

SKRIPSI
PERANCANGAN DAN IMPLEMENTASI WEB SERVICE
PADA SISTEM MANAJEMEN KINERJA ANGGOTA
KEPOLISIAN NEGARA REPUBLIK INDONESIA DENGAN
RESTFUL

Diajukan untuk memenuhi salah satu syarat
memperoleh gelar Sarjana Teknik Informatika



Disusun oleh :

Nama : Farah Luthfi Oktarina
NIM : A11.2014.08644
Program Studi : Teknik Informatika – S1

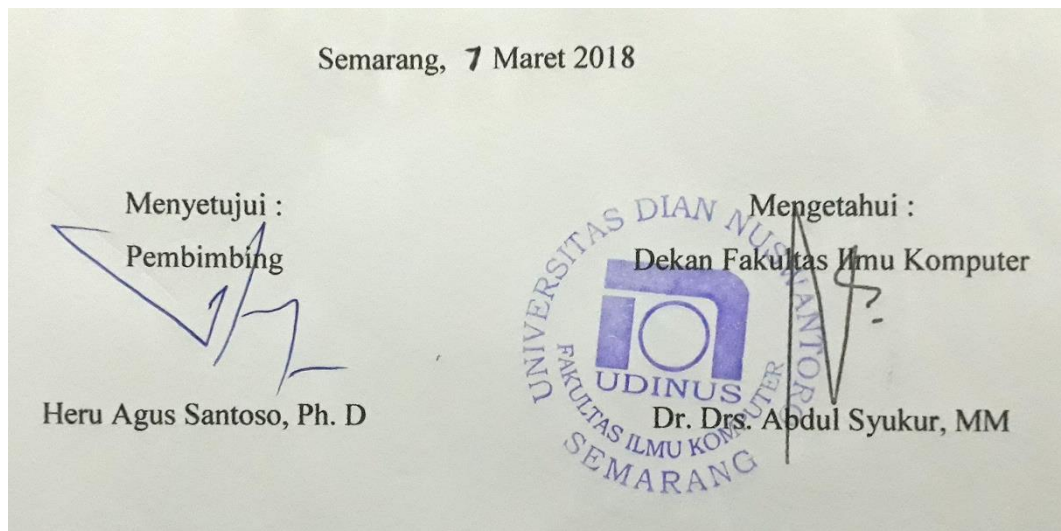
FAKULTAS ILMU KOMPUTER
UNIVERSITAS DIAN NUSWANTORO
SEMARANG

2018

PERSETUJUAN SKRIPSI

Nama : Farah Luthfi Oktarina
NIM : A11.2014.08644
Program Studi : Teknik Informatika
Fakultas : Ilmu Komputer
Judul Tugas Akhir : Perancangan dan Implementasi Web Service pada Sistem
Manajemen Kinerja Anggota Kepolisian Negara Republik
Indonesia dengan RESTful

Tugas Akhir ini telah diperiksa dan disetujui,



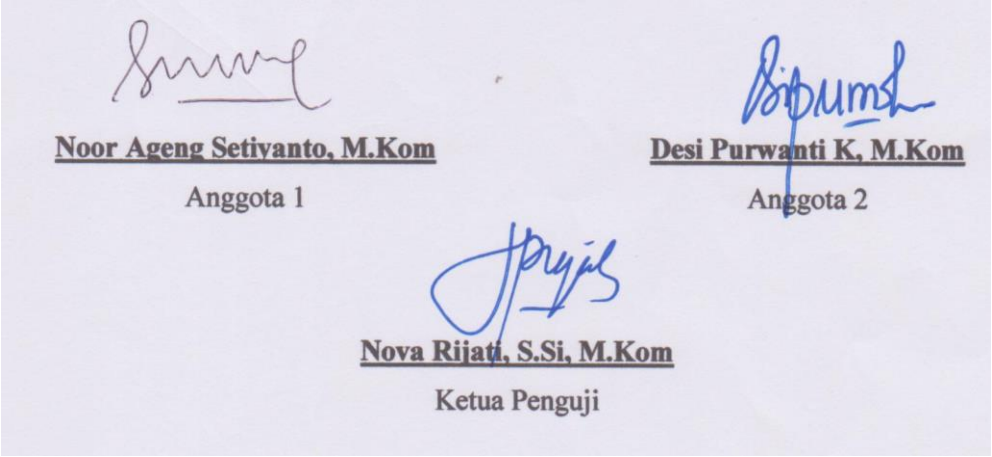
PENGESAHAN DEWAN PENGUJI

Nama Pelaksana : Farah Luthfi Oktarina
NIM : A11.2014.08644
Program Studi : Teknik Informatika
Fakultas : Ilmu Komputer
Judul Tugas Akhir : Perancangan dan Implementasi Web Service pada Sistem
Manajemen Kinerja Anggota Kepolisian Negara Republik
Indonesia dengan RESTful

Tugas akhir ini telah diujikan dan dipertahankan dihadapan Dewan Penguji pada Sidang tugas akhir tanggal 7 Maret 2018. Menurut pandangan kami, tugas akhir ini memadai dari segi kualitas maupun kuantitas untuk tujuan penganugerahan gelar Sarjana Komputer (S.Kom.)

Semarang, 7 Maret 2018

Dewan Penguji :



Noor Ageng Setivanto, M.Kom
Anggota 1

Desi Purwanti K, M.Kom
Anggota 2

Nova Rijati, S.Si, M.Kom
Ketua Penguji

PERNYATAAN KEASLIAN SKRIPSI

Sebagai mahasiswa Universitas Dian Nuswantoro, yang bertanda tangan dibawah ini, saya :

Nama : Farah Luthfi Oktarina

NIM : A11.2014.08644

Menyatakan bahwa karya ilmiah saya yang berjudul :

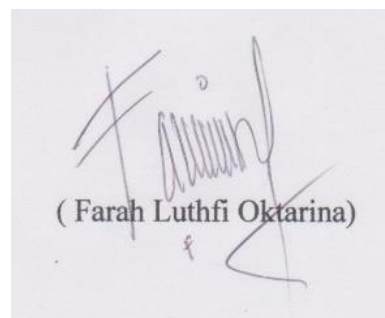
PERANCANGAN DAN IMPLEMENTASI WEB SERVICE PADA SISTEM MANAJEMEN KINERJA ANGGOTA KEPOLISIAN NEGARA REPUBLIK INDONESIA DENGAN RESTFUL

merupakan karya asli saya (kecuali cuplikan dan ringkasan yang masing-masing telah saya jelaskan sumbernya dan perangkat pendukung seperti web cam dll). Apabila di kemudian hari, karya saya disinyalir bukan merupakan karya asli saya, yang disertai dengan bukti-bukti yang cukup, maka saya bersedia untuk dibatalkan gelar saya beserta hak dan kewajiban yang melekat pada gelar tersebut. Demikian surat pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Semarang

Pada tanggal : 7 Maret 2018

Yang menyatakan



(Farah Luthfi Oktarina)

PERNYATAAN PERSETUJUAN PUBLIKASI KARYA ILMIAH UNTUK KEPENTINGAN AKADEMIS

Sebagai mahasiswa Universitas Dian Nuswantoro, yang bertanda tangan dibawah ini, saya :

Nama : Farah Luthfi Oktarina

NIM : A11.2014.08644

Demi mengembangkan Ilmu Pengetahuan, menyetujui untuk memberikan kepada Universitas Dian Nuswantoro Hak Bebas Royalti Non-Eksklusif (*Non-exclusive Royalty-Free Right*) atas karya ilmiah saya yang berjudul :

PERANCANGAN DAN IMPLEMENTASI WEB SERVICE PADA SISTEM MANAJEMEN KINERJA ANGGOTA KEPOLISIAN NEGARA REPUBLIK INDONESIA DENGAN RESTFUL

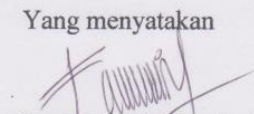
Beserta perangkat yang diperlukan (bila ada). Dengan Hak Bebas Royalti Non-Eksklusif ini Universitas Dian Nuswantoro berhak untuk menyimpan, mengcopy ulang (memperbanyak), menggunakan, mengelolanya dalam bentuk pangkalan data (*database*), mendistribusikannya dan menampilkan/ mempublikasikannya di internet atau media lain untuk kepentingan akademis tanpa perlu meminta ijin dari saya selama tetap mencantumkan nama saya sebagai penulis/ pencipta.

Saya bersedia untuk menanggung secara pribadi, tanpa melibatkan pihak Universitas Dian Nuswantoro, segala bentuk tuntutan hukum yang timbul atas pelanggaran Hak Cipta dalam karya ilmiah saya ini.

Demikian surat pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Semarang

Pada tanggal : 7 Maret 2018

Yang menyatakan

(Farah Luthfi Oktarina)

UCAPAN TERIMA KASIH

Dengan memanjatkan puji syukur kehadiran Allah SWT. Tuhan Yang Maha Pengasih yang telah melimpahkan segala rahmat, hidayah dan inayah-Nya kepada penulis sehingga tugas akhir berjudul “PERANCANGAN DAN IMPLEMENTASI WEB SERVICE PADA SISTEM MANAJEMEN KINERJA ANGGOTA KEPOLISIAN NEGERA REPUBLIK INDONESIA DENGAN RESTFUL” dapat penulis selesaikan sesuai dengan rencana karena dukungan dari berbagai pihak yang tidak ternilai besarnya. Oleh karena itu penulis menyampaikan terimakasih kepada :

1. Prof. Dr. Ir. Edi Noersasongko, M.Kom, selaku Rektor Universitas Dian Nuswantoro Semarang.
2. Dr. Drs. Abdul Syukur, MM, selaku dekan Fakultas Ilmu Komputer Universitas Dian Nuswantoro.
3. Heru Agus Santoso, Ph. D, selaku Ka. Progdi Teknik Informatika dan dosen pembimbing yang telah memberikan bimbingan kepada penulis dalam menyelesaikan tugas akhir ini.
4. Dosen-dosen pengampu di Fakultas Ilmu Komputer Teknik Informatika Universitas Dian Nuswantoro Semarang yang telah memberikan ilmu dan pengalamannya masing - masing, sehingga penulis dapat mengimplementasikan ilmu yang telah disampaikan.
5. Bapak dan Ibu yang tidak pernah lelah untuk memberikan doa serta dorongan kepada penulis untuk terus maju dan berusaha.
6. Tim SMK-Online *developer* yang selalu memberikan semangat dan dukungan moril untuk penulis.
7. Sahabat – sahabat dan teman – teman penulis yang telah membantu dan memberikan semangat kepada penulis.

Semarang, 7 Maret 2018

Penulis

ABSTRAK

Kepolisian Negara Republik Indonesia memiliki beragam sistem guna menunjang aktivitas anggotanya, salah satunya sistem manajemen kinerja berbasis teknologi informasi (SMK-Online) yaitu sistem yang digunakan untuk mengidentifikasi dan mengukur kinerja anggota POLRI agar selaras dengan visi dan misi organisasi yang menjadi landasan dalam terwujudnya SDM POLRI yang unggul dan kompetitif. SMK-Online harus terintegrasi dengan Sistem Informasi Personel POLRI (SIPP) karena seluruh data dari personel POLRI didapatkan dari SIPP. SMK-Online memiliki 2 jenis *platform* yaitu berbasis web dan android yang harus saling terintegrasi dan data harus selalu sinkron satu sama lain. Setiap penilaian kinerja dalam SMK-Online harus dapat dihubungkan dan digunakan bersama pada *platform* yang berbeda. Penerapan *web service* merupakan solusi untuk mengatasi masalah-masalah tersebut. Penelitian ini memanfaatkan *web service* dari *framework* PHP Laravel. *Web service* yang dibangun menerapkan metode REST/RESTful (*Representational State Transfer*) karena *simple*, cepat, terukur, mudah dalam *maintenance*-nya dan memiliki *method* yang mirip dengan web sehingga lebih mudah untuk pembuatannya. Data yang digunakan untuk melakukan pertukaran atau pengiriman dalam format *JSON* (*Javascript Object Notation*). Pada penelitian ini penulis berhasil membangun RESTful *web service* untuk SMK-Online yang terintegrasi dengan SIPP POLRI dan berhasil di implementasikan pada SMK-Online berbasis Android.

Kata kunci : POLRI, *web service*, *RESTful*, Laravel

DAFTAR ISI

PERSETUJUAN SKRIPSI	i
PENGESAHAN DEWAN PENGUJI.....	ii
PERNYATAAN KEASLIAN SKRIPSI.....	iii
PERNYATAAN PERSETUJUAN	iv
UCAPAN TERIMA KASIH.....	v
ABSTRAK	vi
DAFTAR ISI.....	vii
DAFTAR TABEL.....	ix
DAFTAR GAMBAR	x
DAFTAR LAMPIRAN.....	xii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Batasan Masalah.....	3
1.4 Tujuan Penelitian.....	3
1.5 Manfaat Penelitian.....	3
BAB II LANDASAN TEORI.....	5
2.1 Tinjauan Studi	5
2.2 Tinjauan Pustaka	7
2.1.1 Peraturan SMK POLRI	8
2.1.2 Teknologi Pendukung Penelitian	9
2.3 Tinjauan Obyek Penelitian	16

2.2.1	Deskripsi Tempat Penelitian	16
2.2.2	Logo dan Makna Tempat Penelitian	17
2.2.3	Struktur dan Job Deskripsi Tempat Penelitian.....	18
2.2.4	Visi dan Misi Tempat Penelitian.....	19
2.4	Kerangka Pemikiran	20
BAB III METODE PENELITIAN.....		23
3.1	Prosedur Pengambilan atau Pengumpulan data.....	23
3.2	Teknik Analisis Data	24
3.3	Metode yang Diusulkan.....	25
3.4	Eksperimen dan Cara Pengujian Metode	28
BAB IV IMPLEMENTASI SISTEM DAN PEMBAHASAN		33
4.1	Sistem Desain Web Service	33
4.2	Proses Pengambilan atau Pengumpulan Data Sekunder	34
4.3	Alur Kerja Aplikasi Android SMK-Online	35
4.4	Implementasi dan Pembahasan	39
4.3.1	Dashboard Aplikasi Android SMK-Online.....	39
4.3.2	<i>Monitoring</i> Kinerja AYD oleh PP.....	42
4.3.3	Implementasi Teknik Analisis Data dan Metode pada PKA	48
BAB V KESIMPULAN DAN SARAN.....		60
5.1	Kesimpulan.....	60
5.2	Saran	60
DAFTAR PUSTAKA		62
LAMPIRAN		64

DAFTAR TABEL

Tabel 1. 1 Tabel Perbedaan REST dengan SOAP	2
Tabel 2. 1 Tabel State of The Art.....	6
Tabel 2. 2 Tabel Simbol dari Flowchart.....	14
Tabel 2. 3 Tabel Simbol Use Case Diagram	15
Tabel 4. 1 Tabel API RESTful Web Service SMK-Online	37

DAFTAR GAMBAR

Gambar 2. 1 Arsitektur REST Web Service	10
Gambar 2. 2 Logo Framework Laravel.....	11
Gambar 2. 3 Foto Bersama Perwakilan UDINUS dan POLRI	16
Gambar 2. 4 Lab RPL Gedung D. UDINUS.....	17
Gambar 2. 5 Logo POLRI.....	17
Gambar 2. 6 Alur Pikir SMK-Online	20
Gambar 2. 7 Faktor Spesifik	20
Gambar 2. 8 Faktor Generik.....	21
Gambar 2. 9 Nilai Akhir.....	22
Gambar 3. 1 Alur Data Sekunder	23
Gambar 3. 2 Rancangan <i>Head-Detail Table</i>	25
Gambar 3. 3 REST Web Service.....	26
Gambar 3. 4 Alur Pengiriman JSON.....	27
Gambar 3. 5 <i>Header Web Service</i>	28
Gambar 3. 6 <i>Body dan Params Web Service</i> pada <i>Method POST</i>	29
Gambar 3. 7 <i>Flowchart Web Service</i>	30
Gambar 3. 8 Pemanfaatan Fungsi Web.....	30
Gambar 3. 9 Rancangan Struktur JSON	31
Gambar 3. 10 Tampilan Status dan <i>Time</i> pada Postman.....	32
Gambar 4. 1 <i>Use Case Diagram Web Service</i>	33
Gambar 4. 2 Arsitektur Proses Integrasi Data Sekunder.....	34
Gambar 4. 3 <i>Flowchart Login</i>	35
Gambar 4. 4 Panduan SMK-Online	36
Gambar 4. 5 Alur <i>Dashboard AYD</i>	38
Gambar 4. 6 Alur <i>Dahsboard PP</i>	38
Gambar 4. 7 <i>Flowchart User AYD PP</i>	39
Gambar 4. 8 <i>JSON User</i> sebagai AYD	40
Gambar 4. 9 Tampilan Android <i>User</i> sebagai AYD.....	41

Gambar 4. 10 JSON <i>User</i> sebagai AYD dan PP.....	41
Gambar 4. 11 Tampilan Android <i>User</i> sebagai AYD dan PP.....	42
Gambar 4. 12 <i>Flowchat</i> Mengambil Data AYD	43
Gambar 4. 13 JSON Daftar AYD	44
Gambar 4. 14 Daftar AYD pada web SMK-Online	44
Gambar 4. 15 Tampilan Android Daftar AYD di dashboard PP	45
Gambar 4. 16 Status AYD Belum Menyelesaikan Kinerja.....	47
Gambar 4. 17 Status AYD Telah Menyelesaikan Kinerja	48
Gambar 4. 18 <i>Flowchart</i> Mendapatkan Rekan Kerja	51
Gambar 4. 19 JSON <i>User</i> telah Mendapatkan RK dan Belum Menilai.....	53
Gambar 4. 20 JSON <i>User</i> telah Mendapatkan RK dan Sudah Menilai	53
Gambar 4. 21 Tampilan Android AYD telah Memiliki dan Menilai.....	54
Gambar 4. 22 Tampilan Android AYD Telah Memiliki dan Belum Menilai.....	55
Gambar 4. 23 <i>Flowchart</i> Penilaian PKA	56
Gambar 4. 24 Penilaian PKA Rekan Kerja Melalui <i>Postman</i>	58
Gambar 4. 25 Respon <i>Web Service</i> Penilaian PKA Rekan Kerja.....	58
Gambar 4. 26 Penilaian PKA AYD oleh PP Melalui <i>Postman</i>	59
Gambar 4. 27 Respon <i>Web Service</i> Penilaian PKA AYD oleh PP	59

DAFTAR LAMPIRAN

Lampiran 1. Fungsi Login.....	64
Lampiran 2. Fungsi untuk Mendapatkan Data AYD	65
Lampiran 3. Pemanfaatan Fungsi Global.....	66
Lampiran 4. Penerapan <i>Head-Detail</i> Tabel	67
Lampiran 5. Fungsi Update Faktor Generik	68
Lampiran 6. Struktur JSON <i>User</i> Belum dan Sudah Menilai PKA RK	69
Lampiran 7. Penerapan DB <i>Facade</i> Laravel di <i>Set</i> PKA.....	71

BAB I

PENDAHULUAN

1.1 Latar Belakang

Kepolisian Negara Republik Indonesia memiliki beragam sistem guna menunjang aktivitas anggotanya, salah satunya yaitu sistem manajemen kinerja berbasis teknologi informasi (SMK-Online). SMK-Online adalah sistem yang digunakan untuk mengidentifikasi dan mengukur kinerja anggota POLRI agar selaras dengan visi dan misi organisasi yang menjadi landasan dalam terwujudnya SDM POLRI yang unggul dan kompetitif [1]. SMK-Online harus terintegrasi dengan Sistem Informasi Personel POLRI (SIPP). Hal ini dikarenakan seluruh data dari personel POLRI didapatkan dari SIPP.

