

HUBUNGAN ANTARA CACHE, ENERGY CONSUMPTION DAN RUNTIME PERFORMANCE PADA PROGRESSIVE WEB APPS

Wakhid Kurniawan¹, Agung Fatwanto^{*2}

^{1,2}Universitas Islam Negeri Sunan Kalijaga Yogyakarta, Kabupaten Sleman

Email: ¹kurniawan.wk48@gmail.com, ²agung.fatwanto@uin-suka.ac.id

*Penulis Korespondensi

(Naskah masuk: 03 Mei 2021, diterima untuk diterbitkan: 17 Februari 2022)

Abstrak

Progressive Web Apps merupakan teknologi mutakhir di pengembangan web. Kehadiran *Progressive Web Apps* memberikan dampak positif bagi pengembang Web, yang mana keunggulannya memberikan peningkatan performa secara signifikan. *Service Worker* sebagai inti memiliki kemampuan mencegat dan menangani permintaan jaringan, termasuk mengelola *cache respons* lewat program. Semua kemampuan *Service worker* disimpan ke dalam cache. Keunggulan tersebut tentunya mempunyai implikasi pada performance. Disisi lain konsumsi energi perangkat lunak yang berjalan di atasnya, juga merupakan tantangan seorang pengembang aplikasi khususnya Web. Tujuan penelitian ini adalah melakukan investigasi hubungan *Cache* terhadap *Runtime Performance* dan *Energy Consumption*. Metode penelitian menggunakan pendekatan kuantitatif dengan melakukan kajian eksperimen empiris, yang mana menilai 16 Situs yang telah mengadopsi *Progressive Web Apps*. Perancangan eksperimen ini memiliki variabel *Cache*, *Energy Consumption* dan *Runtime Performance* dengan perilaku pengumpulan data ketika *Cache* kosong dan *Cache* terisi yang selanjutnya akan di analisis dengan analisis *Pearson Correlation*. Pelaksanaan eksperimen dijalankan di *Chrome Browser* perangkat android secara langsung. Hasil dari penelitian ini menunjukkan bahwa ada hubungan antara *Cache* dan *Energy Consumption* dengan nilai signifikansi sebesar 0,005 pada *Cache* kosong dan nilai signifikansi sebesar 0,015 pada saat *Cache* terisi. Selain itu, juga ada hubungan antara *Runtime Performance* dan *Energy Consumption* dengan nilai signifikansi sebesar 0,008 pada *Cache* kosong dan nilai signifikansi sebesar 0,003 pada saat *Cache* terisi. Sedangkan pada variabel *Cache* dan *Runtime Performance* menunjukkan bahwa tidak ada hubungan antar variabel, dengan nilai signifikansi lebih besar dari 0,05 yaitu 0,08 pada *Cache* kosong dan 0,098 pada saat *Cache* terisi.

Kata kunci: PWA, Konsumsi Energi, Performa Web

CORRELATION BETWEEN CACHE, ENERGY CONSUMPTION AND RUNTIME PERFORMANCE ON PROGRESSIVE WEB APPS

Abstract

Progressive Web Apps are the latest technology in web development. The presence of *Progressive Web Apps* has a positive impact on Web developers, whose advantages provide a significant increase in performance. *Service Workers* at the core have the ability to intercept and handle network requests, including managing programmatic response caches. All *Service worker* capabilities are cached. These advantages of course have implications for performance. On the other hand, the energy consumption of the software running on it is also a challenge for an application developer, especially the Web. The purpose of this study is to investigate the relationship between *Cache* and *Runtime Performance* and *Energy Consumption*. The research method uses a quantitative approach by conducting an empirical experimental study, which assesses 16 sites that have adopted *Progressive Web Apps*. This experimental design has *Cache*, *Energy Consumption* and *Runtime Performance* variables with data collection behavior when the cache is empty and the cache is filled which will then be analyzed with *Pearson Correlation* analysis. The experiment execution is run in the *Chrome Browser* of the android device directly. The results of this study indicate that there is a relationship between *Cache* and *Energy Consumption* with a significance value of 0,005 in an empty cache and a significance value of 0,015 when the cache is filled. In addition, there is also a relationship between *Runtime Performance* and *Energy Consumption* with a significance value of 0,008 when the cache is empty and a significance value of 0,003 when the cache is filled. While the *Cache* and *Runtime Performance* variables show that there is no relationship between variables, with a significance value greater than 0,05, namely 0,08 when the cache is empty and 0,098 when the cache is filled.

Keywords: PWA, Energy Consumption, Performance

1. PENDAHULUAN

Perkembangan teknologi aplikasi web sudah banyak mengalami perubahan fungsi. Awal mula perkembangan teknologi web dimulai dari web 1.0 yang diperkenalkan tahun 1990-an yang mana bersifat statis hingga menjadi aplikasi web yang dapat menangani masalah pengecekan status baterai, penggunaan mode *offline*, hingga *speech recognition*. Salah satu teknologi yang tengah banyak diperhatikan beberapa tahun terakhir ini adalah teknologi *Progressive Web Apps* (PWA) (Kurniawan dkk., 2018)(Majchrzak dkk., 2018).

