

# Aplikasi Asesmen Anak Berkebutuhan Khusus di SLB Rafaha Arjasari Menggunakan Progressive Web App

Nawan Tutu Syah Lampah<sup>1</sup>, Eko Budi Setiawan<sup>2</sup>

Program Studi Teknik Informatika, Universitas Komputer Indonesia, Jl. Dipatiukur 112 Bandung  
nawantutu@gmail.com  
eko@email.unikom.ac.id

Diterima 14 November 2018

Disetujui 21 Desember 2018

**Abstract**— Rafaha Arjasari Special School is one of the schools that provide special education for children with special needs. Learning will not run optimally and will not be in accordance with the needs of students if previously there was no process of extracting the needs, potential, and challenges needed by students with special needs. This research resolves the problems faced by teachers when the assessment process is for children with special needs. This assessment applications built in the form of web-based applications using the Progressive Web App (PWA) technology to facilitate teachers in conducting the assessments process. This application provides features for assessing visual perceptions and perceived perceptions by utilizing the flash and vibrate features on the android smartphone. The conclusion of this research is that the application built can help the school in the assessment process to be more effective and efficient because the application is still running well, although not connected to the internet.

**Index Terms**— Assessment, Children with Special Needs, Progressive Web App, Android

## I. PENDAHULUAN

Penelitian ini dilakukan di sekolah luar biasa yang befokus untuk mendidik anak berkebutuhan khusus (ABK). Pendidikan tidak akan berjalan dengan optimal apabila pendidik tidak mengetahui hambatan, risiko, dan potensi yang dimiliki oleh setiap siswa. Berdasarkan wawancara dengan Ibu Hj. Djuang Fitriani, M.Pd. sebagai kepala sekolah SLB Rafaha Arjasari Pada bulan Februari 2018, asesmen adalah proses yang mutlak dilakukan agar pendidik dapat mengetahui informasi-informasi yang diperlukan dari setiap siswa.

Anak yang dikategorikan berkebutuhan dalam aspek fisik meliputi kelainan dalam indra penglihatan (tunanetra) kelainan indra pendengaran (tunarungu) kelainan kemampuan berbicara (tunawicara) dan kelainan fungsi anggota tubuh (tunadaksa) [1].

Kekhususan yang mereka miliki menjadikan ABK memerlukan pendidikan dan layanan khusus untuk mengoptimalkan potensi dalam diri mereka secara sempurna [2]. Jumlah anak berkebutuhan khusus di

Indonesia dari tahun ke tahun terus meningkat. Persatuan Bangsa-Bangsa (PBB) memperkirakan bahwa paling sedikit ada 10 persen anak usia sekolah yang memiliki kebutuhan khusus. Di Indonesia, jumlah anak usia sekolah, yaitu 5 - 14 tahun, ada sebanyak 42,8 juta jiwa. Jika mengikuti perkiraan tersebut, maka diperkirakan ada kurang lebih 4,2 juta anak Indonesia yang berkebutuhan khusus [3].

Beberapa ahli juga menyebut anak berkebutuhan khusus adalah mereka yang memiliki disabilitas intelektual dan perkembangan, disabilitas fisik, disabilitas sensoris (indera), hambatan/masalah perilaku, kesulitan belajar, serta cerdas dan bakat [4].

Sejalan dengan pengertian diatas, SLB Rafaha menerima siswa dan memberikan layanan pendidikan bagi anak yang memiliki hambatan fisik, emosi, mental, sosial dan memiliki kecerdasan serta bakat khusus. Sekolah ini bertujuan untuk mengoptimalkan kemampuan yang dimiliki seorang anak berkebutuhan khusus serta untuk memenuhi tujuan pendidikan nasional.

Asesmen merupakan proses mengumpulkan, menganalisis dan menginterpretasikan data atau informasi tentang peserta didik dan lingkungannya [5]. Hal tersebut dilakukan untuk mendapat gambaran berbagai kondisi individu dan lingkungannya sebagai dasar pengembangan program pendidikan dan layanan khusus.

Proses asesmen yang dilakukan oleh SLB Rafaha Arjasari dilakukan di sekolah ataupun langsung berkunjung ke tempat tinggal siswa dengan membawa berkas yang banyak berisi butir-butir instrumen asesmen tiap-tiap aspek dan sejumlah anak yang akan di uji. Wawancara kemudian dilakukan kepada keluarga siswa disertai observasi terkait kondisi calon siswa. Permasalahan muncul ketika data yang diperoleh hilang dikarenakan berkas asesmen hilang atau tercecer. Permasalahan kedua yang muncul adalah asesmen sering kali dilakukan dalam waktu dan dalam kondisi yang tidak terduga, tidak hanya selesai dalam satu waktu saja. Ketiga, kondisi internet di sekitar

sekolah SLB Rafaha Arjasari memiliki kondisi yang kurang stabil, bahkan dalam suatu waktu bisa *offline* sehingga tidak bisa terhubung dengan internet.

Oleh karena itu diperlukan sebuah aplikasi berbasis web yang memiliki performa yang baik meskipun dalam kondisi internet yang buruk masih dapat mengakses serta memanfaatkan fitur-fitur dalam *smartphone* yang digunakan untuk asesment. Salah satu aplikasi berbasis web yang dapat diakses melalui *smartphone* dan mampu berjalan dalam kondisi jaringan yang buruk sekalipun adalah *Progressive Web App* (PWA). PWA adalah *Mobile Web* yang menggunakan teknologi-teknologi terbaru guna menghasilkan *mobile web* yang memiliki pengalaman pengguna yang jauh lebih baik dari *Mobile Web* tradisional [6].