Saat ini SMK-Online memiliki 2 jenis *platform* yaitu berbasis web dan android. Kedua *platform* ini harus saling terintegrasi dan data harus selalu sinkron satu sama lain. Setiap penilaian kinerja dalam SMK-Online harus dapat dihubungkan dan digunakan bersama pada *platform* yang berbeda. Begitu juga pada data anggota POLRI yang begitu banyak harus dapat di *handle* dalam setiap penilaiannya. Masalah-masalah tersebut dapat diatasi dengan mengimplementasikan teknologi *web service*.

Web Service merupakan mekanisme interaksi antar sistem yang menunjang interoperabilitas untuk kepentingan integrasi data yang dapat diakses oleh berbagai *platform* melalui internet [2]. *Web service* dapat menjembatani komunikasi antar program sehingga setiap aplikasi yang terdapat pada satu jaringan sama atau pada jaringan yang berbeda dapat saling berkomunikasi dengan menggunakan protokol yang ditetapkan oleh *web service* seperti HTTP [3]. Salah satu metode dalam *web service* yaitu REST (*Representational State Transfer*) atau disebut RESTful, merupakan *standart* arsitektur komunikasi berbasis web yang sering diterapkan pada pengembangan layanan berbasis web. Terdapat penelitian mengenai performa

RESTful *web service* yang dibandingkan dengan SOAP *web service* pada studi penelitian respon *time* dan ukuran pesan untuk mendapatkan informasi siswa [4] :

Tabel 1. 1 Tabel Perbedaan REST dengan SOAP

Jumlah dari Siswa	Ukuran pesan (byte)		Waktu (ms)	
	SOAP / HTTP	REST (HTTP)	SOAP / HTTP	REST (HTTP)
Informasi ringan dari 1 siswa	3370	374	29.01	13.33
Informasi berat dari 2 siswa	16055	2079	63.82	26.51
Informasi berat dari 3 siswa	29155	5281	100.06	43.38
Informasi berat dari 5 siswa	37001	7566	233.68	86.82
Informasi berat dari 6 siswa	50455	13273	286.57	117.58

Tabel diatas membuktikan bahwa REST *web service* memiliki performa lebih baik dibanding SOAP *web service*. REST sangat populer karena *simple*, cepat, terukur dan mudah dalam *maintenance*-nya karena memiliki *method* yang mirip dengan web (*Create, Read, Update* dan *Delete*). Seluruh *resource* yang telah dibuat di *web service* dengan RESTful dapat digunakan oleh *platform* lain melalui representasi data berupa JSON (*Javascript Object Notation*).

Berdasarkan uraian permasalahan yang telah dijelaskan, maka penulis melakukan penelitian yang berjudul **“Perancangan dan Implementasi Web Service pada Sistem Manajemen Kinerja Anggota Kepolisian Negara Republik Indonesia dengan RESTful”** sebagai salah satu solusi yang diberikan terhadap permasalahan yang muncul dalam SMK-Online.

1.2 Rumusan Masalah

Berdasarkan latar belakang permasalahan yang telah dijelaskan sebelumnya, maka penulis membuat perumusan masalah yang menjadi fokus dari penelitian sebagai berikut :

Bagaimana merancang dan mengimplementasikan *web service* dengan metode RESTful pada Sistem Manajemen Kinerja Anggota Kepolisian Negara Republik Indonesia berbasis teknologi informasi (SMK-Online).

1.3 Batasan Masalah

Dalam menyusun penelitian untuk tugas akhir penulis memberikan batasan masalah, sehingga pembahasan dalam tugas akhir ini tidak terlalu menyimpang pada tujuan penelitian. Berdasarkan perumusan masalah, dapat ditarik batasan masalah sebagai berikut :

1. *Web Service* yang dibangun dengan memanfaatkan *framework* Laravel.
2. Metode yang akan digunakan adalah REST/ RESTful *web service*.
3. Implementasi *web service* yang dibangun hanya pada SMK-Online berbasis Android yang terintegrasi oleh Sistem Informasi Personel POLRI (SIPP).

1.4 Tujuan Penelitian

Tujuan dari penelitian ini yaitu dapat memberikan perancangan dan implementasi *web service* dengan metode RESTful *web service* pada Sistem Manajemen Kinerja Anggota Kepolisian Negara Republik Indonesia berbasis teknologi informasi (SMK-Online) sehingga data dapat terintegrasi dan sinkron.

1.5 Manfaat Penelitian

Manfaat yang diperoleh dari penelitian tugas akhir ini, diantaranya :

1.5.1 Bagi Akademik

- a. Sebagai indikator mahasiswa serta sarana untuk mengukur sejauh mana pemahaman dan penguasaan materi kuliah yang diberikan dalam menuntut ilmu di Universitas Dian Nuswantoro.

- b. Menambah literatur di perpustakaan Universitas Dian Nuswantoro yang dapat digunakan sebagai gambaran atau petunjuk membuat Tugas Akhir.
- c. Menjalin kerjasama antara Universitas Dian Nuswantoro dengan POLRI.

1.5.2 Bagi Instansi

- a. Menjalin kerjasama antara POLRI dengan Universitas Dian Nuswantoro.
- b. Dengan adanya penelitian ini, POLRI sangat terbantu dalam mewujudkan sistem yang dapat memperbaiki generasi POLRI.

1.5.3 Bagi Penulis

- a. Menambah pengetahuan serta pengalaman bekerja di instansi tempat penelitian.
- b. Penulis dapat mengembangkan ilmu yang telah didapatkan pada masa kuliah.

1.5.4 Bagi Pembaca

- a. Penelitian ini dapat digunakan sebagai bahan referensi pada penelitian selanjutnya beserta pengembangannya.

BAB II

LANDASAN TEORI

2.1 Tinjauan Studi

Sebagai referensi tentang *web service* dan metode RESTful pada penelitian yang akan dibuat, peneliti mengambil beberapa referensi penelitian (jurnal) yang sudah ada sebelumnya, diantaranya :

Dr. Digvijaysinh Rathod [4] melakukan perbandingan performa dari gaya arsitektur *web service* antara RESTful (*Representational State Transfer*) dan SOAP (*Simple Object Access Protocol*). Penulis memberikan tiga bagian penelitian atau *parameter* untuk mendapatkan hasil evaluasi performa. Hasilnya bahwa performa dan skalabilitas dari RESTful *web service* lebih baik dibanding dengan SOAP.

Pada penelitian yang dilakukan oleh Ninis Insiyah Masyhur, Kasim dan Irmawati [5] menjelaskan bagaimana penerapan *web service* untuk menjawab kebutuhan akses data yang secara *real time*, fleksibel dan *open platform*. Penelitian ini menggunakan metode RESTful *web service* dan *framework* AngularJS untuk membangun sistem pengolahan data dari objek penelitian tersebut.

Sedangkan penelitian yang dilakukan Penidas Fiodinggo Tanaem, Danny Manongga dan Ade Iriani [6] menggabungkan JWT (*JSON Web Token*) sebagai keamanan web service dengan arsitektur RESTful web service. Melalui token yang dihasilkan dari JWT dapat mengamankan komunikasi yang terjadi dalam proses pemanggilan *web service*.

Dari penelitian yang terkait di atas dapat dirangkumkan ke dalam tabel *State-of-The-Art* dibawah ini :

Tabel 2. 1 Tabel State of The Art

No	Nama Penulis	Tahun	Judul	Masalah	Metode	Hasil
1	Dr. Digvijaysinh Rathod	2017	<i>Performance Evaluation of RESTful Web Service and SOAP/ WSDL Web Service</i>	Mencari tahu perbandingan performa dari dua jenis web service dalam 1 aplikasi mobile yang sama.	Mengukur total pesan pertukaran waktu dari kedua RESTful web service dan SOAP web service kemudian membandingkan untuk dapat mengevaluasi performanya.	RESTful web service memiliki hasil evaluasi performa yang lebih baik dibanding SOAP web service.
2	Ninis Insiyah Masyhur, Kasim dan Irmawati	2015	Rancang Bangun Web Service (Studi Kasus : Layanan SIM Inventaris Barang)	Masih menggunakan sistem konvensional untuk pengiriman data sehingga menghambat kebutuhan data secara realtime pada semua platform.	Membangun web service menggunakan metode RESTful Web Service	Sistem berbasis web dan menggunakan konsep RESTful web service menjadi alternatif dalam pengelolaan manajemen data secara realtime dan menjadi penghubung

						antara sistem yang ada.
3	Penidas Fiodinggo Tanaem, Danny Manongga dan Ade Iriani	2015	RESTFul Web Service Untuk Sistem Pencatatan Transaksi Studi Kasus PT. XYZ	Kebutuhan dalam melakukan integrasi dan distribusi data antar setiap sistem dari objek penelitian	Wawancara untuk mendapatkan data dan informasi sebagai acuan dalam merancang web service yang akan dibangun.	Dapat mengintegrasikan sumber daya dari sistem dengan menggunakan aplikasi yang berbeda dengan memanfaatkan arsitektur RESTful web service yang aman.

Dalam penelitian yang akan dibuat tentang Perancangan dan Implementasi Web Service pada Sistem Manajemen Kinerja Kepolisian Negara Republik Indonesia menggunakan RESTful, peneliti menggunakan API dari *framework* Laravel yang arsitekturnya menjadi satu bagian dari aplikasi web SMK-Online. Sedangkan *web service* yang dibangun menerapkan metode RESTful *web service*. Dalam keamanan transaksi data, menggunakan JWT (*JSON Web Token*) untuk dapat mengakses semua modul dengan aman.

2.2 Tinjauan Pustaka

Untuk tinjauan pustaka pada penelitian ini dibagi menjadi 2 jenis yaitu landasan yang berdasarkan dari aturan yang terkait dengan Sistem Manajemen Kinerja Anggota POLRI beserta paparannya dan landasan teori yang mendukung dalam penelitian ini.

2.1.1 Peraturan SMK POLRI

Penelitian ini berlandaskan pada Peraturan Kepala Kepolisian Negara Republik Indonesia tentang Penilaian Kinerja Anggota Kepolisian Negara Republik Indonesia dengan Sistem Manajemen Kinerja berbasis teknologi informasi [7].

Sesuai dengan PERKAP Penilaian Kinerja Anggota POLRI Bab II tentang Perencanaan Kinerja [7], didasarkan pada 2 faktor yaitu faktor spesifik dan faktor generik. Faktor spesifik terdiri dari penilaian terhadap kontrak kerja yang dibuat dan tugas tambahan. Faktor generik didasarkan atas Perilaku Kerja Anggota (PKA), penghargaan dan hukuman.

SMK-Online ini diperuntukan bagi Anggota Yang Dinilai (AYD) dan Pejabat Penilai (PP), dengan manfaat :

- AYD : sebagai acuan kerja, penilaian kinerja pribadi serta penilaian perilaku rekan kerjanya.
- PP : untuk memantau kinerja serta memberikan penilaian kinerja kepada anggota yang dinilainya.

Penilaian kinerja anggota POLRI melalui SMK-Online meliputi :

1. Perencanaan Kinerja, dimana setiap anggota POLRI harus memahami unsur penilaian kinerja yang terdiri dari faktor spesifik dan faktor generik. Terdapat pada BAB II PERKAP SMK POLRI tentang Perencanaan Kinerja [7].
2. Pemantauan Kinerja, yaitu dapat diwakili oleh pemantauan Pejabat Penilai terhadap Anggota Yang Dinilai nya dan pemantauan PKA oleh PP dan Rekan Kerja (RK). Terdapat pada BAB III PERKAP SMK POLRI tentang Pemantauan Kinerja [7].
3. Pelaksanaan Penilaian Kinerja, yaitu bentuk penanggung jawaban oleh AYD terhadap kontrak kerja yang telah dibuat ataupun penilaian-penilaian dalam sistem kinerja anggota. Terdapat pada

BAB IV PERKAP SMK POLRI tentang Pelaksanaan Penilaian Kinerja [7].

4. Evaluasi kinerja, yaitu diwakili dari hasil nilai akhir yang didapat. Nilai akhir inilah yang digunakan sebagai acuan kinerja anggota POLRI. Evaluasi bisa berupa penghargaan atau pemberian program peningkatan kemampuan sampai adanya mutasi. Terdapat pada BAB V PERKAP SMK POLRI tentang Evaluasi Kinerja [7].

2.1.2 Teknologi Pendukung Penelitian

a. Web Service

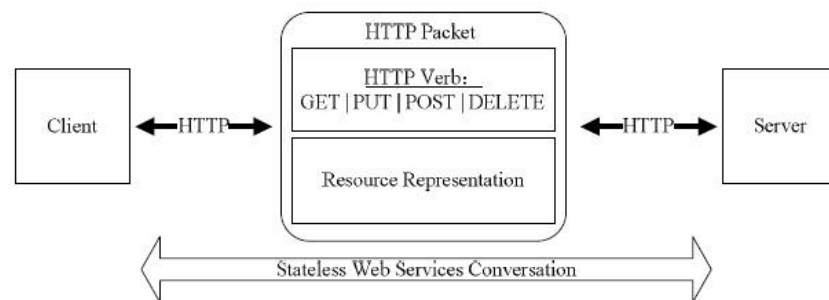
Web Service adalah sebuah entitas komputasi yang dapat diakses melalui jaringan internet maupun intranet dengan standar protokol tertentu dalam *platform* dan antarmuka bahasa pemrograman yang independen. Tujuan adanya web service untuk menjembatani komunikasi antar program sehingga setiap aplikasi yang terdapat pada satu jaringan sama atau pada jaringan yang berbeda dapat saling berkomunikasi dengan menggunakan protokol yang ditetapkan oleh *web service* seperti HTTP [3]

Web Service dapat dibangun pada *platform* maupun sistem operasi yang berbeda dengan memanfaatkan teknologi yang mendukung *web service*, diantaranya *Representation State Transfer (REST)*, *Javascript Object Notation (JSON)* dan *PHP Framework Laravel*.

b. REST

Representation State Transfer (REST) tidak seperti arsitektur tetapi sebagai cara untuk menilai arsitektur. REST merupakan *standart* arsitektur komunikasi berbasis web yang sering diterapkan pada pengembangan layanan berbasis web. REST sangat populer karena *simple*, cepat, terukur dan mudah dalam *maintenance*-nya. REST tidak terkait dengan platform tertentu, melainkan merupakan sebuah ide untuk mengembangkan layanan yang fungsinya mirip dengan web [8]. Istilah “RESTful” menyerupai istilah “*object – oriented*”. Sebuah bahasa, sebuah *framework*, atau sebuah aplikasi yang dapat dirancang dengan cara *object-oriented*, tetapi bukan

untuk membuat arsitektur *object-oriented* nya. [8]. Dalam komunikasi, REST atau RESTful menggunakan protokol HTTP. Untuk menyimpan, mengupdate, membaca dan menghapus data dapat menggunakan HTTP *request* dan *response*. Maka dari itu REST menggunakan HTTP untuk semua operasi CRUD (*Create, Read, Update* dan *Delete*).



Gambar 2. 1 Arsitektur REST Web Service

c. JSON

Javascript Object Notation (JSON) yaitu sebuah paket yang digunakan dalam proses pengiriman data. JSON adalah suatu format ringkas pertukaran data komputer [9]. JSON menggunakan *double quote* atau “ ” untuk menggambarkan string :

“ Ini adalah sebuah string “.

JSON menggunakan kurung persegi [] untuk menggambarkan *list* :

[1, 2, 3]

JSON menggunakan kurung keriting untuk menggambarkan *object* bersamaan dengan pasangan *key* dan nilai :

{ *key* : “nilai” }

d. Laravel



Gambar 2. 2 Logo Framework Laravel

Laravel merupakan sebuah framework PHP yang menganut pola Model, View dan Controller (MVC). Laravel dirancang oleh tim pengembang (Taylor Otwell) untuk meningkatkan kualitas perangkat lunak dengan mengurangi biaya pengembangan awal dan biaya pemeliharaan. Selain itu dengan menggunakan laravel dapat meningkatkan pengalaman bekerja dengan aplikasi dengan menyediakan sintaks yang ekspresif, jelas dan hemat waktu.

Model-View-Controller atau MVC merupakan sebuah pendekatan perangkat lunak yang memisahkan aplikasi logika dari presentasi [10]. MVC memisahkan aplikasi berdasarkan komponen – komponen aplikasi, seperti manipulasi data, controller dan user interface.

- *M-Model*. Mewakili struktur data yang dapat membuat representasi model dengan *class Eloquent*. Yang dimaksud dengan *Eloquent* adalah nama *class* di Laravel yang dapat digunakan untuk keperluan *Create, Read, Update* dan *Delete* (CRUD) suatu *resource* [1].
- *V-View*. Bagian yang mengatur tampilan ke pengguna yang berupa halaman web. Di dalam *Laravel*, terdapat *blade* yaitu *template engine* dari *Laravel*. Pada dasarnya *blade* adalah *view*, namun dengan *blade* programmer dapat memiliki sintak-sintak tambahan yang dapat membantu untuk menampilkan data. *Blade* adalah fitur *template* yang *inheritance*. [11]
- *C-Controller*. Bagian yang menjembatani model dan view. Controller dapat menghandle semua aksi yang perlu dilakukan ketika suatu *request* datang [1].

Laravel dibangun dengan banyak komponen yang dikenal dengan paket, baik itu paket yang dibuat oleh tim pengembang Laravel sendiri, ataupun paket dari pihak ketiga [1]. Faktanya Laravel banyak menggunakan paket dari Symfony dan beberapa paket populer lainnya. Maka dari itu digunakan composer untuk menginstall Laravel beserta semua *dependency*-nya.

