

ANALISIS PERFORMA PROGRESSIVE WEB APPLICATION (PWA) PADA PERANGKAT MOBILE

Antonius Angga Kurniawan
Fakultas Teknologi Informasi Universitas Gunadarma
Jl. Margonda Raya No. 100, Depok 16424, Jawa Barat
anggaku@staff.gunadarma.ac.id

Abstrak

Diantara aplikasi web dan aplikasi native terdapat kesenjangan yang menimbulkan permasalahan. Aplikasi web membutuhkan browser dan koneksi yang baik untuk membuka aplikasi, sedangkan aplikasi native membutuhkan penyedia aplikasi seperti Google PlayStore dan App Store. Pengguna juga memerlukan ruang memori yang cukup untuk sebuah aplikasi yang diinginkan dan tidak semua aplikasi mendukung perangkat yang digunakan. Untuk mengatasi kesenjangan ini, Google menciptakan metode baru dalam pengembangan web. Metode itu adalah Progressive Web Application (PWA). Tujuan dari penelitian ini adalah untuk menganalisis fitur dan kemampuan dari PWA dengan melakukan uji coba pada situs web PWA di mobile dengan aspek penilaian performance, accessibility, best practices, seo, page size, load time, dan penerapan PWA itu sendiri seperti (fast and reliable, installable, pwa optimization). Pengujian juga dilakukan dengan membuktikan fitur dari PWA pada sebuah aplikasi web e-commerce menggunakan mobile. Hasil yang didapatkan menunjukkan bahwa PWA memiliki kemampuan dan fitur yang sangat baik sebagai pengembangan dari aplikasi web dan dapat membantu kesenjangan antara aplikasi web dan aplikasi native.

Kata Kunci: Aplikasi, native, pwa, smartphone, web.

Abstract

Between web applications and native applications there are gaps that cause problems. Web applications require a good browser and connection to open applications, while native applications require application providers such as Google PlayStore and App Store. Users also need enough memory space for each application and not all applications support the device used. To overcome this gap, Google created a new method of web development. That method is Progressive Web Application (PWA). The purpose of this study is to analyze the features and capabilities of PWA by conducting trials on the PWA website on mobile with aspects of performance evaluation, accessibility, best practices, SEO, page size, load time, and the application of PWA itself such as (fast and reliable, installable, and PWA optimization). Testing is also done by proving the features of PWA on an e-commerce web application using a mobile device. The results obtained indicate that PWA has exceptional capabilities and features as the development of web applications and can help reduce the gap between web applications and native applications.

Keywords: Application, native, pwa, smartphone, web.

PENDAHULUAN

Pengembangan sebuah aplikasi dibagi menjadi 2, yaitu aplikasi *native* dan aplikasi

web. Aplikasi *native* memiliki performa yang cepat, dapat mengirim *push notification*, terdapat *icon* untuk membuka aplikasi, dan memiliki tampilan antar muka yang mudah

dipahami. Namun, biaya yang dikeluarkan untuk membangun dan memelihara sebuah aplikasi *native* tidak murah. Aplikasi *native* juga tidak punya kemampuan *cross-platform* (*Android dan IOS*). Masalah lain yang sering muncul adalah keterbatasan ruang memori pada ponsel ketika pengguna ingin mengunduh dan menggunakan sebuah aplikasi *native* [1]. Pengguna dapat membuka aplikasi web melalui *browser* terlebih dahulu, baik di perangkat laptop maupun perangkat *smartphone*. Cepat atau lambatnya aplikasi web yang diakses menjadi salah satu tantangan dalam pengembangan web. Berdasarkan data yang dihimpun oleh Google, sebanyak 53% konsumen akan meninggalkan halaman web jika memakan waktu lebih dari 3 detik untuk memuat halaman web [2]. Dibutuhkan koneksi internet yang baik untuk mengakses aplikasi web. Selain itu, aplikasi web hanyalah web biasa yang tidak bisa diunduh ke dalam perangkat *mobile* layaknya aplikasi *native*. Google menciptakan teknologi baru dalam pengembangan web untuk mengatasi semua masalah, yaitu *Progressive Web Application* (PWA).