Berdasarkan permasalahan dan pengamatan dalam proses asesmen manual yang dilakukan di SLB Rafaha Arjasari maka diperlukan aplikasi asesmen berbasis web sebagai solusi dari permasalahan dan pemanfaatan fasilitas yang ada. Aplikasi tersebut masih dapat berjalan meskipun dalam kondisi internet yang tidak stabil atau bahkan ketika *offline* sekalipun.

## II. TINJAUAN PUSTAKA

### A. Asesmen

Istilah asesmen dalam pendidikan khusus memiliki makna yang berbeda dengan asesmen yang digunakan secara umum dalam dunia pendidikan. Pada umumnya orang sering menterjemahkan istilah asesmen sebagai penilaian, padahal sesungguhnya terjemahan itu tidak cocok, sebab asesmen dalam pendidikan khusus memiliki pengertian yang khas. Sejalan dengan hal ini, Sari mengartikan asesmen sebagai proses mengumpulkan, menganalisis dan menginterpretasikan data atau informasi tentang peserta didik dan lingkungannya [5].

Asesmen dalam pendidikan khusus dapat dikelompokkan menjadi dua kategori yaitu asesmen berazaskan kurikulum (asesmen akademik), dan asesmen berazaskan perkembangan (asesmen non-akademik). Sejalan dengan ruang lingkup asesmen, asesmen diartikan sebagai suatu proses untuk mengambil keputusan dengan menggunakan informasi yang diperoleh melalui pengukuran hasil belajar yaitu menggunakan instrumen test maupun non tes [7].

Sementara itu Hartati [8] menyebutkan bahwa asesmen perkembangan anak merupakan suatu proses yang dilakukan secara sistematis dalam usaha untuk memperoleh informasi tentang kemampuan dan perkembangan anak. Hasil asesmen akan menjadi laporan atau informasi bagi guru dan orangtua dalam merancang program dan kebutuhan yang sesuai dengan pencapaian [8].

Berdasarkan pengertian diatas, maka dapat disimpulkan bahwa asesmen adalah upaya untuk mengetahui kemampuan-kemampuan yang dimiliki, hambatan atau kesulitan yang dialami, mengetahui

latar belakang mengapa hambatan atau kesulitan itu muncul dan untuk mengetahui bantuan apa yang dibutuhkan oleh yang bersangkutan. Berdasarkan data hasil asesmen tersebut dapat dibuat program pembelajaran yang tepat bagi anak itu.

### B. Progressive Web App

*Progressive Web Apps* (PWA) didefinisikan sebagai serangkaian konsep dan kata kunci seperti progresif, responsif, konektivitas yang independen, menyerupai aplikasi *native*, segar, aman, mudah ditemukan, dapat direkayasa ulang, dapat dipasang dan ditautkan [9].

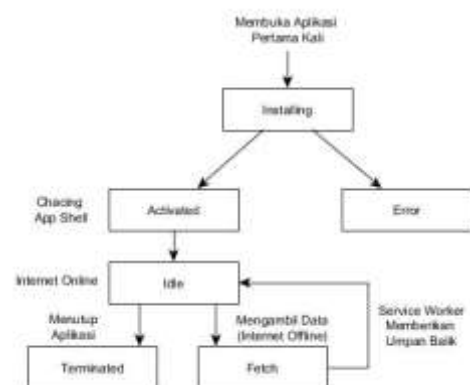
PWA berguna untuk pengguna sejak pertama membuka halaman sebuah web dengan konsep PWA, dan seiring dengan pengguna menggunakan aplikasi web lebih banyak lagi, aplikasi akan menjadi semakin *powerful*. Aplikasi dapat dimuat dengan cepat, bahkan dalam kondisi internet yang kurang baik sekalipun, dapat mengirim *push notifications*, mempunyai ikon aplikasi di *home screen*, dan bisa berjalan dalam mode layar penuh [10].

Adapun beberapa teknologi yang ada dalam *progressive web app* adalah sebagai berikut:

#### 1) Service Worker

*Service worker* adalah salah satu jenis dari web *worker*, yaitu script yang berjalan di belakang browser pengguna. *Service worker* pada dasarnya adalah berkas *JavaScript* yang berjalan pada *thread* yang berbeda dengan *main thread browser*, menangani *network request*, *caching*, mengembalikan *resource* dari *cache*, dan bisa mengirimkan *push message* [10].

*Service worker* sudah mulai berjalan ketika web pertama kali dimuat secara otomatis, dengan kata lain pengguna tidak perlu melakukan aksi untuk menyalakan *service worker*. Penggunaan memori pada *service worker* juga tidak terlalu besar karena *service worker* akan langsung mati apabila sedang tidak digunakan. Gambar 1 berikut merupakan daur hidup (*life cycle*) *service worker*:



Gambar 1. *Life cycle* Service worker

#### 2) Web App Manifest

*Web app manifest* adalah salah satu dari kumpulan teknologi dalam *progressive web app*. *Web app*

*manifest* berbentuk file JSON sederhana yang berfungsi sebagai penyedia informasi aplikasi web. Informasi yang disediakan seperti nama aplikasi, pemilik, ikon, dan deskripsi aplikasi web, hal ini bertujuan agar aplikasi web yang dibuat dapat memberikan fitur *Add To Homescreen*.

### 3) *IndexedDB*

*IndexedDB* adalah media penyimpanan lokal *non-sql* yang ada pada browser. *IndexedDB* memungkinkan kita menyimpan komponen-komponen yang ada pada aplikasi pada saat jaringan internet dalam kondisi *offline*, yang mana hal ini sangat bagus untuk memastikan setiap *request* yang dikirim dapat tersimpan didalam *database* aplikasi.

### 4) *BackgroundSync*

*BackgroundSync* adalah API baru yang memungkinkan aplikasi menunda tindakan hingga pengguna memiliki jaringan internet yang stabil. Hal ini berguna untuk memastikan apapun yang dikirim oleh pengguna telah berhasil terkirim.