Penulis membangun *web service* didalam framework PHP Laravel versi 5.3 yang menjadi satu dalam bagian aplikasi web SMK-Online.

e. API

Application Programming Interface atau disingkat API yang memungkinkan developer untuk mengintegrasikan dua bagian dari aplikasi atau dengan aplikasi yang berbeda secara bersamaan. API terdiri dari berbagai elemen seperti fungsi, protokol dan *tools* lainnya yang memungkinkan pengembang untuk membuat aplikasi. Tujuan penggunaan API yaitu untuk mempercepat proses development dengan menyediakan *function* secara terpisah sehingga pengembang tidak perlu membuat fitur yang serupa. [12].

Dalam membangun API, dibutuhkan response berupa JSON. Di Laravel dapat memanfaatkan method `response()->json()` untuk membuat API. Method ini akan secara otomatis membuat header Content-Type dengan isian *application/json* , contoh :

```
Route::get('users', function(){
    Return response()->json([
        'sukses'=>true,
        'pesan'=> 'Berhasil Mendapatkan Data Users'
    ]);
});
```

Pada penelitian ini, penulis menggunakan Web API dari framework Laravel dimana file `api.php` berada di dalam folder *route*. Web API seperti sebuah alamat web (*end point*) yang dibuat untuk menangani beberapa *task* sesuai dengan *request* yang diterima. Terdapat *parameter* sebagai acuan untuk menampilkan hasil yang diinginkan, yaitu *parameter* : usernrp atau nrp dari Anggota POLRI. Untuk dapat mengakses API membutuhkan kode otentikasi yaitu *JWT Authentication* sebagai perizinan untuk melihat data yang diinginkan.

f. JWT Authentication

Dalam membangun API untuk penelitian ini, salah satu persyaratannya adalah login dengan menggunakan token. Penulis menggunakan teknik dengan menggunakan JSON Web Token (JWT) *Authentication*. Cara kerja dari JWT adalah seperti berikut:

1. Anggota POLRI melakukan *request* token dengan mengirimkan *credentials* berupa data nrp, pin dan tanggal lahir.
2. Server memvalidasi *credentials* yang dikirim dan jika valid maka server mengirimkan token yang digenerate dengan algoritma tertentu. Token ini yang akan digunakan oleh anggota POLRI untuk *request* selanjutnya.
3. Sistem di aplikasi mobile Android akan menyimpak token yang diterima dan untuk setiap *request* selanjutnya menambahkan token tersebut pada headernya.
4. Token ini akan kadaluarsa dan hilang jika Anggota POLRI memilih keluar dari aplikasi (*logout*) dan akan digantikan dengan token yang baru ketika login.




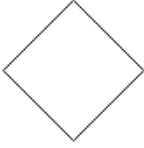

Untuk dapat menggunakan *JWT Authentication*, penulis menggunakan *package* dari “Tyton/jwt-auth”.

g. Flowchart

Flowchart yaitu penggambaran secara grafik dari setiap langkah-langkah dan urutan prosedur dari suatu program. Dengan membuat

flowchart dapat membantu *analyst* dan *programmer* untuk memecahkan masalah ke dalam segmen-segmen yang lebih kecil dan menolong dalam menganalisis alternatif-alternatif lain dalam pengoperasiannya. *Flowchart* di visualkan dengan bentuk gambar/ diagram yang memiliki aliran satu atau dua arah secara sekuensial. Dalam penelitian ini penulis memanfaatkan <https://www.draw.io/> untuk membuat *flowchart*. Berikut deskripsi dari setiap simbol *flowchart* :

Tabel 2. 2 Tabel Simbol dari Flowchart

	terminator permulaan atau akhir program
	process proses perhitungan / proses pengolahan data
	input / output data proses input / output data, parameter dan informasi
	decision perbandingan pernyataan, penyeleksian data yang memberikan pilihan untuk langkah selanjutnya
	flow line arah aliran program


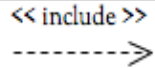
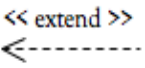



h. Use Case Diagram

Unified Modeling Language (UML) adalah bahasa spesifikasi standar untuk mendokumentasikan, menspesifikasikan, dan membangun sistem perangkat lunak. UML adalah sebuah tool yang dikembangkan dengan

paradigma Object – Oriented. UML menyediakan beberapa diagram untuk memodelkan aplikasi berorientasi objek, salah satunya yaitu *use case diagram*.

Use Case mewakili bagaimana sistem berinteraksi dengan lingkungannya dengan menggambarkan aktivitas yang dilakukan oleh pengguna dan respon sistem. *Use case diagram* merupakan representasi interaksi pengguna dengan sistem dan menggambarkan spesifikasi dari kasus penggunaan. Elemen pada *use case* meliputi *actor*, *use case*, *system boundary*, dan *relationship*.

Tabel 2. 3 Tabel Simbol Use Case Diagram

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
2		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara <i>eksplisit</i> .
3		<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
4		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
5		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
6		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor

2.3 Tinjauan Obyek Penelitian

Objek Penelitian tertuju pada internal Kepolisian Negara Republik Indonesia (POLRI) khususnya pada Kinerja Anggota POLRI.

2.2.1 Deskripsi Tempat Penelitian

SMK-Online merupakan proyek kerjasama Universitas Dian Nuswantoro (UDINUS) dengan Kepolisian Negara Republik Indonesia (POLRI) yang dikembangkan oleh mahasiswa/mahasiswi dari UDINUS. SMK-Online yang dibangun yaitu berupa aplikasi web dan android. Penulis merupakan bagian dari tim pengembang tersebut yang memiliki tugas untuk membuat *web service* SMK-Online pada *mobile application* Android.



Gambar 2. 3 Foto Bersama Perwakilan UDINUS dan POLRI

Aplikasi android SMK-Online ini menjadi proyek yang penulis kerjakan bersama tim *developer* UDINUS dimana objek penelitiannya adalah POLRI. Tempat penelitian penulis telah disepakati tidak harus berada di lingkungan POLRI. Penulis mengembangkan *web service* SMK-Online bersama tim di Laboratorium Rekayasa Perangkat Lunak (RPL) Gedung D lantai 2 Universitas Dian Nuswantoro.



Gambar 2. 4 Lab RPL Gedung D. UDINUS

2.2.2 Logo dan Makna Tempat Penelitian



Gambar 2. 5 Logo POLRI

Logo dari Polisi bernama **Rastra Sewakottama** yang artinya "Polri adalah Abdi Utama dari pada Nusa dan Bangsa." Sebutan itu adalah Brata pertama dari Tri Brata yang diikrarkan sebagai pedoman hidup Polri sejak 1 Juli 1954. [13]

Polri yang tumbuh dan berkembang dari rakyat, untuk rakyat, memang harus berinisiatif dan bertindak sebagai abdi sekaligus pelindung dan pengayom rakyat. Harus jauh dari tindak dan sikap sebagai "penguasa". Ternyata prinsip ini sejalan dengan paham kepolisian di semua Negara yang disebut new modern police philosophy, "Vigilant Quiescant" (kami berjaga sepanjang waktu agar masyarakat tentram).

Prinsip itu diwujudkan dalam bentuk logo dengan rincian makna sbb:

- Perisai bermakna pelindung rakyat dan negara.
- Tiang dan nyala obor bermakna penegasan tugas Polri, disamping memberi sesuluh atau penerangan juga bermakna penyadaran hati nurani masyarakat agar selalu sadar akan perlunya kondisi kamtibmas yang mantap.
- Pancaran obor yang berjumlah 17 dengan 8 sudut pancar berlapis 4 tiang dan 5 penyangga bermakna 17 Agustus 1945, hari Proklamasi Kemerdekaan yang berarti Polri berperan langsung pada proses kemerdekaan dan sekaligus pernyataan bahwa Polri tak pernah lepas dari perjuangan bangsa dan negara.
- Tangkai padi dan kapas menggambarkan cita-cita bangsa menuju kehidupan adil dan makmur, sedangkan 29 daun kapas dengan 9 putik dan 45 butir padi merupakan suatu pernyataan tanggal pelantikan Kapolri pertama 29 September 1945 yang dijabat oleh Jenderal Polisi Raden Said Soekanto Tjokrodiatmodjo.
- 3 Bintang di atas logo bermakna Tri Brata adalah pedoman hidup Polri. Sedangkan warna hitam dan kuning adalah warna legendaris Polri.
- Warna hitam adalah lambang keabadian dan sikap tenang mantap yang bermakna harapan agar Polri selalu tidak goyah dalam situasi dan kondisi apapun; tenang, memiliki stabilitas nasional yang tinggi dan prima agar dapat selalu berpikir jernih, bersih, dan tepat dalam mengambil keputusan.

2.2.3 Struktur dan Job Deskripsi Tempat Penelitian

Organisasi Polri disusun secara berjenjang dari tingkat pusat sampai ke kewilayahan. Organisasi Polri tingkat pusat disebut Markas Besar Kepolisian Negara Republik Indonesia (**Mabes Polri**); sedang organisasi Polri tingkat kewilayahan disebut Kepolisian Negara Republik Indonesia Daerah (**Polda**) di tingkat provinsi, Kepolisian Negara Republik Indonesia Resort (**Polres**) di tingkat kabupaten/kota, dan Kepolisian Negara Republik Indonesia Sektor (**Polsek**) di wilayah kecamatan.

Penelitian ini merupakan penelitian bersama dalam tim dimana setiap tim terdiri dari *project manager*, pembina, *web developer*, *android developer*, *web service* dan *designer*. Dalam hal ini peneliti mendapat posisi *web service* yang bertugas untuk membuat *service* SMK Online atau penghubung antara web dan aplikasi android.

2.2.4 Visi dan Misi Tempat Penelitian

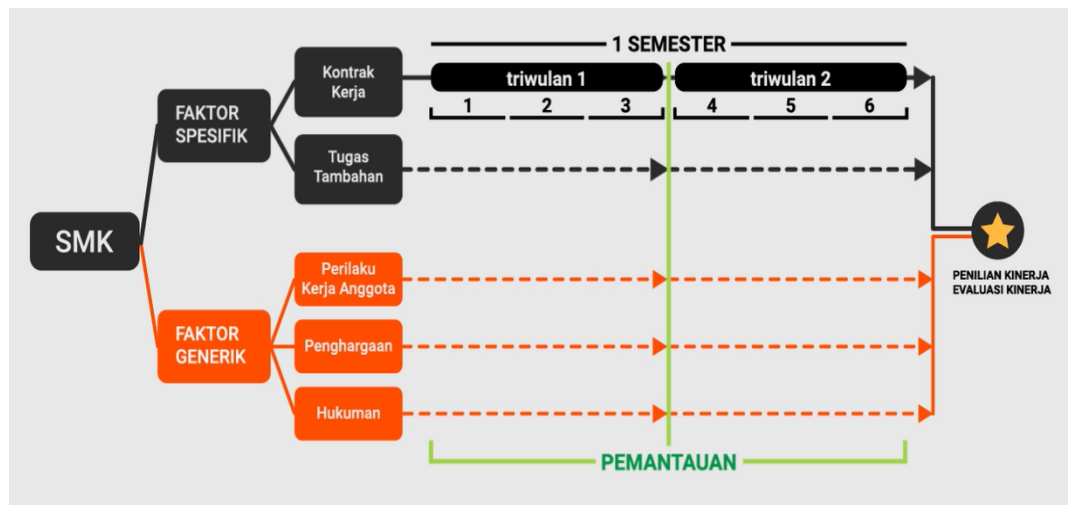
➤ Visi POLRI :

Terwujudnya pelayanan keamanan dan ketertiban masyarakat yang prima, tegaknya hukum dan keamanan dalam negeri yang mantap serta terjalinnya sinergi polisional yang proaktif. [13]

➤ Misi POLRI :

1. Melaksanakan deteksi dini dan peringatan dini melalui kegiatan/operasi penyelidikan, pengamanan dan penggalangan.
2. Memberikan perlindungan, pengayoman dan pelayanan secara mudah, responsif dan tidak diskriminatif.
3. Menjaga keamanan, ketertiban dan kelancaran lalu lintas untuk menjamin keselamatan dan kelancaran arus orang dan barang.
4. Menjamin keberhasilan penanggulangan gangguan keamanan dalam negeri.
5. Mengembangkan perpolisian masyarakat yang berbasis pada masyarakat patuh hukum.
6. Menegakkan hukum secara profesional, objektif, proporsional, transparan dan akuntabel untuk menjamin kepastian hukum dan rasa keadilan.
7. Mengelola secara profesional, transparan, akuntabel dan modern seluruh sumber daya Polri guna mendukung operasional tugas Polri.
8. Membangun sistem sinergi polisional interdepartemen dan lembaga internasional maupun komponen masyarakat dalam rangka membangun kemitraan dan jejaring kerja (*partnership building/networking*).

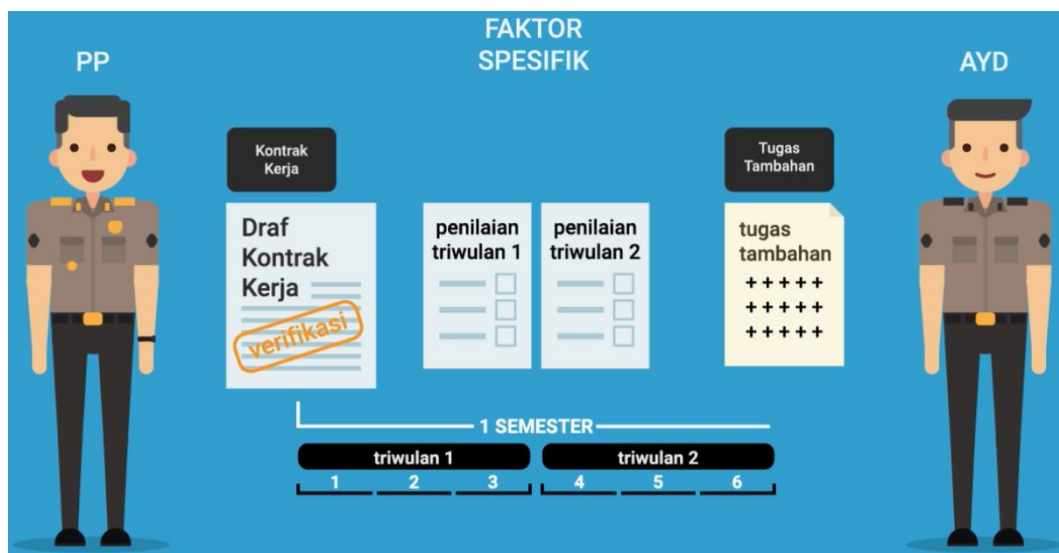
2.4 Kerangka Pemikiran



Gambar 2. 6 Alur Pikir SMK-Online

Aplikasi SMK-Online yang dibangun sesuai dengan peraturan POLRI seperti yang sudah dijelaskan pada bab 2.1.1 *Peraturan SMK POLRI*. Dalam SMK-Online terdiri dari 2 penilaian pokok :

1. Faktor Spesifik (FS)



Gambar 2. 7 Faktor Spesifik

Penelitian ini diawali dengan AYD menginputkan draf kontrak kerja selama 1 semester dengan jangka waktu input selama 15 hari awal di bulan Januari dan

Juli. Kemudian AYD dan PP yang terkait melakukan persetujuan kontrak kerja tersebut. Kontrak kerja 1 semester ini dibagi menjadi 2 triwulan dimana dalam setiap triwulannya AYD harus melakukan input capaian kontrak kerja yang telah dibuat. Masa input capaian kontrak kerja ini selama 3 bulan sekali. Selain itu selama 1 semester AYD dapat melakukan tugas tambahan yang ada hubungannya dengan tugas pokok/ fungsi POLRI dan tidak ada didalam kontrak kerja yang telah ditetapkan.

2. Faktor Generik (FG)



Gambar 2. 8 Faktor Generik

Terdapat 3 penilaian pada faktor generik yaitu PKA, penghargaan dan hukuman. Khusus untuk penilaian kerja anggota dilakukan pada 1 bulan terakhir di setiap semester, yaitu pada bulan Juli dan Desember. Bagi anggota yang dinilai (AYD) melakukan penilaian PKA terhadap rekan kerja nya (RK) sedangkan bagi Pejabat Penilai (PP) melakukan penilaian PKA terhadap setiap anggota yang dinilai nya (AYD). Sedangkan penghargaan dan hukuman berlangsung dari awal hingga akhir semesternya.



Gambar 2. 9 Nilai Akhir

SMK-Online akan menghasilkan nilai akhir yang memiliki peran untuk mengevaluasi kinerja para anggota POLRI. Bagi yang mendapatkan hasil kinerja sangat baik, akan diberikan penghargaan dan pujian. Dan bilamana nilai kerja tidak memenuhi *standart*, maka akan diberikan program peningkatan kemampuan sampai dengan mutasi.

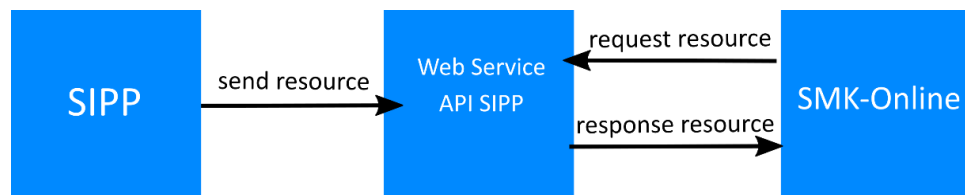
BAB III

METODE PENELITIAN

3.1 Prosedur Pengambilan atau Pengumpulan data

Terdapat dua cara dalam prosedur pengambilan atau pengumpulan data yaitu dengan pendekatan data secara primer dan sekunder. Yang dimaksud dengan data sekunder adalah data yang sudah ada/ tersedia sehingga dapat lebih mudah dan cepat dalam mencari dan mengumpulkan data. Sedangkan yang dimaksud dengan data primer yaitu data yang didapat langsung dari sumber pertama kali yang menjadi responden dari penelitian penulis.

