

RANCANG BANGUN SISTEM INFORMASI MANAJEMEN MASJID MENGUNAKAN PROGRESSIVE WEB APPS (Studi Kasus: Masjid Di Kota Pontianak)

^[1]Ari Yudha Prawira, ^[2]Ikhwan Ruslianto, ^[3]Dian Prawira

^[1]^[2]^[3]Jurusan Sistem Informasi, Fakultas MIPA

Universitas Tanjungpura

Jalan Prof.Dr.H Hadari Nawawi, Pontianak

Telp./Fax.: (0561) 577963

e-mail: ^[1]ari_yudhaprawira@student.untan.ac.id, ^[2]dianprawira@sisfo.untan.ac.id,

^[3]ikhwanruslianto@siskom.untan.ac.id

Abstrak

Masjid memiliki peran penting sebagai pusat peribadahan, dakwah dan peradaban Islam. Namun saat ini masjid di Kota Pontianak masih menggunakan cara manual dalam melakukan pengelolaan masjid serta belum diikuti dengan peningkatan kualitas pengelolaannya baik pada aspek idarah (manajemen), imarah (kegiatan memakmurkan), dan riayah (pemeliharaan). Maka perlu adanya penerapan komputerisasi untuk membantu pengurus masjid dalam pengelolaan masjid agar lebih efisien dan efektif. Oleh karena itu dibangun sebuah Sistem Informasi Masjid yang menerapkan teknologi informasi yang dapat memberikan pengalaman pengguna serta performa layaknya mobile apps. Teknologi yang digunakan dalam penelitian ini adalah PWA (Progressive Web Apps). Hasil implementasi prinsip-prinsip PWA pada penelitian ini yaitu dapat berjalan saat offline, tidak akan memunculkan pesan network error di halaman website pada jaringan yang kurang memadai berkat bantuan dari service worker dan app manifest untuk menghilangkan proses download aplikasi melalui app store agar bisa dipasang pada homescreen secara langsung. Hasil audit PWA menggunakan tools lighthouse pada chrome yaitu installable mencapai angka 1/1 dan optimized mencapai angka 8/8 maka, sistem dapat dikatakan sebagai aplikasi berbasis PWA dikarenakan telah memenuhi berbagai kriteria dari teknologi PWA itu sendiri. Hasil penelitian ini berupa sistem informasi manajemen masjid yang telah dilakukan pengujian fungsional kepada pihak pengurus masjid dan memperoleh hasil sesuai dengan rancangan fungsional sistem. Sedangkan pengujian interface kepada masyarakat umum memperoleh predikat sangat baik dengan persentase 86%.

Kata kunci : PWA, Service Worker, App Manifest

1. PENDAHULUAN

Masjid merupakan bangunan penting dan suci bagi umat muslim. Tidak hanya sebatas rumah ibadah, masjid juga memiliki peran penting dalam usaha pengembangan kualitas, pusat pendidikan dan dakwah serta kesejahteraan maupun pusat kehidupan umat muslim di dunia. Seiring dengan tumbuhnya semangat masyarakat untuk memakmurkan masjid dewasa ini, sesungguhnya masih belum diikuti dengan peningkatan kualitas pengelolaannya baik pada aspek idarah (organisasi dan manajemen), dan riayah (pemeliharaan dan pembangunan).

Pengelolaan masjid oleh para takmir seringkali masih sangat sederhana, belum terkomputerisasi dengan baik bahkan sering kali terkesan apa adanya. Hal ini berdasarkan dengan kualitas SDM pengelola masjid yang belum merata dan perlu peningkatan. Penerapan teknologi informasi membantu memperluas layanan dan bimbingan pemerintahan kepada masyarakat dengan cepat, efektif dan efisien. Pada tahun 2018 Direktorat Urusan Agama Islam dan Pembinaan Syariah melalui program masjid berkualitas memiliki kegiatan nasional Pengelolaan Manajemen Masjid berbasis website, upaya melakukan inovasi dalam

pengelolaan manajemen masjid dengan menghadirkan layanan berbasis aplikasi yang dapat mempermudah para takmir untuk melakukan fungsi manajemen dengan baik dan modern [1].

Masalah masjid di Kota Pontianak yang sering terjadi terletak pada pelayanan kepada jamaah, dikarenakan peningkatan jumlah penduduk yang begitu pesat disuatu lokasi, membuat pengurus harus cekatan dalam memberikan pelayanan. Kurangnya pemahaman dan penggunaan sumber daya komputer masih sangat minim membuat manajemen masjid secara *online* sulit dilakukan. Jaringan internet yang tidak stabil sangat menyulitkan pengurus masjid dalam melakukan pengolahan data. Menurut Aziz [2] Lambatnya proses rekapitulasi maupun dokumentasi membuat kurang efisiensi dalam penulisan maupun perhitungan pengadministrasian masjid yang dilakukan satu persatu secara manual, mengakibatkan sering terjadinya human error seperti redudansi ataupun selisih data rekapitulasi dengan data riil pengadministrasian masjid, yang menyebabkan data tidak valid dan akan membuat jamaah kurang puas terhadap pelayanan kepengurusan masjid karena informasi yang disampaikan tidak valid dan tidak transparan. Proses manajemen masjid yang masih tradisional menimbulkan berbagai masalah seperti pencatatan kegiatan, inventaris, keuangan dan administrasi masih menggunakan buku maupun kwitansi yang rawan akan hilang ataupun rusaknya data.