Alex Russell dan desainer Frances Berriman (2015) dari Google Chrome menciptakan *Progressive Web Application* (PWA). PWA merupakan metode dalam pengembangan *software* dengan mengombinasikan aplikasi *native*, aplikasi web, dan *desktop*. Di dunia kosmetik ada istilah “*your lips but better*”. Di dunia IT, PWA dikenal sebagai “*your website but better*” [3]. Hasil

yang didapat dari penerapan metode PWA adalah kemampuan sebuah aplikasi untuk memiliki berbagai fitur dari aplikasi web pada umumnya, dengan optimasi *mobile experience* yang ada pada aplikasi *native*. PWA juga dapat mengirim *push notification* yang relevan kepada pengguna, dapat memiliki *icon* aplikasi di layar beranda ponsel, dan hanya membutuhkan sedikit ruang memori [4].

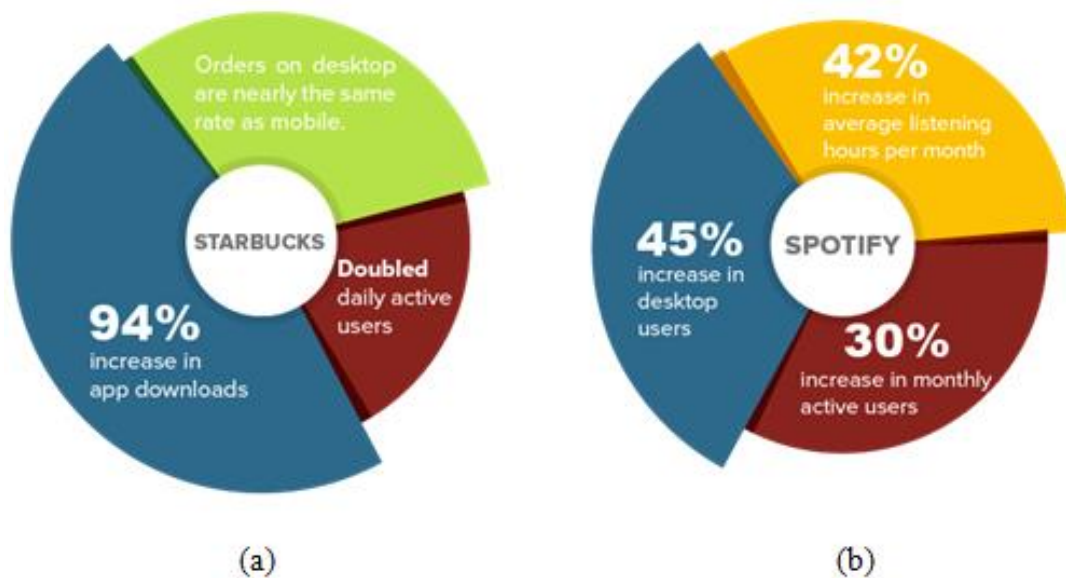
Beberapa penelitian telah dilakukan terhadap metode pengembangan web PWA. Tandel dan Jamadar [5], melakukan penelitian dengan mengidentifikasi masalah yaitu adanya keterbatasan atau kekurangan dari aplikasi *native* dan juga *mobile web* dan menerapkan metode PWA dalam pembuatan sebuah web. Pengujian dilakukan dengan membandingkan kecepatan akses antara web PWA dengan web standar dan membandingkan penghematan memori dalam mengakses sebuah CSS yang selalu dirender saat web dibuka. Hasilnya adalah PWA lebih cepat saat diakses dan lebih menghemat memori saat mengakses web tersebut.

Penelitian lainnya Majchrzak, Hansen dan Gronli [6], menemukan keterbatasan aplikasi yang *support* antara Android dan IOS sehingga mereka mempelajari apakah PWA nantinya bisa menjadi teknologi pemersatu dalam pengembangan web dan aplikasi *native*. Walaupun PWA sudah mendekati sebuah metode pengembangan web secara modern dan memiliki teknologi serta manfaat yang luar biasa. Hasilnya telah berhasil

memperkenalkan beberapa teknologi dan dasar-dasar dari PWA serta *impact* terhadap beberapa perusahaan yang sudah menerapkan PWA. Namun, dari penelitian yang dilakukan belum bisa dikatakan dengan pasti bahwa PWA dapat menjadi solusi dari masalah *cross-platform*. Maka dari itu peneliti telah membuat sketsa perkembangan masa depan dan menyarankan agenda penelitian. Mereka menyarankan pendekatan yang seimbang dari pekerjaan eksperimental dan kualitatif. Peneliti Nurwanto [7] mengidentifikasi adanya masalah kesenjangan antara aplikasi *native* dan web standar dengan setiap kelebihan dan kekurangannya, kemudian mencoba mengimplementasikan metode PWA pada situs web *e-commerce*. Hasilnya adalah PWA merupakan metode pengembangan web yang

luar biasa karena pengguna dapat menjalankan web *e-commerce* tersebut layaknya aplikasi *native* (*push notification*, adanya *icon* aplikasi pada beranda ponsel, dan dapat berjalan saat *offline*).