### C. *React.js*

*React.js* adalah *UI library* yang dikembangkan oleh Facebook untuk memfasilitasi pengembang web interaktif, *stateful*, dan komponen *UI* yang mudah digunakan. *React.js* sudah digunakan pada *website* Facebook dibagian produksi. *React.js* juga merupakan *library* terbaik untuk melakukan *render user interface* yang kompleks dengan performa yang tinggi. Konsep dasar pada *React.js* adalah ada pada *Virtual DOM*.

*React.js* lebih efektif menggunakan *Virtual DOM*, yang mana dapat melakukan *render* pada bagian *client* ataupun bagian server [11].

### D. *Application Programming Interface (API)*

API adalah aplikasi pemrograman yang secara khusus dikembangkan untuk digunakan sebagai perantara komunikasi antara komponen-komponen perangkat lunak. Biasanya hasil *output* dari API dapat berupa data XML ataupun JSON, tergantung dari situs mana yang menyediakan API tersebut [12] [13].

### E. *Lighthouse*

Pada awal tahun 2018, Google merilis aplikasi *SEO* yang bernama *Lighthouse*. *Lighthouse* yang menawarkan bagaimana cara meningkatkan kualitas halaman web. *Lighthouse* adalah aplikasi *open source* yang dibangun oleh Google untuk menguji *Progressive Web App*, *Performance*, *Accessibility*, *Best Practice*, dan *SEO* [10].

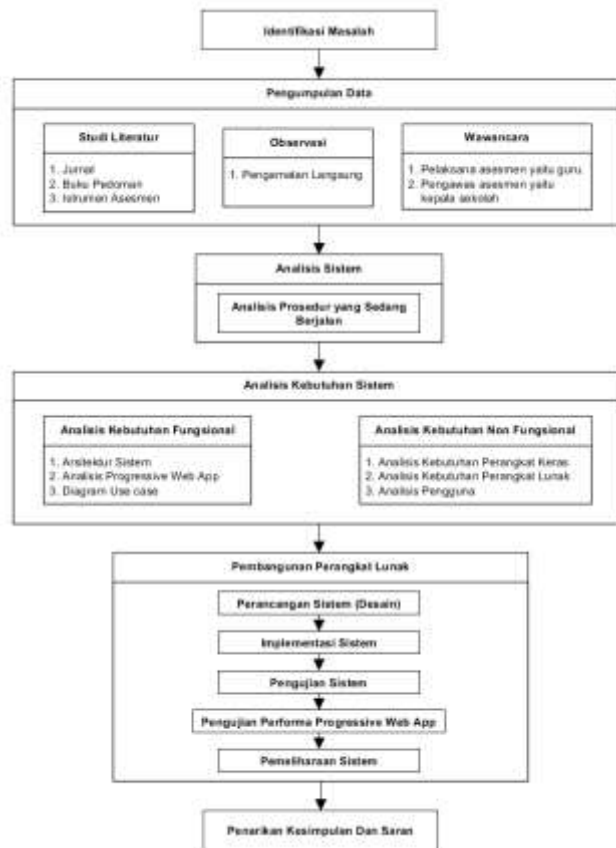
## III. METODOLOGI PENELITIAN

Adapun metodologi yang digunakan dalam penelitian ini menggunakan dua metode, yaitu metode pengumpulan data dan metode pembangunan perangkat lunak dengan model *waterfall*. Metode *waterfall* merupakan model pengembangan sistem informasi yang sistematis dan sekuensial [14].

Tahapan lengkap pada penelitian ini dapat dilihat pada gambar 2.



UMN



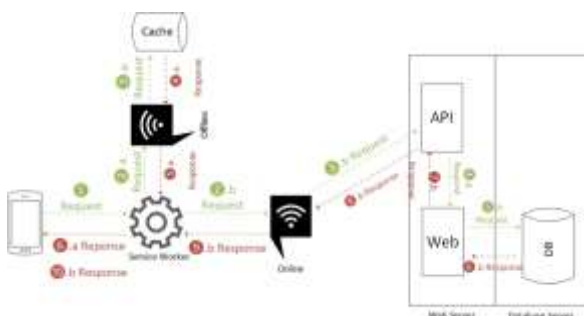
Gambar 2. Tahapan Penelitian

IV. HASIL DAN PEMBAHASAN

Pada tahap ini terdiri dari tahapan penelitian yang dilakukan. Tahapan tersebut terdiri dari analisis dan perancangan sistem, analisis teknologi, implementasi, dan pengujian sistem.

A. Analisis dan Perancangan Sistem

Aplikasi yang akan dibangun yaitu aplikasi asesmen bagi anak berkebutuhan khusus dengan menggunakan teknologi *progressive web app*. Berikut adalah arsitektur aplikasi yang akan dibangun dapat dilihat pada gambar 3.



Gambar 3. Arsitektur Sistem

Dari gambar 3 pada saat pengiriman *request* maka *request* akan dilanjutkan kepada *service worker*, setelah itu *service worker* akan melakukan pengecekan apakah internet dalam keadaan *online* atau *offline*. Jika *offline* maka *request* akan dikirimkan kepada penyimpanan *cache*, dari penyimpanan *cache* akan

dikirim *response* yang nantinya akan diterima oleh pengguna. Sedangkan jika internet *online* maka *request* akan dilanjutkan kepada database server melalui API yang selanjutnya database akan mengembalikan *response* yang ditujukan untuk pengguna.

B. Analisis Teknologi yang Digunakan

Analisis teknologi adalah proses menganalisa cara kerja teknologi-teknologi yang digunakan pada aplikasi ini. Pada tahap ini akan dijelaskan bagaimana teknologi-teknologi apa saja yang ada pada *progressive web app*.

