

# Penerapan Pustaka PolymerJs pada Aplikasi Presensi Dosen

Febianto Arifien

Program Doktor Teknologi Informasi, Universitas Gunadarma  
STMIK Jakarta STI&K

E-mail: febianto@mail.jak-stik.ac.id, febianto@staff.gunadarma.ac.id

## Abstrak

Pengguna UI (User Interface) saat ini sangat penting untuk pengalaman pengguna. Aplikasi yang menarik mata dan fitur yang dimuat harus memberikan pengalaman kepada pengguna. Meskipun Gaya dan desain penting, UI aplikasi dapat memberikan pengalaman pengguna seperti prinsip-prinsip antarmuka pengguna yang berbeda, Display, Input dan beberapa penggunaan perangkat. Penggunaan aplikasi presensi dosen saat ini belum diterapkan di STMIK Jakarta STI&K. Sistem pencatatan saat ini masih dilakukan secara manual tertulis per harinya kemudian diinput kedalam database untuk selanjutnya dapat direkapitulasi laporan kehadiran dosennya. Pengembang aplikasi berusaha mengembangkan aplikasi lintas platform untuk mengatasi permasalahan adanya beberapa platform yang berbeda. Polymer adalah PWA terbaik yang dikembangkan oleh Google. Polymer memberikan standar web yang berbeda yang membantu menjalankan aplikasi di browser dengan sangat cepat. Pembuatan aplikasi informasi presensi dosen diperlukan sebagai proses otomatisasi terhadap akses ke informasi, juga menciptakan akurasi, kecepatan dan kelengkapan sistem terintegrasi, jadi pengolahan data yang terjadi akan menjadi efektif dan efisien. Aplikasi presensi dosen menggunakan pustaka Polymer ini memberikan hasil testing yang cukup baik menggunakan Lighthouse tool dari Google Chrome.

**Kata Kunci:** Component Library, Polymer, User Interface, User Experience, Web Components.

## Pendahuluan

Pemantauan kinerja Sumber Daya Manusia, yang berlaku di suatu institusi salah satunya adalah kehadiran atau presensi SDM yang telah ditetapkan oleh masing-masing institusi, demikian pula termasuk institusi pendidikan. Kehadiran tatap muka dalam proses pembelajaran, merupakan kewajiban dosen pengampu dalam pendidikan tinggi. Hal ini bertujuan untuk membuka wawasan dan terjadi transfer ilmu juga pengalaman, sehingga dapat meningkatkan pemahaman mahasiswa. Pemantauan atau pencatatan kehadiran dosen, khususnya di STMIK Jakarta STI&K, saat ini masih dilakukan secara konvensional, yaitu dengan mencatat jam kehadiran dan saat mengakhiri tugas tatap muka dosen yang telah dijadwalkan. Kondisi ini menjadi tidak efektif jika petugas terlambat hadir atau berhalangan hadir, selain itu sangat rentan terhadap kesalahan pencatatan serta manipulasi data. Saat ini teknologi mobile berkembang pesat, strategi pengembangan mobile dibagi menjadi dua kelompok utama yaitu pengembangan native dan pengembangan berbasis web. Aplikasi mobile native dikembangkan hanya pada satu

platform misalnya, aplikasi Android dengan bahasa pemrograman Java dan Android Software Development Kit (SDK). Aplikasi native memiliki kinerja tinggi dan memiliki performa yang baik dari sisi pengguna (UX). Dikarenakan hasil kode program dari platform tersebut tidak dapat ditulis kembali pada platform lain, perlu untuk mengembangkan dan membuat kembali kode program sehingga proses pengembangan aplikasi akan efektif dan biaya pemeliharaan yang diperlukan lebih rendah. Sebaliknya, pengembangan berbasis web dengan menggunakan teknologi web, seperti HTML 5, CSS, dan JavaScript apabila dapat digunakan pada platform apapun sehingga dapat mengatasi permasalahan dalam pengembangan aplikasi.[1] Berdasarkan permasalahan tersebut, tujuan penelitian ini adalah membuat aplikasi presensi dosen menggunakan teknologi PWA (Progressive Web Apps) dengan model pustaka Polymer Js.