Dalam pengembangan web, pelaku bisnis sudah ada yang mengimplementasikan metode PWA dan merasakan manfaatnya. Gambar 1 menunjukkan dua contoh manfaat yang dirasakan oleh dua pelaku bisnis, yaitu Starbucks dan Spotify. Tujuan dari penelitian ini adalah untuk menganalisis situs web yang sudah mengadopsi PWA dengan cara melakukan pengujian terhadap fitur dan performa dari aplikasi web PWA yang sudah diimplementasikan oleh beberapa pelaku bisnis modern saat ini dalam mengembangkan situs webnya.



Gambar 1. Contoh manfaat dari penerapan PWA (a) Starbucks, (b) Spotify

[sumber: simiCart, 2020 [12]]

METODE PENELITIAN

Metode penelitian dilakukan dengan mengumpulkan data berupa informasi terkait siapa saja pelaku bisnis yang sudah menerapkan metode PWA di dalam aplikasi webnya. Dari banyaknya situs web yang diperoleh, di dalam penelitian ini hanya melakukan analisis terhadap sepuluh web diantaranya adalah yummlly.com, olx.co.id, trivago.co.id, twitter.com, app.starbucks.com, tokopedia.com, makemytrip.com, sudoku.jull.dev, open.spotify.com, dan lofi.news. Kesepuluh web tersebut dipilih karena berasal dari berbagai bidang yang berbeda agar sebuah implementasi PWA dapat lebih terlihat fungsinya bisa diterapkan dibanyak bidang. Bidang tersebut diantaranya adalah situs penyedia resep masakan, *ecommerce*, situs *games*, situs penyedia jasa penginapan dan perjalanan, situs sosial media, situs portal berita, situs *coffee shop*, dan situs pemutar musik. Tahap selanjutnya adalah menyiapkan perangkat yang digunakan untuk melakukan pengujian, yaitu satu perangkat laptop dan satu perangkat *mobile* yang sudah terinstal *browser* Google Chrome. Dari kesepuluh web yang akan diuji, diambil satu contoh web yang digunakan untuk membuktikan bahwa fitur dari komponen utama pada PWA seperti *Manifest* dapat menambahkan situs web ke dalam beranda ponsel berupa *icon* aplikasi layaknya aplikasi *native*. Kemudian komponen *Service Worker* yang berguna dalam melakukan proses *push notification* untuk

memunculkan sebuah notifikasi pada perangkat *mobile* yang sebelumnya kemampuan tersebut tidak dimiliki oleh sebuah aplikasi web [8]. Uji coba dilakukan pada web *ecommerce* dengan melakukan instalasi aplikasi web melalui web *browser* pada *mobile*, kemudian memunculkan *icon* aplikasi pada beranda ponsel, dan terakhir melakukan uji coba dengan fitur *push notification* pada aplikasi web tersebut.

Tahap selanjutnya adalah melakukan pengujian menggunakan alat bantu yang bernama GeekFlare. GeekFlare merupakan sebuah alat bantu yang digunakan untuk mengaudit performa dari sebuah web. Selain GeekFlare, pengujian juga menggunakan alat bantu lain, yaitu *Lighthouse*. *Lighthouse* merupakan alat bantu yang disediakan oleh Google dan biasa digunakan oleh para pengembang web untuk mengaudit performa web terutama PWA. Pengujian dilakukan dengan menggunakan parameter-parameter yang direferensikan di dalam situs resmi Google Developer, parameter itu diantaranya adalah *Performance*, *Accessibility*, *Best Practices*, *SEO*, *Page Size*, *Load Time*, dan penerapan PWA itu sendiri yang terdiri dari beberapa kriteria, yaitu *Fast and Reliable*, *Installable*, *PWA Optimized* [9].

Skor Penilaian dalam Pengujian WEB

Berdasarkan parameter yang direferensikan Google Developer untuk menilai suatu web, alat bantu GeekFlare dan Lighthouse mengkategorikan penilaian skor

dari 0 sampai dengan 100 untuk menentukan bahwa web tersebut terbilang baik atau tidak performanya berdasarkan parameter yang ada [9]. Kategori dari skor dapat dilihat pada Tabel 1.