Pada penelitian ini, penulis memperoleh data sekunder yaitu data Anggota POLRI dari API *web service* Sistem Informasi Personel POLRI (SIPP). Data dari SIPP ini di ambil secara *up to date* ke tabel *Users* di database SMK Online. Hal ini dengan tujuan agar data SMK Online dapat terintegrasi dengan data di aplikasi SIPP :



Gambar 3. 1 Alur Data Sekunder

Sedangkan pendekatan data secara primer didapat langsung dari *request* anggota POLRI di setiap menu SMK Online. Berikut fitur/menu/unit yang harus tersedia dalam aplikasi SMK-Online berdasarkan PERKAP SMK [7]:

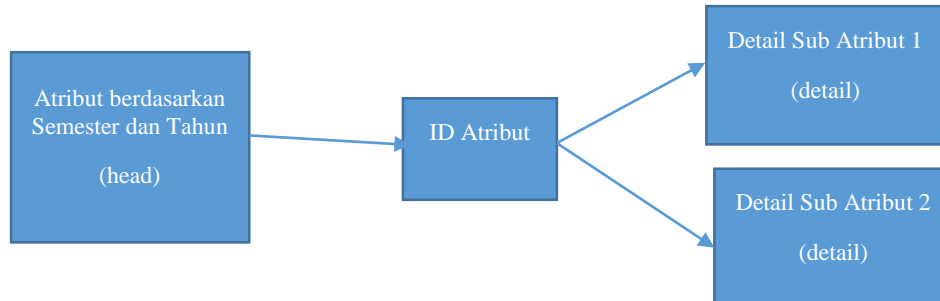
1. Faktor Spesifik
 - a. Kontrak Kerja Semester
 - b. Capaian Kontrak Kerja Triwulan Ganjil
 - c. Capaian Kontrak Kerja Triwulan Genap
 - d. Tugas Tambahan
2. Faktor Generik
 - a. Perilaku Kerja Anggota (PKA)
 - b. Penghargaan
 - c. Hukuman
3. Nilai Akhir
4. Rekam Jejak SMK
 - a. Rekam Penilaian
 - b. Rekam Hukuman
 - c. Rekam Penghargaan
5. Giat Harian

3.2 Teknik Analisis Data

Dalam penelitian ini penulis menggunakan pendekatan analisis data bersifat kuantitatif yaitu dengan menerapkan teknik statistik, setelah semua data seluruh sumber terkumpul. Teknik statistik ini digunakan untuk menganalisa data dengan cara mendeskripsikan data yang sudah terkumpul.

Data yang didapat dari responden yang diisi oleh setiap anggota POLRI akan diolah sesuai dengan bagian-bagian dari sistem manajemen kinerja anggota POLRI. Di setiap bagian tersebut terdapat *head* data dan *detail* data yang disesuaikan dengan semester dan tahun. Parameter ID yang didapat dari data *head* tersebut kemudian dapat digunakan pada sub-atribut. Selain itu untuk menghindari adanya duplikasi data, maka *web service* dapat memanfaatkan ID yang didapat dari

head tabel untuk dilakukan pengecekan diawal proses. Hal ini juga dapat terciptanya normalisasi database.

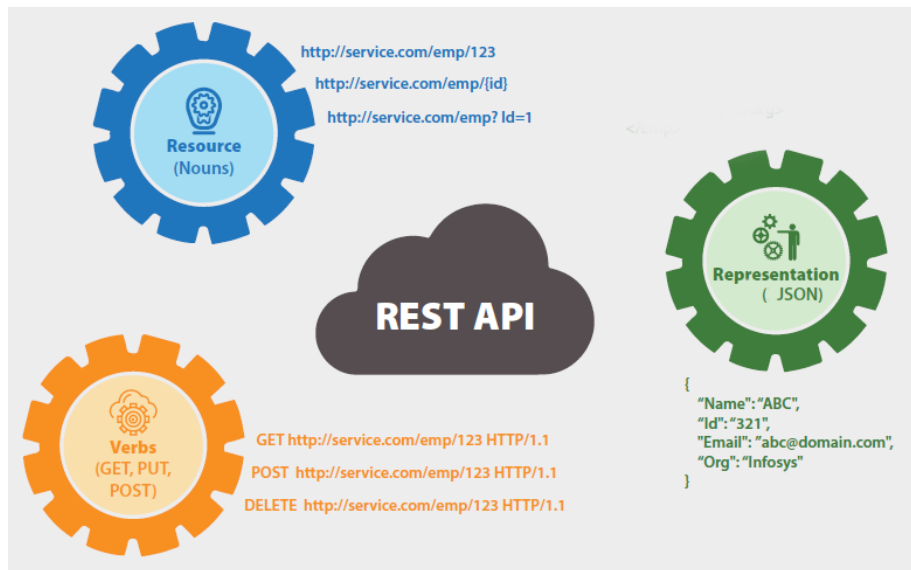


Gambar 3. 2 Rancangan *Head-Detail Table*

Penelitian ini akan menghasilkan nilai akhir yang berfungsi sebagai acuan evaluasi kinerja anggota POLRI. Jika nilai kinerja tidak memenuhi *standart* maka akan diberikan program peningkatan kemampuan sampai dengan mutasi. Dan bagi anggota POLRI yang mendapatkan hasil kinerja yang baik akan mendapatkan penghargaan. Semua nilai akhir ini akan terekam pada rekam jejak anggota POLRI setiap semester nya.

3.3 Metode yang Diusulkan

Dalam membangun kebutuhan web service SMK-Online, penulis menerapkan metode REST/ RESTful. Metode ini cocok digunakan karena fungsinya yang mirip dengan web yaitu adanya fungsi GET, POST, PUT dan DELETE. Proses metode REST web service yang diterapkan [14] :



Gambar 3. 3 REST Web Service

a) Resources

Resources adalah elemen dasar dari web. Ketika bekerja didalam REST, hal pertama yang diidentifikasi adalah *resources* dan menemukan bagaimana *resource-resource* dihubungkan dengan yang lain. Setiap resource memiliki identifikasi yang unik di web *platform* yang disebut *Universal Resource Identifier* (URI) atau pada web berupa *Uniform Resource Locator* (URL). Pada SMK-Online, penulis menggunakan *package* rest api dari Dingo dan menerapkan versi api v1 [15].

Dengan mendefinisikan terlebih dahulu Dingo dalam route API SMK-Online yang menggunakan framework Laravel, seperti dibawah ini :

```
$api = app('Dingo\Api\Routing\Router');
```

Kemudian disetiap *resource* diberikan *authentication* dengan menggunakan JWT agar mendapatkan token sehingga dapat memberikan keamanan dalam *request resource*, seperti dibawah ini :

```
$api->version('v1',['namespace' => 'App\Http\Controllers'],function ($api)
{
    //URI API yang digunakan
})
```

b) Verbs

Verb adalah sebuah aksi HTTP seperti POST, GET, PUT, DELETE. Berikut keterangan mengenai *method-method* yang digunakan dalam RESTful web service [14]:

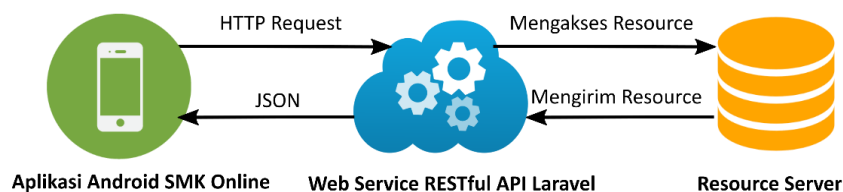
1. GET digunakan untuk mengambil data *resource* dari server, menyediakan hanya akses baca pada *resource*.
2. POST digunakan untuk mengirim data *resource* ke server atau membuat *resource* yang baru.
3. DELETE digunakan untuk menghapus data *resource* yang ada di server.
4. PUT digunakan untuk memodifikasi *resource* yang ada di *server*. Method ini jarang digunakan oleh penulis pada web service SMK-Online.

Contoh HTTP *request* GET untuk mengambil data *resource* pada SMK-Online dengan catatan URI berikut oleh penulis diubah penamaannya untuk kepentingan *privacy* dari POLRI :

GET : domain.com/api-key-point/{ID-user}/end-point

c) Representation

Representasi digunakan untuk menentukan jalan untuk menampilkan *resource-resource* ke *client* dalam hal ini pada aplikasi android SMK-Online. REST mendukung semua format tanpa ada batasan. Pada SMK-Online penulis menggunakan format JSON dalam mengirim *resource – resource* yang dapat digunakan pada aplikasi android. Semua *resource* yang dikirim melalui JSON, disesuaikan dengan kebutuhan dari aplikasi android dan prosesnya sama dengan SMK-Online versi Web. Berikut alur pengiriman JSON :

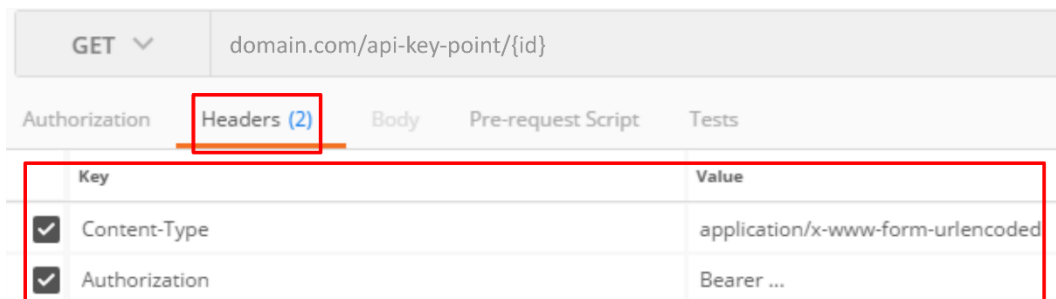


Gambar 3. 4 Alur Pengiriman JSON

3.4 Eksperimen dan Cara Pengujian Metode

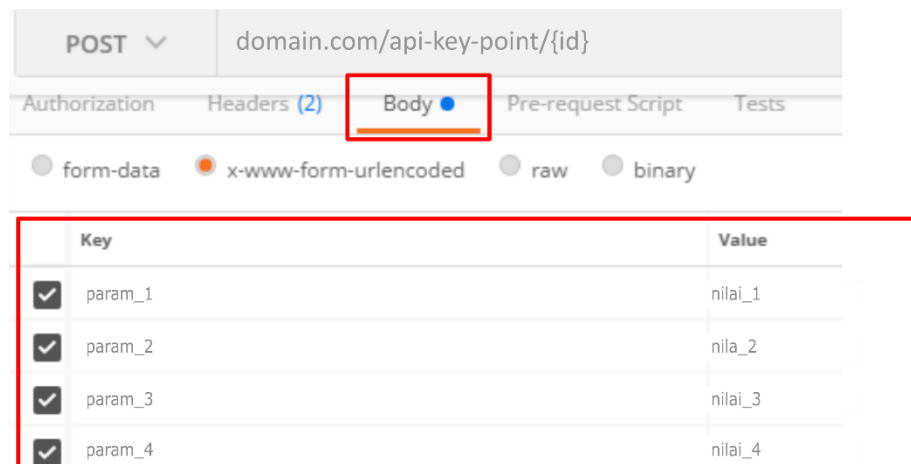
Dalam melakukan eksperimen penelitian, penulis terlebih dahulu harus memahami alur proses setiap *resource* yang dibutuhkan. Jika *resource* yang diminta juga terdapat di aplikasi web, maka penulis harus memahami proses yang akan diterapkan di aplikasi web. Hal ini bertujuan untuk terciptanya integrasi data antara web dengan web service. Karena peng-integrasian fungsi merupakan hal penting dalam membangun web service.

Eksperimen yang dilakukan dimulai dengan memanfaatkan *server* lokal dari XAMPP dan *text-editor* bernama **Atom**. Hal pertama yang harus diperhatikan yaitu mempersiapkan kebutuhan *web service* mulai dari inisialisasi seluruh *resource* dengan menetapkan *header* dan *method* yang digunakan. *Header* dalam penelitian ini terdiri dari **Content-Type** berupa **x-www-form-urlencoded** yaitu standar *request* form yang digunakan pada aplikasi android dan **Authorization** yaitu berupa token *JWT Authentication* dimana token ini akan *expired* jika *user logout* atau melakukan *un-install* aplikasi android.



Gambar 3. 5 Header Web Service

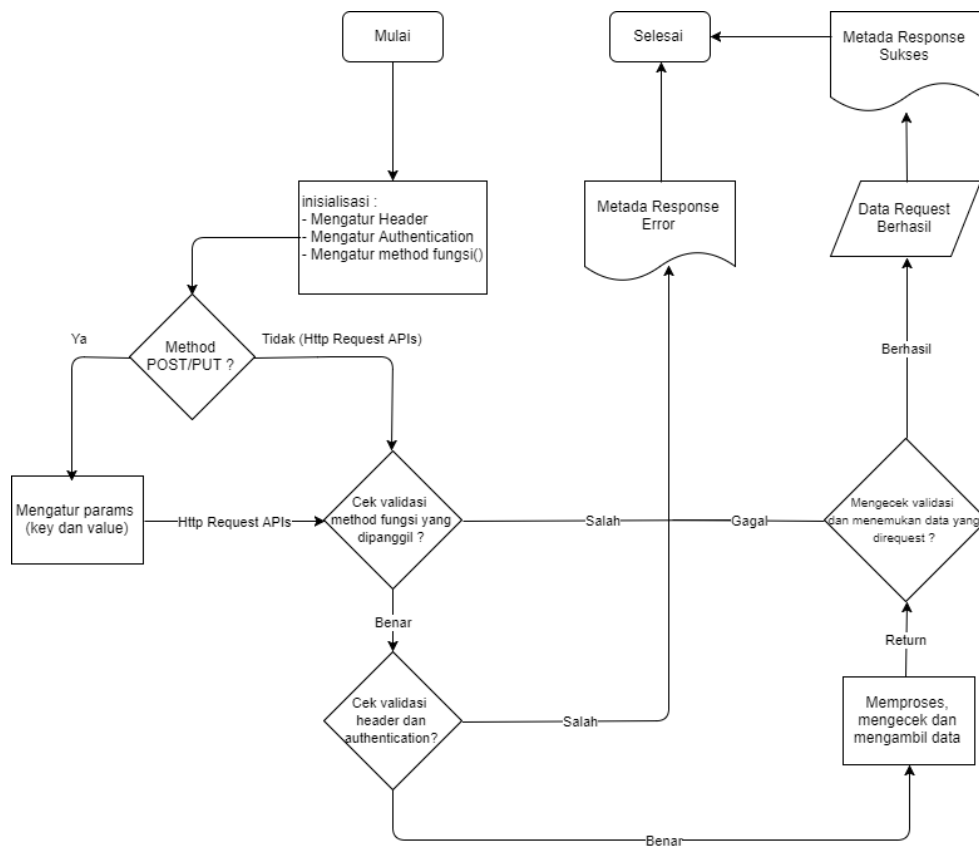
Jika *method* yang digunakan adalah POST atau PUT, maka harus ada **PARAMS** atau elemen yang akan dikirim (*key* dan *value*). Setelah inisialisasi maka melakukan proses *request*. Kemudian memasuki tahap cek validasi *method* dan *header*, *authentication user* yang berupa token yang diawali dengan kunci **Bearer** dari *JWT Authentication*. Jika proses validasi berhasil maka akan lanjut ke tahap berikutnya. Namun jika tidak sesuai maka proses akan berhenti dan mengirimkan *response* eror dan disertai pesan dari masing-masing proses validasi.



Gambar 3. 6 Body dan Params Web Service pada Method POST

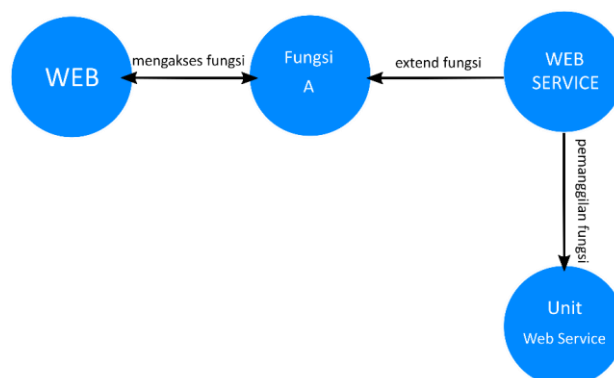
Setelah tahap validasi selesai, kemudian akan berlanjut pada tahap proses atau *checking* atau *get* data, kemudian menuju pada tahap validasi *check data* atau *finding* data. Jika data dapat ditemukan maka akan menuju proses atau tahap selanjutnya, sebaliknya jika tidak maka proses akan berhenti dan mengirimkan *response* error disertai pesan dari masing-masing validasi.

Memasuki tahap terakhir, setelah proses validasi data selesai maka data akan dikirim bersama dengan *response* sukses. Berikut desain alur sistem (*flowchart*) dari *web service* dari awal *request* HTTP *method* hingga mendapatkan hasil data berupa *response*.



Gambar 3. 7 Flowchart Web Service

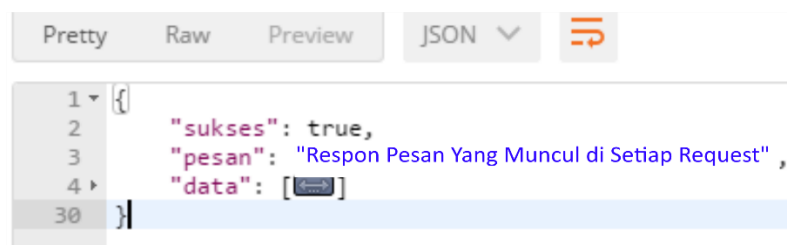
Adapun beberapa fungsi yang penulis ambil dari fungsi web sehingga penulis dapat memanfaatkannya untuk menyederhanakan fungsi web service yang akan dibangun. Adanya fungsi *global* ini dapat dimanfaatkan demi mewujudkan sinkronisasi data antara *platform* web dan android SMK-Online. Berikut desain pemanfaatan fungsi dalam *web service* :



Gambar 3. 8 Pemanfaatan Fungsi Web

Sedangkan pada pengujian, penulis memanfaatkan aplikasi **postman**. Postman merupakan aplikasi yang digunakan bagi developer untuk melakukan pengujian (*testing*) pada API atau *web service*. Postman berfungsi sebagai GUI (*Graphical User Interface*) API caller. Penulis melakukan pengujian di setiap modul atau unit fungsi *web service*. Terdapat 3 faktor sebagai acuan keberhasilan dalam melakukan pengujian yaitu :

1. Nilai kunci **sukses**, **pesan** dan **data**. Ketiga kunci ini sengaja dibuat oleh penulis disetiap JSON dengan nilai respon yang berbeda di setiap kondisinya. Jika nilai sukses *true*, maka sistem pada android akan mengambil data yang dikirimkan melalui JSON tersebut. Namun jika nilai sukses *false*, maka sistem pada android akan mengambil pesan gagal dari JSON. Dalam rancangan JSON yang dikirim, penulis menggunakan model *try-catch* sehingga dapat meminimalisir eror API dari web service. Berikut contoh rancangan struktur JSON pada GET kontrak kerja semester anggota POLRI :



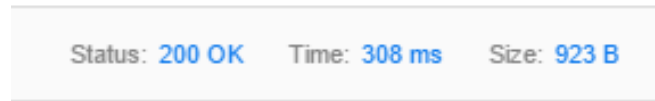
The image shows a screenshot of a JSON editor interface. At the top, there are tabs for 'Pretty', 'Raw', and 'Preview', with 'Pretty' selected. To the right, there is a dropdown menu set to 'JSON' and a red icon. The main area displays a JSON object with the following structure:

```
{
  "sukses": true,
  "pesan": "Respon Pesan Yang Muncul di Setiap Request",
  "data": [ ]
}
```

The JSON is formatted with line numbers on the left (1, 2, 3, 4, 30) and syntax highlighting.

