

# Pengembangan dan Perbandingan Performa Serta Daya Tahan Terhadap Beban Tinggi Pada Website Forum Diskusi Pemrograman: Progressive Web Apps (PWA) vs non-PWA

Muhammad Rizky Cavendio<sup>1</sup>, Asmunin<sup>2</sup>

Manajemen Informatika, Universitas Negeri Surabaya  
Jl. Ketintang, Ketintang, Kec. Gayungan, Kota Surabaya, Jawa Timur 60231

<sup>1</sup>muhammadrizky.20011@mhs.unesa.ac.id

<sup>2</sup>asmunin@unesa.ac.id

**Abstrak**— Penelitian ini membahas Progressive Web Apps (PWA) dan website non-PWA yang identik secara fitur dan fungsi dalam hal performa dan daya tahan terhadap beban tinggi. Tujuannya adalah untuk memberikan pemahaman kepada pengembang, organisasi, dan akademisi tentang perbedaan antara website PWA dan non-PWA dalam hal performa dan daya tahan beban tinggi. Hasil penelitian menunjukkan bahwa PWA unggul dalam beberapa metrik performa, tetapi perbedaan tidak selalu signifikan. Metode tersebut menggunakan desain eksperimental dengan alat seperti Apache JMeter untuk pengujian beban dan Lighthouse, Pingdom, GTmetrix, dan ChromeDevTools untuk mengukur metrik performa. Dalam pengujian beban, PWA memiliki throughput yang lebih besar, meskipun waktu respons rata-rata lebih lambat daripada non-PWA. Meskipun tidak ada perbedaan signifikan pada beberapa parameter performa, throughput menunjukkan perbedaan signifikan ( $p < 0.001$ ). PWA unggul dalam throughput pada pengujian beban tinggi, tetapi non-PWA memiliki waktu respons rata-rata yang lebih lambat. Pilihan antara PWA dan non-PWA bergantung pada prioritas aplikasi yang dibutuhkan.

**Kata kunci**— Progressive Web Apps, performa, pengujian beban, website non-PWA.

**Abstract**—The studies compares the progressive Web Apps (PWAs) and non-PWA websites with similar features and functions are compared in terms of performance and high-load endurance. The goal of this study is to gain an understanding of the differences between PWAs and non-PWA websites and to provide insight to developers, organizations, and academics. In experimental design method, tools such as Apache JMeter for load testing and Lighthouse, Pingdom, GTmetrix, and ChromeDevTools are used to evaluate performance metrics. Results show that PWAs win in several performance metrics, although this difference is not always significant. PWAs show more throughput in load testing, but they have a slower average response time compared to non-PWAs. The Mann-Whitney U test does not show significant differences in performance measures, however there is a significant difference in throughput ( $p < 0.001$ ). The conclusion is that while PWAs perform well in high-load testing, their average response time is slower than that of non-PWAs. The decision

between PWA and non-PWA is determined by the application's priorities, such as throughput and advanced functionality in PWAs against faster reaction times and memory efficiency in non-PWAs.

**Keyword**— Progressive Web Apps, performance, load testing, non-PWA website.

## I. PENDAHULUAN

Dalam beberapa dekade terakhir ini, kemajuan teknologi website telah membentuk tata cara interaksi manusia dengan internet. Sejak pertama kali munculnya World Wide Web pada tahun 1990 dengan versi web 1.0 yang statis, website telah melalui beberapa fase perkembangan yang sangat signifikan. Dimulai dari web 1.0 hingga web 4.0, ini menunjukkan evolusi yang cepat dalam hal fungsi dan juga performanya. Evolusi website telah mencapai titik yang sangat signifikan sehingga bisa mengubah paradigma dan menghasilkan inovasi-inovasi baru dalam pengembangan website. Salah satu inovasi saat ini yang patut dicermati adalah hadirnya Progressive Web Apps (PWA) yang menandai pergeseran menuju user experience yang lebih canggih dan responsif di berbagai platform.

Progressive Web Apps (PWA) merupakan ide yang pertama kali diusulkan oleh insinyur Google, Alex Russel, pada 12 Juni 2015 [1]. PWA merupakan istilah untuk aplikasi web yang dibangun menggunakan teknologi web tercanggih dan sebenarnya merupakan aplikasi web biasa, namun menggunakan fitur-fitur browser modern agar terlihat seperti aplikasi native mobile. PWA digambarkan sebagai seperangkat teknologi, konsep desain, dan Web API (Application Programming Interfaces) yang bekerja sama untuk menghadirkan sentuhan aplikasi web seluler [2]. PWA sendiri menggunakan sejumlah pendekatan, mulai dari pengoptimalan aplikasi web, penggunaan teknologi baru, dan standarisasi mobile web. Hal ini dikatakan mampu meningkatkan performa hingga meningkatkan user experience [3].

Performa pada situs website mengacu pada seberapa cepat situs web dimuat di dalam sebuah browser web dan performa juga dapat menentukan kualitas ergonomi, interaktivitas, dan keandalan dari suatu situs web [4]. Oleh karena itu, performa sebuah situs web sangat penting untuk diperhatikan. Codemotion menunjukkan lambatnya performa dalam pemuatan halaman situs web dapat mengurangi user experience sebesar 16%, dan 79% user cenderung tidak akan berlama-lama menggunakan situs web yang memiliki kinerja atau performa yang buruk [5]. Hal ini juga dapat berdampak negatif pada reputasi 3 situs web, karena 44% user yang tidak puas akan memberi tahu rekan-rekan mereka tentang user experience situs web yang buruk dan berujung dapat mengakibatkan turunnya pengunjung web.

Selain itu, performa pada sebuah website dapat dipengaruhi oleh beberapa faktor, salah satu faktornya yaitu jumlah akses pengguna yang mengakses website di waktu bersamaan. Maka dibutuhkan web yang handal agar semua request dari pengguna dapat ditanggapi dengan baik [6]. Oleh sebab itu, pengujian daya tahan terhadap beban tinggi pada website perlu dilakukan untuk mengetahui sejauh mana sebuah website mampu bertahan terhadap beban kerja yang tinggi tersebut, maka diperlukan pengujian beban pada website atau biasa disebut load testing.

Untuk memperdalam dan mengisi celah pemahaman, penelitian ini memiliki kebaruan yang terletak pada eksplorasi aspek perbandingan performa yang lebih beragam dan perbandingan daya tahan terhadap beban tinggi antara Progressive Web Apps dan Website non-PWA dengan karakteristik website yang identik secara fungsi dan fitur. Tujuannya untuk memahami bagaimana perbedaan performa dan daya tahan terhadap beban tinggi antara keduanya dalam situasi yang sama.

Oleh sebab itu, penelitian ini diharapkan dapat mengisi celah pengetahuan dengan memberikan bukti empiris yang kuat dalam konteks perbandingan performa dan daya tahan terhadap beban tinggi antara Progressive Web Apps dan website non-PWA yang identik secara fitur dan fungsi. Selain itu, hasil penelitian ini diharapkan dapat menjadi wawasan yang mendalam dan sebagai referensi bahan pertimbangan bagi para pengembang software, organisasi, perusahaan, akademisi dan peneliti terkait dengan pemilihan teknologi web yang disesuaikan dengan kebutuhan mereka.

## II. TINJAUAN PUSTAKA

### A. Website

Website merupakan suatu media dengan sejumlah besar halaman yang saling berhubungan (hyperlink) dan kemampuan menyajikan informasi dalam bentuk teks, gambar, video, audio, animasi, atau kombinasi dari semuanya [7]. Aplikasi berbasis website biasanya dibangun dengan HTML (Hypertext Markup Language) dan dipercantik desainnya dengan menggunakan CSS (Cascading Style Sheet). Selain itu, agar website jauh

lebih interaktif menggunakan satu bahasa pemrograman di sisi client yaitu Java Script.