Progressive Web Apps (PWA) merupakan sebuah istilah untuk aplikasi berbasis web yang menggunakan teknologi web mutakhir (Syarifudin dkk., 2019). PWA sebenarnya hanyalah aplikasi berbasis web biasa, tapi memanfaatkan fitur perambanan yang modern agar tampil seolah-olah merupakan aplikasi mobile asli. Diperkenalkan pertama kali pada tahun 2015 oleh Frances Berriman dan Insinyur Google Chrome Alex Russell menciptakan istilah PWA (Khan, Al-Badi and Al-Kindi, 2019). Secara khusus PWA ini memiliki keunggulan antara lain, *progressive enhancement*, *low* atau *no network connectivity*, *background processing capabilities*, dan *push notifications* (Marcillo, Pérez-Medina and Muñoz, 2019). PWA digambarkan sebagai kumpulan dari teknologi, konsep desain dan WEB API (*Application Programming Interface*) yang bekerja secara bersama untuk memberikan sentuhan mobile pada sebuah aplikasi web (Malavolta dkk., 2020). Secara umum inti dari pemanfaatan PWA adalah berjalan seperti aplikasi seluler lainnya tanpa perlu melalui *Apps Market* seperti Google Play Store, App Store, Galaxy Store dan lain-lain.

Berekal keunggulan yang dimiliki, PWA cenderung lebih cepat daripada situs web standar pada umumnya (Karpagam, 2017). Inti dari PWA adalah *service worker*, yang merupakan sekumpulan JavaScript API yang memungkinkan *background processing capabilities*, *push notifications*, sehingga pengembang dapat manipulasi *cache* secara terprogram dan *preload assets* yang digunakan. Fitur inti yang dimaksud ini adalah kemampuan mengendalikan arus permintaan jaringan, termasuk mengelola *cache respons* lewat program.

Saat aplikasi web standar dijalankan pada perangkat mobile mengharuskan adanya internet, menunggu proses memuat halaman web, termasuk memuat *assets*, konten dinamis, gambar, script, dll. Berbeda halnya dengan PWA yang dapat menyimpan sumber daya dan modul javascript dalam *Cache* khusus pada browser setelah pertama kali di akses, sehingga memungkinkan PWA dapat berjalan secara *offline*. Hal ini dikarenakan halaman web yang dimuat sebelumnya di *Client*, *Proxy* dan *Server* dapat disimpan dalam *Cache*.

Mengingat *Mobile Device* memiliki keterbatasan masa pakai baterai, maka kemampuan

programmer dalam memahami karakter atau metode *Energy Consumption* perangkat lunak yang berjalan pada perangkat tersebut, merupakan hal yang penting. Berdasarkan keunggulan PWA dalam mengelola *cache* tentunya ada kemungkinan berimplikasi pada *Performance* perangkat lunak tersebut, disisi lain juga adanya kemungkinan implikasi terhadap *energy consumption* (3PillarGlobal, 2019)(Dutta and Vandermeer, 2017)(Tawalbeh, dkk., 2016)(Pramanik dkk., 2019). Operating System (OS) pada penelitian ini adalah Android yang mana merupakan OS yang populer di dunia, dengan persentase pengguna 71,9% menurut statcounter seperti pada Gambar 1.

Ada beberapa penelitian yang terkait *cache* dan *Energy Consumption*. Penelitian pertama membahas tentang investigasi korelasi antara *Performance Scores* dari 21 situs *mobile web apps* (PWA dan non-PWA) yang dihasilkan dari *Lighthouse* dan *Energy Consumption* yang diukur dengan *Android Profiler* pada *Mobile Web Apps* (Chan-Jong-Chu dkk., 2020). Penelitian kedua, evaluasi tentang dampak *caching* pada *energy consumption* dan *performance Progressive Web Apps* (PWA) dari 9 situs yang mengadopsi PWA, yang mana salah satu variabel yang digunakan adalah *pageload time* yang diasumsikan sebagai *performance* dan pengukuran konsumsi energi menggunakan *Dumpsys*, hasil dari *bug report* android (Malavolta dkk., 2020).

Sehubungan dengan penelitian-penelitian sebelumnya, demi menjaga orisinalitas, perbedaannya dengan penelitian ini adalah membahas tentang investigasi hubungan antara *cache*, *Runtime Performance* dan *Energy Consumption* pada PWA, yang mana *Runtime Performance* dalam penelitian ini adalah performa halaman saat dijalankan, biasanya sering digunakan untuk menganalisis *Respons*, *Animasi*, dan *Idle (RAIL)* pada halaman (Basques, 2020). Selain itu pengukuran konsumsi energi pada penelitian ini menggunakan *Dumpsys Battery Historian*.

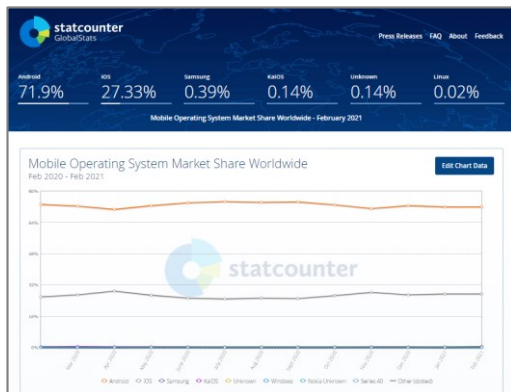
Kontribusi utama dari penelitian ini adalah hasil eksperimen tentang hubungan *caching* pada *Runtime Performance* dan *Energy Consumption* pada PWA bagi pengembang. Manfaat yang di dapat yaitu:

- (i) Pengembang browser (contoh Chrome) dapat mengambil keputusan yang tepat dalam mengembangkan dan merancang browser mereka kedepan dari bukti obyektif tentang *cache* PWA.
- (ii) Memberikan wawasan pengembang web sejauh mana hubungan *cache*, *Runtime Performance* dan *Energy Consumption* pada PWA.