Gambar 3. 9 Rancangan Struktur JSON

2. **Status** dan *time* yang dihasilkan dari setiap *request* API. Dengan bantuan postman, maka penulis dapat mengetahui nilai status dan *time* yang muncul. Status harus bernilai 200 OK yang artinya tidak ada eror saat *request* API. Untuk menguji setiap unit API yang dikirim, hindari status 500 *Internal Server Error*. Sedangkan *time* sangat penting untuk mengetahui berapa lama web service dapat me-load data yang dikirim. Semakin kecil *time* (dalam ms) yang dihasilkan, semakin cepat data dapat dimunculkan. Struktur *coding* dapat mempengaruhi proses *run* dari *web service*.



Gambar 3. 10 Tampilan Status dan *Time* pada Postman

3. **Output** atau kesesuaian *value* yang dihasilkan. Setiap modul *web service* yang dibuat harus dilakukan pengecekan hasil *output* apakah sesuai dengan *output* pada aplikasi web. Pengujian ini dilakukan dengan model *white box*, yaitu pengujian hingga kepada detail pengecekan kode program.

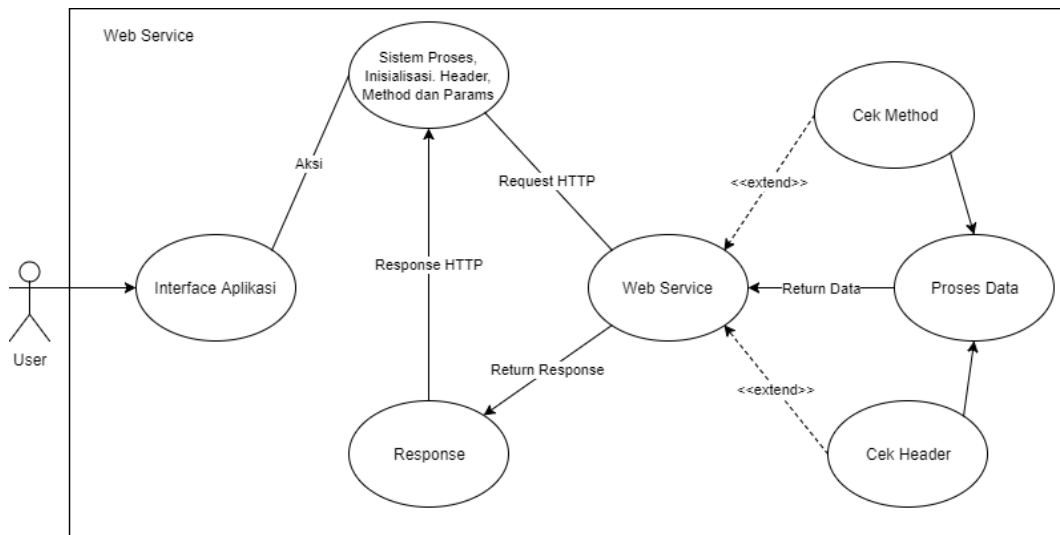
BAB IV

IMPLEMENTASI SISTEM DAN PEMBAHASAN

Tahap implementasi ini merupakan tahap penerapan dari analisa dan metode yang dirancang di BAB III dan berisi langkah-langkah pembuatan web service dari SMK-Online.

4.1 Sistem Desain Web Service

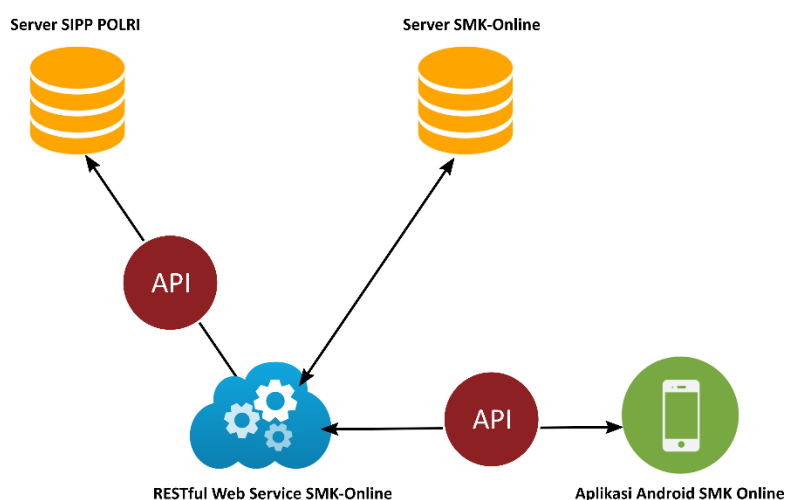
Merupakan lanjutan dari pemetaan eksperimen yang dilakukan penulis pada sub bab 3.4 dengan dibuat dalam bentuk *use case* diagram. Hal ini bertujuan agar dapat mengetahui fungsionalitas dari kinerja sistem *web service* dalam penelitian ini. Pelaku utama yaitu *user* atau dalam hal ini anggota / personel POLRI dengan melakukan aksi interaksi perintah pada aplikasi android SMK-Online yang kemudian akan diproses oleh sistem untuk melakukan *request* dan mendapatkan *response* berupa data yang diinginkan sesuai dengan spesifikasi *web service* pada saat melakukan eksekusi. Berikut merupakan *use case* diagram dari *web service* :



Gambar 4. 1 Use Case Diagram Web Service

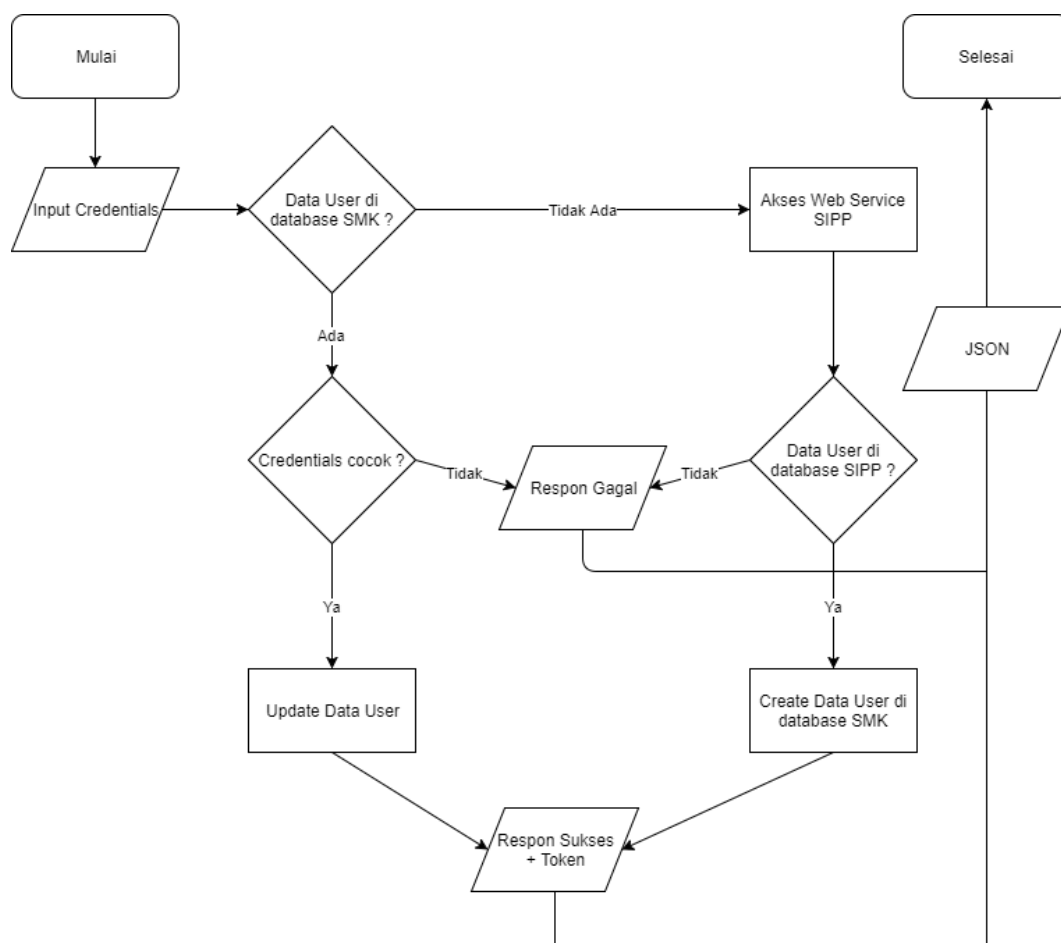
4.2 Proses Pengambilan atau Pengumpulan Data Sekunder

Pada sub bab 3.1 *Prosedur Pengambilan atau Pengumpulan Data* telah dijelaskan bahwa data sekunder didapat dari *service* Sistem Informasi Personel POLRI (SIPP). Hal ini bertujuan agar data SMK-Online dapat terintegrasi dengan data dari SIPP POLRI. Data sekunder ini sebagai data utama dari SMK-Online yang berisi data dari anggota POLRI. Jadi semua data anggota POLRI yang terdapat di database SMK-Online berasal dari SIPP POLRI.



Gambar 4. 2 Arsitektur Proses Integrasi Data Sekunder

Pengecekan data anggota POLRI dimulai dari ketersediaan data di database SMK-Online. Jika *request* data tidak tersedia, maka *service* harus melakukan pengecekan data di SIPP POLRI. Penulis memanfaatkan API dari SIPP POLRI untuk melakukan cek dan *update* data. API ini digunakan pada saat user melakukan *login* di aplikasi android maupun web SMK-Online. Jadi ada atau tidak adanya data anggota POLRI dapat diketahui ketika *user* melakukan *login*. Jika data di database SMK-Online belum ada, maka *service* akan melakukan *create* data user ke database SMK-Online. Dan jika terdapat perubahan data anggota POLRI, maka *service* akan melakukan *update* sesuai dengan data yang ada di SIPP POLRI. *Resource* yang digunakan melalui <http://domain.com/api-key-point/login> dengan method POST dan implementasi terdapat di Lampiran 1. Berikut ini flowchart pengambilan atau pengumpulan data sekunder pada *web service* saat *login*:

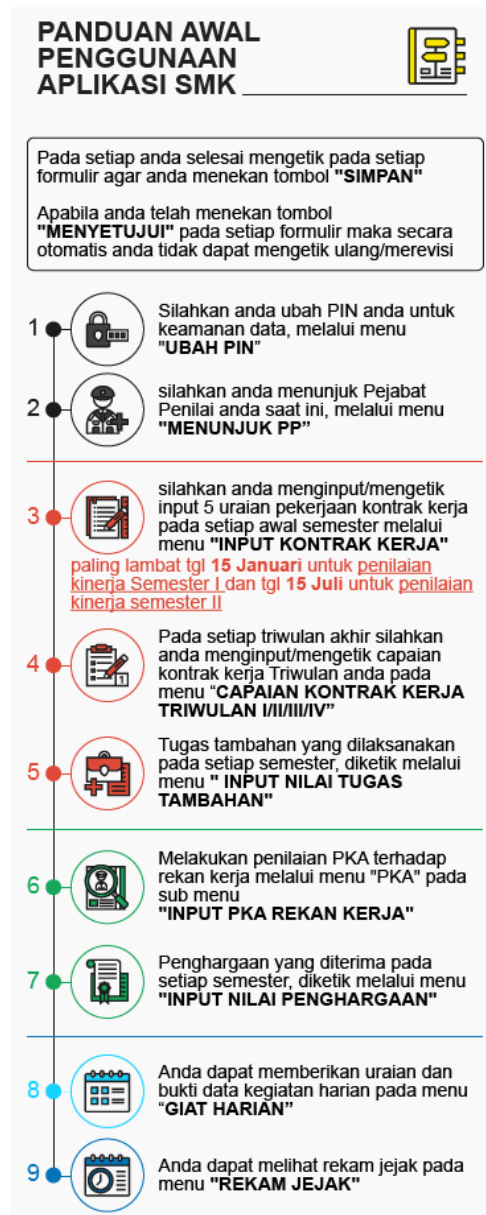


Gambar 4. 3 Flowchart Login

4.3 Alur Kerja Aplikasi Android SMK-Online

Alur Pikir SMK-Online telah dijelaskan pada sub-bab 2.3 *Kerangka Pemikiran*. Terdapat 2 user dalam penggunaan aplikasi android SMK-Online yaitu Anggota Yang Dinilai (AYD) dan Pejabat Penilai (PP) dimana masing-masing memiliki *dashboard*. Fungsi *dashboard* AYD yaitu untuk melakukan perencanaan dan pelaksanaan kinerja AYD. Sedangkan *dashboard* PP berfungsi untuk melakukan pemantauan kinerja terhadap AYD.

Kemudian penelitian ini menyediakan *web service* yang sesuai dengan kebutuhan dari aplikasi android SMK-Online. Adapun panduan yang dibuat guna mempermudah *user* dalam menggunakan aplikasi android SMK-Online :



Gambar 4. 4 Panduan SMK-Online

Maka dari itu berdasarkan panduan SMK-Online yang telah dibuat, *web service* harus dapat menyediakan setiap unit/modul aplikasi android SMK-Online agar sesuai dengan sistem penilaian kinerja anggota POLRI dengan acuan pada PERKAP SMK-Online [7]. Berikut tabel daftar beberapa API RESTful *web service* yang bisa mewakili dari method *CRUD RESTful (resource dan method)* untuk aplikasi android SMK-Online (URI di inisialkan untuk kepentingan *privacy* POLRI) :

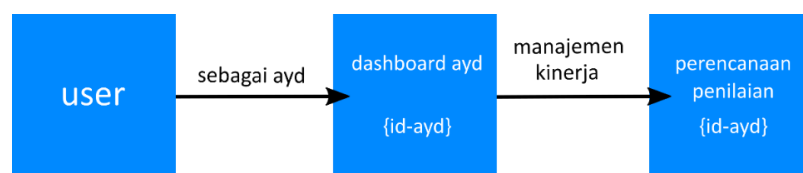
Tabel 4. 1 Tabel API RESTful Web Service SMK-Online

No	Verbs/ Method	Resources	Keterangan
Dashboard AYD			
1	POST	http://domain.com/api-key-point/login	Melakukan login
2	GET	http://domain.com/api-key-point/users	Informasi data user yang sedang login
22	DELETE	http://domain.com/api-key-point/{id}/tugas-tambahan/{nomor}	Menghapus tugas tambahan berdasarkan nomor urut atau id
26	GET	http://domain.com/api-key-point/{id}/pka/menilai	Menampilkan informasi status menilai pka dan rekan kerja yang dimiliki
27	GET	http://domain.com/api-key-point/{id-penilai}/pka/{id-dinilai}/nilai	Menampilkan detail penilaian kerja anggota yang dinilai. Terdapat 9 atau 10 nilai dari faktor kinerja
28	POST	http://domain.com/api-key-point/{id}/pka	Menyimpan nilai penilaian kerja anggota dari 9 atau 10 nilai faktor kinerja
Dashboard PP			
52	GET	http://domain.com/api-key-point/pejabat/{id-pp}/anggota	Menampilkan daftar AYD dari PP
53	GET	http://domain.com/api-key-point/{id-pp}/anggota/status	Menampilkan status kinerja AYD dari PP yang digunakan untuk memantau faktor spesifik, faktor generik dan kegiatan harian anggotanya

Dibawah ini adalah alur pikir *user* dalam mengakses aplikasi android SMK-Online:

a) *User* hanya menjadi AYD

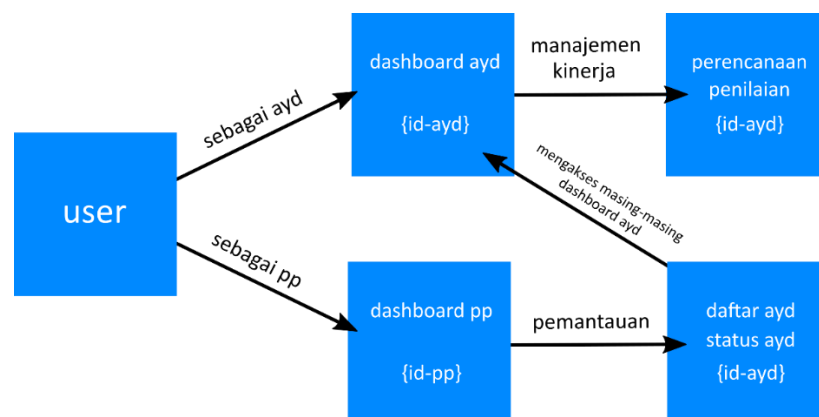
Jika *user* hanya sebagai anggota yang dinilai (AYD), maka setelah melakukan *login* user akan diarahkan kepada *dashboard* AYD dengan token yang disimpan ke dalam *cache* (oleh aplikasi android). Token ini digunakan untuk dapat mengakses seluruh unit penilaian SMK dalam manajemen kinerja yang dimulai dari perencanaan sampai penilaian kinerja.



Gambar 4. 5 Alur Dashboard AYD

b) *User* menjabat PP sekaligus menjadi AYD

Jika *user* memiliki jabatan sebagai pejabat penilai, maka akan muncul 2 *dashboard* yaitu AYD dan PP. Karena pada dasarnya semua anggota POLRI merupakan anggota yang dinilai (AYD) yang memiliki kewajiban untuk memilih pejabat penilai (PP) nya masing-masing. Pada *dashboard* PP disediakan fitur untuk memantau kinerja AYD nya. Untuk melakukan pemantauan kinerja AYD, PP dapat langsung memilih nama AYD dalam daftar yang kemudian sistem akan langsung mengarahkan kedalam *dashboard* AYD.