## Human Computer Interaction

Semakin berkembangnya teknologi komputer dan pemanfaatannya dimasyarakat, model interaksi konvensional (mouse dan keyboard) akan men-

jadi hambatan pemanfaatan arus informasi antara manusia dan komputer. Human Computer Interaction / IMK merupakan bagian disiplin dari ilmu komputer. Ilmu komputer adalah ilmu meliputi teori, metode serta latihan pemrosesan. Inti pembahasan pada IMK adalah pada langkah-langkah pembentukan model dan transformasi menjadi pemrograman komputer, dan menggunakan bahasa pemrograman untuk menulis program, menjalankannya pada mesin komputer, sifat-sifat terstruktur dari program, arsitektur dari perancangan program dan masih banyak lainnya.[2]

## Progressive Web App (PWA)

Pengembang aplikasi berusaha mengembangkan aplikasi lintas platform untuk mengatasi permasalahan adanya beberapa platform yang berbeda. Saat ini perkembangan web modern sudah dapat diakses secara offline, lain halnya seperti aplikasi native, setiap perpindahan halaman pada aplikasi tidak dapat menampilkan halaman secara keseluruhan, penggunaan web modern sudah dapat dipasang pada homescreen perangkat pengguna, dan menampilkan splashscreen sehingga ketika awal aplikasi dapat dibuka melalui icon pada homescreen perangkat pengguna. Penggunaan Progressive Web App (PWA) yang menggabungkan hal terbaik dari web modern dan fitur-fitur yang telah dimiliki oleh aplikasi native. Beberapa hal yang dapat dilakukan PWA antara lain website dapat dipasang pada ponsel pengguna dan ditambahkan ke homescreen layaknya aplikasi native, website yang telah diinstalasi dapat berfungsi secara offline dengan menggunakan offline cache APIs, dan cache aplikasi akan diperbaharui dari sisi server[3].

## Pustaka Polymer Javascript

Polymer adalah PWA terbaik yang dikembangkan oleh Google. Polymer memberikan standar web yang berbeda (sebagai contoh: webkomponen, variabel CSS) yang membantu menjalankan aplikasi di browser dengan sangat cepat. Menggunakan Polymer, dapat membuat sendiri elemen dan digunakan kembali pada aplikasi lain. Template yang disediakan merupakan kombinasi HTML, CSS, dan JavaScript yang mencakup fungsionalitas yang akan tersedia saat akan digunakan elemen itu. Polymer juga menyediakan materi Google desain untuk UI sehingga bisa digunakan dalam membangun hybrid aplikasi seluler[4].

## Metode Penelitian

Gambaran penelitian yang difokuskan pada pembuatan aplikasi, dapat dijelaskan pada Gambar 1.



Gambar 1: Gambaran Penelitian

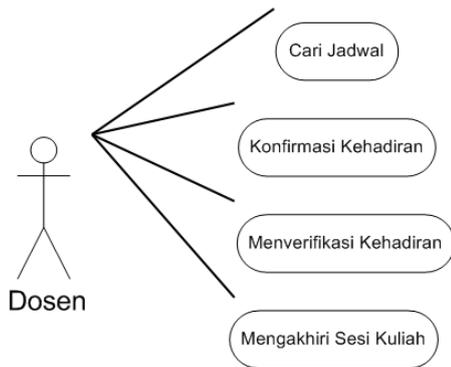
## Analisis Kebutuhan

Analisis kebutuhan digunakan untuk mengetahui kondisi yang harus dipenuhi dalam pembuatan aplikasi, yang meliputi:

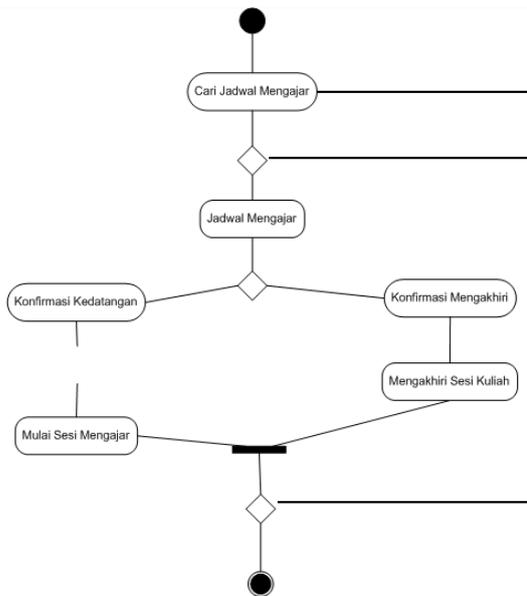
1. Kebutuhan internal yakni kebutuhan yang harus dipenuhi aplikasi, seperti aplikasi harus bisa menampilkan jadwal mengajar dosen, harus bisa menginput jam mengajar, dan juga harus bisa memverifikasi kehadiran dosen.
2. Kebutuhan eksternal yakni kebutuhan pendukung dalam pembuatan aplikasi seperti perangkat lunak (NodeJs, Microsoft Visual Studio, Git Repository) serta perangkat keras (PC, Monitor, Keyboard dan Mouse).