Sehubungan dengan kejelasan arah dan cakupan pada penelitian ini maka, perlu dibuat pertanyaan penelitian, adapun pertanyaan penelitian ini sebagai berikut:

- (i) Bagaimana hubungan antara *Cache* dan *Energy Consumption* pada saat *cache* kosong?

- (ii) Bagaimana hubungan antara *Cache* dan *Runtime Performance* pada saat cache kosong?
- (iii) Bagaimana hubungan antara *Cache* dan *Energy Consumption* pada saat cache terisi?
- (iv) Bagaimana hubungan antara *Cache* dan *Runtime Performance* pada saat cache terisi?
- (v) Bagaimana hubungan antara *Energy Consumption* dan *Runtime Performance* pada saat cache kosong?
- (vi) Bagaimana hubungan antara *Energy Consumption* dan *Runtime Performance* pada saat cache terisi?



Gambar 1. Data Pengguna Android Di Seluruh Dunia

2. METODE PENELITIAN

Pendekatan yang digunakan penelitian ini adalah pendekatan kuantitatif dan metode yang digunakan adalah eksperimen empiris. Adapun perancangan dan pelaksanaan eksperimen yang dilakukan sebagai berikut:

2.1 Perancangan Eksperimen

2.1.1 Seleksi Subyek Penelitian

Proses seleksi situs PWA dilakukan beberapa tahap, pada awalnya total situs yang terkoleksi terdapat 27 situs. Kemudian situs akan diuji, yaitu (i) harus lulus uji PWA di *Google Lighthouse*; (ii) halaman login tidak sebagai halaman utama; (iii) bukan merupakan situs Games; hal ini dilakukan untuk menjaga konsistensi pengukuran halaman utama, yang pada akhirnya yang telah lulus seleksi adalah 16 situs (Chan-Jong-Chu dkk., 2020).

2.1.2 Variabel Eksperimen

Pada penelitian ini terdiri dari enam variabel yaitu:

- a. *Empty Cache* = data *cache size* saat cache kosong.
- b. *Populate Cache* = data *cache size* saat cache telah terisi.
- c. *Energy Consumption^{ec}* = data penggunaan *energy consumption* saat cache kosong.
- d. *Rutime Performance^{ec}* = data waktu *Rutime Performance* saat cache kosong.
- e. *Energy Consumption^{pc}* = data penggunaan *energy consumption* saat cache terisi.

- f. *Rutime Performance^{pc}* = data waktu *Rutime Performance* saat cache terisi.

Tabel 1. Situs PWA

No	Situs PWA	Link
1	AliExpress	https://m.id.aliexpress.com/
2	BookMyShow	https://in.bookmyshow.com/
3	Olx	https://www.olx.co.id/
4	MakeMyTrip	https://www.makemytrip.com/
5	Trivago	https://www.trivago.co.id/
6	George	https://global.direct.asda.com/
7	Treebo	https://www.treebo.com/
8	Biggerpicture	https://www.biggerpicture.agency/
9	Starbucks	https://app.starbucks.com/
10	Petlove	https://www.petlove.com.br/
11	Nau	https://www.nau.ch/
12	Slate	https://slate.com/
13	Edgy	https://edgy.app/
14	Smashingmagazine	https://www.smashingmagazine.com/
15	Soundslice	https://www.soundslice.com/
16	Wego	https://www.wego.co.id/

2.1.3 Hipotesis

Penelitian ini merumuskan hipotesis dengan dua model. Pertama pengujian hipotesis ketika *cache* kosong dan kedua pengujian hipotesis ketika *cache* telah terisi. Adapun variabel yang digunakan, yaitu: Sehingga di dapatkan hipotesis dibawah ini:

H_0 = Tidak ada hubungan antara *Empty Cache* dan *Energy Consumption^{ec}*.

H_1 = Ada hubungan antara *Empty Cache* dan *Energy Consumption^{ec}*.

H_0 = Tidak ada hubungan antara *Empty Cache* dan *Rutime Performance^{ec}*.

H_1 = Ada hubungan antara *Empty Cache* dan *Rutime Performance^{ec}*.

H_0 = Tidak ada hubungan antara *Populate Cache* dan *Energy Consumption^{pc}*.

H_1 = Ada hubungan antara *Populate Cache* dan *Energy Consumption^{pc}*.

H_0 = Tidak ada hubungan antara *Populate Cache* dan *Rutime Performance^{pc}*.

H_1 = Ada hubungan antara *Populate Cache* dan *Rutime Performance^{pc}*.

H_0 = Tidak ada hubungan antara *Energy Consumption^{ec}* dan *Rutime Performance^{ec}*.

H_1 = Ada hubungan antara *Energy Consumption^{ec}* dan *Rutime Performance^{ec}*.

H_0 = Tidak ada hubungan antara *Energy Consumption^{pc}* dan *Rutime Performance^{pc}*.

H_1 = Ada hubungan antara *Energy Consumption*^{pc} dan *Runtime Performance*^{pc}.

2.1.4 Analisis Data

Data eksperimen yang diperoleh yang selanjutnya di analisis secara kuantitatif. Hal pertama yang dilakukan adalah melakukan uji normalitas mengingat data ini kurang dari 50 sampel, maka pengujian normalitas dilakukan dengan *Shapiro-Wilk* (Elliott AC, 2007). Selain itu di tampilkan secara visual berupa Histogram dan PP-Plot. Apabila data tidak normal maka langkah kedua perlu dilakukannya transformasi data menggunakan Lg^{10} , Ln dan sebagainya (Perdana K, 2016). Kemudian setelah diketahui bahwa data telah terdistribusi normal, maka langkah ketiga adalah Uji Linieritas, hal ini dilakukan untuk menentukan apakah data mempunyai hubungan secara linier antar variabel atau tidak (Verbeek, 2017). Langkah keempat adalah Uji *Pearson Correlations*, untuk melihat nilai signifikansi keeratan hubungan antar variabel (Fitri, 2017)(Mohd dkk., 2020).