Beralih dari permasalahan yang telah di rumuskan, masjid di Kota Pontianak memerlukan SIMM (Sistem Informasi Manajemen Masjid). SIMM adalah perangkat lunak yang mendukung proses manajemen data pengadministrasian masjid secara online dengan preview laporan yang dikelola langsung oleh pengurus masjid yang membuat data dapat cepat diupdate dengan valid. Riset ini mengimplementasikan teknologi *Progressive Web Apps* (PWA) karena website akan lebih cepat maupun mudah meminta data melalui fetch API dan dapat berjalan secara *offline* atau jaringan yang tidak stabil berkat *cache local* yang tersimpan sebagai aset web oleh *Service Worker*. Hal tersebut berupaya agar pengguna dapat melakukan pengecekan informasi masjid secara instan dimanapun dan kapanpun. Sehingga tidak perlu ke masjid untuk

mendapatkan informasi pada masing masjid. Kemudian demi kemudahan dalam mengakses sistem, web juga diberikan kemampuan untuk menginstall dan menyimpan *bookmark* ke *homescreen* perangkat melalui *Web App Manifest* dengan begitu pengguna akan lebih mudah untuk menginstall sistem tanpa harus melalui playstore dan dapat menghemat ruang penyimpanan. Berdasarkan dari konsep PWA tersebut menghasilkan sistem yang dapat memberikan pengalaman pengguna aplikasi web progressif layaknya *mobile native* agar mudah untuk dipahami dalam pengoperasiannya oleh berbagai kalangan yang dapat digunakan secara *responsive* di platform manapun seperti android, IOS, windows, ataupun linux. Sistem ini diharapkan dapat menciptakan pengelolaan masjid yang berkualitas, akuntabel dan modern untuk pelayanan umat yang lebih baik.

2. LANDASAN TEORI

2.1. Sistem Informasi

Suatu sistem di dalam organisasi yang mengimpun keperluan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi serta menyediakan pihak luar tertentu dengan memberikan laporan yang diperlukan adalah sistem informasi [3].

2.2. Manajemen Masjid

Pengelolaan atau sering disebut manajemen ialah sebuah proses penetapan dan pencapaian tujuan melalui penerapan empat fungsi dasar yaitu perencanaan, pengorganisasian, bimbingan, dan pengendalian dalam penggunaan sumber daya organisasi. Dari definisi tersebut dapat diketahui bahwa didalam manajemen terdapat beberapa bagian yaitu tujuan, dan SDM. Aspek tersebutlah yang dikelola secara efektif dan efisien demi mencapai tujuan dalam pemberdayaan ekonomi masyarakat [4].

2.3. Progressive Web Apps

Progressive Web Application (PWA) merupakan suatu website yang dapat berlaku seperti *mobile app* namun dibangun dengan menerapkan teknologi web modern. Pada tahun 2015, *Google Engineer Alex Russel & Frances Berriman* berusaha memastikan bahwa konsep web apps ini dapat memberikan *users experience* (kepuasan pengguna) seperti

Coding : Jurnal Komputer dan Aplikasi
Volume 11, No. 03 (2023), hal 313-324
aplikasi *native* dalam hal *speed* (kecepatan),
reliability (keandalan), dan *users engagment*
(keterlibatan pengguna). Dimana kata
progresif mengarah kepada kemajuan secara
bertahap. Perkembangan fitur aplikasi *native*
akan diikuti oleh PWA dengan memanfaatkan
web browser[5].

2.4. Basis Data

Perancangan basis data (*database*) merupakan bagian dari siklus hidup mikro. Sedangkan aktivitas yang terdapat di dalam proses tersebut meliputi: Pengumpulan data dan analisis, pemilihan DBMS (*database management system*) dan implementasi sistem database. Sedangkan aktivitas utama dalam perancangan suatu database meliputi perancangan database secara konseptual (*conceptual scheme design*), perancangan database menurut logika (*logical design*), perancangan database menurut fisik (*phisycal design*).

Perancangan database memiliki tujuan utama yaitu, untuk memenuhi informasi yang berisikan keperluan pengguna tersendiri dan aplikasi-aplikasinya, kedua, memudahkan pemahaman susunan informasi, dan yang ketiga ialah untuk membantu keperluan pemrosesan dan beberapa bahan penampilan seperti waktu tanggap, waktu proses, dan ruang penyimpanan[4].

2.5. Web Service

Menurut standarisasi *web service* yang dibangun oleh W3C yang bergerak di lembaga internasional *web service* merupakan sistem perangkat lunak yang dirancang dapat dioperasikan oleh mesin ke mesin lewat jaringan. *Web service* digunakan oleh Pruter et al untuk menghubungkan secara dinamis perangkat-perangkat yang sudah terdeteksi ataupun yang tidak terdeteksi pada suatu jaringan komputer. Dengan begitu, *web service* dapat menjadi sarana operasi standar antar perangkat lunak pada berbagai platform yang berbeda[7].

2.6. REST API

REST API merupakan suatu penerapan API (*Application Programming Interface*). Sedangkan REST (*Representational State Transfer*) merupakan suatu arsitektur metode komunikasi yang menggunakan protokol

p-ISSN : 2338-493X

e-ISSN : 2809-574X

HTTP untuk pertukaran data. REST adalah teori desain yang mendorong kita untuk menggunakan aturan dan fitur yang sudah ada pada web untuk memetakan permintaan terhadap sumber daya pada berbagai macam manipulasi dan representasi data di internet[8].

2.7. Analisis PIECES

Kerangka analisis ini disebut dengan analisis PIECES (*Performance, Information, Economy, Controll, Efeciency, & Service*). Didalam perancangan perangkat lunak analisis PIECES berfungsi untuk mengetahui permasalahan terhadap kinerja, informasi, ekonomi, keamanan, efisiensi dan pelayanan. Berikut merupakan penjelasan dari beberapa komponen analisis PIECES [3]:

1. Analisis Kinerja, diukur berdasarkan waktu tanggap dan jumlah produksi. Waktu tanggap merupakan waktu yang diperlukan dari suatu transaksi terhadap tanggapan yang diberikan kepada transaksi tersebut. Sedangkan jumlah produksi ialah jumlah dari pekerjaan yang bisa diperoleh dalam jangka waktu tertentu.