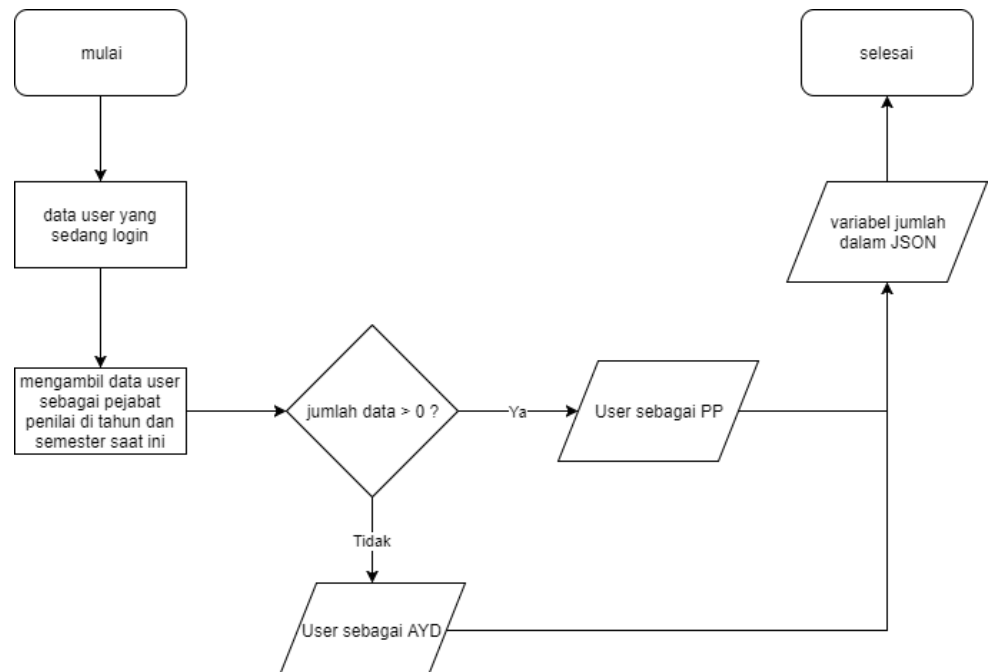


Gambar 4. 6 Alur Dashedboard PP

4.4 Implementasi dan Pembahasan

Dalam aplikasi android SMK-Online terdapat 2 jenis *user* yaitu *user* sebagai AYD dan *user* sebagai AYD sekaligus PP. Maka dari itu, *web service* harus membuat penanda *user* apakah sebagai AYD saja atau juga sebagai PP.

4.3.1 Dashboard Aplikasi Android SMK-Online



Gambar 4. 7 Flowchart User AYD PP

Resource yang digunakan yaitu <http://domain.com/api-key-point/users> dengan method GET. Dimulai dengan proses mendapatkan informasi *user* yang sedang *login*. Untuk mendapatkan jumlah anggota yang dinilai (AYD) dari *user* tersebut, dapat diketahui dari jumlah anggota yang memilih *user* tersebut sebagai pejabat penilainya di semester dan tahun saat ini. Kemudian dihitung jumlahnya lalu dimasukkan kedalam variabel tertentu dalam JSON.

Berikut hasil implementasi *web service* beserta *resource* nya :

a) *User* sebagai AYD

```
{
  "sukses": true,
  "pesan": "berhasil",
  "data": {
    "num_ayd": 0,
  }
}
```



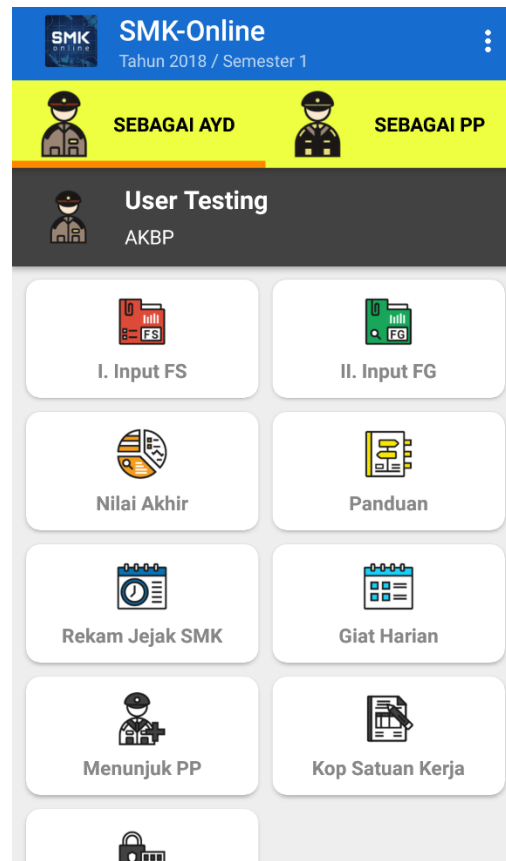
Gambar 4. 9 Tampilan Android *User* sebagai AYD

b) *User* sebagai AYD dan PP



Gambar 4. 10 JSON *User* sebagai AYD dan PP

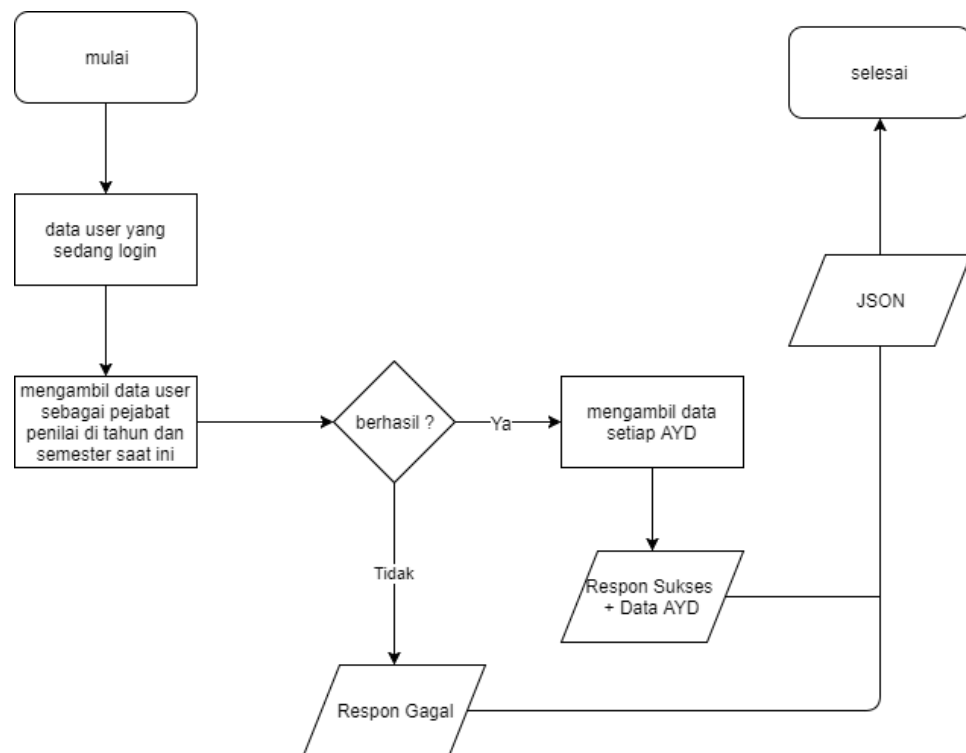
Sebagai contoh *user* yang memiliki variabel **num_ayd** bernilai 3 yang artinya *user* tersebut memiliki 3 anggota yang dinilai (AYD) atau terdapat 3 anggota yang memilih *user* tersebut sebagai PP. Data tersebut didapat dari *resource* <http://domain.com/api-key-point/users>. Sehingga tampilan android SMK-Online menjadi seperti pada gambar berikut :



Gambar 4. 11 Tampilan Android *User* sebagai AYD dan PP

4.3.2 *Monitoring Kinerja AYD oleh PP*

Bagi *user* yang berperan menjadi PP diwajibkan untuk memantau kinerja AYD nya. Maka dari itu aplikasi android SMK-Online harus menyediakan tampilan yang dapat memberikan informasi / status kinerja AYD selama masa penilaian kinerja berlangsung.



Gambar 4. 12 Flowchat Mengambil Data AYD

Gambar 4.12 merupakan flowchart dengan method GET untuk mendapatkan daftar AYD dari PP. Dimulai dengan mendapatkan data *user* yang sedang login yaitu *user* sebagai PP. Dilanjutkan proses untuk mendapatkan data AYD dari PP pada tahun dan semester saat ini. Jika berhasil, maka dilakukan proses pengulangan agar mendapatkan data setiap AYD (Lampiran 2). Agar data AYD yang dikirim melalui JSON tidak berat, maka *web service* melakukan *selection* data dengan memanfaatkan fungsi yang dibuat di Model Laravel (Lampiran 2). *Web service* ini diwakili dalam resource <http://domain.com/api-key-point/pejabat/{id-pp}/anggota>.

Contoh struktur data yang dikirim melalui JSON seperti dibawah ini :


```

{
  "sukses": true,
  "pesan": "Berhasil Mendapatkan Data Anggota",
  "data": [
    {
      "column_1": "column_1_A",
      "column_2": "column_2_A",
      "column_3": "column_3_A",
      "column_4": "column_4_A",
      "column_5": "column_5_A",
      "column_6": "column_6_A",
    },
    {
      "column_1": "column_1_B",
      "column_2": "column_2_B",
      "column_3": "column_3_B",
      "column_4": "column_4_B",
      "column_5": "column_5_B",
      "column_6": "column_6_B",
    },
    {
      "column_1": "column_1_C",
      "column_2": "column_2_C",
      "column_3": "column_3_C",
      "column_4": "column_4_C",
      "column_5": "column_5_C",
      "column_6": "column_6_C",
    }
  ]
}

```

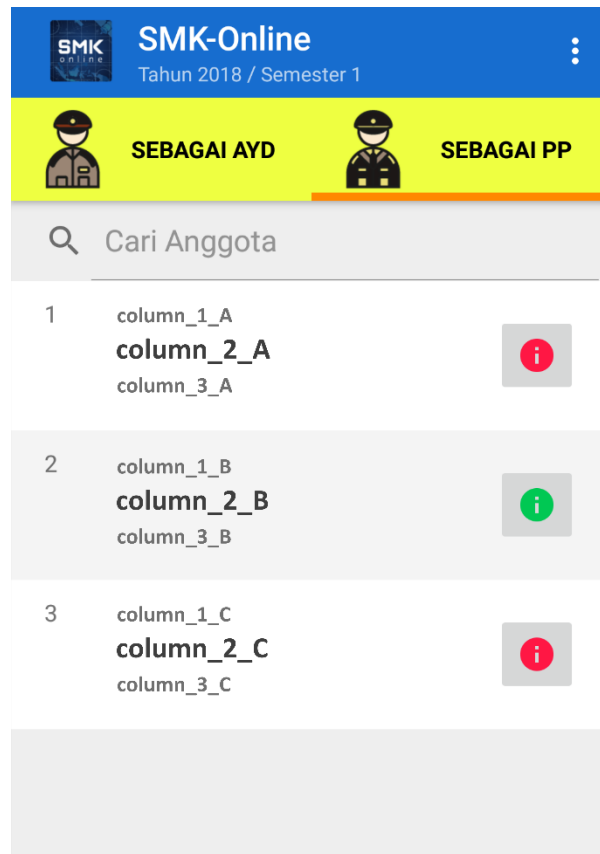
Gambar 4. 13 JSON Daftar AYD

Berikut pengecekan data yang muncul di tampilan aplikasi web SMK-Online :

SMK-Online						
HOME SEBAGAI PEJABAT PENILAI						
User Testing () Keluar						
Daftar Anggota						
Show 10 entries			Search:			
NO	NAMA AYD	PANGKAT	NRP	JABATAN	SATUAN KERJA	PENILAIAN AYD
1	column_1_A	column_2_A	column_3_A	column_4_A	column_5_A	Formulir Penilaian AYD
2	column_1_B	column_2_B	column_3_B	column_4_B	column_5_B	Formulir Penilaian AYD
3	column_1_C	column_2_C	column_3_C	column_4_C	column_5_C	Formulir Penilaian AYD
Showing 1 to 3 of 3 entries						Previous 1 Next

Gambar 4. 14 Daftar AYD pada web SMK-Online

Aplikasi android SMK-Online dapat menggunakan data pada JSON di Gambar 4.12 yang menampilkan 3 data AYD untuk ditampilkan ke dalam *dashboard* PP berupa daftar AYD :



Gambar 4. 15 Tampilan Android Daftar AYD di dashboard PP

Pada gambar diatas merupakan tampilan pada aplikasi Android SMK-Online yang memberikan informasi daftar anggota yang dinilai oleh pejabat penilai. Untuk mempermudah PP dalam memantau kinerja AYD, maka dalam tampilan aplikasi android SMK-Online diberikan penanda warna. *Web service* telah merancang 3 index nilai warna sebagai penanda status setiap unit penilaian kinerja yang diwakili dengan variabel **nilai** :

- nilai = 0 untuk abu-abu, artinya AYD belum mengisi atau belum mengisi lengkap setiap penilaiannya. AYD dan PP belum menandatangani.

- nilai = 1 untuk warna merah, artinya AYD telah mengisi dan menyelesaikan setiap penilaiannya. AYD telah menandatangani namun PP belum.
- nilai = 2 untuk warna hijau, artinya AYD telah mengisi dan menyelesaikan setiap penilaiannya. AYD dan PP telah menandatangani.

Perlu diketahui sesuai dengan PERKAP SMK-Online [7], selama 1 semester berjalan kinerja yang wajib dilakukan dan diselesaikan yaitu untuk faktor spesifik dan faktor generik. Tugas tambahan dan Giat harian tidak harus ada dalam 1 semester namun akan menjadi nilai tambah jika kedua faktor ini dikerjakan. *Web service* status AYD tersebut akan mempermudah PP dalam memantau kinerja setiap AYD nya. Pada a) *Status AYD Belum Menyelesaikan Kinerja* (Gambar 4.16) terlihat 3 warna di setiap pesan yaitu abu-abu, merah dan hijau. Sedangkan pada b) *Status AYD Telah Menyelesaikan Kinerja* (Gambar 4.17) hanya terlihat 1 warna yaitu hijau yang menandakan bahwa status AYD telah menyelesaikan seluruh kewajiban kinerjanya dan PP telah menyetujui seluruh kinerja AYD. Berikut realisasi dari implementasi status *web service* dalam tampilan aplikasi android SMK-Online :

a) Status AYD Belum Menyelesaikan Kinerja

PROGRES ANGGOTA

Faktor Spesifik

- ☒ Kontrak Kerja
Anda telah menyetujui
- ☒ Capaian Kontrak Kerja Triwulan 1
Anda telah menyetujui
- ☒ Capaian Kontrak Kerja Triwulan 2
Anda telah menyetujui
- ☐ Tugas Tambahan
Anggota belum menyelesaikan

Faktor Generik

- ☐ Nilai Penghargaan
Anda belum menyetujui

Kegiatan Harian

- 0 Sudah Diverifikasi
- 0 Belum Diverifikasi

Gambar 4. 16 Status AYD Belum Menyelesaikan Kinerja

b) Status AYD Sudah Menyelesaikan Kinerja

PROGRES ANGGOTA

Faktor Spesifik

- ☒ Kontrak Kerja
Anda telah menyetujui
- ☒ Capaian Kontrak Kerja Triwulan 1
Anda telah menyetujui
- ☒ Capaian Kontrak Kerja Triwulan 2
Anda telah menyetujui
- ☒ Tugas Tambahan
Anda telah menyetujui

Faktor Generik

- ☒ Nilai Penghargaan
Anda telah menyetujui

Kegiatan Harian

0 Sudah Diverifikasi
0 Belum Diverifikasi

Gambar 4. 17 Status AYD Telah Menyelesaikan Kinerja

4.3.3 Implementasi Teknik Analisis Data dan Metode pada PKA

Salah satu modul terpenting dalam sistem manajemen kinerja anggota POLRI yaitu penilaian Perilaku Kerja Anggota (PKA) yang terdiri dari 9 atau 10 faktor kinerja yaitu :

1. Kepemimpinan
2. Orientasi pelayanan
3. Komunikasi
4. Pengendalian emosi
5. Integritas
6. Empati
7. Komitmen terhadap organisasi
8. Inisiatif

9. Disiplin

10. Kerjasama

PKA sebagaimana telah dijelaskan dalam PERKAP SMK [7] merupakan indikator penilaian yang terkait dengan faktor kinerja dan standar kinerja yang telah ditentukan, yang diberlakukan sama untuk seluruh anggota POLRI. Penilaian PKA dilakukan oleh PP dan RK terhadap AYD sesuai dengan kriteria yang ditentukan. Rentang nilai PKA ditentukan dari angka 0-100. Khusus penilaian kepemimpinan (a) hanya dilakukan bagi AYD yang menduduki jabatan struktural.