*Service worker* adalah *script* yang berjalan di belakang browser pengguna. *Service worker* tidak membutuhkan sebuah halaman ataupun interaksi dari pengguna untuk menjalankan tugasnya, dengan begitu *service worker* akan terus berjalan walaupun halaman webnya tidak terbuka. *Service worker* memiliki beberapa *event* untuk menangani setiap kondisi pada *service worker*. Salah satunya adalah *event fetch* yang memungkinkan *service worker* dapat tetap menjalankan setiap *request* yang dikirim oleh pengguna walaupun dalam kondisi internet yang sedang *offline*. Arsitektur *lifecycle* dan penjelasan setiap *event* pada *service worker* dapat diilustrasikan seperti pada Gambar 1.

Berikut adalah deskripsi dari gambar *lifecycle service wrker* pada sistem perangkat lunak:

a) Installing

Proses ini adalah proses pendaftaran *service worker* pada browser agar browser dapat mengenali *service worker* yang digunakan oleh aplikasi. Disamping itu *event install* adalah *event* yang berfungsi untuk menyimpan nama file/url yang akan di simpan didalam penyimpanan *cache*.

b) *Activated*

*Event* ini dijalankan apabila proses pendaftaran *service worker* pada browser sudah berhasil. *Activated* berarti *service worker* telah aktif dan berjalan di aplikasi. Disamping itu kita dapat menyimpan beberapa kondisi pada *event* ini. Berikut adalah *source code* yang ada pada *event activated*.

c) *Error*

*Event* ini adalah *event* yang menunjukkan kesalahan. *Event* ini berjalan apabila ada kesalahan pada pendaftaran *service worker* di browser.

d) *Idle*

*Event* ini adalah *event* yang menandakan bahwa *service worker* dalam keadaan menunggu.

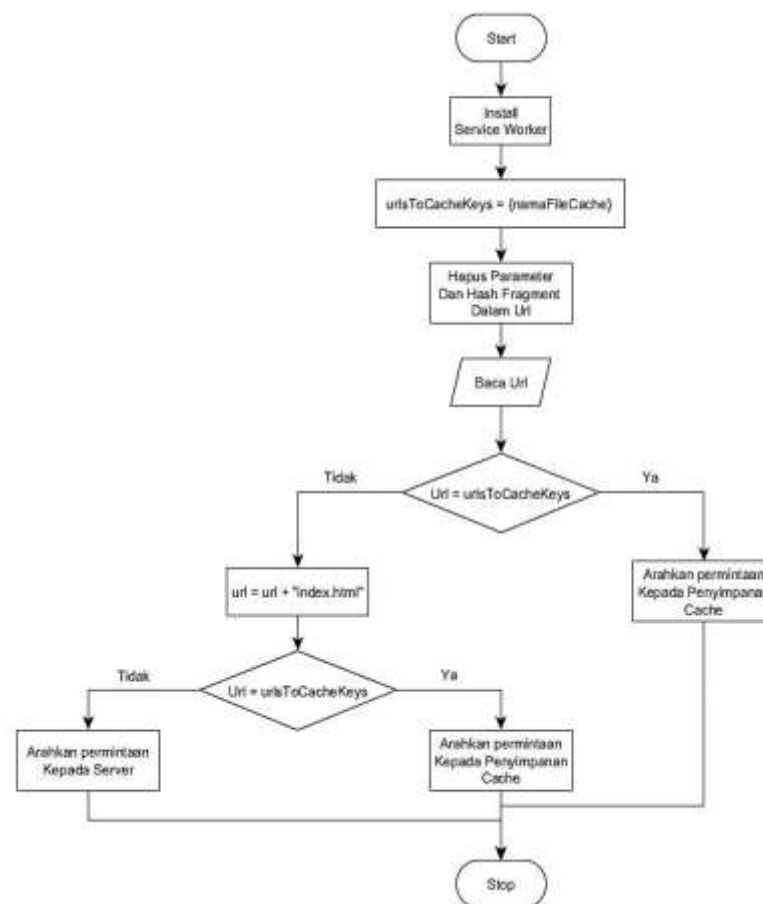
e) *Terminated*

*Event* ini berjalan ketika aplikasi ditutup. *Event* biasanya berfungsi untuk menghapus *service worker* yang telah didaftarkan pada *browser* sebelumnya.

f) *Fetching*

*Event* ini bekerja pada saat kondisi internet sedang mati atau *offline*. Pada umumnya *event* ini mengarahkan semua request pada *cache*.

Gambar 4 berikut merupakan *flowchart* dari *event fetch* pada *service worker*.



Gambar 4. *Flowchart event fetch service worker*

Setelah *service worker* berhasil diinstall maka setiap pengguna membuka halaman atau memuat ulang halaman, *service worker* akan mulai menjalankan *event fetch*. Variabel *urlsToCacheKeys* adalah variabel yang menampung nama file atau url yang akan kita masukan ke dalam *cache*. Setelah itu *service worker* akan memeriksa apakah url tersebut ada di penyimpanan *cache*. Jika ada maka *service worker* akan mengambil data dari penyimpanan *cache*. Tetapi

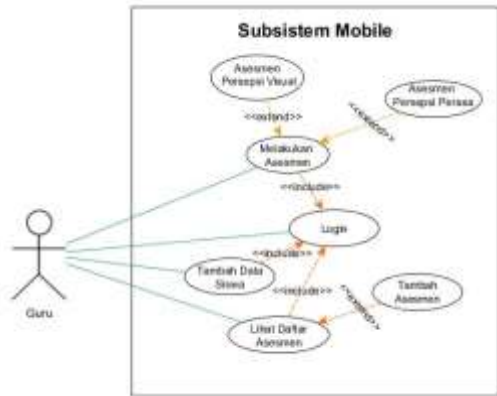
jika tidak maka *service worker* akan memeriksa penyimpanan *cache* lagi dengan menambahkan kata *index.html* pada url yang akan dicari. Jika masih tidak ada maka *service worker* akan melanjutkan permintaan kepada server.