2. Analisis Informasi, merupakan analisis evaluasi terhadap kemampuan sistem informasi demi menghasilkan informasi yang bermanfaat. Dalam hal tersebut meningkatkan kualitas informasi tidak dengan penambahan jumlah informasi, karena terlalu banyak informasi akan berdampak terhadap bagian lainnya.

3. Analisis Ekonomi, merupakan analisis untuk memperkirakan pengaruh ekonomi terhadap pengembangan perangkat lunak seperti biaya tidak dapat dilacak ke sumber, biaya yang tidak diketahui, biaya terlalu tinggi, penambahan pemasaran, dan keuntungan pasar-pasar baru yang dapat di riset.

4. Analisis Keamanan, berdasarkan keamanan atau kontrol yang lemah seperti masukkan data tidak memenuhi persyaratan, kejahatan pencurian data, pelanggaran etika informasi dan peraturan privasi data, terjadi kesalahan saat membuat keputusan dan diproses oleh manusia, mesin, atau perangkat lunak. Sedangkan keamanan dan pengendalian yang berlebihan akan dapat mengakibatkan prosedur birokrasi memperlambat pemrosesan sistem.

5. Analisis Efisiensi, memperkirakan bagaimana cara menciptakan *output* sebanyak mungkin dengan *input* yang sangat minimal. Berikut merupakan indikasi bahwa suatu

sistem dapat dikatakan tidak efisien:

- a. Banyaknya waktu yang terbuang.
 - b. Terlalu banyak data masukan atau disalin.
 - c. Terlalu banyak informasi yang dihasilkan.
 - d. Terlalu banyak usaha yang dibutuhkan untuk tugas-tugas.
 - e. Terlalu banyak bahan yang dibutuhkan untuk tugas-tugas.
6. Analisis Layanan, memperkirakan kriteria penilaian kualitas sistem yang buruk yaitu menghasilkan produk yang tidak akurat dan tidak konsisten serta tidak dapat dipercaya, tidak mudah untuk digunakan maupun dipelajari, canggung untuk digunakan dan tidak fleksibel.

2.8. UML (*Unified Modelling Language*)

Permodelan atau model piranti lunak dapat diibaratkan seperti pembuatan cetak biru (*blueprint*) untuk pembangunan gedung. Permodelan merupakan proses merancang piranti lunak sebelum melakukan pengkodean atau coding. Membangun model berdasarkan suatu sistem yang kompleks begitu penting karena kita tidak dapat memahami sistem semacam itu secara menyeluruh. Semakin kompleks sebuah sistem, semakin membutuhkan teknik pemodelan yang baik. Dengan menggunakan model, diharapkan pengembangan *software* dapat memenuhi berbagai keinginan pengguna secara tepat dan lengkap, termasuk aspek-aspek seperti ketahanan, keamanan, dan sebagainya. Kesuksesan sebuah pemodelan *software* berdasarkan tiga aspek, yang sering disebut segitiga sukses (*the triangle for success*). Ketiga aspek tersebut ialah metode pemodelan (*notation*), proses (*process*) dan alat (*tool*) yang digunakan[7].

2.9. *Framework Laravel*

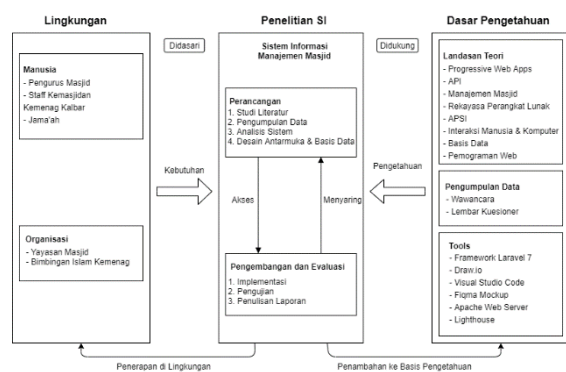
Suatu pengembangan website akan dapat lebih mudah jika menggunakan *tools* yang tepat, seperti pemilihan *framework* yang baik yaitu sesuai dengan kebutuhan aplikasi yang ingin dibangun. Dimana *framework* itu adalah sebuah kerangka kerja yang terdiri dari beberapa folder yang berisikan file PHP yang menyajikan *class libraries*, *plugins*, *helpers* dan sebagainya. *Framework* menyajikan teknik coding tertentu serta konfigurasi. Adanya penggunaan *framework* bertujuan untuk menyederhanakan proses pembuatan dengan menghasilkan kinerja web yang

maksimal.

Salah satu *framework* yang banyak diminati saat ini ialah laravel karena dengan kesederhanaannya sudah dapat menghasilkan aplikasi web yang powerful. Laravel merupakan *framework* aplikasi web open source yang di bangun menggunakan bahasa pemograman PHP, dengan menerapkan konsep MVC (*Model, View, Controller*) codingan akan lebih terstruktur. Adanya *framework* laravel bertujuan untuk memudahkan developer untuk membangun aplikasi web dinamis secara mudah dan cepat dibandingkan harus membuatnya dari awal [2].

3. METODE PENELITIAN

Metode penelitian yang digunakan mengacu pada kerangka kerja IS Research yang membagi penelitian kedalam tiga bagian yang saling terhubung. Ketiga bagian tersebut adalah Lingkungan, Penelitian Sistem Informasi, dan Dasar Pengetahuan. Metode Penelitian dapat dilihat pada Gambar 1.