## Desain Aplikasi

Pada tahapan ini menampilkan beberapa rancangan menggunakan diagram UML meliputi use case, activity diagram, sequence diagram, dan class diagram. Selain itu juga ditampilkan perancangan antarmuka dan perancangan Progressive Web App. Gambar 2 merupakan diagram usecase yang menggambarkan interaksi aktor yaitu dosen dengan aplikasi.

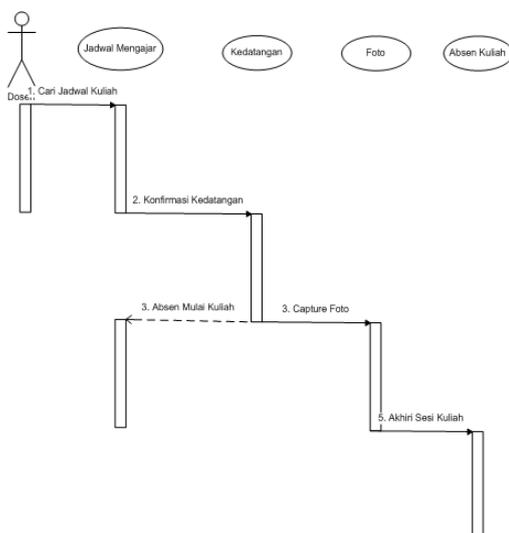


Gambar 2: Use Case Diagram Aplikasi



Gambar 3: Activity Diagram Aplikasi

Gambar 3 merupakan activity diagram yang menggambarkan aliran aktivitas atau aliran kerja pada aplikasi.



Gambar 4: Sequence Diagram Aplikasi

Gambar 4 merupakan sequence diagram yang menggambarkan interaksi objek dan komunikasi antar obyek-obyek tersebut.

## Implementasi

Antarmuka pada sistem akan dibangun secara responsif, diharapkan website dapat diakses pada perangkat dengan berbagai ukuran. Penggunaan teknologi service worker diharapkan membuat website tidak bergantung pada konektivitas jaringan dan selalu up-to-date. Menggunakan manifestweb app diharapkan akan membuat website bersifat installable, dan perancangan sistem dengan gaya architecture app shell diharapkan akan membuat website terasa seperti aplikasi native pada umumnya.

## Testing

Testing kualitas PWA menggunakan bantuan lighthouse tool yang terdapat pada browser Google Chrome, yang akan memverifikasi beberapa item di daftar baseline progressive web app checklist. Lighthouse yaitu perangkat lunak open-source untuk peningkatan kualitas aplikasi web. Lighthouse dapat dijalankan dengan ekstensi Chrome atau dari baris perintah (command line). Langkah pengujian dengan menginputkan URL website. Lighthouse menjalankan serangkaian pengujian terhadap website tersebut, kemudian menghasilkan laporan mengenai seberapa efektif website tersebut. Lighthouse saat ini memfokuskan pada fitur PWA, misalnya add to homescreen dan dukungan offline. Referensi pengujian yang diaudit menggunakan lighthouse tool meliputi[5]:

1. Performance, audit untuk melihat pertama akses aplikasi dan waktu interaktif untuk menentukan waktu jeda.
2. Accessibility, memeriksa masalah umum yang dapat mencegah pengguna mengakses konten aplikasi.
3. Best practices, mencari segala sesuatu dari penggunaan HTTPS untuk memperbaiki rasio aspek gambar.
4. SEO, memeriksa cara terbaik untuk memastikan situs Anda dapat ditemukan.

