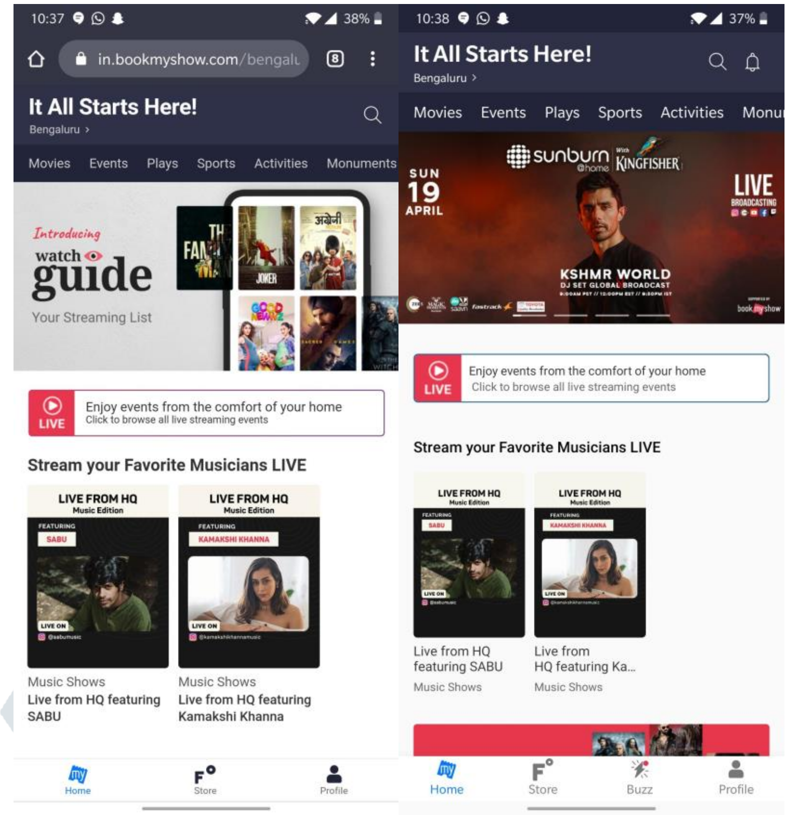

10:37
in.bookmyshow.com/bengalu
It All Starts Here!
Bengaluru >
Movies
Events
Plays
Sports
Activities
Monuments
Introducing watch guide
Your Streaming List
LIVE
Enjoy events from the comfort of your home
Click to browse all live streaming events
Stream your Favorite Musicians LIVE
LIVE FROM HQ Music Edition
SABU
KAMAKSHI KHANNA
Music Shows
Live from HQ featuring SABU
Music Shows
Live from HQ featuring Kamakshi Khanna
Home
Store
Profile
10:38
It All Starts Here!
Bengaluru >
Movies
Events
Plays
Sports
Activities
Monu
sunburn @home With KINGFISHER
SUN 19 APRIL
LIVE BROADCASTING
KSHMR WORLD
DJ SET GLOBAL BROADCAST
LIVE
Enjoy events from the comfort of your home
Click to browse all live streaming events
Stream your Favorite Musicians LIVE
LIVE FROM HQ Music Edition
Live from HQ featuring SABU
Music Shows
Live from HQ featuring Ka...
Music Shows
Home
Store
Buzz
Profile

Figure 4.3(a)BookMyShow interface(PWA,Android)

In Figure 4.3(a) It can be observed that the User Interface of Progressive Web Application(Left) is the same as native Android Application

5.0 FINDINGS AND CONCLUSION

- Progressive Web Apps have a number of benefits so far, rendering improved user interface and incorporating the necessary PWA features is not quite difficult.Considering the large amount of people accessing the internet through mobile browsers, the prevalence of Progressive Web Applications is expected to increase considerably.
- Several native applications are likely to be replaced in the future, particularly apps that do not really need native features or hardware.Nevertheless, Progressive Web Apps have not yet been widely known to end-users, so there is a shortage of browser support to some degree.PWAs focus mostly on the accessible web APIs, and although the amount of them has expanded, this ensures that PWAs have a much broader variety of functionality which is not yet noticeable to a larger audience
- With Microsoft's upcoming plans to have PWAs in its Windows Store, Google's plans to exploit them on the ChromeOS framework, and the existing opportunity for downloading PWAs on mobile computers e.g. the Chrome browser, a change can be seen in how smartphone apps are created.The Progressive Web App app will be further tracked by more study and market acceptance.

Summarizing,developing a web app implementing the techniques and concepts of PWA is an important method to design web apps .PWA can be used to overcome storage and installation challenges in native apps, as well as to overcome drawbacks linked to lack of connectivity and user interface with web applications.

REFERENCES

[1] AliExpress Mobile Team , AliExpress use case , https://developers.google.com/web/showcase/2016/aliexpress Accessed 18 Mar 2020

[2] Twitter Lite PWA significance https://developers.google.com/web/showcase/2017/twitter Accessed 28 April 2020

[3] T. Majchrzak, A. Biørn-Hansen, and T. Grønli. 2018. Progressive Web Apps: the Definite Approach to Cross-Platform Development?. In Hawaii International Conference on System Sciences

[4] David Fortunato and Jorge Bernardino,"Progressive web apps: An alternative to the native mobile Apps", IEEE/13th Iberian Conference on Information Systems and Technologies (CISTI), 2018.

[5] P. Kinlan, "Installable web Apps with the web App manifest in chrome for Android", Google Developers, 2017.

[6] Bob Frankston," Progressive Web Apps", IEEE Consumer Electronics Magazine, vol.7, no 2. pp.106, 2018.

[7] Parbat Thakur, "Evaluation and Implementation of Progressive Web Application", bachelor's thesis, Helsinki Metropolia University of Applied Sciences, 2018.

[8] Ivano Malavolta, Giuseppe Procaccianti, Paul Noorland andPeter Vukmirovic," Assessing the Impact of Service Workers on the Energy Efficiency of Progressive Web Apps", IEEE/ACM 4th International Conference on Mobile Software Engineering and Systems (MOBILE Soft),2017

[9] Guant, M., Kinlan, P.: The web app manifest (2016). https://developers.google.com/web/fundamentals/engage- and- retain/web-app- manifest/. Accessed 4 Mar 2020

[10] Caniuse.com. (2019). Can I use... Support tables for HTML5, CSS3, etc. [online] Available at: https://caniuse.com/#feat=push-api [Accessed 19 Apr. 2020].

[11] Google Developers: Lighthouse (2017).https://developers.google.com/web/tools/lighthouse/. Accessed 30 April 2020

[12] .Hajian M. Progressive Web Apps with Angular: Create Responsive, Fast and Re- liable PWAs Using Angular. 1st ed. Oslo: Apress; 2019

[13] Osmani, A., Gaunt, M.: Instant loading web apps with an application shell archi- tecture (2017).https://developers.google.com/web/updates/2015/11/app-shell