### C. Use Case Diagram

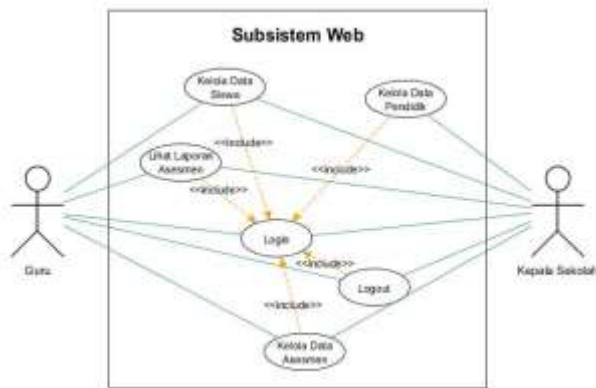
Diagram *usecase* pada aplikasi *mobile* yang dibangun terdiri dari tujuh fungsional, yaitu login, melakukan asesmen, tambah data siswa, lihat daftar

asesmen, asesmen persepsi visual, asesmen persepsi perasa dan tambah asesmen. Diagram *usecase* pada aplikasi *web* yang dibangun terdiri dari enam fungsional, yaitu login, lihat laporan asesmen, logout, kelola data asesmen, kelola data siswa, dan kelola data pendidik. Sedangkan *actor* yang terlibat yaitu guru dan kepala sekolah.

*Use case* diagram mobile dapat dilihat pada gambar 5, sedangkan untuk web *use case* diagramnya dapat dilihat pada gambar 6.



Gambar 5. *Use case* Aplikasi *Mobile*



Gambar 6. *Use Case* Aplikasi Web

**D. Implementasi Sistem**

Implementasi sistem merupakan tahap penterjemah perancangan berdasarkan hasil analisis ke dalam suatu bahasa pemrograman tertentu serta penerapan perangkat lunak yang dibangun pada lingkungan yang sesungguhnya.

Berikut adalah perangkat keras aplikasi *mobile* dan web yang dapat dilihat pada tabel 1.

Tabel 1. Implementasi Perangkat Keras yang Direkomendasikan

Nama Perangkat Keras	Spesifikasi
RAM <i>Mobile</i>	1 GB
Memori Internal <i>Mobile</i>	8 GB
Kamera	5 MP
RAM PC	2 GB
Koneksi Internet	5 mbps
Processor	Core i3 – 2,3Ghz
VGA	512 MB

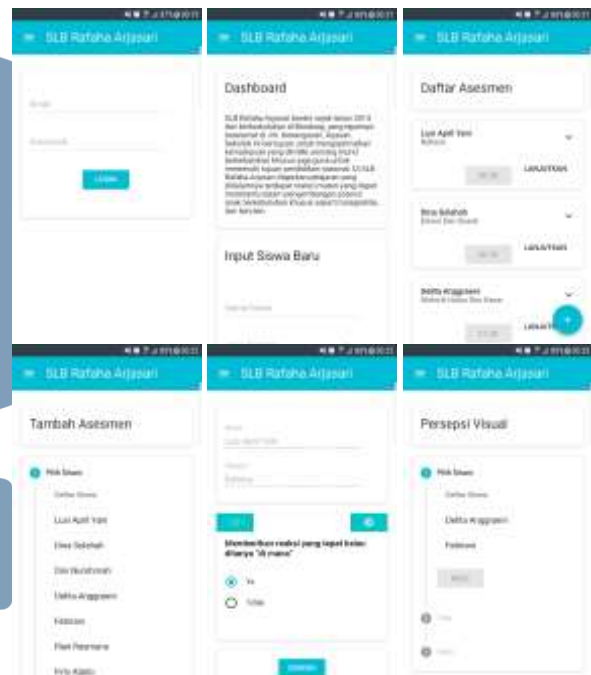
Nama Perangkat Keras	Spesifikasi
Harddisk	500 GB

Berikut adalah perangkat lunak aplikasi *mobile* dan web yang dapat dilihat pada Tabel 2.

Tabel 2. Implementasi Perangkat Lunak

No.	Jenis Perangkat Lunak	Spesifikasi
1.	Sistem Operasi <i>Mobile</i>	Android 4.1 Jelly Bean
2.	Perambah <i>Mobile</i>	Chrome v.64
3.	Sistem Operasi Web	Windows 7 32-bit
4.	Peramban Web	Google Chrome, Mozilla Firefox

Implementasi antarmuka berisi pemaparan setiap tampilan perangkat lunak yang dibangun sebagai layout dari masing-masing antarmuka. Berikut adalah tampilan implementasi antarmuka pada aplikasi *mobile* yang dapat dilihat pada gambar 7.



Gambar 7. Antarmuka Aplikasi *Mobile*

Dari gambar 7 antarmuka untuk *login* ditunjukkan pada gambar atas kiri. Pada halaman ini terdapat form yang harus diisi oleh pengguna agar dapat masuk kedalam aplikasi. Halaman *dashboard* ditunjukkan oleh gambar atas kedua, dimana dalam halaman ini terdapat informasi tentang sekolah dan juga terdapat fitur untuk menambah data siswa. Sedangkan untuk halaman asesmen terdapat pada gambar atas kanan. Halaman ini menampilkan daftar siswa yang sedang melaksanakan proses asesmen. Halaman tambah asesmen ditunjukkan pada posisi kiri bawah yang berfungsi untuk menambahkan siswa yang akan melakukan proses asesmen. Halaman detail asesmen berada pada posisi bawah tengah yang menampilkan soal-soal instrument asesmen yang akan ditanyakan kepada siswa. Terakhir merupakan halaman persepsi

visual pada posisi kanan bawah berfungsi untuk melakukan asesmen persepsi visual pada siswa.