### B. Progressive Web Apps

Progressive Web Apps (PWA) adalah sebuah konsep yang memungkinkan aplikasi web memberikan pengalaman terbaik kepada pengguna dalam hal keandalan (reliability), kecepatan (speed), dan keterlibatan pengguna (user engagement). Hal ini dicapai dengan memanfaatkan fitur dan layanan yang didukung oleh perangkat native device di browser web modern [8]

Progressive Web Apps (PWA) adalah aplikasi yang dibuat menggunakan teknologi platform web, namun memberikan pengalaman pengguna (user experience) seperti layaknya aplikasi native mobile. Layaknya sebuah website, PWA dapat berjalan di berbagai platform dan perangkat, seperti desktop, ponsel cerdas, tablet, dan lain sebagainya. Mirip dengan aplikasi mobile native, aplikasi yang dibuat dengan PWA dapat diinstal pada perangkat yang digunakan, dijalankan secara offline, dapat melakukan push notification, memiliki ikon pada homescreen, serta terintegrasi dengan perangkat dan aplikasi lain yang diinstal pada perangkat tersebut.[9]

### C. Pengujian Website

Pengujian website termasuk ke dalam pengujian perangkat lunak. Pengujian perangkat lunak adalah proses mengevaluasi dan memverifikasi apakah produk perangkat lunak atau aplikasi berfungsi seperti yang diharapkan [10]. Pengujian perangkat lunak mengidentifikasi kesalahan dan masalah dalam proses pengembangan dan memperbaikinya sebelum produk diluncurkan. Pendekatan ini memastikan bahwa hanya produk berkualitas tinggi yang sampai ke konsumen, sehingga meningkatkan kepuasan dan kepercayaan pelanggan [11].

### D. Performa Website

Performa website mengacu pada seberapa cepat website dimuat di browser. Ini juga menentukan kualitas kegunaan, interaktivitas, dan keandalan dari website tersebut. Performa website sering kali menjadi evaluasi pertama suatu situs web oleh pengunjung. Saat situs website dimuat dengan cepat, pengunjung website dapat menjelajahi website dengan lebih cepat, sehingga meningkatkan pengalaman pengguna mereka atau meningkatkan user experience. Website yang berkinerja baik membantu dalam berkomunikasi dan membangun citra merek, yang pada akhirnya menguntungkan pemilik website tersebut. Hal ini juga dapat meningkatkan peringkat website di halaman hasil mesin pencari atau bisa dikatakan meningkatkan SEO dari website tersebut [4]

Terdapat beberapa faktor yang dapat mempengaruhi performa website. Faktor yang mempengaruhi performa situs web biasanya adalah ukuran gambar dan file yang sangat besar, server atau hosting yang digunakan, serta penulisan kode yang buruk mulai dari kode Javascript, HTML yang tidak baik, atau CSS yang tidak terkompresi.

Selain itu, biasanya tidak memanfaatkan browser caching untuk me-caching data yang sering digunakan dan terlalu banyaknya jumlah request HTTP yang dilakukan [12].

Perlunya mengetahui aspek-aspek pengujian performa website sebagai acuan perhitungan yang nantinya digunakan dalam penelitian. Aspek-aspek pengujian performa antara lain:

1. Pingdom Tools: Page size, load time, dan request.
2. Lighthouse: Performance, accessibility, best practice, dan SEO.
3. Chrome DevTools: Memory resource.
4. GTmetrix: First contentful paint, Time to interactive, Total blocking time, Cumulative layout shift, dan Speed index.

#### E. Pengujian Beban Tinggi (Load Testing)

Pengujian beban atau load testing adalah pengujian performa yang dapat mengukur respon sistem dalam kondisi beban yang berbeda (Intan Permatasari & Santoso, 2019). Pengujian beban dilakukan untuk melihat bagaimana aplikasi dapat mengatasi masalah dan pengujian beban yang disesuaikan dengan kondisi dan situasi sebenarnya [6].

Dalam melakukan pengujian beban atau load testing menggunakan tools Apache JMeter. Apache JMeter merupakan software open source yang dibangun menggunakan 100% bahasa Java dan dirancang untuk 39 memuat perilaku fungsional pengujian dan untuk mengukur kinerja. Pada awalnya aplikasi ini dirancang untuk pengujian aplikasi berbasis website tetapi telah diperluas ke fungsi pengujian lainnya [13]. Berikut aspek-aspek pengujian beban antara lain:

1. Response Time.
2. Throughput.
3. Error Rate.

#### F. Metode Eksperimental Design

Metode Eksperimen merupakan penelitian kuantitatif murni, karena semua prinsip dan kaidah-kaidah penelitian kuantitatif berlaku pada metode ini. Penelitian eksperimental termasuk ke dalam penelitian laboratorium, meskipun penelitian ini dapat dilakukan di luar laboratorium, pelaksanaannya menerapkan prinsip-prinsip penelitian laboratorium, terutama dalam melakukan pengontrolan terhadap hal-hal yang mempengaruhi proses percobaan. Metode ini menguji pengaruh satu atau lebih variabel terhadap variabel lainnya. Variabel yang mempengaruhi disebut sebagai variabel bebas (independen) sedangkan variabel yang dipengaruhi disebut sebagai variabel terikat (dependen) [14].

true experimental desain bertujuan untuk mengeksplorasi kemungkinan keterkaitan sebab-akibat dengan cara menerapkan perlakuan dan membandingkan hasilnya dengan kelompok kontrol yang tidak mendapatkan perlakuan. Pengambilan sampel menggunakan desain True Experimental baik kelompok eksperimen maupun kelompok kontrol harus diambil sampelnya secara acak dari populasi. Peneliti memiliki

kendali atas semua variabel eksternal yang mempengaruhi jalannya percobaan. Oleh karena itu, validitas internal (kualitas pelaksanaan desain penelitian) menjadi tinggi [15].

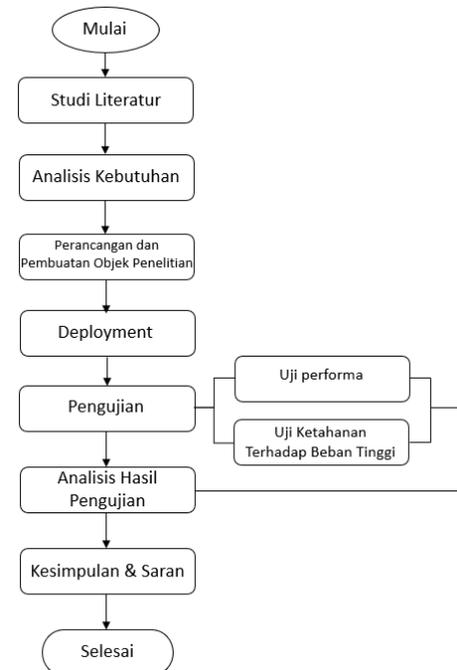
#### G. Hipotesis

Hipotesis berasal dari dua kata yaitu hypo yang artinya sebelum atau di bawah dan thesis yang artinya pernyataan, pendapat, dan kebenaran. Jadi arti hipotesis adalah sebuah pernyataan yang pada waktu diungkapkan belum terbukti kebenarannya atau bisa diartikan sebagai jawaban sementara terhadap rumusan masalah penelitian [16].

Dari pengertian hipotesis di atas, dapat dipahami penggunaan hipotesis dalam penelitian, hipotesis digunakan sebagai jawaban sementara terhadap hasil penelitian yang akan dilaksanakan. Adapun hipotesis dalam penelitian ini sebagai berikut:

1. Performa Aplikasi.
  - a) H0: Tidak ada perbedaan signifikan dalam performa antara Progressive Web Apps dan website non-PWA.
  - b) H1: Terdapat perbedaan signifikan dalam performa antara Progressive Web Apps dan website non-PWA.
2. Daya Tahan Terhadap Beban Tinggi.
  - a) H0: Tidak ada perbedaan signifikan dalam hasil uji beban tinggi antara Progressive Web Apps dan website non-PWA.
  - b) H1: Terdapat perbedaan signifikan dalam hasil uji beban tinggi antara Progressive Web Apps dan website non-PWA.

