

Web Service untuk Transaksi Data pada Aplikasi Sistem Informasi Kelembagaan Petani dengan Metode Rest

Fitrah Satyra Fajar Kusumah¹, Safaruddin Hidayat Al Ikhsan², Risky Adit Pratama³

Informatic Engineering, Ibn Khaldun University, Jl. Sholeh Iskandar, RT.01/RW.10, Kedungbadak, Kec. Tanah Sereal, Kota Bogor, Jawa Barat 16162
e-mail: ¹fitrah@uika-bogor.ac.id, ²safaruddin@uika-bogor.ac.id, ³risky.aditp06@gmail.com

Submitted Date: October 10th, 2021
Revised Date: July 28th, 2022

Reviewed Date: January 05th, 2022
Accepted Date: August 16th, 2022

Abstract

In the current era of technology, there are many systems that are integrated with each other even though they involve different platforms. To support this, it is necessary to have web service technology as an interaction mechanism between systems that can support interoperability for the benefit of data integration processes and future application development. In this study, a web service will be created for data transactions on the application of the farmer institutional information system with the REST method so that the interoperability process can run smoothly and make it easier for future application development. The purpose of this research is to apply the REST method to the web service of the Farmer Institutional Information System by using the Lumen Micro Framework as a technology to develop a web service. The output of this web service is in the form of JSON so that the communication process and data transactions in the Farmer Institutional Information System application run well according to the results of the tests carried out.

Keywords: Web Service; REST; Lumen; JSON

Abstrak

Di era teknologi saat ini banyak sekali sistem yang saling terintegrasi meskipun melibatkan *platform* yang berbeda-beda. Untuk mendukung hal tersebut perlu adanya sebuah teknologi *web service* sebagai prosedur interaksi antar sistem yang dapat menunjang interoperabilitas untuk kepentingan proses integrasi data dan pengembangan aplikasi kedepannya. Pada penelitian ini akan dibuat sebuah *web service* untuk transaksi data pada aplikasi sistem informasi kelembagaan petani dengan metode REST agar proses interoperabilitas dapat berjalan dengan lancar dan memudahkan untuk pengembangan aplikasi kedepannya. Tujuan dari penelitian adalah menerapkan metode REST pada *web service* Sistem Informasi Kelembagaan Petani dengan menggunakan *Lumen Micro Framework* sebagai teknologi untuk mengembangkan *web service*. Adapun *output* dari *web service* ini berupa JSON sehingga proses komunikasi dan transaksi data pada aplikasi Sistem Informasi Kelembagaan Petani berjalan dengan baik sesuai dengan hasil pengujian yang dilakukan.

Kata Kunci: *Web Service; REST; Lumen; JSON*

1 Pendahuluan

Sektor pertanian memegang peranan penting dalam pembangunan nasional. Dalam lima tahun terakhir sektor pertanian telah tumbuh rata-rata sebesar 3,4 persen (BAPPENAS, 2015). Melihat besarnya pengaruh sistem perekonomian nasional dan kesejahteraan masyarakat, maka strategi untuk meningkatkan produktifitas dan kualitas khususnya

di sektor pertanian menjadi sangat penting. Namun dari hal tersebut ada satu yang diabaikan oleh pemerintah yaitu kelembagaan petani. Lemahnya kelembagaan petani cenderung membuat petani bekerja secara individualistik dan petani tidak mampu merespon permintaan pasar (BAPPENAS, 2015). Dari adanya hal tersebut perlu adanya sebuah sistem informasi kelembagaan petani agar

petani dan kelompok tani termonitoring oleh pemerintah sehingga proses kebutuhan dan potensi komoditas yang dihasilkan petani dapat diinformasikan secara lebih luas dan proses penyaluran bantuan menjadi lebih mudah dan tepat. Pemanfaatan teknologi informasi dan komunikasi (TIK) dapat menjadi salah satu strategi penting untuk mewujudkan hal tersebut.

Oleh karena itu pemanfaatan TIK sangat penting di era teknologi saat ini. Sistem harus mampu saling terintegrasi sehingga proses transaksi data dapat berjalan dengan lancar meskipun saat terintegrasi tersebut harus melibatkan *platform* yang berbeda-beda. Untuk mendukung komunikasi data antar *platform* perlu adanya teknologi yang dapat memfasilitasi hal tersebut. Berdasarkan permasalahan di atas perlu adanya teknologi *web service* menjadi prosedur interaksi antar sistem yang bisa menunjang interoperabilitas untuk kepentingan proses integrasi data dan pengembangan perangkat lunak kedepannya (Komputer, 2021). *Web service* sendiri adalah sebuah sistem perangkat lunak yang dibuat untuk mendukung komunikasi serta hubungan antar mesin ke mesin melalui sebuah jaringan (Firdaus et al., 2019). Teknologi *web service* nantinya digunakan untuk melakukan transaksi data yang berkaitan dengan pendataan petani dan kelompok tani.