Tabel 1 menunjukkan bahwa apabila suatu web setelah diuji mendapatkan skor 0-49, maka status dari web tersebut adalah lambat atau kurang baik. Jika web setelah diuji mendapatkan skor 50-89, maka status dari web tersebut adalah standar atau rata-rata seperti web pada umumnya. Jika suatu web setelah diuji mendapatkan skor 90-100, maka status dari web tersebut adalah cepat atau sangat baik. Warna dari status yang lambat sampai cepat adalah merah, jingga, dan hijau.

Bobot Pengujian Berdasarkan Kriteria dari PWA

Pengujian terkait seberapa berhasil sebuah web mengimplementasikan metode pengembangan PWA diuji dengan menggunakan *Lighthouse*.

Tiga buah parameter utama yang dinilai oleh *Lighthouse* untuk memenuhi kriteria dari

sebuah PWA, yaitu *Fast and Reliable*, *Installable*, dan *PWA Optimized* [10]. *Fast and Reliable* berarti suatu web harus dapat menanggapi dengan cepat interaksi pengguna dengan animasi atau tampilan yang halus dan memuat secara instan bahkan dalam kondisi jaringan yang tidak pasti. *Installable* berarti suatu web harus bisa ditambahkan ke dalam perangkat *mobile* melalui halaman web dan menampilkan *icon* pada beranda ponsel, serta sebuah web harus dapat melakukan fungsi-fungsi layaknya aplikasi *native* seperti *push notification* (seperti mengirim notifikasi ke dalam ponsel). Yang terakhir *PWA Optimized* dimaksudkan agar web yang dibangun dapat memperhatikan hal-hal seperti mengarahkan halaman http menuju https, kemudian agar peringkat pada mesin pencarian pada saat dilakukan pencarian oleh pengguna menjadi peringkat teratas.

Masing-masing kriteria memiliki jumlah persyaratan yang harus dipenuhi agar suatu web itu dikatakan benar-benar sukses mengimplementasikan pengembangan web PWA.

Tabel 1. Kategori Skor dalam Penilaian Web

Skor	Status	Warna Status
0-49	Kurang Baik (biasanya lambat juga)	Merah
50-89	Standar/Rata-rata	Jingga
90-100	Sangat Baik (biasanya cepat)	Hijau

Masing-masing dari persyaratan memiliki bobot persentase dari 0 sampai dengan 100 [11]. Bobot ini dapat dilihat pada Tabel 2, Tabel 3, dan Tabel 4. Tabel 2 menunjukkan bahwa kriteria dari *fast and reliable* memiliki 3 syarat utama. Syarat pertama memiliki bobot skor 53.8% di mana web harus memuat halaman cukup cepat di jaringan seluler dan syarat ini memiliki pengaruh besar pada kriteria *fast and reliable* karena mengambil bagian 53.8 dari 100, syarat kedua memiliki bobot skor 38.5% di mana dalam hal ini suatu web harus tetap bisa berjalan dan memberikan respon sukses meskipun sedang *offline*, pada syarat ini mengambil bagian 38.5 dari 100, dan syarat ketiga memiliki bobot skor 7.7%,

di mana bobot ini merupakan bobot yang paling rendah, yaitu mengambil bagian 7.7 dari 100. Tabel 3 menunjukkan bahwa kriteria dari *installable* memiliki 3 syarat utama. Syarat pertama memiliki bobot skor 40% yaitu harus menggunakan HTTPS, di mana syarat ini punya pengaruh penting dalam penilaian karena mengambil bagian 40 dari 100, syarat kedua memiliki bobot skor 20% yaitu *service worker* harus terdaftar pada halaman web dan mengambil bagian 20 dari 100, dan syarat ketiga memiliki bobot skor 40% yaitu web *manifest* yang dibuat harus memenuhi persyaratan PWA, syarat ini juga memiliki peran penting karena mengambil bagian 40 dari 100.