Gambar 1. Metode Penelitian.

a. Lingkungan

Lingkungan penelitian menjelaskan dua bagian utama yaitu manusia, dan organisasi. Pada bagian manusia menjelaskan siapa saja yang terlibat dan menjadi salah satu sumber data, diantaranya ialah pengurus masjid, serta jamaah masjid di Kota Pontianak dan staff kemasjidan kemenag kalbar. Pada bagian organisasi menjelaskan tentang tempat dilakukannya penelitian yaitu yayasan masjid dan Ditjen Bimas islam kemenag. Manfaat yang didapatkan bagi lingkungan yaitu membantu mempermudah dalam pengelolaan informasi masjid dengan tepat dan cepat yang kemudian dapat membantu meringankan pekerjaan pengurus masjid serta dapat membantu kemenag kalbar dalam melakukan pendataan

masjid yang ada di Kota Pontianak. Sedangkan teknologi pendukung dalam pengerjaan penelitian ini adalah framework laravel sebagai pengembangan web dan penerapan teknologi PWA.

b. Penelitian Sistem Informasi

Pada penelitian sistem informasi ini menjelaskan bagian utama dalam penelitian, terbagi menjadi dua bagian utama yaitu perancangan serta pengembangan dan evaluasi. Dimana kedua bagian ini bisa dikerjakan secara fleksibel, tidak harus menyelesaikan bagian perencanaan terlebih dahulu kemudian bagian pengembangan dan evaluasi maupun sebaliknya.

c. Perancangan

1. Studi Literatur

Studi literatur diperlukan dengan cara mencari dokumen penelitian yang telah dilakukan sebelumnya dengan keterkaitan masalah SIMM menggunakan PWA. Studi literatur ini dilakukan guna mencari referensi dan mengambil perbandingan dalam pembuatan sistem.

2. Pengumpulan Data

Pengumpulan data adalah bagian dimana peneliti harus memperoleh data dan informasi yang diperlukan dalam penelitian. Data dan informasi ini digunakan untuk menentukan kebutuhan sistem. Metode yang digunakan adalah observasi langsung dan wawancara.

3. Analisis Sistem

Tahap ini bertujuan untuk menemukan masalah-masalah yang sering dihadapi dalam proses manajemen masjid, kebutuhan apa saja yang diperlukan dalam perancangan sistem informasi manajemen, serta layak atau tidaknya sistem yang akan dirancang.

4. Desain Antarmuka & Basis Data

Tujuan dari tahapan ini adalah merancang seperti apa nantinya tampilan sistem ini dibuat, tampilan sistem yang efektif dapat memudahkan pengguna dalam menggunakannya. Sedangkan desain basis data bertujuan untuk memenuhi informasi yang dibutuhkan pengguna dan memudahkan pemrosesan dalam sistem.

Pengembangan dan Evaluasi:

1. Implementasi

Tahap ini merupakan proses penerapan

hasil dari perancangan analisis sistem, desain antarmuka dan basis data serta langkah-langkah dalam metode yang digunakan dalam kode program sehingga menjadi sebuah aplikasi berbasis web.

2. Pengujian

Pengujian bertujuan untuk mengevaluasi sistem yang telah dibuat baik dari segi kinerja maupun tampilan antarmuka. Metode yang digunakan adalah pengujian Black Box dimana pada metode ini bagian sistem yang dievaluasi hanya bagian-bagian fungsional sistem bukan kode program. Selain pengujian fungsional sistem, pengujian juga dilakukan pada hasil dari penerapan PWA apakah berhasil di implementasikan pada sistem.

3. Penulisan Laporan

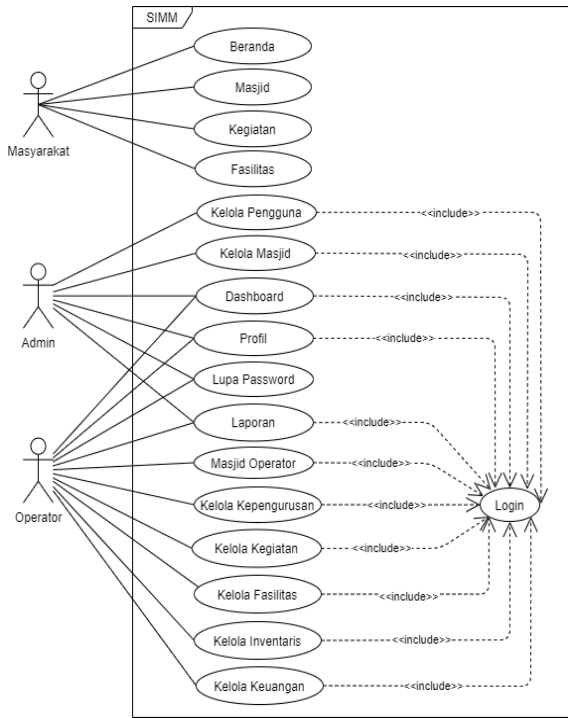
Tahapan ini dilakukan penyusunan laporan tugas akhir sebagai salah satu syarat lulus program studi Sistem Informasi FMIPA Untan dengan judul penelitian Sistem Informasi Manajemen Masjid Menggunakan Progressive Web Apps. Dalam tahap ini dituliskan proses dari awal hingga akhir dalam melakukan penelitian termasuk kesimpulan dan saran yang dapat dijadikan masukan bagi penulis ataupun penelitian selanjutnya.

Dasar Pengetahuan

Dasar pengetahuan terdiri dari dua bagian yaitu landasan teori dan metodologi. Landasan teori berisi teori yang dapat digunakan untuk mendukung penulisan dan pengerjaan penelitian, sedangkan metodologi merupakan metode-metode yang digunakan untuk mengumpulkan data, menganalisis data, serta menganalisis data.