2.2 Pelaksanaan Eksperimen

Tabel 2. Spesifikasi Devices Android Model SM-A507FN/DS

Display	6.4 inch Super AMOLED
Chipset	Exynos 9611 (10nm)
OS	Android 10
RAM	4 GB
Internal	64 GB
Battery	Li-Po 4000 mAh, non-removable

Pelaksanaan Eksperimen Penelitian ini, di lakukan secara langsung pada *Android Device* yang menjalankan Chrome Browser dengan spesifikasi dijelaskan pada tabel 2. Langkah pertama yang dilakukan adalah bagaimana cara mengendalikan *android device* pada Windows, dalam hal ini menggunakan alat bantu *Minimal ADB and Fastboot* dan fasilitas *chrome://inspect/#devices*. Langkah kedua setiap kali pelaksanaan eksperimen pastikan *dummysys* pada *Minimal ADB and Fastboot* telah mereset baterai terlebih dahulu. Selain itu hal yang perlu diperhatikan adalah waktu, yang mana waktu memuat masing-masing situs PWA adalah tiga menit. Waktu tersebut merupakan satu rangkaian untuk mencatat nilai pengukuran *Runtime Performance* pada *empty cache* maupun *populate* sekaligus (Malavolta dkk., 2020).

Adapun proses pengumpulan data yang diperoleh pada pelaksanaan eksperimen ini adalah sebagai berikut:

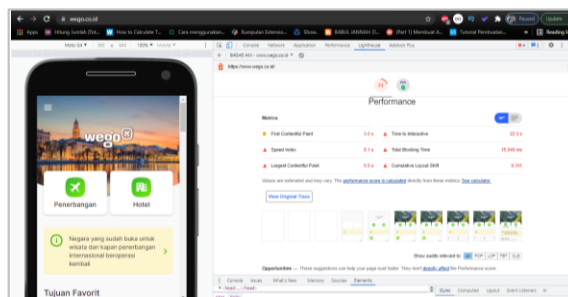
2.2.1 Data Cache

Proses pengumpulan data *cache* dilakukan dengan cara mengkosongkan terlebih dahulu *cache*, hal ini dilakukan dengan cara inspect halaman situs,

cek pada *tab application*, kemudian *clear cache*. Selanjutnya yang dilakukan adalah muat ulang halaman situs untuk mendapatkan data *empty cache* melalui *Google Lighthouse* yang merupakan ekstensi tambahan dari chrome. Setelah mendapatkan data *empty cache* muat ulang situs tersebut melalui *Google Lighthouse*, sehingga data *populate cache* berhasil di dapatkan.

2.2.2 Data Runtime Performance

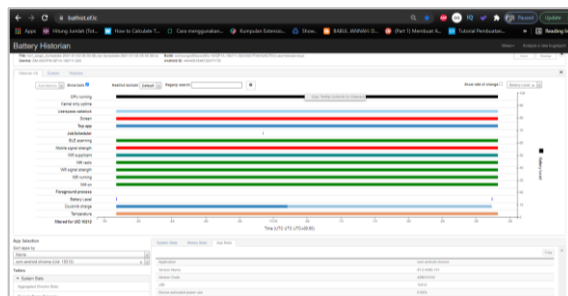
Seperti penjelasan sebelumnya pelaksanaan eksperimen ini merupakan satu rangkaian, sehingga data yang di peroleh sesuai dengan keadaan pada waktu itu. Untuk memperoleh data *Runtime Performance* pada halaman *Google Lighthouse* masuk ke *View Original Trace* untuk melihat satuan waktu yang dibutuhkan dalam memuat halaman situs tersebut lihat gambar 2.



Gambar 2. Proses Pengumpulan Data Runtime Performance

2.2.3 Data Energy Consumption

Hal yang perlu diperhatikan saat pengumpulan data *Energy Consumption* adalah *Minimal ADB and Fastboot* telah mereset baterai. Hal ini dilakukan karena pada *bug report* perangkat selalu mengumpulkan statistik baterai dan informasi debug lainnya di latar belakang. *Reset* akan menghapus data pengumpulan baterai lama. Jika tidak melakukan *reset* baterai, hasilnya akan sangat besar. Setelah waktu berjalan tiga menit, lakukan *bug report* pada *android device* maka akan menghasilkan *file type .zip*. Lakukan *submit* pada *Battery Historian* untuk melihat besaran *energy* yang digunakan, selanjutnya di konversi dalam satuan *Joule (J)*. Hasil submit seperti gambar 3. Langkah ini dilakukan setiap kali melakukan muat halaman situs satu rangkaian dengan pengukuran *cache* dan *runtime performance*.