Tabel 2. Bobot Persentase dari *Fast and Reliable*

No	Syarat	Bobot (%)
1	Web harus memuat halaman cukup cepat di jaringan seluler	53.8
2	Halaman web terakhir harus merespons dengan kode 200 ketika offline	38.5
3	start_url merespons dengan kode 200 ketika offline	7.7

Tabel 3. Bobot Persentase dari *Installable*

No	Syarat	Bobot (%)
1	Harus menggunakan HTTPS	40
2	Mendaftarkan <i>service worker</i> yang mengontrol halaman dan start_url	20
3	Web app <i>manifest</i> yang dibuat memenuhi persyaratan PWA	40

Tabel 4. Bobot Persentase dari PWA Optimized

No	Syarat	Bobot (%)
1	Arahkan lalu lintas http ke https	22.2
2	Konfigurasi untuk <i>splash screen</i> khusus	11.1
3	Set warna tema untuk <i>address bar</i>	11.1
4	Membuat ukuran konten yang tepat untuk <i>viewport</i>	11.1
5	Mempunyai tag <meta name="viewport"> dengan <i>width</i> atau <i>initial-scale</i>	22.2
6	Memuat sebagian konten pada saat <i>Javascript</i> tidak tersedia	11.1
7	Memberikan <i>apple-touch-icon</i> yang valid	11.1

Tabel 4 menunjukkan bahwa kriteria dari *PWA optimized* memiliki 7 syarat utama. Syarat pertama dan syarat kelima memiliki bobot skor 22.2% yaitu suatu situs web harus dapat mengarahkan lalu lintas jaringan dari http menuju https dan suatu web harus memiliki meta tag *viewport*, kedua syarat ini mengambil bagian yang paling tinggi diantara syarat yang lain, yaitu masing-masing 22.2 dari 100. Sisanya memiliki bobot skor 11.1%, di mana suatu web harus mengkonfigurasi *splash screen* khusus, mengeset warna tema dari *address bar*, membuat ukuran konten yang tepat, memuat halaman konten pada saat *Javascript* tidak tersedia, dan suatu web harus memberikan *icon* sentuh yang valid pada perangkat *Apple*, beberapa syarat ini mengambilm bagian masing-masing 11.1 dari 99.9.

HASIL DAN PEMBAHASAN

Hasil dari pengujian yang dilakukan didapat dari proses pengujian suatu aplikasi

web. Hasil yang disajikan terdiri dari hasil pengujian untuk membuktikan fitur dari komponen PWA yaitu *manifest* dan *service worker* benar-benar berjalan atau tidak, kemudian hasil dari pengujian web PWA pada perangkat *mobile* berdasarkan parameter yang direferensikan oleh Google Developer, dan hasil dari pengujian web PWA berdasarkan kriteria dari PWA itu sendiri.

Hasil Pengujian Fitur dari Komponen PWA

Pengujian dilakukan pada salah satu web *ecommerce*, yaitu *tokopedia.com*. Pemilihan salah satu web yang diuji ini dipilih secara acak.

Setelah web sudah berhasil mengimplementasikan PWA, maka pada saat akan mengakses situs web tersebut akan tampil *banner* untuk menambahkan aplikasi ke dalam perangkat *mobile* seperti ditunjukkan pada Gambar 2. Kemudian jika *banner* diklik maka akan tampil pertanyaan konfirmasi apakah yakin ingin menambahkan aplikasi ke

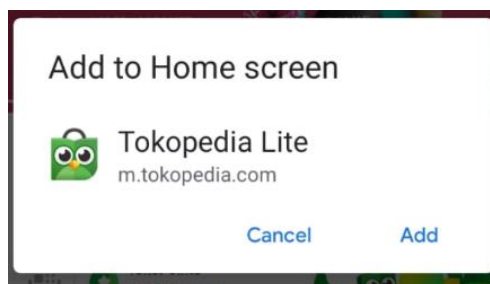
dalam *mobile*, hal ini dapat dilihat pada Gambar 3.

Gambar 2 menunjukkan bahwa fitur dari komponen PWA yaitu *manifest* terbukti benar karena pada aplikasi web PWA akan muncul sebuah *banner* yang berada di posisi bawah untuk menambahkan aplikasi web

pada perangkat *mobile* yang digunakan. Pada Gambar 3, jika suatu *banner* yang terlihat pada gambar 2 ditekan, maka akan muncul sebuah kotak untuk mengkonfirmasi apakah Anda yakin akan menambahkan aplikasi web tersebut pada beranda ponsel Anda.