4. HASIL DAN PEMBAHASAN

4.1. *Use Case Diagram*

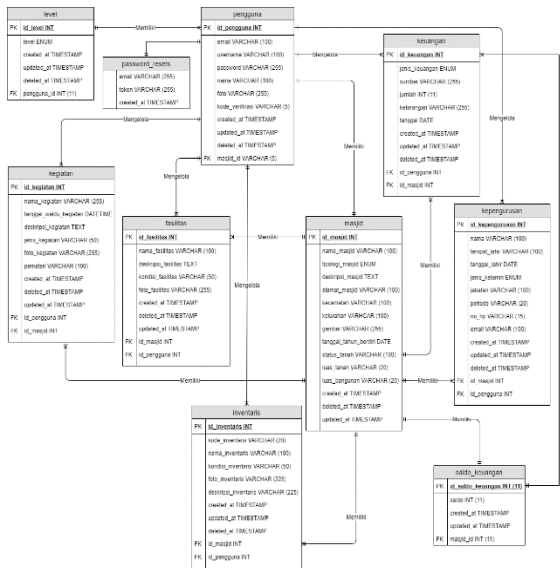


Gambar 2. Use Case Diagram

Gambar 2 merupakan diagram *use case* yang menggambarkan interaksi antara aktor dan fungsi-fungsi didalam sistem.

4.2. ERD

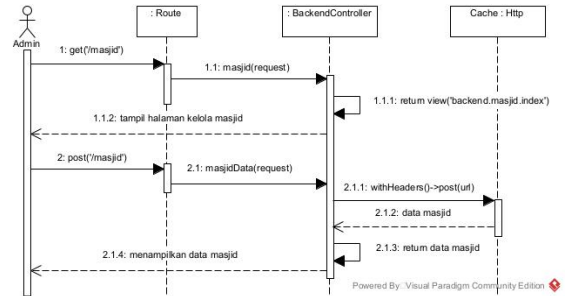
Gambar 3 merupakan rancangan *Entity Relationship Diagram (ERD)* pada sistem informasi manajemen masjid dengan menggunakan notasi *crow's foot*.



Gambar 3. Entity Relationship Diagram (ERD)

4.3. Sequence Diagram

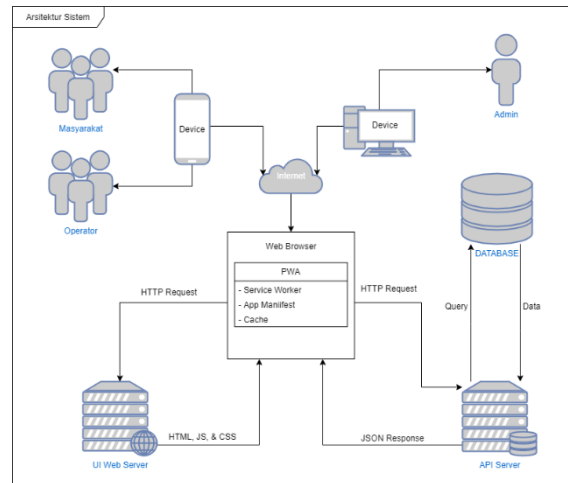
Adapun *Sequence Diagram* Kelola Masjid dapat dilihat pada gambar 4.



Gambar 4. Sequence Diagram

4.4. Arsitektur Sistem

Adapun Arsitektur Sistem SIMM dapat dilihat pada gambar 5.



Gambar 5. Arsitektur Sistem

4.5. Implementasi

Implementasi sistem merupakan implementasi hasil rancangan antarmuka ke dalam sistem yang dibangun menggunakan *framework laravel* dengan bahasa pemrograman PHP untuk *frontend* dan *backend*. Berikut implementasi dari setiap fungsional pada sistem informasi manajemen masjid.

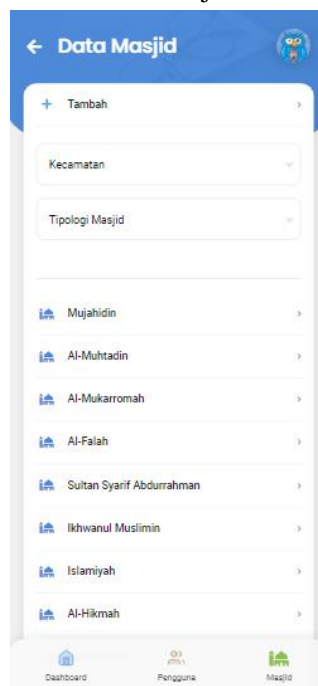
Antarmuka Beranda



Gambar 6. Antarmuka Beranda

Halaman beranda menampilkan data masjid, kegiatan, dan fasilitas yang telah tersimpan pada *database* sistem. Pengguna mampu menjalankan beberapa fungsi lainnya dengan memilih menu pada *tabs*. Pada sisi *topbar* ada menu *sidebar* yang berisikan fitur untuk *login*, tentang, dan faq sistem.

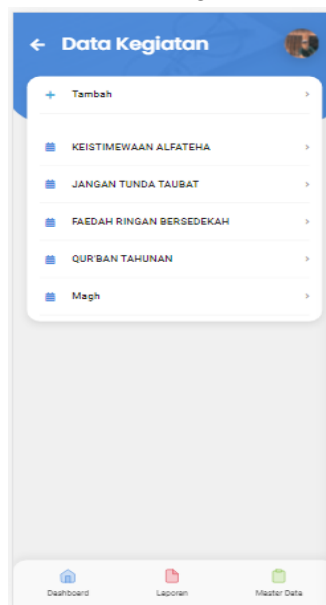
1. Antarmuka Kelola Masjid



Gambar 7. Antarmuka Kelola Masjid

Halaman kelola masjid menampilkan *list* data masjid dari basis data. Pada halaman ini admin dapat melakukan beberapa fungsi diantaranya tambah masjid, lihat *detail* masjid, ubah masjid dan hapus masjid.