### E. Pengujian Performance

Pada tahap ini akan dilakukan pengujian yang bertujuan untuk mengetahui waktu respon, penggunaan memori, dan penggunaan media penyimpanan pada aplikasi.

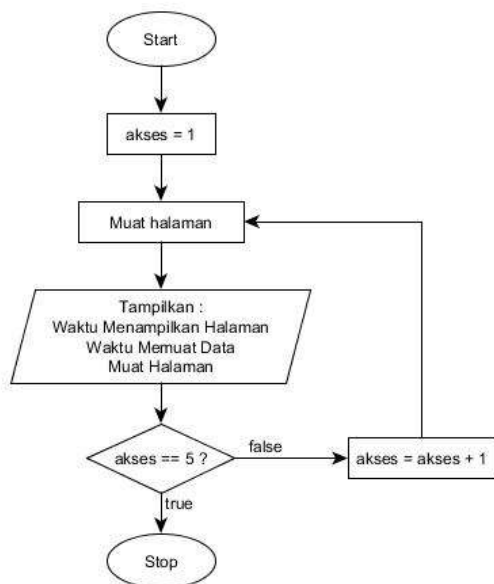
#### E.1. Variabel Pengujian

Terdapat tiga variabel pengujian *performance* pada penelitian ini, yaitu jenis akses, *profile* jaringan serta pengaksesan halaman. Variabel-variabel ini akan digunakan sebagai acuan dalam membuat skenario pengujian.

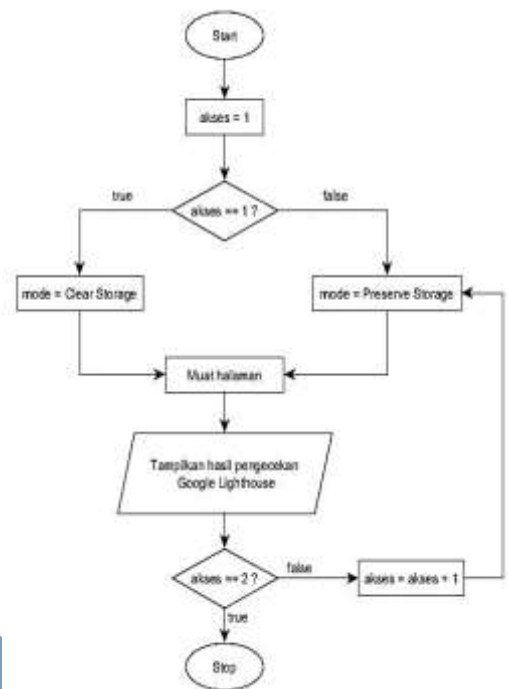
1. Jenis Akses : *Mobile*, Desktop
2. Profil Jaringan : Indonesia Avg. Speed (5,9 Mbps)
3. Pengaksesan Halaman : Pengaksesan ke-1, pengaksesan ke-2, ke-3, ke-4, ke-5.

#### E.2. Skenario Pengujian

Pada penelitian ini terdapat 2 skenario untuk menjalankan pengujian *performance*. Skenario yang pertama adalah variabel pengujian berupa tipe aplikasi *mobile*, sedangkan skenario kedua berupa tipe aplikasi desktop. Kedua skenario kecepatan internetnya dibatasi menjadi maksimal 5,9 Mbps. Kedua skenario pengujian dapat dilihat pada gambar 8 dan gambar 9.



Gambar 8. Diagram alir pengujian skenario 1



Gambar 9. Diagram alir pengujian skenario 1

Pada skenario yang dijelaskan di gambar 9 memerlukan bantuan aplikasi untuk melakukan penilaian performa, aplikasi yang digunakan adalah Google Lighthouse dengan pengecekan pertama menggunakan *mode clear storage* dan pengecekan kedua dengan menggunakan *preserve storage*.

#### E.3. Hasil Pengujian

Setelah pengujian dilakukan, data-data hasil pengujian dijabarkan berdasarkan skenario pengujian yang dilakukan. Terdapat dua satuan nilai pada hasil pengujian penelitian ini, yaitu satuan waktu yang dijabarkan dalam milidetik (mdtk) dan satuan ukuran berkas yang dijabarkan dalam ukuran *KiloByte* (KB). Terdapat beberapa nama variabel pengujian dan singkatannya yang digunakan pada tabel hasil pengujian yaitu # untuk nomor urut pengaksesan halaman, WMH untuk Waktu Menampilkan Halaman, WMD untuk Waktu Memuat Data dan MD untuk Muat Data.

Berikut adalah hasil pengujian pada skenario 1 yang dapat dilihat pada tabel 3.

Tabel 3. Tabel Hasil Pengujian Skenario 1

#	WMH (mdtk)	WMD (mdtk)	MD (KB)
1	3090	3530	165
2	1230	1930	0,8
3	1250	1890	0,8
4	1220	1600	0,8
5	1140	1570	0,8

Pada langkah #1 adalah pengujian aplikasi yang diakses pertama kali tanpa peranan dari *service worker*, dan *cache* halaman belum tersedia. Sedangkan pada

langkah #2 sampai #5 *service worker* sudah berjalan dan *cache* halaman sudah tersedia.