## Hasil dan Pembahasan

Pada tahapan ini merupakan implementasi terhadap perancangan aplikasi ke dalam bahasa pemrograman.

## Beranda

Pada beranda menampilkan keterangan aplikasi dan tabel jadwal, lihat Gambar 5.

## Cari Jadwal Kuliah

Pencarian jadwal kuliah dilakukan berdasarkan kelas, lihat Gambar 6.

## Konfirmasi Kedatangan

Apabila jadwal sudah didapatkan, maka dosen melakukan konfirmasi kedatangan, lihat Gambar 7.

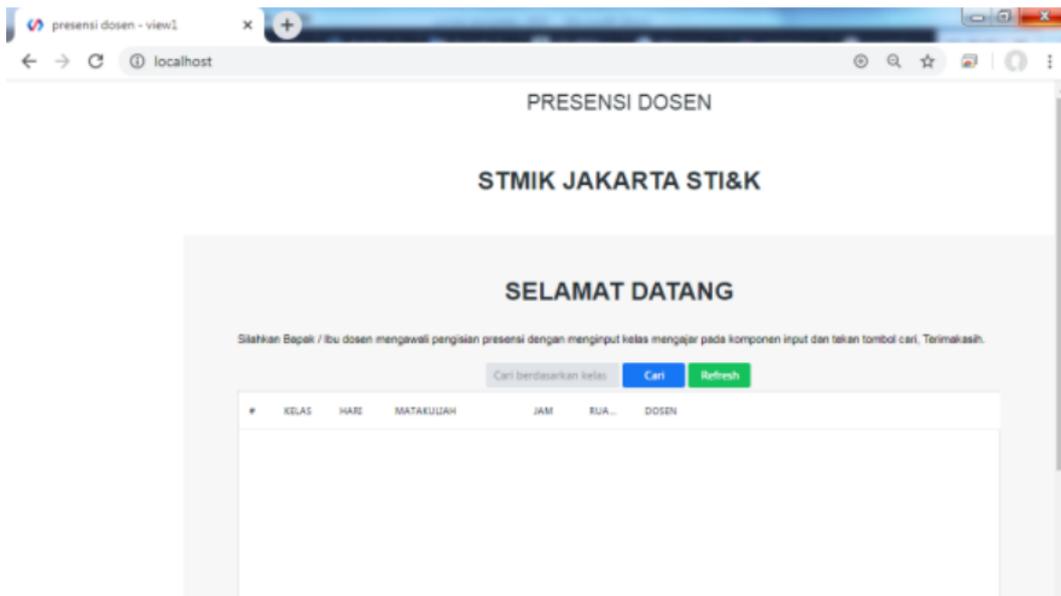
## Capture Foto

Untuk konfirmasi kedatangan, dosen harus mengambil foto untuk verifikasi apakah benar dosen

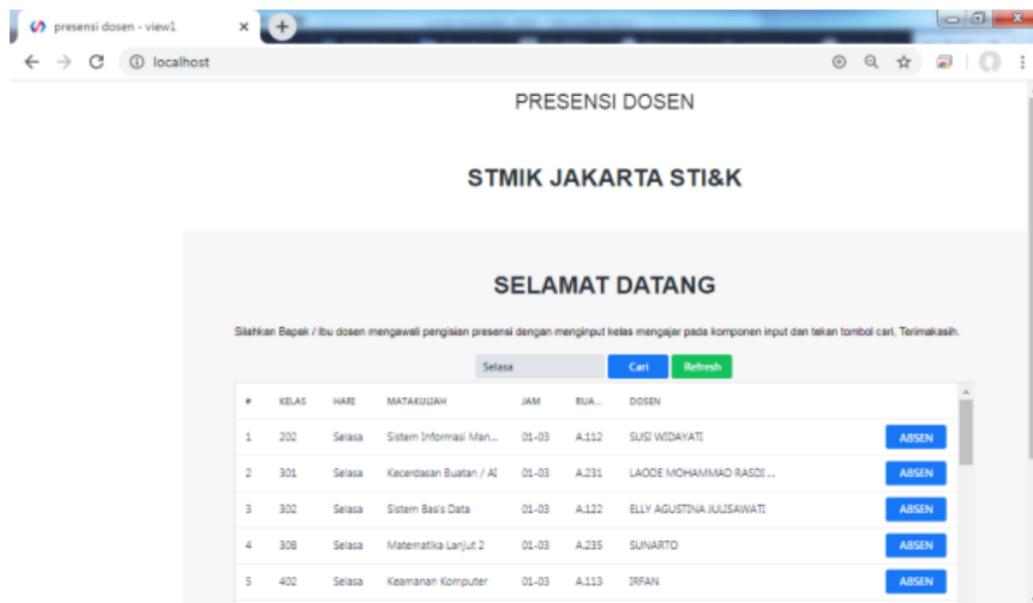
yang mengajar sesuai dan tidak diwakilkan, lihat Gambar 8.