Berdasarkan PERKAP terhadap PKA tersebut, penulis dapat menyimpulkan sebagai berikut :

1. Penilaian PKA dilakukan oleh 2 *user* yaitu pejabat penilai (PP) terhadap anggota yang dinilainya (AYD) dan rekan kerja (RK) dari AYD. Jadi dapat disimpulkan bahwa setiap AYD mendapatkan penilaian PKA dari PP dan RK nya.
2. Bagi yang menduduki jabatan struktural mendapatkan penilaian PKA sebanyak 10 faktor kinerja kecuali bagi yang menduduki jabatan non struktural hanya mendapatkan penilaian PKA sebanyak 9 faktor kinerja (kepemimpinan tidak termasuk).
3. PKA dinilai dengan range 0 – 100 yang dibagi ke dalam 5 standar kerja yaitu :
 - Standar 1 : 91 – 100
 - Standar 2 : 76 – 90
 - Standar 3 : 61 – 75
 - Standar 4 : 51 – 60
 - Standar 5 : 50 ke bawah

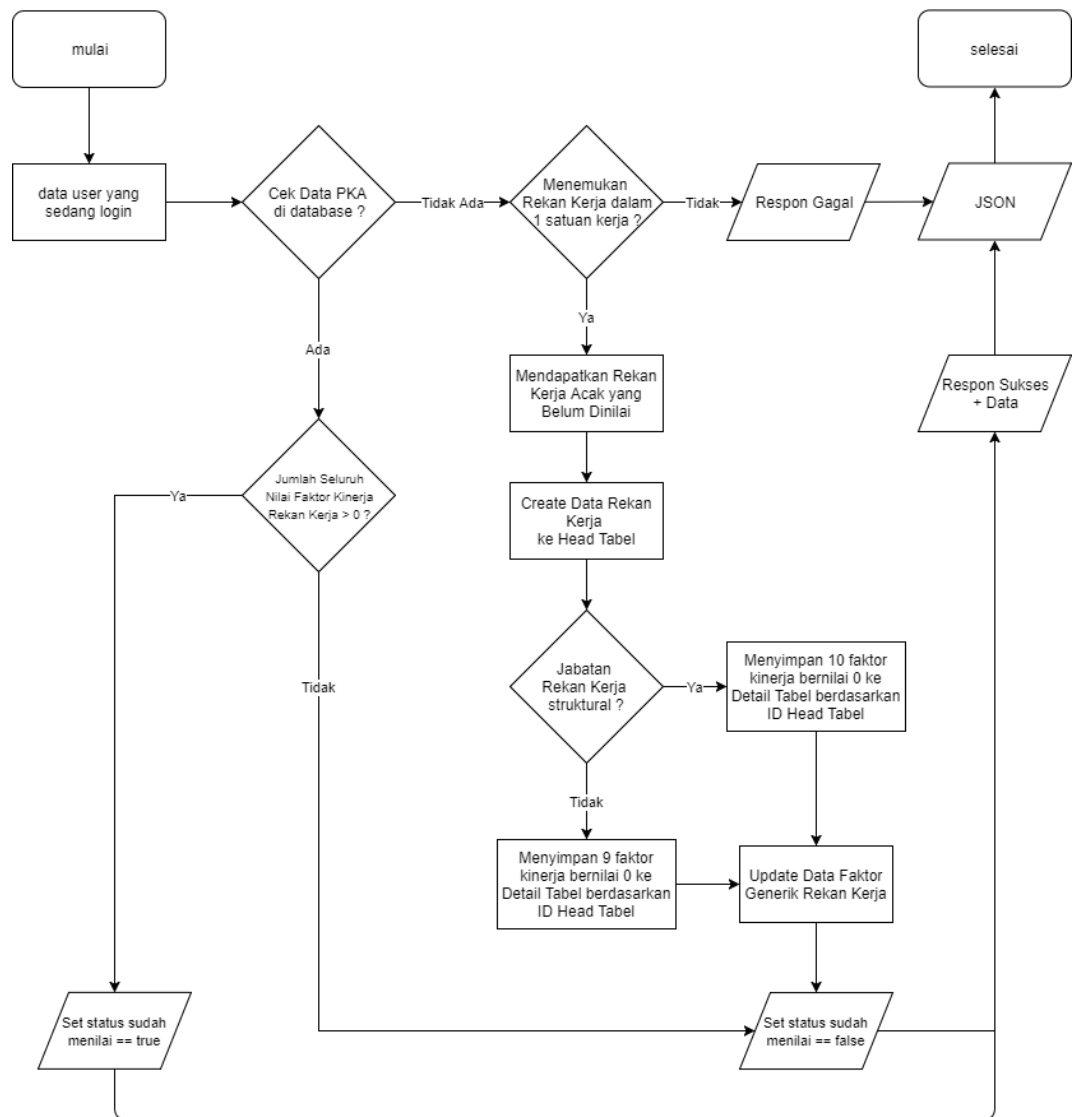
Bagi AYD diwajibkan melakukan penilaian Perilaku Kerja Anggota (PKA) milik Rekan Kerja (RK) yang **memiliki pangkat dan satuan kerja yang sama**. Maka dari itu untuk membedakan jenis PKA apakah dilakukan

oleh AYD terhadap RK nya atau PP terhadap AYD nya, maka *web service* menyediakan variabel **jenis**. *Web service* menerima nilai jenis ini dari *request* yang dikirim oleh aplikasi android SMK-Online. Nilai dari variabel jenis ini yaitu :

- jenis = 1 untuk penilaian PKA yang dilakukan PP kepada AYD nya. Dalam hal ini PP sebagai penilai dan AYD sebagai yang dinilai.
- jenis = 2 untuk penilaian PKA yang dilakukan AYD kepada RK nya. Dalam hal ini AYD sebagai penilai dan RK sebagai yang dinilai.

Sesuai dengan PERKAP SMK-Online, untuk mendapatkan RK dengan cara **acak** namun masih dalam satu pangkat dan satuan kerja penilai. Dalam hal mencari RK ini, *web service* memanfaatkan beberapa fungsi yang dibuat oleh tim *web* SMK-Online. Ini dilakukan sesuai dengan metode yang telah dijelaskan pada BAB III.

Proses pengacakan RK ini dilakukan ketika AYD pertama kali membuka menu **Input Nilai PKA Rekan Kerja** di aplikasi android maupun web. Jadi data RK akan otomatis tersimpan kedalam tabel *Head* dan jika *user* membuka kembali menu tersebut, maka akan otomatis muncul penilaian yang ditujukan pada RK milik AYD tersebut. Setiap AYD memiliki 1 RK. Unit *web service* ini terdapat pada *resource* <http://domain.com/api-key-point/{id}/pka/menilai> dengan method GET. Namun dalam method GET ini juga berlaku proses pengiriman data ke server (POST). Proses penyimpanan dalam method GET ini diperbolehkan karena hasil akhir dari *resource* adalah mengirimkan data dalam bentuk JSON. Berikut flowchart nya :



Gambar 4. 18 Flowchart Mendapatkan Rekan Kerja

a) Jika Belum ada data PKA Rekan Kerja di Database

Dalam proses mendapatkan Rekan Kerja terdapat pemanfaatan fungsi global kedalam fungsi *web service* yaitu ***findRekanKerja()*** untuk menemukan data rekan kerja dalam satuan kerja dan pangkat yang sama dan ***findPkaRk()*** untuk mengecek apakah rekan kerja yang didapat dari proses acak sudah dipilih oleh anggota lain (Lampiran 3). Kedua fungsi ini dapat mempermudah dalam pembangunan *web service* sehingga dapat lebih efisien.

Selain itu dalam proses ini menerapkan model penyimpanan data *Head – Detail* (Lampiran 4) yaitu dimulai dengan menyimpan data *head* di tabel **Head Model_perilaku_head**. Kemudian dilanjutkan dengan menyimpan data *detail* di tabel **Model_perilaku_detail** dengan memanfaatkan id dari *head* tabel. Jika status jabatan Rekan Kerja adalah struktural (*column_status* = 1), maka mendapatkan 10 faktor kinerja (*id_fk* dimulai dari 2). Sebaliknya jika non struktural (*column_status* = 0), maka mendapatkan 9 faktor kinerja yang tidak termasuk kepemimpinan (*id_fk* dimulai dari 2). Semua nilai faktor kinerja bernilai 0.

PKA merupakan salah satu aspek dari penilaian faktor generik. Maka dari itu setiap ada *progress* penilaian harus selalu *up-to-date* terhadap data di tabel Faktor Generik. *Web service* memiliki peran dalam memproses hal tersebut (Lampiran 5). Secara otomatis *user* sudah memiliki data rekan kerja namun belum menilai PKA rekan kerjanya, maka dari itu data yang dikirim dalam bentuk JSON array memiliki variabel **sudah_menilai** bernilai *false* (Lampiran 6).

b) Jika Sudah Ada Data PKA Rekan Kerja di Database

Web Service hanya melakukan pengecekan variabel **sudah_menilai** dengan nilai **true** jika seluruh faktor kinerja telah diberi nilai (tidak sama dengan 0) dan data **rekan_kerja** langsung diambil dari database (Lampiran 6).

Contoh struktur JSON yang dikirim jika AYD baru pertama kali mendapatkan Rekan Kerja dan belum melakukan penilaian PKA (Gambar 4.19) . Kemudian kondisi AYD telah mendapatkan Rekan Kerja dan sudah melakukan penilaian PKA (Gambar 4.20) :

```

1 {
2   "sukses": true,
3   "pesan": "Belum Menilai dan Sudah Memiliki Rekan Kerja",
4   "data": {
5     "column_1": "column_1_A",
6     "column_2": "column_2_A",
7     "column_3": "column_3_A",
8     "column_4": "column_4_A",
9     "column_5": "column_5_A",
10    "sudah_menilai": false,
11    "punya_rk": true,
12    "rekan_kerja": {
13      "column_1": "column_1_B",
14      "column_2": "column_2_B",
15      "column_3": "column_3_B",
16      "column_4": true
17    },
18    "ttd": {}
19  },
20 }
33
34 }

```

Gambar 4. 19 JSON User telah Mendapatkan RK dan Belum Menilai

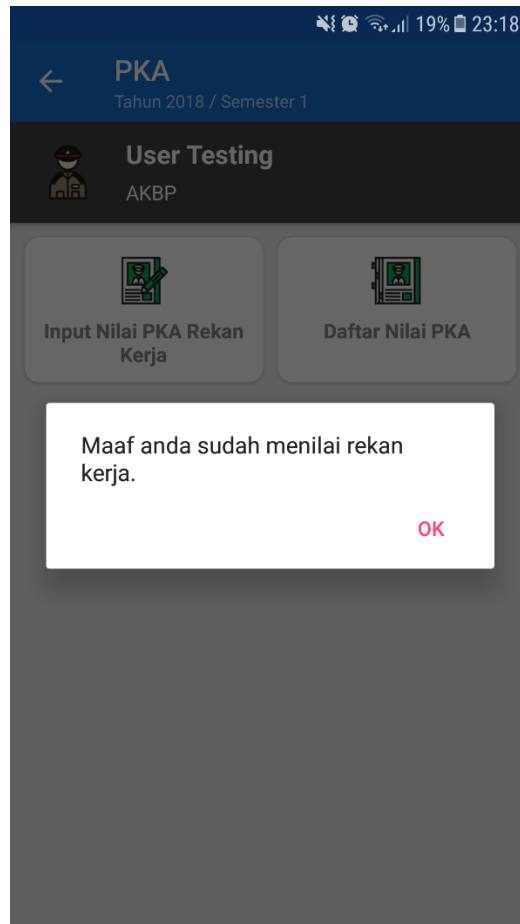
```

1 {
2   "sukses": true,
3   "pesan": "Sudah Menilai dan Sudah Memiliki Rekan Kerja",
4   "data": {
5     "column_1": "column_1_A",
6     "column_2": "column_2_A",
7     "column_3": "column_3_A",
8     "column_4": "column_4_A",
9     "column_5": "column_5_A",
10    "pka": {
11      "sudah_menilai": true,
12      "punya_rk": true,
13      "rekan_kerja": {
14        "column_1": "column_1_B",
15        "column_2": "column_2_B",
16        "column_3": "column_3_B",
17        "column_4": true
18      }
19    },
20    "ttd": {}
21  },
22 }
33
34 }

```

Gambar 4. 20 JSON User telah Mendapatkan RK dan Sudah Menilai

Berikut tampilan realisasi dari proses mendapatkan Rekan Kerja pada aplikasi android SMK-Online :



Gambar 4. 21 Tampilan Android AYD telah Memiliki dan Menilai Rekan Kerja

Pesan dialog pada gambar diatas akan muncul jika AYD membuka kembali menu **Input Nilai PKA Rekan Kerja** dimana android memanfaatkan nilai dari variabel **sudah_menilai** dan **punya_rk** yang bernilai **true**. Artinya AYD telah memiliki RK dan melakukan penilaian PKA terhadap RK nya.

Form PKA
Tahun 2018 / Semester 1

Mabes Polri
Mabes Polri

Penilaian terhadap anggota/rekan kerja pada aspek faktor kerja dibawah ini.
Berikan nilai antara 0 sampai dengan 100.
[Sentuh form input nilai untuk mengisi nilai anggota.](#)

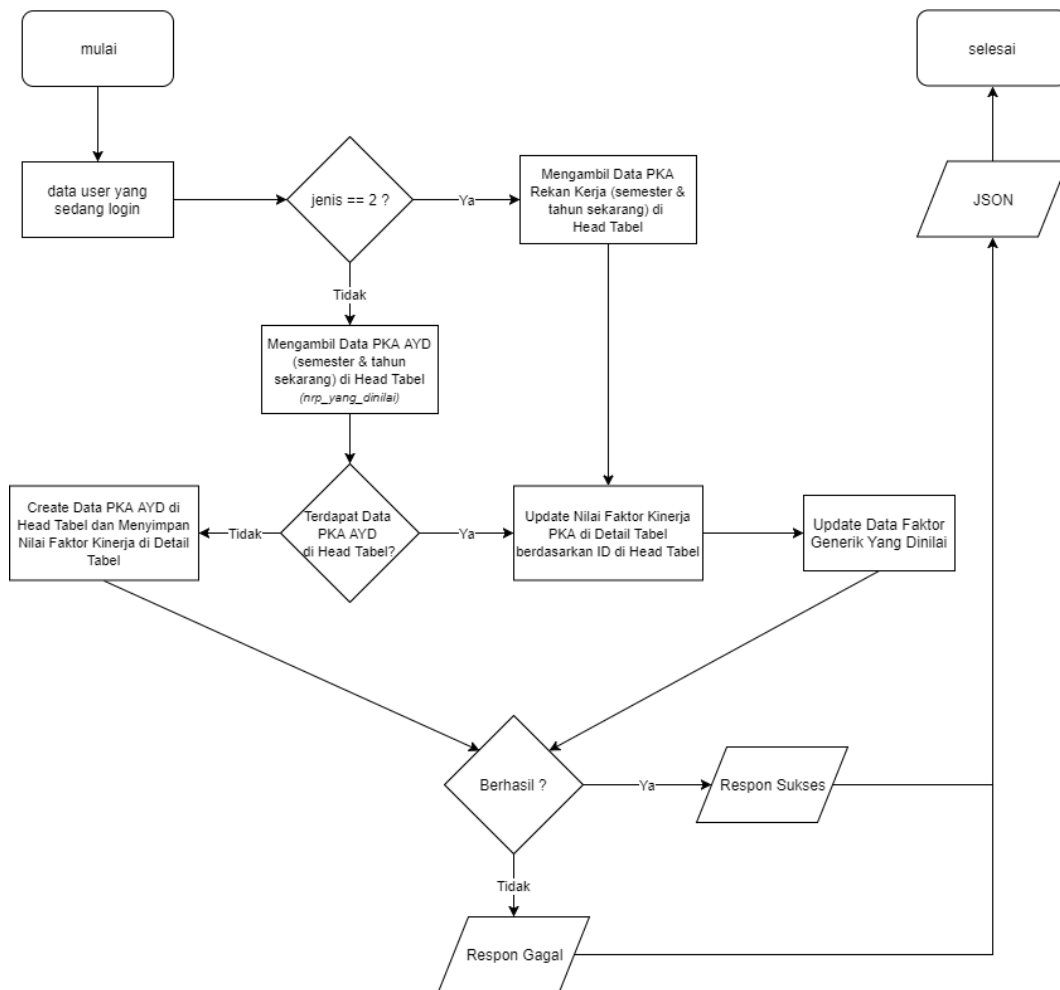
Aspek Penilaian

- Input Nilai Kepemimpinan
- Input Nilai Orientasi Pelayanan
- Input Nilai Komunikasi
- Input Nilai Pengendalian Emosi
- Input Nilai Integritas

Gambar 4. 22 Tampilan Android AYD Telah Memiliki dan Belum Menilai Rekan Kerja

Tampilan pada gambar diatas akan muncul jika AYD telah memiliki rekan kerja namun belum melakukan penilaian PKA. Aplikasi Android memanfaatkan nilai dari variabel **sudah_menilai = false** dan **punya_rk = true**. Maka akan muncul tampilan untuk menginput 9 atau 10 faktor kinerja Rekan Kerja.

Kemudian untuk menyimpan penilaian PKA menggunakan method POST yang akan mengirimkan data ke database SMK-Online melalui *resource* <http://domain.com/api-key-point/{id}/pka> .Berikut flowchart dari proses penilaian PKA :



Gambar 4. 23 Flowchart Penilaian PKA

Web service menerima nilai dari *param jenis* yang dikirim oleh android. *Param jenis* digunakan untuk mengambil data PKA yang dinilai dari *head* tabel. Khusus untuk jenis 1 yaitu *user* sebagai pejabat penilai, *web service* menerima *params ayd_yang_dinilai* dari android yang digunakan untuk mendapatkan id ayd yang dinilainya. Untuk penilaian PKA Rekan Kerja, dalam tahap ini status AYD sudah memiliki rekan kerja karena telah melalui proses mendapatkan rekan kerja ketika membuka menu **Input Nilai PKA Rekan Kerja**.

Jika PP belum mendapatkan data PKA AYD nya, maka *web service* melakukan proses *create* data AYD di *head* tabel dan menyimpan seluruh nilai faktor kinerja di *detail* tabel seperti pada proses sebelumnya. Sedangkan

jika terdapat data PKA di *head* tabel, maka *web service* melakukan update nilai faktor kinerja pada *detail* tabel, yang awalnya bernilai 0 menjadi sesuai dengan nilai yang diinputkan/ dinilai. *Web service* menerima 9-10 params sesuai dengan jumlah faktor kinerja yang dinilai.

Di setiap fungsi *web service*, wajib menerapkan model *try-catch*. Di dalam proses *create* atau *update* nilai PKA di *resource* dengan *method* POST kali ini, *try web service* memanfaatkan *method* **DB::beginTransaction()** dari *DB facade* Laravel. Fungsi ini berfungsi sebagai penanda awal dimulainya transaksi pengiriman data yang dilakukan. Kemudian *web service* menghitung rata-rata dari seluruh nilai yang diinput dan melakukan *update* total nilai PKA di *head* tabel dan tabel Faktor Generik yang dinilai.

Setelah semua proses penilaian PKA selesai, kemudian *web service* melakukan *commit* dengan menggunakan *method* **DB::commit()**. *Method* ini untuk mengunci seluruh proses yang dimulai dari *DB::beginTransaction()* kemudian melakukan penyimpanan data ke database. Jika terjadi proses yang belum selesai atau terjadi eror, maka akan diarahkan pada *catch Exception*. Didalam kondisi *catch* ini terdapat *method* **DB::rollback()** yang berfungsi untuk mengembalikan data ke pada kondisi awal atau data yang di *commit* tidak jadi tersimpan ke dalam database. Ketiga *DB facade* ini wajib diterapkan pada setiap *resource* dengan *method* POST (Lampiran 7).

Berikut contoh penilaian PKA melalui *postman* terhadap Rekan Kerja yang memiliki jabatan struktural atau terdapat 10 faktor kinerja yang harus dinilai dan respon *web service* :

The screenshot shows the 'Body' tab in Postman with 'x-www-form-urlencoded' selected. A table lists the parameters to be sent in the request body.