Gambar 3. Proses Pengumpulan Data Energy Consumption

3. HASIL DAN PEMBAHASAN

Sesuai dengan tahapan-tahapan analisis data pada penelitian ini maka di dapatkan hasil sebagai berikut:

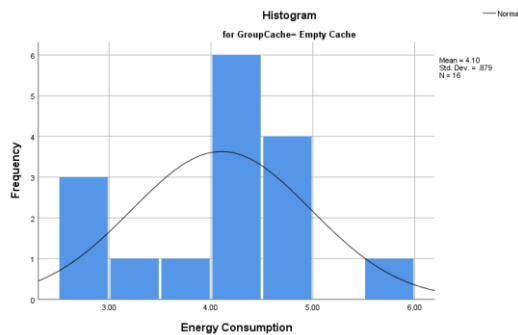
3.1 Uji Normalitas

Pada Uji Normalitas ini menggunakan *Shapiro-Wilk*.

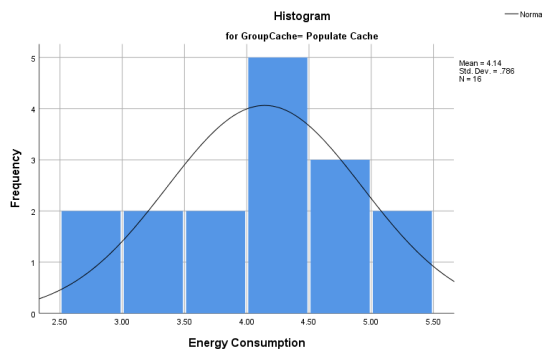
Tabel 3. Uji Normalitas

Tests of Normality				
Shapiro-Wilk				
	Group Cache	Statistic	df	Sig.
Energy	Empty Cache	.937	16	.315
Consumption	Populate Cache	.965	16	.748
Runtime	Empty Cache	.936	16	.307
Performance	Populate Cache	.959	16	.645

Berdasarkan Tabel 3. diketahui bahwa nilai signifikansi *energy consumption* adalah sebesar 0,315 pada *empty cache* dan 0,748 pada *populate cache*, keduanya lebih besar dari 0,05. Sehingga dapat disimpulkan bahwa data *energy consumption* berdistribusi normal. Sedangkan pada *runtime performance* menunjukkan nilai sebesar 0,307 pada *empty cache* dan 0,645 pada *populate cache*, keduanya lebih besar dari 0,05. Sehingga dapat disimpulkan bahwa data *runtime performance* juga berdistribusi normal.

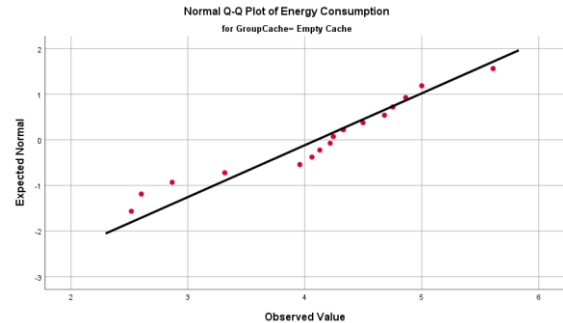


Gambar 4. Histogram Normalitas *Energy Consumption* ketika *Empty Cache*

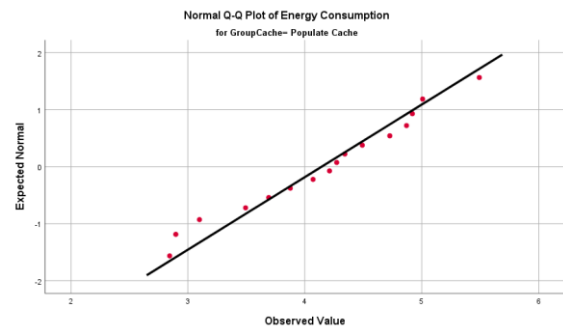


Gambar 5. Histogram Normalitas *Energy Consumption* ketika *Populate Cache*

Melihat Gambar 4. terdapat *outlier* sedangkan pada Gambar 5. kurva cenderung sedikit ke kanan namun secara garis besar distribusi data masih mengikuti kurva normal, sehingga dapat disimpulkan bahwa kedua data tersebut berdistribusi normal.

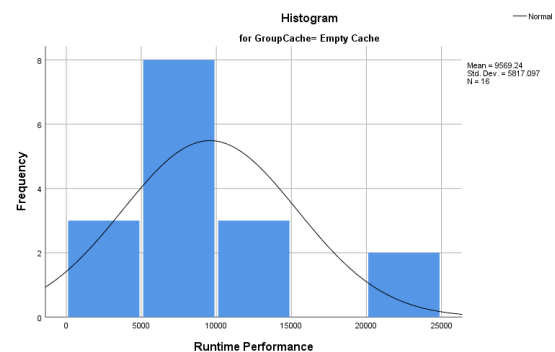


Gambar 6. PP-Plot Normalitas *Energy Consumption* ketika *Empty Cache*



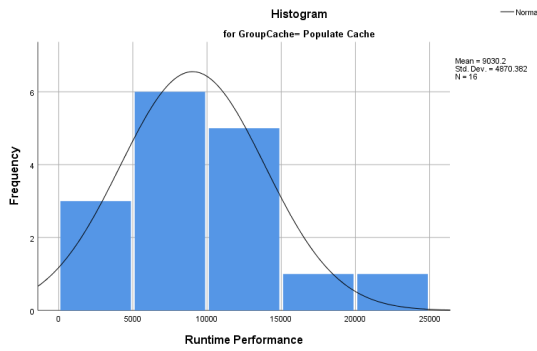
Gambar 7. PP-Plot Normalitas *Energy Consumption* ketika *Populate Cache*

Berdasarkan PP-Plot pada Gambar 6. dan Gambar 7. sebaran data cenderung mengikuti garis normal. Sehingga secara visual dapat disimpulkan data menunjukkan terdistribusi normal.