## Pengujian

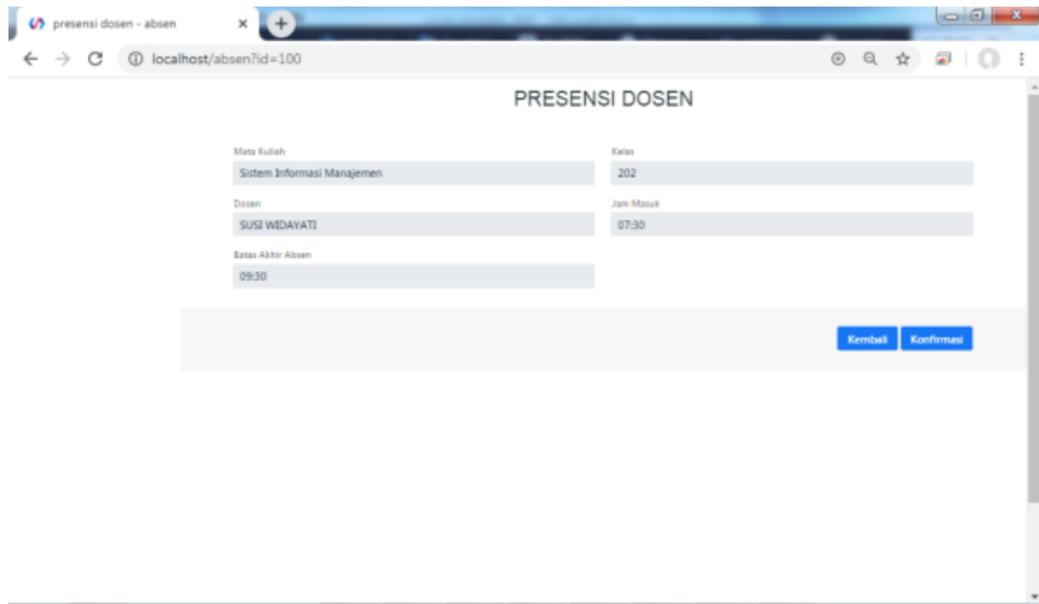
Pengujian aplikasi dilakukan berdasarkan platform yang biasa digunakan yaitu menggunakan desktop ataupun mobile. Fitur pengujian ini sudah tersedia pada browser Chrome pada modul audit. Pada gambar 9 terlihat bahwa dari 5 item yang diuji, yakni performance, accessibility, best practice, SEO, Progressive Web Apps, hanya performance yang memiliki nilai yang rendah, 28.