### III. METODOLOGI PENELITIAN



Gambar. 1 Metodologi Penelitian

### A. Jenis Penelitian

Jenis penelitian yang digunakan adalah penelitian eksperimen yang tergolong penelitian kuantitatif. Pemilihan jenis penelitian didasari pada tujuan utama penelitian, yaitu untuk membandingkan performa dan daya tahan terhadap beban tinggi antara PWA dan website non-PWA. Dalam penelitian ini menggunakan bentuk true eksperimental design, bentuk tersebut dipilih karena memungkinkan peneliti untuk mengontrol semua variabel luar yang dapat mempengaruhi jalannya eksperimen. Mengingat fokus penelitian pada perbandingan performa dan daya tahan terhadap beban tinggi antara Progressive Web Apps (PWA) dan website non-PWA, kontrol penuh terhadap faktor-faktor eksternal penting dilakukan. Hal ini untuk memastikan bahwa perbedaan hanya pada penerapan teknologi antara kedua jenis website bukan pada faktor-faktor lainnya. Dengan demikian, peneliti dapat memastikan bahwa hasil penelitian tidak dipengaruhi oleh faktor-faktor eksternal yang tidak relevan dengan tujuan penelitian, sehingga meningkatkan validitas internalnya.

Hal ini sesuai dengan pendapat Sugiyono yang menjelaskan bahwa metode eksperimen cocok digunakan untuk mencari pengaruh perlakuan tertentu terhadap yang lain dalam kondisi yang terkendali [15]. Dalam konteks penelitian ini, mencari perlakuan yang diteliti antara penggunaan teknologi PWA dan website non-PWA, sementara variabel yang diamati adalah performa dan daya tahan terhadap beban tinggi dari kedua jenis website tersebut.

Oleh sebab itu, dengan menggunakan metode eksperimen memungkinkan penelitian ini dapat dikendalikan dengan baik, seperti membuat dua website yang identik kecuali dalam penerapan teknologinya (PWA dan website non-PWA) dan mengujinya dalam lingkungan yang seragam dan terkondisikan.

### B. Studi Literatur

Tahap pertama dalam penelitian ini yaitu mengumpulkan dan mempelajari literatur-literatur yang berkaitan dengan topik penelitian ini. Studi literatur diperlukan untuk mengumpulkan dan mempelajari mengenai perbandingan performa dan daya tahan terhadap beban tinggi antara progressive web apps dan website non-PWA dengan metode eksperimen. Literatur-literatur yang sudah dipelajari diharapkan dapat membantu dalam proses penelitian kedepannya. Adapun pustaka studi literatur yang dipelajari terakait dengan topik penelitian ini sebagai berikut:

1. Website
2. Progressive Web Apps
3. Pengujian Website
4. Performa Website
5. Pengujian Beban Tinggi (Load Testing)
6. Metode Eksperimen

### C. Analisis Kebutuhan

Analisis kebutuhan perlu dilakukan untuk menyiapkan hal-hal yang dibutuhkan dalam melakukan penelitian. Ini dilakukan agar penelitian berjalan dengan lancar dan sesuai dengan perencanaan. Dalam penelitian ini, analisis kebutuhan fungsional dan non-fungsional dari website forum diskusi yang akan dijadikan objek penelitian.

#### 1. Analisis Kebutuhan Fungsional:

- a) Autentikasi pengguna: Sistem memungkinkan pengguna melakukan login dan logout dari platform forum diskusi.
- b) Manajemen Konten: Sistem memungkinkan untuk menambah, mengubah, mengedit, dan menghapus diskusi, jawaban diskusi pada forum diskusi.
- c) Responsif dan Interaktif: Tampilan pengguna yang responsive di berbagai perangkat.

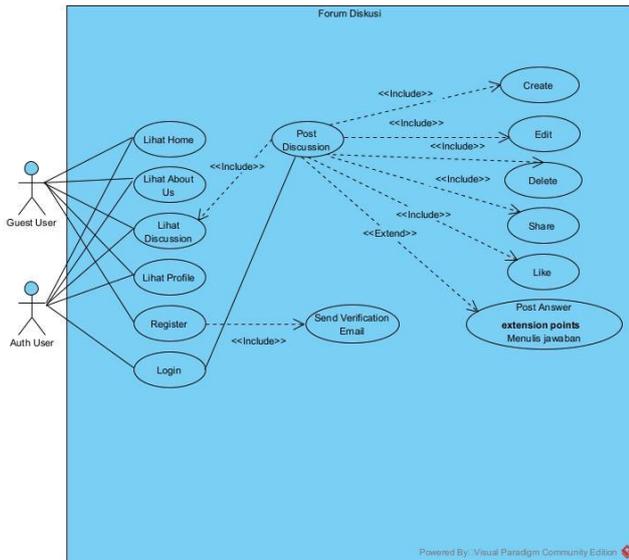
#### 2. Analisis Kebutuhan Non Fungsional:

- a) Sistem dapat diakses dengan baik
- b) Sistem mudah digunakan oleh pengguna
- c) Sistem responsive dan antarmuka pengguna yang cepat dan mulus tanpa hambatan.

### D. Perancangan dan Pembuatan Objek Penelitian.

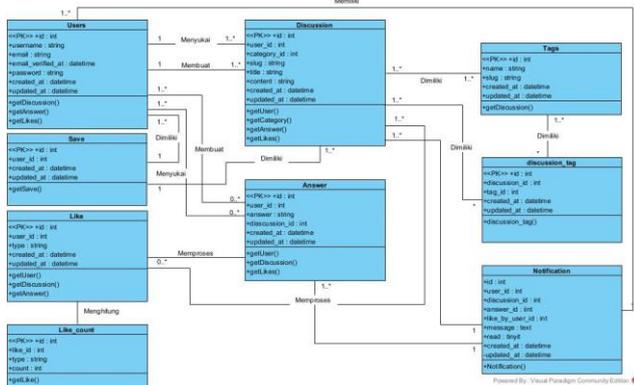
Perancangan dan pembuatan objek penelitian dilakukan dengan membuat dua website yang identik, satu sebagai representasi situs web biasa dan satu lainnya sebagai progressive web apps. Website yang dibuat merupakan website forum diskusi yang keduanya nanti akan di hosting di tempat yang sama. Proses desainnya melibatkan pemilihan struktur halaman, asset dan resource yang sama antara keduanya. Dalam perancangannya benar-benar yang membedakan hanya pada penerapan teknologinya saja, aspek-aspek lainnya akan disamakan antar progressive web apps dengan website non-PWA. Selain itu, nanti akan dipastikan bahwa progressive web apps memenuhi standar dan pedoman yang berlaku. Berikut di bawah ini merupakan diagram sistem dan antarmuka dari website forum diskusi:

#### 1. Use Case Diagram:



Gambar. 2 Use Case Diagram.

2. Class Diagram:



Gambar. 3 Class Diagram.

E. Deployment

Pada tahap deployment ini merupakan tahap di mana setelah melakukan perancangan dan pembuatan objek penelitian berupa dua buah website yakni PWA dan website non-PWA selanjutnya dua website tersebut akan di deploy ke server yang sama sebelum dilakukan pengujian performa dan beban tinggi.

F. Pengujian

Pada tahap ini yaitu melakukan pengujian performa dan daya tahan terhadap beban tinggi dari kedua jenis aplikasi baik PWA maupun website non-PWA. Pengukuran akan dilakukan sesuai dengan parameter-parameter yang telah ditentukan sebelumnya.

1. Variabel Pengujian
2. Skenario Pengujian

G. Analisis Hasil Pengujian

Analisis hasil pengujian akan melibatkan perbandingan antara performa dan hasil load testing pada kedua kelompok. Data yang terkumpul akan

dianalisis secara rinci untuk mengevaluasi sejauh mana PWA dapat meningkatkan performa dan daya tahan terhadap beban tinggi dibandingkan dengan website non-pwa. Sumber data merupakan data primer yaitu hasil yang didapat langsung dari pengujian kedua jenis website. Metode pengumpulan data dilakukan secara observasi pengamatan kedua website secara langsung selama pengujian dilakukan.