Gambar 2. Terdapat *Banner Install Aplikasi ke dalam Beranda Mobile*



Gambar 3. Konfirmasi Apakah Ingin Menambahkan Aplikasi ke dalam *Mobile*

Jika sudah menambahkan aplikasi web ke dalam perangkat *mobile*, maka akan muncul sebuah *icon* dari aplikasi yang ditampilkan pada Gambar 4. Adanya *icon* aplikasi yang tampil semakin membuktikan bahwa fitur dari komponen *manifest* pada PWA benar valid sesuai dengan fungsinya. Jika sudah pada saat membuka aplikasi tersebut maka tampilannya akan terlihat

seperti aplikasi *native* biasa namun sebenarnya itu adalah sebuah web di mana web tersebut sudah berhasil diimplementasikan PWA.

Hal ini dapat dilihat pada Gambar 5 yang membuktikan bahwa aplikasi web PWA membuat sebuah web standar memiliki optimasi layaknya aplikasi *native* yang terpasang di dalam perangkat *mobile*.



Gambar 4. *Icon* dari Aplikasi Web yang Ditambahkan Melalui *Browser*



Gambar 5. Aplikasi Web PWA Sudah Seperti Layaknya Aplikasi *Native*

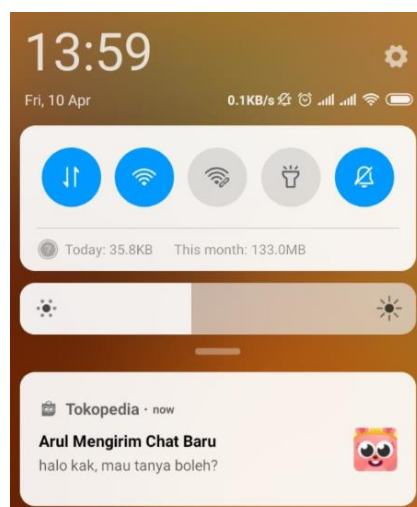
Hasil yang terakhir adalah pembuktian dari fitur komponen PWA yaitu *service worker* dengan kemampuannya adalah sebuah web PWA dapat melakukan *push notification*, hal ini dapat dilihat pada Gambar 6 yang menunjukkan bahwa aplikasi web PWA dapat mengirimkan sebuah notifikasi, salah satunya adalah notifikasi *chat*.

Hasil Pengujian Web (PWA)

Pengujian menggunakan parameter-parameter seperti *performance*, *accessibility*, *best practices*, *seo*, *page size*, *load time*. *Performance* adalah proses audit yang menilai kinerja web seperti seberapa lama web diakses dan membuat pengguna menunggu, kemudian halaman sebuah web dapat diakses atau tidak sama sekali, selain itu tidak responsive terhadap input dan interaksi yang dilakukan oleh pengguna di dalam web. *Accessibility* adalah proses audit yang memeriksa masalah umum yang dapat mencegah pengguna sulit untuk mengakses

konten dan fungsionalitas pada web, pada umumnya audit ini akan melihat seberapa mudah sebuah web dapat diakses oleh semua pengguna, terutama pengguna yang memiliki kebutuhan khusus.

Best Practices adalah proses audit yang memeriksa sebuah situs web sudah dibangun dengan cara-cara yang sudah tepat atau belum dalam pembuatan kodenya, misalnya memperbaiki rasio aspek pada gambar saat ditampilkan. *Seo* adalah proses audit yang memeriksa apakah sebuah web sudah melakukan praktik terbaik untuk memastikan situs web dapat dengan mudah ditemukan di dalam mesin pencarian. *Page size* adalah proses audit yang melihat seberapa besar ukuran halaman web yang diakses. *Load Time* adalah proses audit yang memeriksa kecepatan sebuah web dalam memuat sebuah halaman web. Pengujian ini dilakukan menggunakan *Geek Flare tools* untuk mendapatkan nilai *performance*, *best practices*, *seo*, *page size*, dan *load time*.



Gambar 6. Contoh *Push Notification* pada Web PWA

Pengujian dilakukan menggunakan server yang berposisi di United States. Lighthouse tools digunakan untuk menguji dan mendapatkan nilai dari *accessibility* dan tingkat keberhasilan suatu web dalam penerapan PWA mengikuti kriteria PWA seperti *fast and reliable*, *installable*, dan *PWA optimized*. Tabel 5 dan Tabel 6 menunjukkan hasil pengujian yang sudah dilakukan pada beberapa situs web. Tabel 5 menunjukkan bahwa hampir keseluruhan web PWA memiliki skor yang terbilang sangat baik dari *performance*, *accessibility*, *best practices*, dan

juga *seo*. Hal ini dibuktikan dengan rata-rata dari masing-masing web 86 sampai dengan 97.5. Hampir semua web PWA yang dilakukan pengujian mendekati angka 90 dan sudah mencapai angka lebih besar sama dengan 90. Selain itu web PWA memiliki hasil *load time* dan *page size* yang rendah sehingga dapat diakses dengan cepat. Tabel 6 menunjukkan hasil yang diperoleh dari pengujian seberapa berhasil implementasi PWA dilakukan berdasarkan kriteria yang harus dipenuhi suatu web PWA, yaitu *fast and reliable*, *installable*, dan *pwa optimized*.