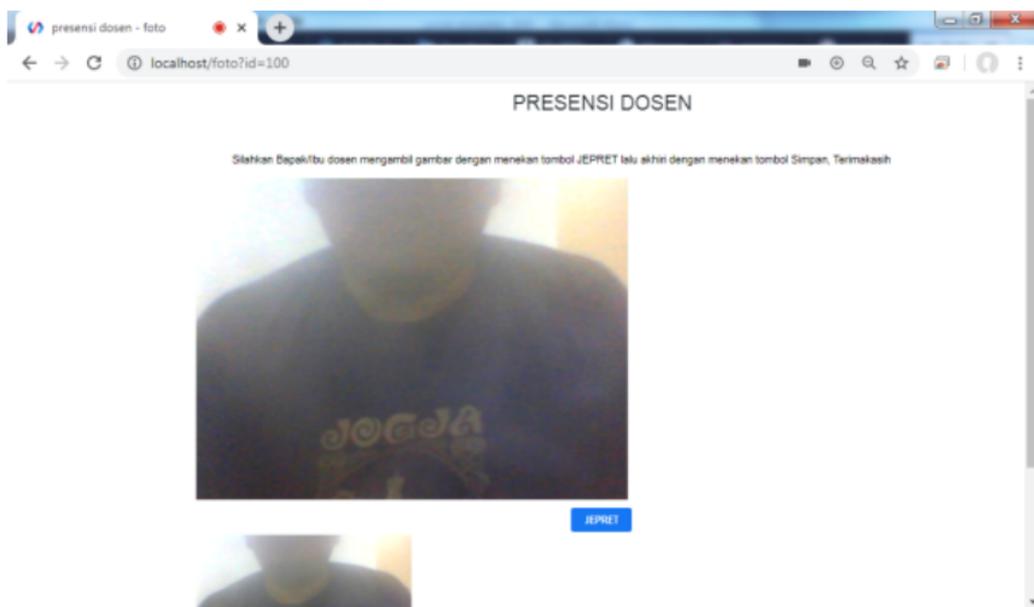
Gambar 5: Beranda



Gambar 6: Pencarian Jadwal



Gambar 7: Konfirmasi Jadwal



Gambar 8: Foto

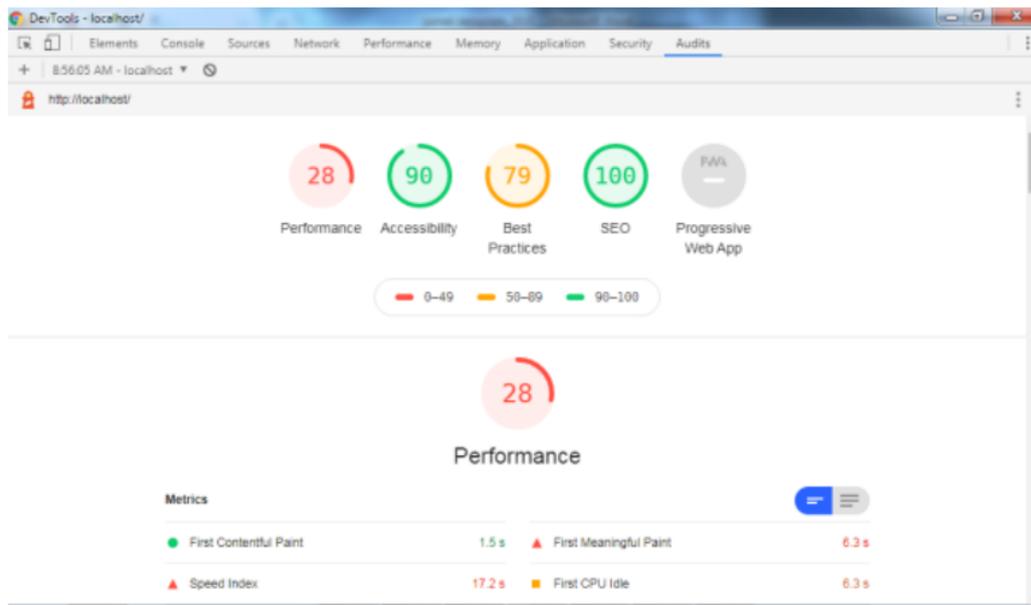
Pada gambar 10 juga terlihat pada 5 item yang diuji, yakni performance, accessibility, best practice, SEO, Progressive Web Apps, juga hanya performance yang memiliki nilai yang rendah, yaitu 4. Performance yang rendah menurut matriks dari lighthouse tool audit Google Chrome dikarenakan waktu yang relatif lama dari item berikut:

1. First Contentful Paint, waktu awal teks atau gambar dimuat.
2. Speed index, seberapa cepat memuat semua halaman.
3. Time to Interactive, waktu aplikasi berfungsi