1. Instrumen Pengumpulan Data

TABEL I

Instrumen Pengumpulan Data Performa

No.	Aspek Pengujian	GTmetrix	Lighthouse	Chrome DevTools	Pingdom Tools
1	Performance		✓		
2	Load Time				✓
3	Page Size				✓
4	Request				✓
5	Best Practice Score		✓		
6	Accessibility		✓		
7	SEO Score		✓		
8	First Contentful Paint	✓			
9	Time To Interactive	✓			
10	Total Blocking Time	✓			
11	Cumulative Layout Shift	✓			
12	Speed Index	✓			
13	Memory Usage			✓	

TABEL II

Instrumen Pengumpulan Data Load Testing

No	Aspek Pengujian	Apache JMeter
1	Response Time	✓
2	Concurrent User	✓
3	Throughput	✓
4	Error Rate	✓

2. Teknik Pengumpulan Data

Dalam tahap ini pengumpulan data performa dan daya tahan terhadap beban tinggi antara PWA dan website non-PWA dikumpulkan melalui observasi saat pengujian performa dan daya tahan terhadap beban tinggi kedua jenis website dilakukan. Peneliti tidak terlibat langsung pada aspek yang diteliti yaitu observasi non partisipan dimana peneliti hanya melakukan pengamatan terhadap kedua jenis website (PWA dan website non-PWA) dengan menggunakan tools seperti GTmetrix, lighthouse, chrome devtools, Pingdom tools, dan apache JMeter. Dengan memasukkan link kedua website (PWA dan non-PWA) pada tools tersebut. Durasi waktu yang diperlukan dalam melakukan observasi ini bervariasi tergantung pada kompleksitas pengujian dan jumlah sampel yang diobservasi.

3. Hasil Pengujian

Dalam tahap ini akan dijabarkan data-data hasil pengujian performa dan load testing antara progressive web apps dengan website non-PWA. Hasil pengujian akan dijabarkan sesuai dengan aspek-aspek

pengujian baik performa maupun daya tahan terhadap beban tinggi.

#### 4. Analisis Hasil Pengujian

Dalam tahap ini, setelah data-data tadi dijabarkan kemudian dianalisis. Teknik analisis data yang digunakan dalam penelitian ini adalah analisis deskriptif dan analisis komparatif. Analisis deskriptif digunakan untuk merangkum dan mendeskripsikan data yang didapatkan dari hasil pengujian, seperti statistik deskriptif, grafik, dan tabel. Sementara analisis komparatif digunakan untuk membandingkan hasil antara kelompok yang berbeda (PWA dan website non-PWA) dalam hal performa dan daya tahan terhadap beban tinggi sesuai dengan aspek-aspek pengujiannya masing-masing.

### H. Kesimpulan dan Saran

Kesimpulan dari penelitian ini merangkum temuan berdasarkan analisis hasil pengujian. Dalam penelitian ini, dilakukan evaluasi sejauh mana implementasi PWA menguntungkan performa dan daya tahan terhadap beban tinggi.

## IV. HASIL DAN PEMBAHASAN

Pada tahap ini akan membahas langkah-langkah dalam mengembangkan dan melakukan pengujian performa serta daya tahan terhadap beban tinggi antara PWA dan non-PWA. Di mulai dengan analisis kebutuhan, rancangan objek penelitian, deployment, pengujian, hingga analisis hasil pengujian.

### A. Analisis Kebutuhan

#### 1. Perangkat Lunak

TABEL III  
Perangkat Lunak Untuk Pengembangan.

No	Perangkat Lunak	Keterangan
1	IDE (Integrated Development Environment)	Visual Studio Code 1.90.0 (64 Bit)
2	Web Server Lokal	Laragon 6.0 220916 (64 Bit)
3	Browser	Google Chrome 125.0.0622.142 (64 Bit)
4.	Framework	Laravel 10, Vite
5.	Bahasa Pemrograman	PHP 8.10, JavaScript
6	Sistem Operasi	Windows 11

TABEL IV  
Perangkat Lunak Untuk Pengujian.

No	Perangkat Lunak	Versi
1	Browser	Google Chrome 125.0.0622.142 (64 Bit)
2	Tools	Apache jMeter 5.6.3 (64 Bit)

#### 2. Perangkat Keras

TABEL IV  
Perangkat Lunak Untuk Pengujian.

No	Komponen	Spesifikasi
1	CPU	Intel(R) Core (TM) i5-10500H CPU @ 2.50GHz.
2	RAM	16.0 GB.
3	Penyimpanan	500GB (SSD NVMe) + 256GB (SSD SATA).
4	Koneksi Jaringan Internet	FirstMedia (30 Mbps).

### B. Rancangan Objek Penelitian

#### 1. Pengembangan Dua Website Forum Diskusi

Pada pengembangan website forum diskusi, menggunakan beberapa teknologi pengembangan website seperti HTML, CSS, Javascript, dan PHP. Penulisan logika bisnis dari website forum diskusi pemrograman yang dibangun menggunakan bahasa pemrograman PHP dengan menggunakan framework Laravel. Framework Laravel dipilih karena sudah menyediakan fitur-fitur yang kuat untuk mempercepat pengembangan seperti routing, session manajemen, validasi, autentikasi, otorisasi, blade templating engine, ORM (Object-Relational Mapping), dan memiliki komunitas yang besar. Dalam website forum diskusi pemrograman menggunakan menggunakan RDBMS (Relational Database Manajemen System) untuk menyimpan dan mengelola data. RDBMS yang digunakan yaitu MySQL. MySQL memberikan kemudahan dalam mengelola basis data relasional dan memiliki performa berkecepatan tinggi, memiliki cache memori khusus, dan keamanan data yang baik. Security yang diterapkan dalam website forum diskusi dijaga dengan menggunakan praktik keamanan pada setiap tingkatan seperti sanitasi input pengguna untuk mencegah serangan injeksi, enkripsi data sensitif, hashing password, dan pengelolaan autentikasi dan otorisasi yang tepat dengan mekanisme yang sudah disediakan oleh framework Laravel.

Proses pengembangan yang dilakukan pertama kali yaitu mengembangkan website non-PWA terlebih dahulu. Setelah website forum diskusi non-PWA selesai dibuat, selanjutnya dilakukan implementasi PWA pada website forum diskusi pemrograman. Implementasi PWA dilakukan dengan menggunakan package dari laravel-pwa.

#### 2. Fitur-fitur

Fitur-fitur yang dikembangkan untuk kedua jenis website antara lain:

- a) Landing page.
- b) Login dan register
- c) Email verification register
- d) Mengajukan diskusi atau pertanyaan
- e) Melihat detail diskusi
- f) Mengedit, menghapus, dan share diskusi
- g) Memberikan jawaban
- h) Mengedit, menghapus, dan share jawaban
- i) Like/Unlike diskusi dan jawaban
- j) Melihat dan mengedit profile
- k) Bookmark untuk menyimpan diskusi
- l) Searching bar untuk mencari diskusi berdasarkan judul, isi, dan tag
- m) Filter data berdasarkan tag, paling banyak dilihat, paling banyak disukai,

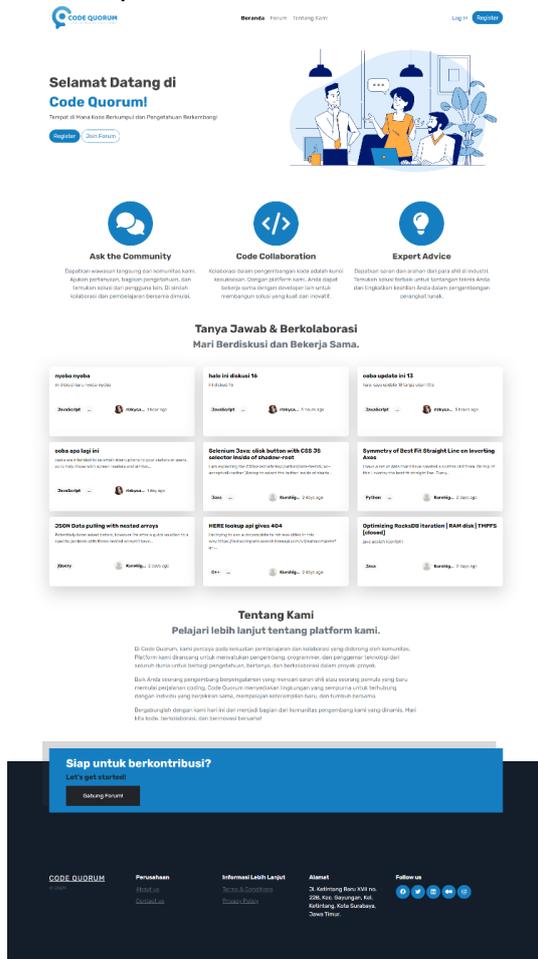
- paling banyak dijawab, diskusi terbaru dan diskusi terlama
- n) Notifikasi jika diskusi disukai atau dijawab oleh user lain