Tabel 5. Hasil Pengujian pada Situs Web (PWA)

Situs Web	Performance	Accessibility	Best Practices	Seo	Rata-rata skor	Page Size	Load Time
olx.co.id	89	77	86	97	87.25	0.9 mb	4.8 s
trivago.co.id	96	95	79	88	89.5	1.3 mb	3.5 s
makemytrip.com	100	95	93	94	95.5	2.7 kb	205 ms
yummly.com	76	98	100	100	93.5	3.9 mb	570 ms
app.starbucks.com	98	92	79	75	86	1.8 mb	2.7 s
sudoku.jull.dev	100	66	100	100	91.5	0.2 mb	0.4 s
open.spotify.com	100	90	100	100	97.5	0.1 mb	0.5 s
twitter.com	97	87	86	91	90.25	1.1 mb	4.1 s
tokopedia.com	98	78	86	90	88	1.1 mb	2.6 s
lofi.news	100	84	93	100	94.25	0.9 mb	1 s

Tabel 6. Hasil Pengujian Berdasarkan Kriteria Web PWA

Situs Web	Fast and Reliable (%)	Indeks syarat yang tidak terpenuhi pada Tabel 2	Installable (%)	Indeks syarat yang tidak terpenuhi pada Tabel 3	PWA Optimized (%)	Indeks syarat yang tidak terpenuhi pada Tabel 4
olx.co.id	46,2	No 1	100	-	100	-
trivago.co.id	38,5	No 1, 3	100	-	100	-
makemytrip.com	7,7	No 1, 2	100	-	100	-
yummly.com	46,2	No 1	100	-	100	-
app.starbucks.com	46,2	No 1	100	-	100	-
sudoku.jull.dev	100	-	100	-	100	-
open.spotify.com	0	No 1,2,3	100	-	77,7	No 3, 7
twitter.com	46,2	No 1, 2	100	-	100	-
tokopedia.com	46,2	No 1	100	-	100	-
lofi.news	100	-	100	-	88,8	No 7

Hasil pengujian menunjukkan semua web yang dilakukan pengujian sudah memenuhi kedua kriteria tersebut hanya saja open.spotify.com dan lofi.news masih sedikit kurang memenuhinya. Selain itu, untuk *fast and reliable*, web PWA yang diuji hampir seluruhnya belum memenuhi kriteria tersebut terutama memenuhi persyaratan nomor 1, sehingga skor dari masing-masing web sebesar 46,2%.

KESIMPULAN DAN SARAN

Berdasarkan hasil pengujian yang didapatkan terhadap situs web yang sudah mengimplementasikan pengembangan web dengan metode PWA, dapat ditarik kesimpulan bahwa PWA dapat ditambahkan ke dalam perangkat *mobile* melalui *browser*, terdapat *icon* pada beranda ponsel, dan bisa melakukan *push notification* bahkan saat aplikasi sedang ditutup.

Berdasarkan skor penilaian yang diperoleh dari pengujian web PWA dapat disimpulkan bahwa pengembangan web dengan menerapkan PWA dapat memberikan dampak yang sangat baik. Dilihat dari skor yang diperoleh dengan parameter *performance*, *accessibility*, *best practices*, *seo*, terlihat bahwa skor yang dihasilkan mempunyai rata-rata yang cukup tinggi dengan rata-rata 86 sampai dengan 97,5. Di mana jika dilihat dari skor penilaian yang direkomendasikan Google, suatu web dapat terbilang sangat baik jika skor yang didapat adalah 90 sampai

dengan 100. Dari sisi *load time* web PWA menunjukkan waktu yang sangat cepat pada saat memuat sebuah halaman web pada *mobile* dengan waktu tercepat adalah 205ms dan yang paling lama adalah 4.8s.

Dari sisi *page size* berdasarkan web PWA yang diuji memperoleh angka terendah sebesar 0.1mb dan tertinggi 3.9mb. Rata-rata pengembang web PWA sudah memenuhi persyaratan dalam membangun web PWA tersebut dari sisi *installable* dan *pwa optimized*.