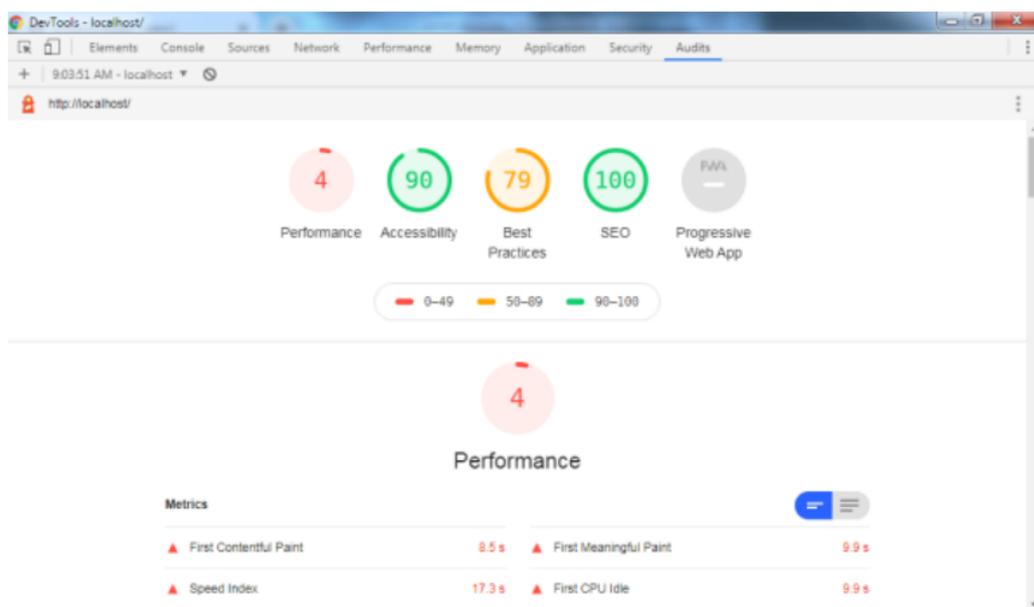
keseluruhan.

4. First Meaningful Paint, waktu halaman depan selesai dimuat.
5. First CPU Idle, waktu user dapat akses setelah halaman dimuat.
6. Max Potential First Input Delay, waktu paling lama pengguna menggunakan aplikasi.

Untuk selanjutnya dapat dicari solusi untuk mempercepat waktu performance berdasarkan indikasi dan analisis awal yang didapat dari hasil testing menggunakan lighthouse tool audit Google Chrome.



Gambar 9: Pengujian Platform Desktop



Gambar 10: Pengujian Platform Mobile

## Penutup

Penggunaan Progressive Web App memberikan pengalaman antarmuka seperti perbedaan platform dan penggunaan kembali elemen atau komponen pada aplikasi lainnya. Aplikasi Presensi Dosen di STMIK Jakarta STI&K menggunakan Polymer sudah membantu proses administrasi yang tadinya dilakukan manual. Performa setelah diuji menggunakan library Polymer cukup baik memberikan pengalaman pengguna yaitu dosen dalam melakukan interaksi dengan aplikasi. Selanjutnya diperlukan pengembangan pada sisi autentifikasi pengguna aplikasi yang saat ini masih manual mengambil foto

pengguna. Perlu dilakukan penambahan elemen atau komponen face detection atau face recognition pada aplikasi sehingga autentifikasi aplikasi bisa lebih baik.

## Daftar Pustaka

- [1] N. Nurwanto, "Penerapan Progressive Web Application (PWA) pada E-Commerce", *Techno.Com*, vol. 18, no. 3, pp. 227–235, Aug. 2019.
- [2] John M. Carroll, "Human-Computer Interac-  
290

- tion: Psychology as a Science of Design”, In International Journal of Human-Computer Studies, 46 (4) pp. 501-522, 1997.
- [3] A. Aminudin, B. Basren, dan I. Nuryasin, “Perancangan Sistem Repositori Tugas Akhir Menggunakan Progressive Web App (PWA)”, Techno.Com, vol. 18, no. 2, pp. 154–165, 2019.
- [4] Shabnam Shaikh and Lavina Jadhav, “Polymer JavaScript”, International Research Journal of Engineering and Technology (IRJET), Volume: 05, Issue: 06, pp. 633–638, June 2018.
- [5] I. A. Rahman dan I. Ikbali, “Perancangan Lite-speed Cache Menggunakan Metode PPDIIO di PT. ABC”, Komputa Jurnal Ilmu Komputer dan Informasi, vol. 8, no. 2, pp. 61–68, Oct. 2019

Halaman ini sengaja dikosongkan.