### 3. Perbandingan Kedua Rancangan Website

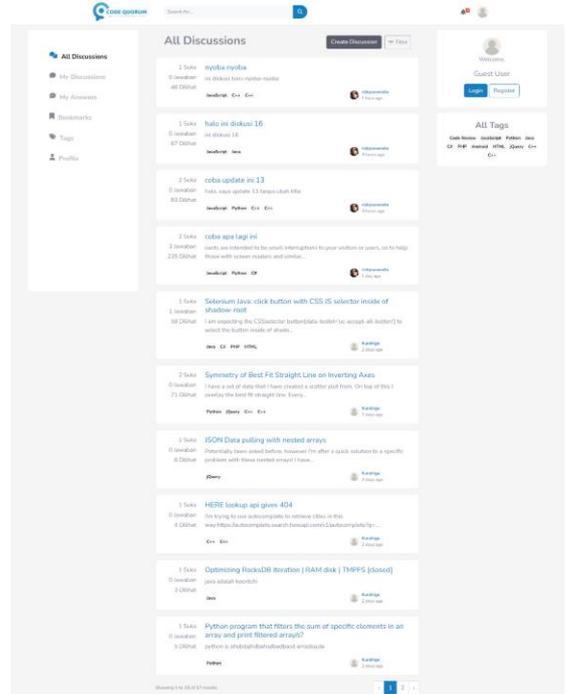
**TABEL V**  
Perbandingan Rancangan Aplikasi.

No	Aspek	Webiste PWA	Website non-PWA
1	Offline Access	Ya	Tidak
2	Caching	Ya	Tidak
3	Responsive Design	Ya	Ya
4	Installable	Ya	Tidak

### 4. Hasil Tampilan Antarmuka



Gambar. 4 Landing page.



Gambar. 5 Discussions page.

### C. Deployment

Proses deployment untuk kedua jenis website di deploy di Virtual Private Server (VPS) dengan spesifikasi yang sama antara keduanya. Dalam penelitian ini, menggunakan layanan Cloud VPS (Virtual Private Server) dari idCloudHost. Untuk rincian spesifikasi dari VPS dapat dilihat pada berikut ini:

**TABEL VI**  
Spesifikasi Server Kedua Aplikasi.

No.	VPS	Spesifikasi	PWA	Non-PWA
1	Id CloudHost	Ubuntu 24.04 (Sistem Operasi)	Ya	Ya
2		2 CPU	Ya	Ya
3		RAM 2GB	Ya	Ya
4		20 GB Disk (SSD)	Ya	Ya
5		Nginx 1.24.0 (Web Server)	Ya	Ya

### D. Pengujian

Tahap pengujian dilakukan dibagi menjadi dua bagian, yaitu pengujian performa dan pengujian daya tahan terhadap beban tinggi antara website forum dikusi pemrograman PWA dan non-PWA.

#### 1. Black Box Testing

**TABEL VII**  
Hasil Pengujian Black Box

No	Fitur	Deskripsi Uji	Hasil Yang Diharapkan	Status Uji (Pass/Fail)
1	Landing page	- Memastikan halaman muncul tanpa kesalahan	Halaman utama tampil tanpa error	Pass
2	Login dan register	-Coba login dengan akun yang valid -Coba register dengan email yang valid	User dapat login dan register dengan sukses	Pass
3	Mengajukan diskusi atau pertanyaan	-Ajukan diskusi baru dengan judul, isi, dan tag yang valid	Diskusi tersimpan dengan benar dan tampil di halaman utama	Pass

4	Melihat detail diskusi	-Buka detail diskusi dan pastikan informasi diskusi tampil lengkap	Halaman detail diskusi menampilkan semua informasi yang relevan	Pass
5	Mengedit, menghapus, dan share diskusi	- Edit diskusi yang sudah ada -Hapus diskusi yang sudah ada -Bagikan diskusi	erubahan (jika ada) disimpan dengan benar, diskusi dihapus dengan sukses, dan sharing berfungsi dengan baik	Pass
6	Memberikan jawaban	-Beri jawaban pada diskusi yang ada dengan teks yang valid	Jawaban tersimpan dan terlihat pada halaman diskusi	Pass
7	Mengedit, menghapus, dan share jawaban	- Edit jawaban yang sudah ada - Hapus jawaban yang sudah ada - Bagikan jawaban ke media sosial	Perubahan (jika ada) disimpan dengan benar, jawaban dihapus dengan sukses, dan sharing berfungsi dengan baik	Pass
8	Like/Unlike diskusi dan jawaban	- Klik like dan unlike pada diskusi dan jawaban	Jumlah like/unlike berubah sesuai dengan aksi yang dilakukan	Pass
9	Melihat dan mengedit profile	- Buka halaman profile pengguna - Edit informasi profil	Informasi profil pengguna diperbarui dengan benar	Pass
10	Bookmark untuk menyimpan diskusi	- Tandai diskusi untuk bookmark - Buka daftar bookmark dan lihat diskusi yang disimpan	Diskusi berhasil ditandai dan bisa dilihat kembali dalam daftar bookmark	Pass
11	Searching bar untuk mencari diskusi	- Cari diskusi berdasarkan judul, isi, dan tag yang valid	Hasil pencarian menampilkan diskusi sesuai dengan kata kunci	Pass
12	Filter data berdasarkan tag, dll.	- Gunakan filter untuk memilih diskusi berdasarkan tag, jumlah views, jumlah likes, jumlah jawaban, diskusi terbaru, dan terlama	Diskusi ditampilkan sesuai dengan filter yang dipilih	Pass
13	Notifikasi jika diskusi disukai atau dijawab	- Dapatkan notifikasi jika diskusi disukai atau dijawab	Notifikasi muncul secara tepat waktu dan sesuai dengan tindakan yang dilakukan	Pass

## 2. Skenario Pengujian Performa dan Beban Tinggi



Gambar. 6 Skenario Pengujian Performa.

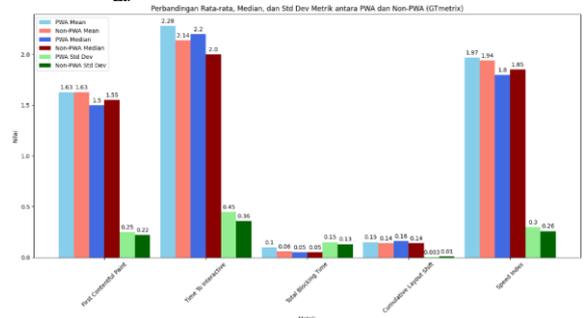
TABEL VIII  
Skenario Pengujian Beban Tinggi

Skenario 1	Number of User	10
	Ramp-up Period	10
	Loop Count	20
Skenario 2	Number of User	20
	Ramp-up Period	20
	Loop Count	30
Skenario 3	Number of User	30
	Ramp-up Period	30
	Loop Count	40
Skenario 4	Number of User	50
	Ramp-up Period	100
	Loop Count	60
Skenario 5	Number of User	100
	Ramp-up Period	200
	Loop Count	80

## E. Hasil dan Analisis Perbandingan Pengujian

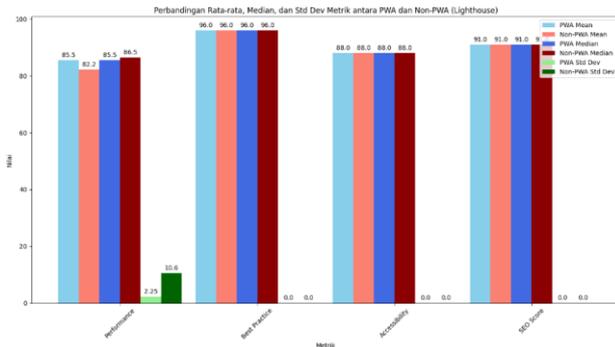
Pada bagian ini akan memaparkan hasil pengujian dari kedua jenis website anatar PWA dan non-PWA baik pengujian performa maupun beban tinggi. Bagian performa akan dipecah menjadi bagian perhalaman untuk menjelaskan hasil data yang diperoleh dari berbagai tools pengujian. Selain itu, juga akan dilakukan analisis deskriptif dan komparatif dari hasil uji yang sudah dilakukan.