Namun, jika dilihat dari sisi *fast and reliable*, para pengembang perlu melakukan focus dan perbaikan agar implementasi untuk *fast and reliable* ini dapat terpenuhi persyaratannya agar implementasi web PWA yang dilakukan dapat lebih terasa manfaatnya jika semua kriterianya sudah berhasil diterapkan. Dengan adanya penelitian ini diharapkan para pelaku bisnis dan pengembang aplikasi mendapatkan pengetahuan yang lebih dalam tentang fitur dan kemampuan teknologi PWA yang nantinya dapat dimanfaatkan sebagai bahan pertimbangan apakah ingin mengadopsi metode dari PWA atau tidak ke dalam sebuah aplikasi yang ingin dibangun.

Penelitian lebih lanjut dapat dilakukan untuk mengetahui apakah teknologi PWA mampu menjadi solusi dari kesenjangan antara aplikasi *native* dan aplikasi web, serta mengatasi kesenjangan antar perangkat *mobile* dalam membangun suatu aplikasi (*cross-platform*).

DAFTAR PUSTAKA

- [1] A. Bhilare, Y. Gaikwad, K. Varsha and K. Satish, "Review on: Progressive Web App for Organization System", *International Journal of Management, Technology And Engineering* Volume 8, Issue IX, September/2018 ISSN NO : 2249-745, vol. 8, no. 9, p. 988, 2018. [Accessed 3 April 2020].
- [2] D. Kirkpatrick, "Google: 53% of mobile users abandon sites that take over 3 seconds to load", Marketing Dive, 2016. [Online]. Available: <https://www.marketingdive.com/news/google-53-of-mobile-users-abandon-sites-that-take-over-3-seconds-to-load/426070/>. [Accessed: 10- Apr- 2020].
- [3] SoftwareSeni, "Membuat Aplikasi Dengan Progressive Web Application (PWA)", *SoftwareSeni*, 2019. [Online]. Available: <https://www.softwareseni.co.id/progressive-web-application-pwa/>. [Accessed: 03- Apr- 2020].
- [4] R. Mishra, "Progressive WEBAPP: Review", *International Research Journal Of Engineering and Technology (IRJET)*, vol. 3, no. 6, 2016. [Accessed 3 April 2020].
- [5] S. Tandel and A. Jamadar, "Impact of Progressive Web Apps on Web App Development", *International Journal of Innovative Research in Science, Engineering and Technology*, vol. 7, no. 9, 2018. [Accessed 4 April 2020].
- [6] T. Majchrzak, A. Hansen and T. Gronli, "Progressive Web Apps: the Definite Approach to Cross-Platform Development?", in *Proceedings of the 51st Hawaii International Conference on System Sciences*, 2018.
- [7] Nurwanto, "Penerapan Progressive Web Application (PWA) pada E-Commerce", *Techno.Com*, vol. 18, no. 3, pp. 227-235, 2019. Available: 10.33633/tc.v18i3.2400.
- [8] A. Mhaske, A. Bhattad, P. Khamkar and R. More, "Progressive Web App for Educational System", *International Research Journal of Engineering and Technology (IRJET)*, vol. 5, no. 1, 2018. [Accessed 4 April 2020].
- [9] "Lighthouse Scoring Guide | Tools for Web Developers", *Google Developers*. [Online]. Available: <https://developers.google.com/web/tools/lighthouse/v3/scoring?hl=id>. [Accessed: 10- Apr- 2020].
- [10] "PWA audits", *web.dev*. [Online]. Available: <https://web.dev/lighthouse-pwa/>. [Accessed: 11- Apr- 2020].
- [11] "Lighthouse v5 Score Weighting [MAKE YOUR OWN COPY]", *Google Docs*, 2020. [Online]. Available: <https://docs.google.com/spreadsheets/d/1up5rxd4EMCoMa>

- xH8cppcK1x76n6HLx0e7jxb0e0FXvc/edit#gid=0. [Accessed: 10- Apr- 2020].
- [12] "12 Best Examples of Progressive Web Apps (PWAs) in 2020 - SimiCart", *SimiCart Blog | eCommerce Insights & Magento Tutorials*, 2020. [Online]. Available: <https://www.simicart.com/blog/progressive-web-apps-examples/>. [Accessed: 10- Apr- 2020].