2. Antarmuka Kelola Kegiatan



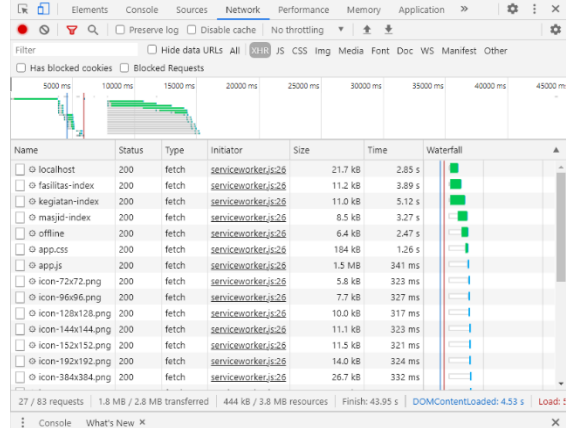
Gambar 8. Antarmuka Kelola Kegiatan

Halaman kelola kegiatan menampilkan *list* berisi data kegiatan dari basis data. Pada antarmuka ini operator dapat melakukan beberapa fungsi diantaranya tambah kegiatan,

lihat *detail* kegiatan, ubah kegiatan dan hapus kegiatan.

3. Pengujian *Chrome Devtools*

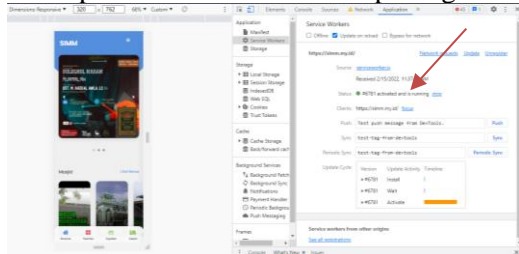
Pengujian kinerja *service worker* pada Sistem Informasi Manajemen Masjid (SIMM) digunakan *Chrome Devtools* dengan berbagai kondisi. Pegujian ini bertujuan untuk menunjukkan kinerja *service worker* pada aplikasi web dengan melihat jumlah *request*, waktu yang dibutuhkan untuk *load* semua konten halaman, dan jumlah data yang dikirim.



Gambar 9. Hasil pengujian Network Response pada *Chrome DevTools*

Apabila terlihat status *running* dan *activated* maka penambahan *service worker* telah berhasil dilakukan. Alasan kenapa sistem web ini bisa berjalan secara *offline* dikarenakan adanya dukungan dari *service worker* yang sudah terpasang pada sistem. Hal demikianlah yang membuat sistem meskipun dalam keadaan *offline* maupun koneksi jaringan yang buruk, halaman masih dapat dimuat seketika secara utuh.

4. Tampilan *Service Worker* Terpasang

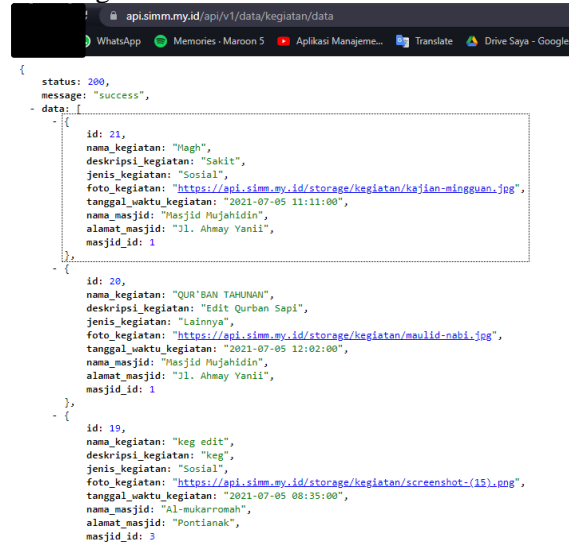


Gambar 10. Tampilan *Service Worker* Terpasang

Data yang diperoleh berupa *file/assets* yang bisa disimpan oleh *service worker* pada saat sistem dijalankan dimana data tersebut berasal dari sebuah API yang kemudian

dianalisis demi memperlihatkan kinerja dari *service worker* dalam melakukan *caching* berkas sehingga data masjid yang diakses sistem tetap dapat ditampilkan walaupun ketika keadaan *offline*. Pada proses halaman SIMM ada beberapa pilihan yang mendukung proses kerja sistem.

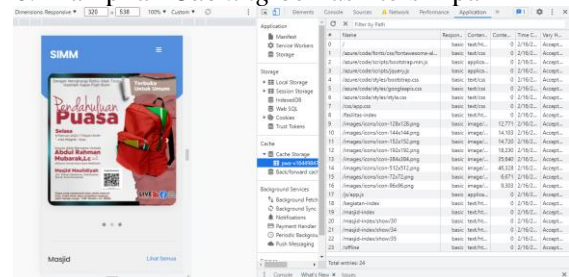
5. Pengambilan Data API



Gambar 11. API Kegiatan dalam format JSON

Proses pengambilan data masjid yang dijalankan dalam pengambilan data kegiatan masjid dengan menjalankan url API yang kemudian menghasilkan data dalam bentuk format JSON yang terdiri dari id, nama_kegiatan, foto, jenis_kegiatan, tanggal_kegiatan dan pemateri.

6. Tampilan *Caching* berhasil tersimpan

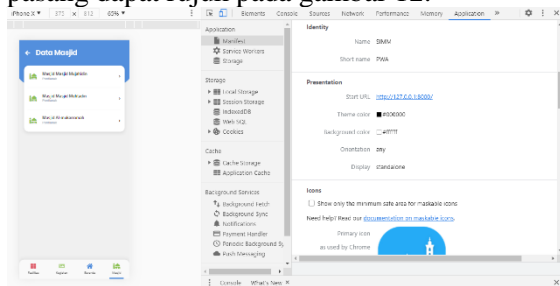


Gambar 12. Tampilan *Caching* berhasil tersimpan

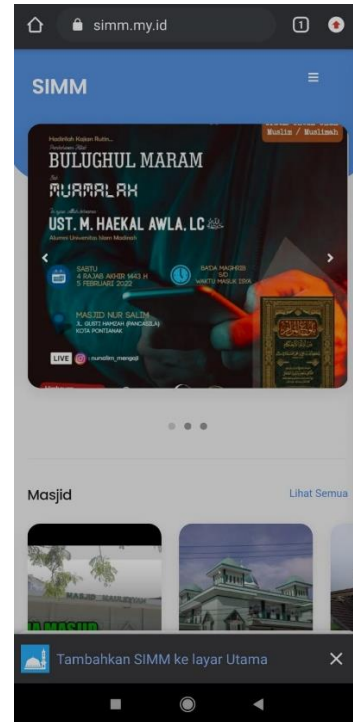
Sistem menjalankan proses *caching* dari *service worker*. Pada saat jaringan ketika dalam keadaan *offline*, sistem bisa dijalankan dengan cepat karena hanya memuat data yang sudah disimpan oleh *service worker*.