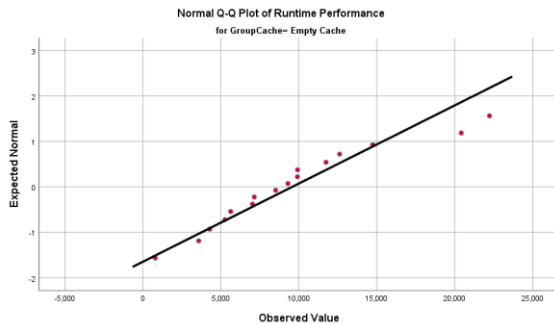


Gambar 8. Histogram Normalitas *Runtime Performance* ketika *Empty Cache*

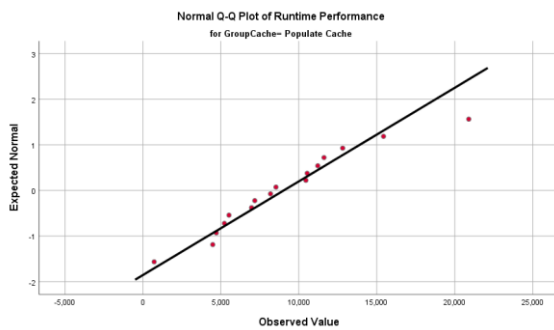
Pada Gambar 8. menunjukkan bahwa pola histogram sedikit condong ke kiri dan terdapat outlier, sedangkan pada Gambar 9. menunjukkan kurva sedikit condong ke kiri, namun secara garis besar distribusi kedua data tersebut masih berbentuk lonceng atau mengikuti kurva normal, sehingga dapat disimpulkan bahwa data berdistribusi normal.



Gambar 9. Histogram Normalitas *Runtime Performance* ketika *Populate Cache*



Gambar 10. PP-Plot Normalitas *Runtime Performance* ketika *Empty Cache*



Gambar 11. PP-Plot Normalitas *Runtime Performance* ketika *Populate Cache*

Melihat PP-Plot uji normalitas pada *runtime performance* ditampilkan pada Gambar 10. terdapat dua data sedikit melebar dari garis diagonal dan pada Gambar 11. terdapat satu data yang melebar dari

garis diagonal meskipun demikian menurut data statistik uji *Shapiro-Wilk* menunjukkan kedua data tersebut masih berdistribusi normal. Sehingga secara visual masih dapat disimpulkan data menunjukkan terdistribusi normal.

3.2 Uji Linearitas

Dasar pengambilan keputusan pada dalam uji linieritas dapat dilakukan dengan dua cara yaitu nilai signifikansi dan nilai F. Namun pada pengambilan keputusan ini cukup menggunakan nilai signifikansi.

Berdasarkan hasil Tabel 4. diketahui *Empty Cache* dan *Energy Consumption^{ec}* diperoleh nilai *Deviation from Linearity Sig.* 0,644 lebih besar dari 0,05, maka dapat disimpulkan bahwa ada hubungan linear secara signifikan antara *Empty Cache* dan *Energy Consumption^{ec}*. Sedangkan pada *Populate Cache* dan *Energy Consumption^{pc}* memiliki nilai *Deviation from Linearity Sig.* 0,451 lebih besar dari 0,05, maka dapat disimpulkan bahwa ada hubungan linear secara signifikan antara *Populate Cache* dan *Energy Consumption^{pc}*.

Selanjutnya, diketahui *Empty Cache* dan *Runtime Performance^{ec}* memiliki nilai *Deviation from Linearity Sig.* 0,734 lebih besar dari 0,05, maka dapat disimpulkan bahwa ada hubungan linear secara signifikan antara *Empty Cache* dan *Runtime Performance^{ec}*. Sedangkan pada *Populate Cache* dan *Runtime Performance^{pc}* diketahui bahwa memiliki nilai *Deviation from Linearity Sig.* 0,880 lebih besar dari 0,05, maka dapat disimpulkan bahwa ada hubungan linear secara signifikan antara *Populate Cache* dan *Runtime Performance^{pc}*.

Selanjutnya, diketahui *Runtime Performance^{ec}* dan *Energy Consumption^{ec}* memiliki nilai *Deviation from Linearity Sig.* 0,266 lebih besar dari 0,05, maka dapat disimpulkan bahwa ada hubungan linear secara signifikan antara *Runtime Performance^{ec}* dan *Energy Consumption^{ec}*. Sedangkan pada *Runtime Performance^{pc}* dan *Energy Consumption^{pc}* diketahui bahwa memiliki nilai *Deviation from Linearity Sig.* 0,361 lebih besar dari 0,05, maka dapat disimpulkan bahwa ada hubungan linear secara signifikan antara *Runtime Performance^{pc}* dan *Energy Consumption^{pc}*.