### 1. Hasil Pengujian Performa



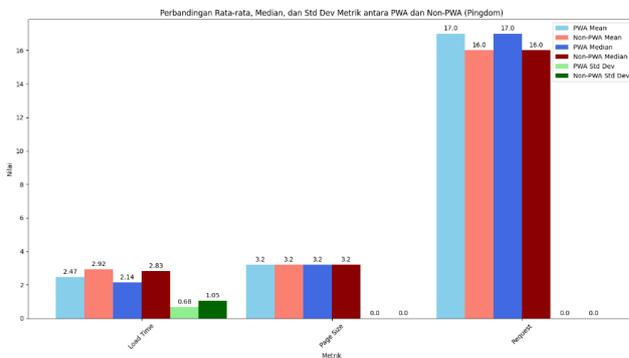
Gambar. 7 Perbandingan Performa dengan GTMetric

Berdasarkan grafik pada gambar 7 menunjukkan hasil pengujian menggunakan GTMetric. Perbandingan antara PWA dan non-PWA menunjukkan bahwa rata-rata metrik performa seperti First Contentful Paint, Time To Interactive, Total Blocking Time, Cumulative Layout Shift, dan Speed Index hampir serupa antara keduanya. Meskipun terdapat perbedaan kecil dalam median dan standar deviasi, tidak terlihat perbedaan yang signifikan antara aplikasi PWA dan non-PWA dalam hal performa pada halaman home berdasarkan metrik yang diukur.



Gambar. 8 Perbandingan Performa dengan Lighthouse

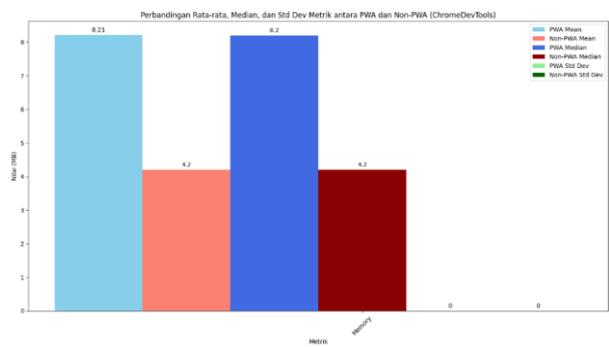
Berdasarkan gambar 8 di atas menunjukkan hasil uji di yang menggunakan Lighthouse. Aplikasi PWA menunjukkan performa yang sedikit lebih baik dibandingkan dengan aplikasi non-PWA dalam hal performa, dengan rata-rata skor performa sebesar 85.5 untuk PWA dan 82.2 untuk non-PWA. Namun, perbedaan ini tidak terlalu signifikan karena median keduanya cukup serupa, yaitu 85.5 untuk PWA dan 86.5 untuk non-PWA. Standar deviasi yang lebih tinggi pada aplikasi non-PWA menunjukkan adanya variasi yang lebih besar dalam performa halaman. Selain itu, baik aplikasi PWA maupun non-PWA memiliki skor yang sama untuk metrik Best Practice, Accessibility, dan SEO, yaitu 96 tanpa deviasi, menunjukkan bahwa keduanya memiliki nilai best practice, aksesibilitas, dan optimalisasi search engine yang baik.



Gambar. 9 Perbandingan Performa dengan Pingdom

Berdasarkan gambar 9 di atas menunjukkan hasil uji menggunakan Pingdom. Aplikasi PWA menunjukkan performa yang sedikit lebih baik dibandingkan dengan aplikasi non-PWA dalam hal waktu muat (load time), dengan rata-rata load time sebesar 2.47 detik untuk PWA dan 2.92 detik untuk non-PWA. Meskipun rata-rata load time aplikasi PWA lebih rendah, median keduanya cukup serupa, yaitu 2.14 detik untuk PWA dan 2.83 detik untuk non-PWA.

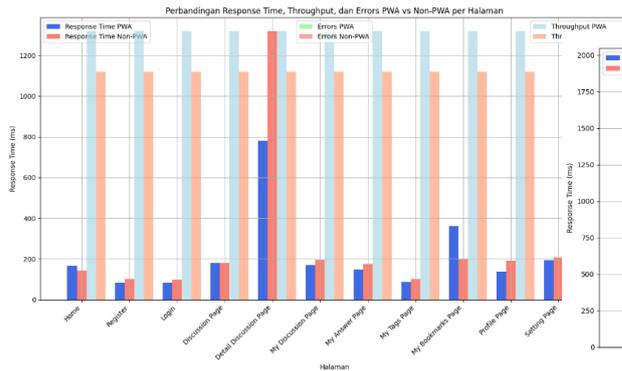
Standar deviasi yang lebih tinggi pada aplikasi non-PWA menunjukkan adanya variasi yang lebih besar dalam waktu muat halaman. Selain itu, baik aplikasi PWA maupun non-PWA memiliki ukuran halaman (page size) dan jumlah permintaan (requests) yang sama, yaitu 3.20 MB dan 17 request, menunjukkan bahwa keduanya memiliki karakteristik yang serupa dalam hal ukuran dan jumlah elemen yang dimuat.



Gambar. 10 Perbandingan Performa dengan Chrome DevTools

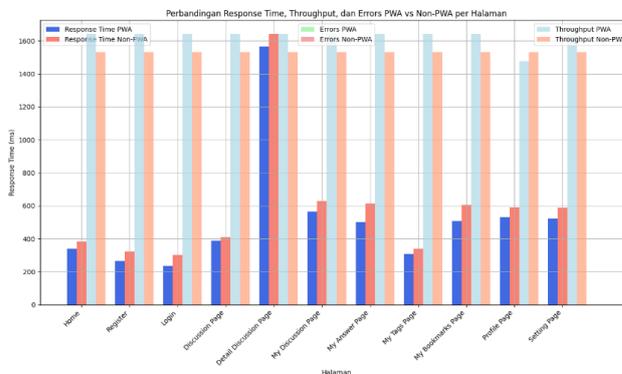
Berdasarkan gambar 10 di atas menunjukkan hasil pengujian menggunakan Chrome DevTools untuk halaman home, terlihat bahwa aplikasi PWA memiliki penggunaan memori yang lebih tinggi dibandingkan dengan aplikasi non-PWA. Rata-rata penggunaan memori pada aplikasi PWA adalah 8.21 MB, sementara pada aplikasi non-PWA hanya 4.20 MB. Median untuk kedua aplikasi adalah 8.20 MB untuk PWA dan 4.20 MB untuk non-PWA, tanpa adanya deviasi pada nilai tersebut. Hal ini menunjukkan bahwa secara konsisten, aplikasi PWA memiliki kebutuhan memori yang lebih besar daripada aplikasi non-PWA, yang dapat mempengaruhi kinerja dan pengalaman pengguna secara keseluruhan.

## 2. Hasil Pengujian Beban Tinggi (Load testing)



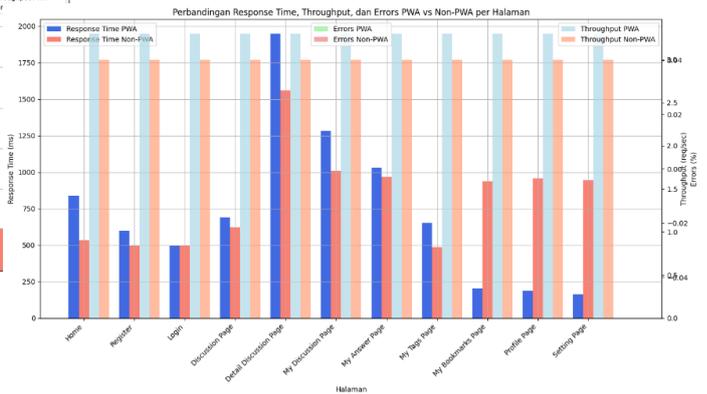
Gambar. 11 Perbandingan Load Testing Skenario 1

Berdasarkan gambar 11 di atas menunjukkan bahwa PWA (Progressive Web App) secara konsisten menunjukkan waktu respons lebih cepat dibandingkan non-PWA pada sebagian besar halaman seperti Home, Register, Login, My Discussion Page, My Answer Page, My Tags Page, dan Profile Page. Meskipun pada halaman My Bookmarks Page non-PWA memiliki waktu respons lebih baik, PWA tetap menunjukkan throughput yang lebih tinggi secara umum (3.3 per detik dibanding 2.7-2.8 per detik). Kedua jenis aplikasi memiliki tingkat kesalahan yang sama rendahnya yakni 0.00%. Secara keseluruhan, PWA lebih unggul karena performa lebih baik dalam waktu respon dan throughput.