Pada skenario ini dilakukan dua pengecekan, yang mana pengujian pertama menggunakan *mode Clear Storage* dengan maksud agar semua *request* yang dilakukan oleh aplikasi dikirim ke server tanpa ada peranan *service worker*. Sedangkan pengujian kedua akan menggunakan *mode Preserve Storage*. Mode ini adalah mode yang memanfaatkan penyimpanan aplikasi dengan ini *service worker* dapat mengatur setiap *request* yang harus dikirim ke server atau penyimpanan *cache*.

Pada pengujian pertama nilai index *performance* yang didapatkan adalah 79 dengan total waktu pengaksesan halaman adalah 4,1 detik. Gambar 10 berikut adalah hasil penilaian menggunakan *Google Lighthouse* :



Gambar 10. Nilai Indeks Google Lighthouse Pengujian Pertama

Gambar 11 berikut adalah matriks yang menunjukkan aktivitas muat halaman yang terjadi pada waktu 4,1 detik tersebut :



Gambar 11. Matriks *Performance* Pengujian Pertama

Selanjutnya pada pengujian kedua terjadi peningkatan yang signifikan yang mana nilai indeks *performance* meningkat menjadi 100 dengan total waktu muat halaman selama 1.1 detik. Secara detail dapat dilihat pada gambar 12.



Gambar 12. Nilai Indeks Google Lighthouse Pengujian Kedua

Gambar 13 berikut adalah matriks aktifitas yang dilakukan aplikasi selama 1.1 detik :



Gambar 13. Matriks *Performance* Pengujian Kedua

Berdasarkan pengujian skenario 1 dan skenario 2 yang dilakukan dapat diartikan bahwa pengujian

berhasil dan *progressive web app* sangat cocok digunakan untuk aplikasi dengan frekuensi pengecekan berulang kali.

#### F. Pengujian Black-Box

Metode *black-box* adalah metode pengujian yang berfokus pada persyaratan fungsional perangkat lunak. Tujuan dari metode *black-box* ini adalah untuk menemukan kesalahan fungsi pada aplikasi. Pengujian dilakukan dengan mencoba semua kemungkinan yang terjadi secara berulang-ulang, jika dalam pengujian ditemukan kesalahan maka akan dilakukan penelusuran untuk memperbaiki kesalahan yang terjadi. Pengujian fungsional subsistem web dapat dilihat pada tabel 4.

Tabel 4. Hasil Pengujian Subsistem Web

Item Uji	Detail Pengujian	Hasil
Login	Memasukan email dan password yang benar	(✓)
	Memasukan email dan password yang salah	Berhasil
Kelola Data Asesmen	Menambah data soal asesmen	(✓)
	Mengubah data soal asesmen	
	Menghapus data soal asesmen	
	Menambah data kategori	(✓)
	Mengubah data kategori	
	Menghapus data kategori	
Kelola Data Siswa	Menambah data siswa	(✓)
	Mengubah data siswa	
	Menghapus data siswa	
Kelola Data Pendidik	Menambah data pendidik	(✓)
	Mengubah data pendidik	
	Menghapus data pendidik	
Lihat Laporan Asesmen	Menampilkan laporan asesmen	(✓)
Logout	Menekan tombol logout pada aplikasi.	(✓)

Sedangkan hasil pengujian subsistem *mobile* dapat dilihat pada tabel 5.

Tabel 5. Hasil Pengujian Subsistem *Mobile*

Item Uji	Detail Pengujian	Hasil
Melakukan Asesmen	Menampilkan daftar siswa	(✓)
	Memilih dan menyimpan jawaban asesmen	
Asesmen Persepsi Visual	Menampilkan daftar siswa	(✓)
	Menampilkan kamera	
	Menyalakan flash kamera	



Item Uji	Detail Pengujian	Hasil
Asesmen Persepsi Perasa	Menampilkan daftar siswa	(✓)
	Menyalakan getar <i>smartphone</i>	Berhasil
Tambah Data Siswa	Menambahkan data siswa	(✓) Berhasil
Lihat Daftar Asesmen	Menampilkan daftar asesmen	(✓) Berhasil
Tambah Asesmen	Menambahkan data asesmen	(✓) Berhasil
Login	Memasukkan email dan password yang benar	(✓)
	Memasukkan email dan password yang salah	Berhasil

Berdasarkan hasil pengujian *black-box* yang dilakukan, semua fungsional sistem sudah berhasil berjalan seperti yang diharapkan.

#### G. Pengujian Beta

Pengujian beta merupakan pengujian yang dilakukan secara objektif dan dilakukan secara langsung oleh pengguna yang nantinya akan menggunakan aplikasi asesmen bagi anak berkebutuhan khusus ini. Jumlah responden pada pengujian ini berjumlah 13 orang berdasarkan jumlah guru yang ada di SLB Rafaha Arjasari. Pengujian ini menggunakan kuesioner dengan skala likert yang hasilnya dapat dilihat pada tabel 6.

Tabel 6. Hasil Pengujian Kuesioner

Pertanyaan Pertama: Apakah Anda setuju bahwa perangkat lunak ini dapat mempermudah Anda dalam melakukan proses asesmen?				
SS	S	CS	TS	STS
3	10	0	0	0
Rata – Rata = $(15 + 40 + 0 + 0 + 0) / 65 \times 100\% = 84\%$				
Pertanyaan Kedua: Apakah Anda setuju bahwa pencatatan jawaban asesmen lebih efektif dengan menggunakan aplikasi ini?				
SS	S	CS	TS	STS
3	10	0	0	0
Rata – Rata = $(15 + 40 + 0 + 0 + 0) / 65 \times 100\% = 84\%$				
Pertanyaan Ketiga: Apakah Anda setuju bahwa aplikasi ini dapat memudahkan guru dalam menggunakan aplikasi dengan kondisi internet yang kurang stabil?				
SS	S	CS	TS	STS
2	9	1	0	0
Rata – Rata = $(10 + 36 + 3 + 0 + 0) / 65 \times 100\% = 75\%$				
Pertanyaan Keempat: Apakah Anda setuju bahwa aplikasi ini berhasil menggunakan fitur <i>smartphone</i> dalam melakukan asesmen persepsi visual dan asesmen persepsi perasa?				
SS	S	CS	TS	STS
1	5	7	0	0
Rata – Rata = $(5 + 20 + 21 + 0 + 0) / 65 \times 100\% = 70\%$				
Pertanyaan Kelima: Apakah Anda setuju bahwa aplikasi ini mudah digunakan?				