Tabel 4. Uji Linieritas

ANOVA Table						
		Sum of Squares	df	Mean Square	F	Sig.
Energy Consumption ^{ec} * Empty Cache	Deviation from Linearity	2.269	5	.454	.689	.644
Energy Consumption ^{pc} * Populate Cache	Deviation from Linearity	1.755	4	.439	1.001	.451
Runtime Performance ^{ec} * Empty Cache	Deviation from Linearity	1.668	5	.334	.553	.734
Runtime Performance ^{pc} * Populate Cache	Deviation from Linearity	.633	4	.158	.286	.880
Runtime Performance ^{ec} * Energy Consumption ^{ec}	Deviation from Linearity	1.745	2	.872	1.481	.266
Runtime Performance ^{pc} * Energy Consumption ^{pc}	Deviation from Linearity	.867	2	.433	1.111	.361

Tabel 5. Uji Pearson Correlations Saat Cache Kosong

		Correlations		
		Empty Cache	Energy Consumption ^{ec}	Runtime Performance ^{ec}
Empty Cache	Pearson Correlation	1	.669**	.450
	Sig. (2-tailed)		.005	.080
	N	16	16	16
Energy Consumption ^{ec}	Pearson Correlation	.669**	1	.637**
	Sig. (2-tailed)	.005		.008
	N	16	16	16
Runtime Performance ^{ec}	Pearson Correlation	.450	.637**	1
	Sig. (2-tailed)	.080	.008	
	N	16	16	16

** . Correlation is significant at the 0.01 level (2-tailed).

* . Correlation is significant at the 0.05 level (2-tailed).

Tabel 6. Uji Pearson Correlations Saat Cache Terisi

		Correlations		
		Populate Cache	Energy Consumption ^{pc}	Runtime Performance ^{pc}
Populate Cache	Pearson Correlation	1	.594*	.428
	Sig. (2-tailed)		.015	.098
	N	16	16	16
Energy Consumption ^{pc}	Pearson Correlation	.594*	1	.684**
	Sig. (2-tailed)	.015		.003
	N	16	16	16
Runtime Performance ^{pc}	Pearson Correlation	.428	.684**	1
	Sig. (2-tailed)	.098	.003	
	N	16	16	16

** . Correlation is significant at the 0.01 level (2-tailed).

* . Correlation is significant at the 0.05 level (2-tailed).

Berdasarkan hasil Tabel 5. yang mana hasil pengukuran saat *cache* kosong menunjukkan nilai signifikansi antara *Empty Cache* dan *Energy Consumption^{ec}* 0,005 yang mana lebih kecil dari 0,05, maka dapat disimpulkan bahwa ada hubungan yang signifikan antara *Empty Cache* dan *Energy Consumption^{ec}*. *Pearson Correlations* ini bernilai positif yang artinya semakin meningkatnya *Empty Cache* maka akan meningkat pula *Energy Consumption^{ec}*.

Selanjutnya, hubungan antara *Empty Cache* dan *Runtime Performance^{ec}* memiliki nilai sebesar 0,08 yang mana lebih besar dari 0,05 maka dapat disimpulkan bahwa tidak ada hubungan yang signifikan antara *Empty Cache* dan *Runtime Performance^{ec}*.

Sedangkan, hubungan antara *Runtime Performance^{ec}* dan *Energy Consumption^{ec}* memiliki nilai sebesar 0,008 yang mana lebih kecil dari 0,05. hal ini menunjukkan ada hubungan yang signifikan antara *Energy Consumption^{ec}* dan *Runtime Performance^{ec}*. Mempehatikan *Pearson Correlations* yang bernilai positif, artinya semakin meningkatnya *Runtime Performance^{ec}* maka akan meningkat pula *Energy Consumption^{ec}*.

Berdasarkan hasil Tabel 6. yang mana hasil pengukuran saat *cache* terisi menunjukkan nilai

signifikansi antara *Populate Cache* dan *Energy Consumption^{pc}* 0,015 yang mana lebih kecil dari 0,05 maka dapat disimpulkan bahwa ada hubungan yang signifikan antara *Populate Cache* dan *Energy Consumption^{pc}*. *Pearson Correlations* ini bernilai positif yang artinya semakin meningkatnya *Populate Cache* maka akan meningkat pula *Energy Consumption^{pc}*.

Selanjutnya, hubungan antara *Populate Cache* dan *Runtime Performance^{pc}* memiliki nilai signifikansi sebesar 0,098 yang mana lebih besar dari 0,05 maka dapat disimpulkan bahwa tidak ada hubungan yang signifikan antara *Populate Cache* dan *Runtime Performance^{pc}*.

Sedangkan, hubungan antara *Runtime Performance^{pc}* dan *Energy Consumption^{pc}* memiliki nilai signifikansi sebesar 0,003 yang mana lebih kecil dari 0,05. hal ini menunjukkan ada hubungan yang signifikan antara *Energy Consumption^{pc}* dan *Runtime Performance^{pc}*. Mempehatikan *Pearson Correlations* yang bernilai positif, artinya semakin meningkatnya *Runtime Performance^{pc}* maka akan meningkat pula *Energy Consumption^{pc}*.

4. KESIMPULAN

Dari hasil investigasi beberapa situs PWA dapat disimpulkan bahwa terdapat beberapa variabel yang berhubungan dan variabel yang tidak berhubungan.

Adapun variabel-variabel yang berhubungan seperti pada variabel *Cache* dan *Energy Consumption* dengan perilaku berbeda antara *Cache* kosong dan *Cache* terisi menunjukkan bahwa terdapat hubungan dengan nilai positif, artinya semakin meningkatnya *Cache* maka akan meningkat pula *Energy Consumption* dengan masing-masing nilai signifikansi sebesar 0,005 dengan *Pearson Correlations* 66,9% pada saat *cache* kosong dan nilai signifikansi sebesar 0,015 dengan *Pearson Correlations* sebesar 59,4% pada saat *Cache* terisi.

Selanjutnya pada variabel *Runtime Performance* dan *Energy Consumption* terdapat hubungan bernilai positif, artinya semakin meningkatnya satuan waktu *Runtime Performance* maka akan meningkat pula *Energy Consumption*. Dengan masing-masing nilai signifikansi sebesar 0,008 dengan *Pearson Correlations* sebesar 63,7% pada saat *cache* kosong dan nilai signifikansi sebesar 0,003 dengan *Pearson Correlations* sebesar 68,4% pada saat *Cache* terisi.