7. Tampilan *App Manifest* Terpasang

Browser chrome pada *development tools* menyajikan varian khusus properti manifest agar mudah untuk dibaca atau bisa juga dilakukan pengujian pada tools audit PWA yaitu *lighthouse* yang telah tersedia juga pada ekstensi *chrome* dengan cara memberikan URL pada *lighthouse*. Selanjutnya *lighthouse* menjalankan serangkaian audit terhadap laman tersebut yang kemudian akan menampilkan hasil laporan audit. Serangkaian audit *lighthouse* terkait *app manifest* akan memeriksa apakah sistem dapat ditambahkan ke layar utama *smartphones*, kemudian sesudah ditambahkan apakah aplikasi akan diluncurkan bersama layar pembuka khusus (*splashscreen*), dan sistem harus berada di *HTTPS* yang merupakan prasyarat untuk *Add to Homescreen*. Adapun tampilan *App manifest* berhasil di pasang dapat rujuk pada gambar 12.



Gambar 12. Tampilan *App Manifest* Terpasang
Apabila terlihat pada point manifest maka dapat dikatakan sistem telah berhasil menambakan app manifest. Perlu adanya fakta bahwa pengguna dapat melakukan penambahan situs web pada layar utama (*homescreen*) *smartphones*, adapun caranya yaitu pengguna dapat membuka URL sistem pada *google* yang kemudian akan muncul suatu *pop up* notifikasi berisi apakah pengguna ingin menambahkan ikon pada *homescreen smartphones* untuk digunakan langsung pada layar utama. Adapun tampilan notifikasi permintaan *Add to Homescreen* dan tampilan *homescreen* berhasil di pasang pada *smartphones* dapat rujuk pada gambar 13 dan 14.



Gambar 13 Tampilan Permintaan Add Homescreen pada *Smartphones*

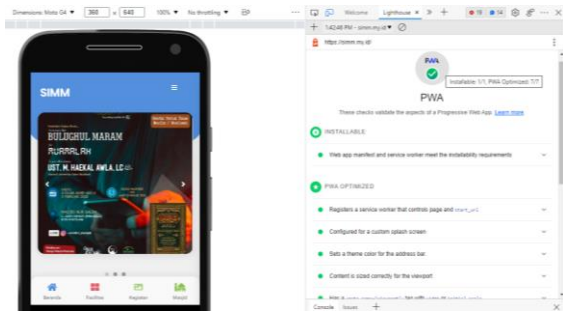


Gambar 14 Tampilan *Homescreen* Aplikasi Berhasil Pada *Smartphones*

8. Pengujian Teknologi PWA (*Progressive Web Application*)

Setelah didapatkan hasil pengujian menggunakan *lighthouse* sesudah menambahkan *app manifest* serta *service worker*. Sistem dapat dikatakan sebagai aplikasi berbasis PWA dikarenakan sudah memenuhi

berbagai kriteria dari teknologi *progressive web app* itu sendiri yaitu *installable* mencapai angka 1/1 dan *optimized* mencapai angka 7/7. Misalnya sistem belum memenuhi beberapa point kriteria audit pada *lighthouse* atau masih memiliki *point* yang berwarna merah, sehingga perlu dilakukan perbaikan demi pengembangan aplikasi.



Gambar 15. Hasil Pengujian PWA menggunakan *Lighthouse*

Berikut merupakan pemeriksaan dari validasi aspek PWA berdasarkan beberapa indikator pengujian *lighthouse*:

Tabel 1 Indikator Pengujian *Lighthouse*

<p>(<i>Installable</i>) DAPAT DIINSTAL</p>	<p>1. Pekerja layanan dan aplikasi web manifes memenuhi persyaratan kemampuan instal</p> <p><i>Service worker</i> merupakan teknologi yang memungkinkan aplikasi untuk menggunakan banyak fitur PWA, seperti <i>offline</i>, menambah layar beranda, dan <i>push notifikasi</i>. Dengan pekerja layanan yang tepat dan implementasi manifes, browser dapat secara proaktif meminta pengguna untuk menambahkan aplikasi pada layar beranda mereka, yang dapat menghasilkan interaksi yang lebih tinggi.</p>
<p>(<i>Optimized</i>) DIOPTIMALKAN</p>	<p>1. Mendaftarkan <i>service worker</i> yang mengontrol</p>

	<p>halaman dan <code>start_url</code></p> <p><i>Service worker</i> merupakan teknologi yang memungkinkan aplikasi Anda menggunakan banyak fitur PWA, seperti <i>offline</i>, menambah layar utama, dan <i>push notification</i>.</p> <p>2. Dikonfigurasi untuk layar splash khusus</p> <p>Layar pembuka bertema memastikan pengalaman pengguna berkualitas tinggi pada saat menjalankan aplikasi pada layar beranda mereka.</p> <p>3. Menetapkan warna tema untuk bilah alamat.</p> <p>Bilah alamat browser dapat bertema agar sesuai dengan situs Anda</p> <p>4. Konten berukuran benar untuk area pandang</p> <p>Apabila lebar konten aplikasi tidak sesuai dengan lebar area pandang, aplikasi mungkin tidak dioptimalkan untuk layar <i>smartphones</i>.</p> <p>5. Memiliki tag <code><meta name="viewport"></code> dengan lebar atau skala awal</p> <p><code><meta name="viewport"></code> tidak hanya mengoptimalkan aplikasi untuk ukuran layar <i>smartphones</i>, tetapi juga mencegah penundaan 300</p>
--	---