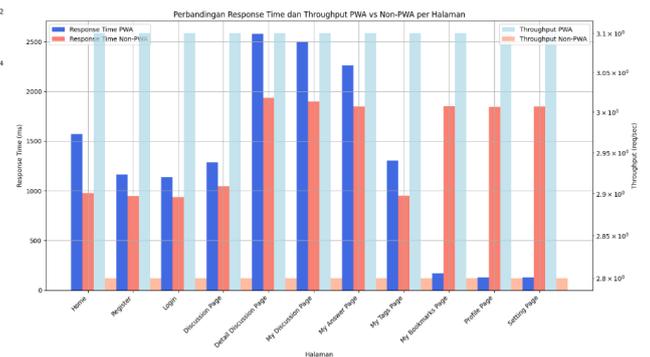
Gambar. 12 Perbandingan Load Testing Skenario 2

Berdasarkan gambar 12 di atas menunjukkan bahwa waktu respons rata-rata PWA (Progressive Web App) lebih rendah dibandingkan dengan non-PWA pada sebagian besar halaman yang diuji. Hal ini terlihat pada halaman seperti Home, Register, Login, Discussion Page, My Discussion Page, My Answer Page, My Tags Page, My Bookmarks Page, Profile Page, dan Setting Page. PWA juga konsisten dalam throughput yang lebih tinggi (3.0 per detik) dibandingkan non-PWA (2.7-2.8 per detik) di sebagian besar halaman. Tidak ada error yang tercatat untuk kedua jenis aplikasi. Secara keseluruhan, PWA menunjukkan performa yang lebih baik dalam respons waktu dan throughput dibandingkan dengan non-PWA.



Gambar. 13 Perbandingan Load Testing Skenario 3

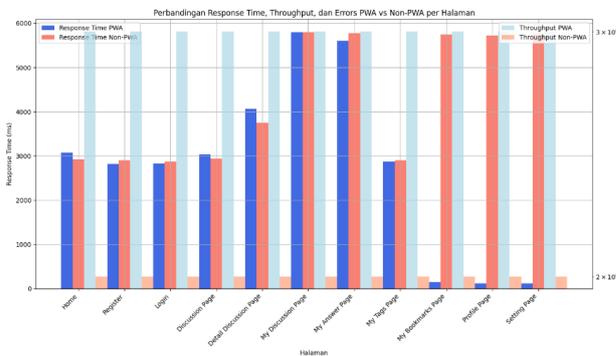
Berdasarkan gambar 13 di atas menunjukkan bahwa waktu respons rata-rata PWA (Progressive Web App) lebih lambat dibandingkan non-PWA pada hampir semua halaman yang diuji, seperti Home, Register, Login, Discussion Page, dan lainnya. Akan tetapi, keduanya memiliki throughput yang sama (3.3 per detik), hal ini menunjukkan kemampuan yang seimbang dalam menangani jumlah permintaan. Tidak ada error yang tercatat untuk kedua jenis aplikasi. Secara keseluruhan, dari segi waktu respons, non-PWA lebih baik dalam skenario ini. Perbedaan antara keduanya cukup jauh dalam hal waktu respons, meskipun throughput keduanya sama.



Gambar. 13 Perbandingan Load Testing Skenario 4

Berdasarkan gambar 13, non-PWA secara konsisten menunjukkan waktu respons rata-rata yang lebih baik dibandingkan PWA pada semua halaman yang diuji. Perbedaan waktu responsnya cukup signifikan, terlihat dari halaman Home, Register, Login, Discussion Page, Detail

Discussion Page, My Discussion Page, My Answer Page, My Tags Page, My Bookmarks Page, Profile Page, dan Setting Page. Meskipun throughput keduanya sama (3.1 per detik) dan tidak ada error yang tercatat untuk keduanya, PWA tetap kalah dalam hal waktu respons. Hal ini menandakan bahwa dalam skenario ini, non-PWA lebih baik dalam hal waktu respons. Secara keseluruhan, perbedaan antara PWA dan non-PWA dalam hal waktu respons cukup jauh, dengan non-PWA menunjukkan performa yang lebih baik dalam skenario ini.



Gambar. 14 Perbandingan Load Testing Skenario 5

Berdasarkan gambar 15 terlihat bahwa baik PWA maupun non-PWA memiliki throughput yang sama (3.0 per detik) dan tidak ada error yang tercatat untuk kedua jenis aplikasi. Namun, perbedaan terbesar terlihat pada waktu respons rata-rata antara PWA dan non-PWA. PWA menunjukkan waktu respons rata-rata yang lebih tinggi dibandingkan non-PWA pada semua halaman yang diuji. Misalnya, pada halaman Home, Register, Login, Discussion Page, Detail Discussion Page, My Discussion Page, My Answer Page, My Tags Page, My Bookmarks Page, Profile Page, dan Setting Page, PWA secara konsisten memiliki waktu respons yang lebih lama dibandingkan non-PWA. Secara keseluruhan, non-PWA menunjukkan performa yang lebih baik

dalam hal waktu respons dibandingkan PWA dalam skenario ini.

### 3. Uji Statistik Mann-Whitney U

#### a) Performa

TABEL IX  
Hasil Uji Statistik Performa

No	Parameter	Asymp. Sig. (2-tailed)
1	First Contentful Paint	0.653
2	Time To Interactive	0.749
3	Total Blocking Time	0.644
4	Cumulative Layout Shift	1.000
5	Speed Index	0.949
6	Performance	0.180
7	Best Practice	0.949
8	Accessibility	0.949
9	SEO	1.000
10	Load Time	0.898
11	Page Size	1.000
12	Request	0.565
13	Memory	0.141

Berdasarkan tabel IX hasil menunjukkan uji statistik Mann-Whitney U, tidak ada perbedaan signifikan antara PWA dan non-PWA pada berbagai parameter performa. Parameter seperti First Contentful Paint, Time to Interactive, Total Blocking Time, Cumulative Layout Shift, Speed Index, dan Performance menunjukkan nilai p yang tidak signifikan, menandakan bahwa waktu respons dan interaksi halaman, stabilitas tata letak, kecepatan muat, dan performa keseluruhan adalah serupa antara PWA dan non-PWA alias tidak signifikan. Selain itu, parameter Best Practice, Accessibility, SEO, Load Time, Page Size, Request, dan Memory juga tidak menunjukkan perbedaan signifikan.

#### b) Load Testing

TABEL X  
Hasil Uji Statistik Load Testing

No	Parameter	Asymp. Sig. (2-tailed)
1	Response Time	0.069
2	Troughput	<0.001

Berdasarkan tabel X hasil uji statistik yang dilakukan, dapat disimpulkan bahwa terdapat perbedaan signifikan antara aplikasi Progressive Web App (PWA) dan non-PWA dalam beberapa

parameter. Untuk waktu respons, nilai signifikansi asimtotik (p-value) sebesar 0.069 menunjukkan bahwa tidak terdapat perbedaan yang signifikan antara kedua jenis aplikasi dalam hal waktu respons. Namun, untuk throughput, nilai p sangat rendah ( $<0.001$ ), mengindikasikan adanya perbedaan signifikan dalam kemampuan menangani beban antara PWA dan non-PWA. Hal ini menegaskan bahwa PWA mungkin lebih efisien dalam menangani volume trafik yang tinggi dibandingkan dengan aplikasi non-PWA.