Adapun variabel yang tidak berhubungan yaitu antara *Cache* dengan *Runtime Performance* dengan nilai signifikansi lebih besar dari 0,05 yaitu 0,08 pada *Cache* kosong dan 0,098 pada *Cache* terisi. Dari hasil analisis data ini, kedepan penelitian ini dapat dikembangkan untuk investigasi seberapa pengaruh variabel – variabel yang berhubungan tersebut.

DAFTAR PUSTAKA

- 3PILLARGLOBAL, 2019. *How Web Caching Improves Internet Performance*. [online] Tersedia di: <<https://www.3pillarglobal.com/insights/how-web-caching-improves-internet-performance>> [Diakses 9 Jan. 2021].
- BASQUES, K., 2020. *Get Started With Analyzing Runtime Performance*. [online] Tersedia di: <<https://developers.google.com/web/tools/chrome-devtools/evaluate-performance>> [Diakses 28 Des. 2020].
- CHAN-JONG-CHU, K., ISLAM, T., EXPOSITO, M.M., SHEOMBAR, S., VALLADARES, C., PHILIPPOT, O., GRUA, E.M. AND MALAVOLTA, I., 2020. Investigating the Correlation between Performance Scores and Energy Consumption of Mobile Web Apps. *ACM International Conference Proceeding Series*, (Mei), pp.190–199.
- DUTTA, K. AND VANDERMEER, D., 2017. Caching to reduce mobile app energy consumption. *ACM Transactions on the Web*, 12(1), pp.1–30.
- ELLIOTT AC, W.W., 2007. *Statistical analysis quick reference guidebook with SPSS examples*. 1st ed ed. London: Sage Publications.
- FITRI, G.A., 2017. Analisis Pengaruh Financial Distress, Leverage dan Kepemilikan Manajerial

Perusahaan Terhadap Penerapan Konservatisme dalam Akuntansi. *Jurnal Ekonomi dan Akuntansi Universitas Negeri Padang*.

- KARPAGAM, V., 2017. Performance Enhancement of Webpage Using Progressive Web App Features. *International Journal of Innovative Research in Advanced Engineering*, 03(4), pp.2349–2163.
- KHAN, A.I., AL-BADI, A. AND AL-KINDI, M., 2019. Progressive web application assessment using AHP. *Procedia Computer Science*, [online] 155, pp.289–294. Tersedia di: <<https://doi.org/10.1016/j.procs.2019.08.041>>.
- KURNIAWAN, A., ARENI, I.S. AND ACHMAD, A., 2018. Implementasi Progressive Web Application pada Sistem Monitoring Keluhan Sampah Kota Makassar. *Jurnal Penelitian Enjiniring*, 21(2), pp.34–38.
- MAJCHRZAK, T.A., BIØRN-HANSEN, A. AND GRØNLI, T.-M., 2018. Progressive Web Apps: the Definite Approach to Cross-Platform Development? *Proceedings of the 51st Hawaii International Conference on System Sciences*, pp.5735–5744.
- MALAVOLTA, I., CHINNAPPAN, K., JASMONTAS, L., GUPTA, S. AND SOLTANY, K.A.K., 2020. Evaluating the impact of caching on the energy consumption and performance of progressive web apps. *Proceedings - 2020 IEEE/ACM 7th International Conference on Mobile Software Engineering and Systems, MOBILESoft 2020*, (Oktober), pp.109–119.
- MARCILLO, W., PÉREZ-MEDINA, J.L. AND MUÑOZ, C., 2019. A progressive web application for control and reading of water meters. *RISTI - Revista Iberica de Sistemas e Tecnologias de Informacao*, 2019(19), pp.415–427.
- MOHD, T., JAMIL, S., MASROM, S., ARCHITECTURE, F. AND MARA, U.T., 2020. Multiple Linear Regression on Building Price Prediction with Green Building Determinant. *International Journal of Advance Science and Technology*, 29(9), pp.1137–1148.
- PERDANA K, E., 2016. *Olah Data Skripsi Dengan SPSS 22. Lab Kom Manajemen Fe Ubb*. Bangka Belitung: LAB KOM MANAJEMEN FE UBB.
- PRAMANIK, P.K.D., SINHABABU, N., MUKHERJEE, B., PADMANABAN, S., MAITY, A., UPADHYAYA, B.K., HOLM-NIELSEN, J.B. AND CHOUDHURY, P., 2019. Power Consumption Analysis, Measurement, Management, and Issues: A State-of-the-Art Review of Smartphone Battery and Energy Usage. *IEEE Access*, 7(Desember), pp.182113–182172.
- SYAIFUDIN, K., PRANATA, E., A, W.N. AND DIAN, A.R., 2019. Analisis Usability pada Perbandingan Web-Native dengan Web

Berbasis Progressive Web App. *Informatika*.

- TAWALBEH, M., EARDLEY, A. AND TAWALBEH, L., 2016. Studying the Energy Consumption in Mobile Devices. *Procedia Computer Science*, [online] 94(MobiSPC), pp.183–189. Tersedia di:<<http://dx.doi.org/10.1016/j.procs.2016.08.028>>.
- VERBEEK, M., 2017. Using linear regression to establish empirical relationships. *IZA World of Labor*, (Januari).

Halaman ini sengaja dikosongkan