	<p>milidetik untuk input pengguna.</p> <p>6. Menyediakan ikon sentuh-apel yang valid Demi standar tampilan pada iOS saat pengguna menambahkan aplikasi web progresif ke layar beranda, penambahan `apple-touch-icon`. Harus mengarah ke PNG persegi 192px (atau 180px) yang tidak transparan.</p> <p>7. Manifes memiliki ikon yang dapat ditutup-tutupi Ikon yang dapat ditutup-tutupi memastikan bahwa gambar memenuhi seluruh bentuk tanpa dikotak-kotakkan saat memasang aplikasi di perangkat.</p>
--	--

5. KESIMPULAN

Berdasarkan hasil penelitian sistem informasi manajemen masjid bahwa diperoleh beberapa kesimpulan yaitu:

1. Berdasarkan hasil penelitian penulis dapat menyimpulkan bahwa dengan di banggunya Sistem Informasi Manajemen Masjid (SIMM) di Kota Pontianak dapat membantu proses manajemen masjid serta memberikan informasi dan layanan berbasis aplikasi yang dapat mempermudah para takmir untuk melakukan fungsi manajemen dengan baik dan modern. Melalui penggunaan teknologi Progressive Web Apps (PWA) dengan penerapan prinsip-prinsip dari PWA itu sendiri seperti, sistem dapat berjalan lebih cepat maupun mudah meminta data melalui fetch API serta dapat berjalan pada jaringan yang tidak stabil maupun offline sekalipun karena adanya Service Worker. Kemudian dengan bantuan app manifest membuat aplikasi dapat menghilangkan proses download melalui app store dengan

langsung bisa dipasang pada homescreen secara langsung, membuat sistem menjadi lebih ramah terhadap penyimpanan pada smartphones pengguna dan memberikan web progressif layaknya mobile native agar mudah untuk digunakan dan dipahami.

2. Dari hasil pengujian teknologi PWA menggunakan *tools lighthouse* pada chrome, dimana sistem sudah dapat dikatakan sebagai aplikasi berbasis PWA dikarenakan sudah memenuhi berbagai kriteria dari teknologi progressive web app itu sendiri yaitu installable mencapai angka 1/1 dan optimized mencapai angka 7/7. Misalnya sistem belum memenuhi beberapa point kriteria audit pada lighthouse atau masih memiliki point yang berwarna merah, sehingga perlu dilakukan perbaikan demi pengembangan aplikasi.
3. Dari hasil pengujian fungsional sistem yang dilakukan kepada pengurus masjid di Kota Pontianak, sistem yang telah dibangun bisa berjalan sesuai dengan fungsi yang telah dirancang, sistem juga dapat menunjukkan tranparansi dalam memberikan informasi terkait pengelolaan masjid termasuk kegiatan, keuangan, fasilitas, kepengurusan, inventaris dan pelaporan. Sistem yang di bangun juga mempunyai kekurangan seperti terdapat pengguna yang masih sulit memahami fungsi dari sistem. Sedangkan pada pengujian antarmuka sistem kepada 47 responden diperoleh persentase 86% dan masuk kedalam kategori baik sekali.

6. SARAN

Adapun saran untuk penelitian yang akan datang adalah sebagai berikut:

1. Diharapkan penelitian selanjutnya dapat menambahkan TWA (*Trusted Web Activity*) pada sistem informasi manajemen masjid agar dapat didistribusikan melalui *Google Play Store*.
2. Pada penelitian ini masih memiliki berbagai kekurangan pada antarmuka sistem seperti masih terdapat pengguna yang belum memahami fungsi sistem serta kemudahan dalam mengenali ikon-ikon pada sistem. Kedepanya peneliti diharapkan bisa menciptakan sebuah tampilan pengguna yang lebih mudah dimengerti atau *friendly*.

DAFTAR PUSTAKA

- [1] H. Santoso, *Membangun Aplikasi Mobile dengan PWA (Progressive Web Apps)*, Yogyakarta : Lokomedia, 2019.
- [2] A. Aziz, "Rancang Bangun Sistem Informasi Manajemen Zakat, Infak dan Sedekah Masjid Berbasis PWA Menggunakan Metode Extreme Programming," Skripsi, Institutional Repository UIN Sunan Kalijaga Yogyakarta, 2020.
- [3] H. A. Fatta, *Analisis & Perancangan Sistem Informasi untuk Keunggulan Bersaing Perusahaan & Organisasi Modern*, Yogyakarta: Andi, 2007.
- [4] A. Pellu, "Manajemen Pemberdayaan Ekonomi Masyarakat Berbasis Masjid (Studi Kasus Pada Masjid Jogokariyan Yogyakarta)," Tesis, UIN SUNAN AMPEL, 2020.
- [5] Dirjen Bimas Islam, Nomor 52 Tahun 2018 (2018, Januari 3). *Petunjuk Pelaksanaan Pengelolaan Manajemen Masjid Berbasis Sistem Informasi Masjid*. [Online].
- [6] M. I. Perkasa and E. B. Setiawan, "Pembangunan Web Service Data Masyarakat Menggunakan REST API dengan Access Token," *ULTIMA Computing*, vol. 10, no.1, pp. 19-26, 2018.
- [7] S. Dharwiyanti, and R. S. Wahono, *Pengantar Unified Modeling Language (UML)*. Kuliah Umum IlmuKomputer.Com, 2003.
- [8] E. Kurniawan, "Implementasi Rest Web Service Untuk Sales Order Dan Sales Tracking Berbasis Mobile," *Jurnal EKSIS*, vol. 7, no. 1, pp. 1-12, 2014.