SS	S	CS	TS	STS
4	9	0	0	0
Rata – Rata = $(20 + 39 + 0 + 0 + 0) / 65 \times 100\% = 86\%$				
Rata-rata Akhir = $(84 + 84 + 75 + 70 + 86) / 7 = 79.8\%$				

Berdasarkan hasil pengujian beta pada tabel 6 yang dilakukan kepada guru-guru SLB Rafaha Arjasari selaku pengguna aplikasi pada bulan Juli 2018, dapat disimpulkan bahwa mayoritas responden menjawab setuju mengenai fungsi sistem asesmen bagi anak berkebutuhan khusus dalam mempermudah proses asesmen.

Berdasarkan hasil yang didapat dari semua tahapan pengujian yang dilakukan, maka didapatkan hasil sebagai berikut:

1. Fitur *flash* dan alat getar yang diimplementasikan pada aplikasi, dapat memudahkan ketika proses asesmen karena tidak perlu lagi membawa alat bantu yang lain.
2. Proses pencatatan hasil asesmen dapat dilakukan dimana saja dan kapanpun, ketika tidak ada akses internet sekalipun masih dapat dilakukan, sehingga proses asesmen menjadi lebih efisien.
3. *Progressive web app* dalam aplikasi ini dapat berjalan dengan baik sehingga memberikan kemudahan pengguna ketika menggunakan aplikasi

#### V. SIMPULAN

Adapun kesimpulan dari penelitian ini didapatkan bahwa aplikasi asesmen bagi anak berkebutuhan khusus yang dibahas dalam penelitian ini dapat memenuhi kebutuhan yang ada pada proses asesmen. Hal ini dikarenakan fitur asesmen persepsi visual dan persepsi perasa dapat membantu guru dalam melakukan asesmen. Selain itu aplikasi juga dapat berjalan pada kondisi internet yang tidak stabil atau *offline*.

#### DAFTAR PUSTAKA

- [1] A. Nandiyah, "Menenal Anak Berkebutuhan Khusus," *Magistra*, vol. 25, no. 86, p. 1, 2013.
- [2] L. R. Smiley, R. L. Taylor and S. B. Richards, *Exceptional Students (Preparing Teachers for the 21st Century) Second Edition*, New York: McGraw-Hill Education, 2015.
- [3] D. R. Desiningrum, *Psikologi Anak Berkebutuhan Khusus*, Yogyakarta: Psikosain, 2016.
- [4] S. Aziz, "Pendidikan Seks Bagi Anak Berkebutuhan Khusus," *Jurnal Kependidikan*, vol. 2, no. 2, pp. 182-204, 2014.
- [5] N. Sari, "Pola Pelaksanaan Bimbingan Dan Konseling Untuk Mengoptimalkan," *Jurnal Bimbingan dan Konseling Indonesia*, vol. 1, no. 1, pp. 31-35, 2016.
- [6] M. R. Ridho, A. Pinandito and R. K. Dewi, "Perbandingan Performa Progressive Web Apps dan Mobile Web Terkait," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 2, no. 10, pp. 3483-3491, 2018.
- [7] Wahyudi, "Assesment Pembelajaran Berbasis Portofolio di Sekolah," *Jurnal Visi Ilmu Pendidikan*, vol. 2, no. 1, pp. 288-296, 2010.

- [8] S. Hartati, "Pengembangan Model Asesmen Perkembangan," *Jurnal Pendidikan Anak Usia Dini*, vol. 1, no. 1, pp. 19-30, 2017.
- [9] A. Bjørn-Hansen, T. A. Majchrzak and T.-M. Grønli, "Progressive web apps: The possible web-native unifier for mobile development," *Proceedings of the 13th International Conference on Web Information Systems and Technologies*, vol. 1, no. 1, pp. 344-351, 2017.
- [10] L. Adi, R. J. Akbar and W. N. Khotimah, "Platform E-Learning untuk Pembelajaran Pemrograman Web Menggunakan Konsep Progressive Web Apps," *Jurnal Teknik ITS*, vol. 6, no. 2, pp. A579-A583, 2017.
- [11] A. Kumar and R. K. Singh, "Comparative Analysis of AngularJS and ReactJS," *International Journal of Latest Trends in Engineering and Technology*, vol. 7, no. 4, pp. 225-227, 2016.
- [12] L. O. M. Hasan, M. Sholeh and C. Iswahyudi, "Pemanfaatan Twitter API Untuk Mengakses Jadwal Bimbingan Dosen Pada IST AKPRIND Yogyakarta," *Jurnal Script*, vol. 3, no. 2, pp. 139-147, 2016.
- [13] M. Ichwan and F. Hakiky, "Pengukuran Kinerja Goodreads Application Programming Interface (Api) Pada Aplikasi Mobile Android (Studi Kasus Untuk Pencarian Data Buku)," *Jurnal Informatika*, vol. 2, no. 2, pp. 13-21, 2016.
- [14] R. S. Pressman, *Rekayasa perangkat lunak : pendekatan praktisi*; buku II, Yogyakarta: Yogyakarta : Andi, 2002.