Key	Value
<input checked="" type="checkbox"/> param_1	80
<input checked="" type="checkbox"/> param_2	89
<input checked="" type="checkbox"/> param_3	70
<input checked="" type="checkbox"/> param_4	81
<input checked="" type="checkbox"/> param_5	80
<input checked="" type="checkbox"/> param_6	70
<input checked="" type="checkbox"/> param_7	80
<input checked="" type="checkbox"/> param_8	70
<input checked="" type="checkbox"/> param_9	70
<input checked="" type="checkbox"/> param_10	78
<input checked="" type="checkbox"/> jenis	2
<input type="checkbox"/> nrp_yang_dinilai	

Gambar 4. 24 Penilaian PKA Rekan Kerja Melalui *Postman*

The screenshot shows the 'JSON' response view in Postman. The response is a JSON object indicating a successful update.

```

{
  "sukses": true,
  "pesan": "Penilaian Di Update"
}

```

Gambar 4. 25 Respon *Web Service* Penilaian PKA Rekan Kerja

Sedangkan bagi PP yang melakukan penilaian PKA terhadap AYD nya, terdapat param `nrp_yang_dinilai` untuk menampung id AYD yang akan dinilai dan respon *web service* :

Authorization	Headers (2)	Body	Pre-request Script	Tests
<input type="radio"/> form-data	<input checked="" type="radio"/> x-www-form-urlencoded	<input type="radio"/> raw	<input type="radio"/> binary	
Key		Value		
<input checked="" type="checkbox"/>	param_1	77		
<input checked="" type="checkbox"/>	param_2	60		
<input checked="" type="checkbox"/>	param_3	72		
<input checked="" type="checkbox"/>	param_4	80		
<input checked="" type="checkbox"/>	param_5	78		
<input checked="" type="checkbox"/>	param_6	75		
<input checked="" type="checkbox"/>	param_7	82		
<input checked="" type="checkbox"/>	param_8	72		
<input checked="" type="checkbox"/>	param_9	75		
<input checked="" type="checkbox"/>	param_10	83		
<input checked="" type="checkbox"/>	jenis	1		
<input checked="" type="checkbox"/>	nrp_yang_dinilai	id_AYD		

Gambar 4. 26 Penilaian PKA AYD oleh PP Melalui *Postman*

```

Pretty Raw Preview JSON
{
  "sukses": true,
  "pesan": "Penilaian Di Simpan"
}

```

Gambar 4. 27 Respon *Web Service* Penilaian PKA AYD oleh PP

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Setelah melakukan analisis, perancangan dan melakukan implementasi *web service* dengan metode RESTful pada Sistem Manajemen Kinerja (SMK) Online, dapat diperoleh kesimpulan sebagai berikut :

- a. Berdasarkan hasil penelitian, REST/RESTful *web service* yang diimplementasikan dapat mewujudkan integrasi data pada SMK-Online. *Web service* yang dibangun berhasil mengintegrasikan data yang diambil dari Sistem Informasi Personel POLRI (SIPP) dengan data yang terdapat di SMK-Online.
- b. RESTful *Web service* yang dibangun berhasil menyediakan *resource* yang sama dan dapat diakses bersama dengan data yang sinkron dan berhasil diimplementasikan pada SMK-Online berbasis android sehingga mempermudah penilaian kinerja yang dilakukan oleh personel POLRI.

5.2 Saran

Saran penulis terhadap pengembangan *web service* SMK-Online adalah sebagai berikut :

1. Disediakan fungsi bersama (global) yang dapat dimanfaatkan untuk aplikasi web dan *web service*. Jika terjadi perubahan pada unit tertentu, maka semua fungsi akan mengikuti dengan perubahan tersebut baik pada *web service* maupun *web*. Saat ini setiap *resource web service* yang dibangun belum sepenuhnya efisien. Masih ada proses dari fungsi web yang dibangun kembali pada proses *web service*.

2. Diterapkannya *repository design pattern* Laravel. *Web Service* yang dibangun masih belum bersifat *clean code*. Beberapa fungsi menghabiskan banyak baris *code* di dalam *Controller*. Struktur dalam membangun *web service* dapat mempengaruhi waktu dan ukuran data yang dikirim melalui *JSON*.
3. Untuk keamanan data pada penelitian ini menggunakan *JSON Web Token (JWT) Authentication* dalam setiap akses penilaian kinerja anggota POLRI. Sedangkan *web service* dibangun didalam *framework* Laravel. Untuk penelitian selanjutnya dapat dikembangkan dengan *OAuth2* dan *framework* lainnya.

DAFTAR PUSTAKA

- [1] K. N. R. Indonesia, “Paparan SMK-Online,” KAPOLRI, Jakarta, 2017.
- [2] E. S. a. K. Mustofa, “KEBUTUHAN WEB SERVICE UNTUK SINKRONISASI DATA ANTAR SISTEM INFORMASI DALAM E-GOV DI PEMKAB BANTUL YOGYAKARTA,” *Tek. Inform - STMIK Bandung*, 2012.
- [3] I. M. d. P. J. Siregar, *Membongkar Teknologi Web Service*, Yogyakarta: Gava Media, 2012.
- [4] D. Dr. Rathod, “PERFORMANCE EVALUATION OF RESTFUL WEB SERVICES AND SOAP / WSDL WEB SERVICES,” *International Journal of Advanced Research in Computer Science*, vol. 8, p. 7, 2017.
- [5] N. I. K. d. I. Masyhur, “Rancang Bangun Web Service (Studi Kasus : Layanan SIM Inventaris Barang),” *Prosiding Seminar Nasional Komunikasi dan Informatika (SNKI)*, 2015.
- [6] P. F. M. D. d. I. A. Tanaem, “RESTFul Web Service Untuk Sistem Pencatatan Transaksi Studi Kasus PT. XYZ,” *Jurnal Teknik Informatika dan Sistem Informasi*, vol. 2, 2016.
- [7] K. N. R. Indonesia, “Peraturan KAPOLRI tentang Penilaian Kinerja Anggota POLRI dengan Sistem Manajemen Kinerja,” KAPOLRI, Jakarta, 2017.
- [8] L. d. R. S. Richardson, *RESTful Web Services*, United State of America: O'Reilly Media, Inc, 2011.
- [9] R. Awaludin, *Menyelami Framework Laravel*, Leanpub, 2016.
- [10] D. Rees, *Laravel : Code Bright*, Leanpub, 2013.

- [11] A. Pribadi Basuki, Konsep & Implementasi Pemrograman Laravel 5, Yogyakarta: CV. Lokomedia, 2016.
- [12] A. Sandi, “Codepolitan,” 2017. [Online]. Available: <https://www.codepolitan.com/mengenal-apa-itu-web-api-5a0c2855799c8>. [Diakses Sabtu Januari 2018].
- [13] K. N. R. Indonesia, “POLRI (Kepolisian Negara Republik Indonesia),” 2018. [Online]. Available: <https://www.polri.go.id>. [Diakses Selasa Januari 2018].
- [14] D. Kumar, Best Practices for Building RESTful Web Services, Bengaluru: Infosys, 2017.
- [15] L. Jason, “Github,” 2015. [Online]. Available: <https://github.com/dingo/api>. [Diakses Monday January 2018].
- [16] L. d. A. M. Richardson, RESTful Web APIs, United States of America: O'Reilly Media, Inc, 2013.

LAMPIRAN

Lampiran 1. Fungsi Login

```
public function login(Request $request)
{
    $credentials = $request->only('param_1', 'param_2', 'param_3');
    $var_1 = Model_user::where('param_1', '=', $request->param_1)->first();
    try {
        ❶ if(empty($var_1)){
            ❷ $user= json_decode(file_get_contents('http://domain.com/api-key-point-sipp/&param_1='.$request->param_1,true));
            ❸ if ($var_1[0]['column_1'] != "found")
            {
                // create user
                $data = new Model_user();
                $data->column_1 = $request->param_1;
                $sanggota->column_2 = $var_1[0]['column_1'];
                $sanggota->column_3 = \Carbon\Carbon::parse($var_1[0]['param_2'])->format('Y-m-d');
                $sanggota->column_4 = bcrypt($request->param_3);
                // save
                $data->save();
                $token = JWTAuth::attempt($credentials);
                return response()->json(['sukses'=>true,'pesan'=>'Berhasil Diregistrasi','token'=>$token]);
            }
        }
        else{
            ❹ $token = JWTAuth::attempt($credentials);
            ❺ if(!$token){
                return response()->json(['sukses' => false,'pesan'=>'Mohon Maaf, Kombinasi Tidak Cocok'], 401);
            }
            else{
                ❻ $this->update_data($var_1->param_1);
                return response()->json(['sukses'=>true,'pesan'=>'Berhasil Login','token'=>$token]);
            }
        }
    }
    catch(\Exception $e){}
```

Terdapat 3 *request* yang harus diinput oleh *user* yaitu *param_1*, *param_2* dan *param_3* yang dimasukan kedalam variabel *credentials*. Pertama kali *service* mencari data *user* dengan memanfaatkan *request* *param_1* yang diinput. Pada nomor 1 digunakan ketika kondisi tidak ada data *user* di database SMK-Online. Kemudian pada nomor 2 *web service* mengakses *internal service* API dari SIPP POLRI untuk mengecek ketersediaan data *user* di database SIPP. Masuk ke nomor 3 merupakan kondisi dimana nama *user* yang di *request* ditemukan di database SIPP. Lalu *web service* meng-create data *user* ke database SMK-Online sesuai dengan yang ada di data SIPP dan mengirimkan *response* token dalam bentuk JSON. Lain halnya jika data *user request* sudah ada di database SMK-Online, maka pada nomor 4 *credentials* yang disimpan di cek pada class JWTAuth *attempt* apakah terdapat kecocokan pada *credentials* tersebut. Jika terdapat ketidakcocokan, maka memasuki kondisi pada nomor 5. Jika *credentials* cocok, maka langsung diarahkan ke kondisi nomor 6 dimana terdapat fungsi **update_data()**. Fungsi ini bertujuan untuk melakukan update data *user*. Jika terdapat perubahan data dari SIPP, maka *web service* akan mengupdate data tersebut ke dalam database SMK-Online. Kemudian *web service* mengirimkan token *response* berupa JSON.

Lampiran 2. Fungsi untuk Mendapatkan Data AYD

```
public function hasAyd($id)
{
    $user = Model_user::where('column_1', $id)->first();
    try{
        1 $var_pp = Model_pp::where('column_1','=', $id)
            ->where('tahun','=', $this->tahun)
            ->where('semester','=', $this->semester)
            ->get();

        $cek = $user->numAYD();
        if(empty($cek)){
            //get AYD
            $i=0;
            2 foreach ($var_pp as $pp) {
                3 $data[$i]=$pp->getAyd;
                $i++;
            }
            $pesan = "Berhasil Mendapatkan Data Anggota";
        }

        return response()->json([
            'sukses'=>true,
            'pesan'=> $pesan,
            'data' => $data,
        ]);
    }catch(\Exception $e){
    }
}
```

Nomor 1 bertujuan untuk mendapatkan data AYD pada tahun dan semester saat ini dari tabel Model_pp dimana **column_1** sama dengan id dari *user* yang sedang *login* tersebut dengan memanfaatkan fungsi PHP **get()**. Nomor 2 merupakan kondisi pengulangan yang diwakili oleh fungsi PHP **foreach** agar dapat memberikan informasi AYD secara berulang sampai AYD terakhir. Nomor 3 pemanggilan fungsi **getAyd()** pada model Model_pp untuk menyeleksi data AYD yang dikirim.

```
namespace App;

use Illuminate\Database\Eloquent\Model;
class Model_pp extends Model
{
    public function getAyd(){
        return $this->hasOne('App\Model_user', 'id', 'id')
            ->select('column_1','column_2','column_3','column_4','column_5','column_6');
    }
}
```

Lampiran 3. Pemanfaatan Fungsi Global

```
public function getPkaMenilai($id)
{
    try {
        $user = Model_user::where('column_1', $id)->first();
        $pka = $this->getMenilaiPka($user->id);//mengecek rekan kerja

        if(empty($pka)){
            $dataRekanKerja = $this->pka->findRekanKerja($user->column_1,$user->column_2,$user->id);
            if(count($dataRekanKerja)==0){
                return response()->json(['sukses'=>false,'pesan'=>'Maaf Belum ada rekan kerja dalam 1 satuan kerja!']);
            }

            //Proses Mendapatkan ekan Kerja secara Acak
            $getRK = array();
            foreach ($dataRekanKerja as $data) {
                $rekanKerjaToArray = $dataRekanKerja->toArray();
                $getRekanKerja = array_rand($rekanKerjaToArray);
                $dataAYD = $rekanKerjaToArray[$getRekanKerja];
                $cek = $this->pka->findPkaRk($dataAYD['id'], $this->tahun, $this->semester);
                if (count($cek) == 0) {
                    $getRK = $dataAYD;
                    break;
                }
            }
        }
    }
}
```

Lampiran 4. Penerapan *Head-Detail* Tabel

```
//Save Data RK Baru ke table Model_perilaku_head
$perilaku = new Model_perilaku_head();
$perilaku->column_1 = $getRK['column_1'];
$perilaku->column_2 = $user->id;
$perilaku->column_3 = 0;
$perilaku->column_jenis = 2;
$perilaku->column_status = $getRK['column_status'];
$perilaku->save();

//Membuat Data Detail Perilaku Rekan Kerja
$status_jabatan = $getRK['column_status'];
if($status_jabatan==0){
    $pka = Model_perilaku_head::where('column_1','=', $getRK['id'])
        ->where('jenis','=',2)
        ->where('semester','=', $this->semester)
        ->where('tahun','=', $this->tahun)
        ->first();

    $perilakuD = new Model_perilaku_detail();
    $perilakuD->id_pka = $pka->id;
    $perilakuD->id_fk = 1;
    $perilakuD->nilai = 0;
    $perilakuD->save();
    $perilakuD = new Model_perilaku_detail();
    $perilakuD->id_pka = $pka->id;
    $perilakuD->id_fk = 2;
    $perilakuD->nilai = 0;
    $perilakuD->save();
    $perilakuD = new Model_perilaku_detail();
    $perilakuD->id_pka = $pka->id;
    $perilakuD->id_fk = 3;
    $perilakuD->nilai = 0;
    $perilakuD->save();
    $perilakuD = new Model_perilaku_detail();
    $perilakuD->id_pka = $pka->id;
    $perilakuD->id_fk = 4;
    $perilakuD->nilai = 0;
    $perilakuD->save();
    $perilakuD = new Model_perilaku_detail();
    $perilakuD->id_pka = $pka->id;
    $perilakuD->id_fk = 5;
    $perilakuD->nilai = 0;
    $perilakuD->save();
}
```


Lampiran 5. Fungsi Update Faktor Generik

```
//update data pka rk ke tabel faktor generik rk
if(!empty($this->cekFg($getRK['id']))){
    $this->updateFg($getRK['id'],'fg_rk',0);
    $this->updateHitungFg($getRK['id']);
}
else{
    $this->createFg($getRK['id']);
    $this->updateFg($getRK['id'],'fg_rk',0);
    $this->updateHitungFg($getRK['id']);
}
```

Lampiran 6. Struktur JSON *User* Belum dan Sudah Menilai PKA RK

```
$data = array(
    'column_1'    => $user->id,
    'column_2'    => $user->column_2,
    'column_3'    => $user->column_3,
    'column_4'    => $user->column_4,
    'column_5'    => $user->column_5,
    'pka'         => array(
        'sudah_menilai' => false,
        'punya_rk'     => true,
        'rekan_kerja' => array(
            'column_1' => $getRK['id'],
            'column_2' => $getRK['column_2'],
            'column_3' => $getRK['column_3'],
            'column_4' => ($getRK['column_status'] == 0) ? false : true,
        ),
    ),
    'ttd' => array(
    );
$sukses = true;
$pesan = "Belum Menilai dan Sudah Memiliki Rekan Kerja";
return response()->json([
    'sukses' => $sukses,
    'pesan'  => $pesan,
    'data'   => $data,
]);
}
```

```

else{
    $sudah_menilai = Model_perilaku_detail::where('id_pka','=', $pka->id)->sum('nilai');
    $rk = Model_user::where('column_1','=', $pka->id)->first();
    $data = array(
        'column_1' => $user->id,
        'column_2' => $user->column_2,
        'column_3' => $user->column_3,
        'column_4' => $user->column_4,
        'column_5' => $user->column_5,
        'pka' => array(
            'sudah_menilai' => ($sudah_menilai==0)?false:true,
            'punya_rk' => true,
            'rekan_kerja'=>array(
                'column_1'=> $pka->id,
                'column_2'=> $rk->column_2,
                'column_3'=> $rk->column_3,
                'column_4'=>($rk->column_status==0)?false:true,
            ),
        ),
        'ttd'=>array(=
    );
    $sukses = true;
    $pesan = "Sudah Menilai dan Sudah Memiliki Rekan Kerja";
    return response()->json([
        'sukses' => $sukses,
        'pesan' => $pesan,
        'data' => $data,
    ]);
}
} catch (\Exception $e) {=
}

```

Lampiran 7. Penerapan DB *Facade* Laravel di Set PKA

```
else{
    try{
        DB::beginTransaction();
        if((int)$r->jenis==2){
            $ayd_rk = Model_perilaku_head::where('column_penilai','=', $id)
                ->where('semester','=', $this->semester)
                ->where('tahun','=', $this->tahun)
                ->where('jenis','=',2)
                ->first();

            $id_ayd = $ayd_rk->id;
        }
        else if((int)$r->jenis==1){
            $id_ayd = $r->ayd_yang_dinilai;
        }
        $ayd = Model_user::where('column_1','=', $id_ayd)->first();
        //mengambil data head pka
        $pka = Model_perilaku_head::where('column_1','=', $id_ayd)
            ->where('column_penilai','=', $id)
            ->where('semester',$this->semester)
            ->where('tahun',$this->tahun)
            ->where('jenis','=', (int)$r->jenis)
            ->first();

        //update PKA
        if(!empty($pka)){
            $perilaku = $pka;
            $status_jabatan = $ayd->column_status;
            if($status_jabatan==1){
                Model_perilaku_detail::where('id_pka','=', $perilaku->id)
                    ->where('id_fk','=',1)
                    ->update(['nilai'=>$r->fk_1]);
                Model_perilaku_detail::where('id_pka','=', $perilaku->id)
                    ->where('id_fk','=',2)
                    ->update(['nilai'=>$r->fk_2]);
                Model_perilaku_detail::where('id_pka','=', $perilaku->id)
                    ->where('id_fk','=',3)
                    ->update(['nilai'=>$r->fk_3]);
            }
        }
    }
}
```

```

//update jumlah nilai fg ke tabel faktor generik anggota yang dinilai
if(!empty($this->cekFg($id_ayd)){
    if((int)$r->jenis==1){
        $this->updateFg($id_ayd,'fg_pp',$rata_rata);
        $this->updateHitungFg($id_ayd);
    }
    else if((int)$r->jenis==2){
        $this->updateFg($id_ayd,'fg_rk',$rata_rata);
        $this->updateHitungFg($id_ayd);
    }
}
else{
    if((int)$r->jenis==1){
        $this->createFg($id_ayd);
        $this->updateFg($id_ayd,'fg_pp',$rata_rata);
        $this->updateHitungFg($id_ayd);
    }
    else if((int)$r->jenis==2){
        $this->createFg($id_ayd);
        $this->updateFg($id_ayd,'fg_rk',$rata_rata);
        $this->updateHitungFg($id_ayd);
    }
}
$pesan="Penilaian Di Update";
$sukses=true;
DB::commit();
return response()->json(['sukses' => $sukses,'pesan'=>$pesan]);
}
else{
}
} catch(\Exception $e){
    DB::rollBack();
    $pesan = "Error ".$e;
    $sukses = false;
    return response()->json(['sukses' => $sukses,'pesan'=>$pesan]);
}
}
}

```