## V. KESIMPULAN DAN SARAN

Berdasarkan penelitian yang telah dilaksanakan, dapat disimpulkan beberapa hal berikut ini:

1. Untuk menguji perbandingan antara Progressive Web App (PWA) dan website non-PWA dari segi performa dan daya tahan terhadap beban tinggi, dapat dilakukan dengan menggunakan metode dan skenario pengujian yang sudah ditentukan. Pertama, uji performa dapat dilakukan dengan menggunakan alat seperti Lighthouse, Pingdom, GTmetrix, atau ChromeDevTools untuk mengukur metrik seperti waktu muat halaman, waktu interaktif, total blocking time, dan lain-lain. Hasil dari pengujian ini akan memberikan gambaran tentang seberapa efisien dan cepat kinerja aplikasi PWA dan non-PWA dalam berbagai kondisi. Kedua, uji beban (load testing) dapat dilakukan untuk menguji kemampuan aplikasi dalam menangani beban tinggi. Ini melibatkan simulasi peningkatan jumlah pengguna atau permintaan ke server secara bersamaan untuk melihat bagaimana aplikasi PWA dan non-PWA menanggapi. Alat yang digunakan Apache JMeter yang berupa

Testing Tools yang dapat digunakan untuk tujuan ini. Dengan melakukan kedua jenis pengujian ini, mendapatkan pemahaman yang lebih komprehensif tentang kekuatan dan kelemahan masing-masing jenis aplikasi dalam konteks performa dan daya tahan terhadap beban tinggi.

2. Berdasarkan hasil analisis perbandingan antara Progressive Web Apps (PWA) dan non-PWA, ditemukan beberapa perbedaan dalam hal performa dan daya tahan terhadap beban tinggi. Dari segi performa, PWA menunjukkan keunggulan dalam beberapa metrik utama, namun perbedaannya tidak selalu signifikan. Pada pengujian Lighthouse, PWA memiliki skor kinerja (Performance) yang sedikit lebih tinggi dibandingkan non-PWA, dengan skor yang sama dalam Best Practice, Accessibility, dan SEO. Pengujian menggunakan Pingdom menunjukkan bahwa PWA memiliki waktu muat (Load Time) yang sedikit lebih lama dibandingkan non-PWA, meskipun perbedaannya tidak signifikan. Kedua jenis aplikasi memiliki ukuran halaman (Page Size) dan jumlah permintaan (Requests) yang sama. Pengujian menggunakan ChromeDevTools menunjukkan bahwa PWA menggunakan lebih banyak memori dibandingkan non-PWA, dan dalam pengujian GTmetrix, PWA memiliki waktu interaktif (Time to Interactive) yang sedikit lebih lama. Dari segi load testing, PWA umumnya menawarkan throughput yang lebih tinggi, menunjukkan kemampuan yang lebih baik dalam menangani beban tinggi. Namun, waktu respons rata-rata PWA lebih lambat dibandingkan non-PWA di hampir semua halaman yang diuji, seperti

Home, Register, Login, dan Discussion Page. Kedua jenis aplikasi menunjukkan keandalan yang sama tinggi dengan error rate nol. Kesimpulannya, meskipun PWA dapat menangani lebih banyak permintaan per detik dan memiliki beberapa keunggulan performa, perbedaannya dengan non-PWA tidak selalu signifikan. Non-PWA menawarkan waktu respons yang lebih cepat dan penggunaan memori yang lebih efisien. Penggunaan PWA berdampak positif pada daya tahan terhadap beban tinggi tetapi bisa mengorbankan kecepatan respons dibandingkan dengan website non-PWA. Oleh karena itu, pilihan antara PWA dan non-PWA sebaiknya didasarkan pada prioritas spesifik aplikasi, apakah lebih mementingkan throughput dan fitur canggih dari PWA atau waktu respons yang lebih cepat dan efisiensi memori dari non-PWA.

Dalam penelitian ini tentunya masih memiliki kekurangan yang perlu diatasi dan dilengkapi, sehingga diperlukan penelitian lanjutan untuk meningkatkan manfaat penelitian ini di masa depan. Berikut adalah beberapa saran untuk penelitian lebih lanjut:

1. Studi Perbandingan di Berbagai Platform dan Perangkat: Melakukan pengujian lebih lanjut untuk membandingkan performa PWA dan non-PWA di berbagai platform (desktop, mobile, tablet) dan berbagai perangkat (Android, iOS, PC) untuk memahami bagaimana platform dan perangkat yang berbeda mempengaruhi kinerja kedua aplikasi.
2. Pengukuran Pengalaman Pengguna (User Experience): Memasukkan aspek

pengalaman pengguna (user experience) ke dalam pengujian, seperti kecepatan persepsi, kecepatan interaksi, dan faktor-faktor lain yang memengaruhi kepuasan pengguna dalam menggunakan aplikasi PWA dan non-PWA.

#### REFERENSI

- [1] Rochim, R. V., Rahmatulloh, A., Akbar, R. El, & Rizal, R. (2023). Performance Comparison of Response Time Native, Mobile and Progressive Web Application Technology ARTICLE INFORMATION ABSTRACT (Vol. 5, Issue 1). <http://innovatics.unsil.ac.id>.
- [2] Kurniawan, A., Areni, I. S., & Achmad, A. (2017). Implementasi Progressive Web Application pada Sistem Monitoring Keluhan Sampah Kota Makassar. In *Jurnal Penelitian Enjiniring* (Vol. 21, Issue 02). Fakultas Teknik. [www.google.com](http://www.google.com).
- [3] Ridho, M. R., Pinandito, A., & Dewi, R. K. (2018). Perbandingan Performa Progressive Web Apps dan Mobile Web Terkait Waktu Respon, Penggunaan Memori dan Penggunaan Media Penyimpanan (Vol. 2, Issue 10). <http://j-ptiik.ub.ac.id>.
- [4] Aris B. (2023, July 28). What Is Website Performance and Why It's Important. [Hostinger.Com](https://www.hostinger.com).
- [5] Codemotion. (2022, October 3). Why is Web Performance More Important than Ever? [Codemotion.Com](https://www.codemotion.com).
- [6] Putri Agustika, C., & Saputra, W. S. (2021). PENGUJIAN APLIKASI GREENWALLET DENGAN METODE LOAD 177 TESTING DAN APACHE JMETER. In *Jurnal Informatika dan Sistem Informasi (JIFoSI)* (Vol. 2, Issue 2). <https://scholar.archive.org/work/fdkktnhxzzh7xo5cl6xpn2zqbe/access/wayback/http://jifosi.upnjatim.ac.id/index.php/jifosi/article/download/357/170>.
- [7] Elgamar. (2020). BUKU AJAR KONSEP DASAR PEMROGRAMAN WEBSITE DENGAN PHP. In Ndari Pangesti (Ed.), *CV. Multimedia Edukasi* (Vol. 1). CV. Multimedia Edukasi.
- [8] Erisu. (2017, September 14). Why Progressive Web Apps. <https://medium.com/the-web-tub/why-progressive-web-apps-a2c74bd6afdc> MDN Contributors. (2023a, October 26). Progressive web apps. 176 [https://developer.mozilla.org/en-US/docs/Web/Progressive\\_web\\_apps](https://developer.mozilla.org/en-US/docs/Web/Progressive_web_apps).
- [9] MDN Contributors. (2023a, October 26). Progressive web apps. 176 [https://developer.mozilla.org/en-US/docs/Web/Progressive\\_web\\_apps](https://developer.mozilla.org/en-US/docs/Web/Progressive_web_apps).
- [10] IBM. (2019, January 26). What is software testing? <https://www.ibm.com/topics/software-testing>.
- [11] Kinza Yasar. (2022, August 12). Definition software testing. <https://www.techtarget.com/whatis/definition/software-testing>.
- [12] Bronagh. (2022, June 23). 5 factors affecting website performance. <https://eyekiller.com/blog/5-factors-affecting-website-performance>.
- [13] The Apache Software Foundation. (n.d.). Apache JMeter. <https://jmeter.apache.org/>.
- [14] Dr. Untari Tyas, D. SE., MM. (2018). *Buku Ajar Metodologi Penelitian Kontemporer Bidang Ekonomi Dan Bisnis*. Penapersada. [www.penapersada.com](http://www.penapersada.com).
- [15] Sugiyono. (2013). *METODE PENELITIAN KUANTITATIF KUALITATIF DAN R&D*. ALFABETA BANDUNG. 178 [www.cvalfabeta.com](http://www.cvalfabeta.com).
- [16] Hartono. (2019). *METODOLOGI PENELITIAN*. Zanafa Publishing.