Penelitian sebelumnya terkait penggunaan *web service* pernah dilakukan oleh Laili Nur Asiyah dkk di mana pada penelitian tersebut diterapkan *Restfull web service* untuk mengoptimalkan kecepatan akses pada aplikasi berbasis android (Asiyah et al., 2020). Penelitian lainnya juga pernah dilakukan oleh I Gusti Ngurah Ady Kusuma penelitian yang dilakukan adalah dengan membuat model interopebalitas *web service* pada Feeder PDDIKTI dengan menggunakan *Enterprise Javabeans (EJB)* dan *REST-API* (Arifien dan Riastuti, 2019). Penelitian lainnya juga pernah dilakukan oleh Rofiq dan Ajib Susanto mereka mengimplementasikan *RESTfull web service* untuk sistem perhitungan suara cepat pada pilkada (Rofiq, 2019). Penelitian lainnya terkait metode *REST* pernah dilakukan juga oleh Diki Darmawan di mana penelitian ini terkait *web service* untuk proses transaksi data aplikasi fasilitas keuangan (Komputer, 2021).

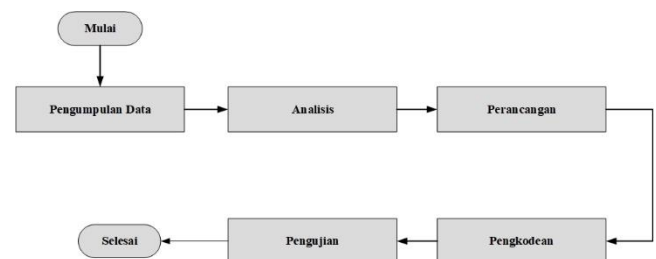
Berdasarkan latar belakang di atas, maka pada penelitian ini akan dibuat suatu "*Web Service* untuk Transaksi Data pada Aplikasi Sistem Informasi Kelembagaan Petani dengan

menggunakan Metode *REST*". *REST (Representational State Transfer)* adalah suatu arsitektur metode komunikasi yang sering diterapkan dalam pengembangan layanan berbasis *web* (Firdaus et al., 2019). Untuk dapat mengakses *resource* pada *REST*, diperlukan *URI (Uniform Resource Identifier)* sebagai pengenal ketika melakukan request ke *HTTP*. Terdapat beberapa *HTTP Method* seperti *GET, PUT, DELETE, POST* (Safitri dan Putro, 2021).

Adanya teknologi *web service* yang akan diterapkan pada aplikasi ini, diharapkan dapat mengoptimalkan interoperabilitas Aplikasi Sistem Informasi Kelembagaan Petani sehingga lebih mudah diakses meskipun berbeda *platform* dan juga dapat melakukan proses transaksi data (*input/output*) yang stabil sehingga dapat memudahkan untuk pengembangan aplikasi kedepannya.

2 Metodologi

Metode penelitian yang digunakan mengadopsi metode *Waterfall*. Meliputi pengumpulan data, analisis, perancangan, pengkodean, pengujian.



Gambar 1. Metode *Waterfall*

2.1 Pengumpulan Data

Data yang diambil dari sumber lain seperti *website* atau dokumen pemerintah. Adapun data yang digunakan pada penelitian ini adalah data yang diperoleh dari Kementerian Pertanian Tahun 2019. Data-data yang digunakan dalam penelitian ini yaitu data Poktan yang berjumlah 124 data, data Gapoktan yang berjumlah 43 data dan Petani yang berjumlah 6.000 data.

2.2 Analisis (Requirement Definition)

Pada tahap ini dilakukan proses analisis kebutuhan sistem di antaranya:

1. Analisis kebutuhan fungsional sistem,
2. Analisis kebutuhan non-fungsional sistem, dan
3. Analisis arsitektur sistem.

2.3 Perancangan (System and Software Design)

Pada tahap ini dilakukan proses perancangan desain dengan menggunakan UML, perancangan basis data yang didapatkan pada tahap analisis, merancang pemasangan *web service* dengan metode REST. Perancangan yang akan disajikan yaitu:

1. Perancangan *use case diagram*
2. Perancangan Basis Data.

2.4 Pengkodean (Implementation and Unit Testing)

Untuk tahap ini dilakukan proses implementasi ke dalam kode program menggunakan bahasa pemrograman berbasis PHP yang didukung oleh Lumen *micro-framework*. Proses ini merupakan penerjemahan dari desain sistem ke dalam bahasa yang dikenali oleh komputer. Setelah dilakukan pengkodean, maka akan dilakukan proses *testing* terhadap sistem yang sudah dibuat.

2.5 Pengujian (Integration and System Testing)

Tahap pengujian ini dilakukan proses pengujian sistem dengan menggunakan metode *black box* untuk melihat fungsi dari setiap *web service* yang telah dibuat. Pengujian dilakukan menggunakan *software* Insomnia. Insomnia adalah aplikasi desktop lintas *platform* gratis yang digunakan untuk berinteraksi dengan dan merancang API berbasis *HTTP* (Insomnia, n.d.).

3 Hasil Dan Pembahasan

3.1 Analisis

3.1.1 Analisis Kebutuhan Fungsional

Kebutuhan fungsional adalah kebutuhan pada sistem yang merupakan layanan dalam *web service* yang harus disediakan untuk proses transaksi data aplikasi Sistem Informasi Kelembagaan Petani berbasis *web*. Berdasarkan kebutuhan aplikasi Sistem Informasi Kelembagaan Petani berbasis *web* maka modul-modul yang akan disediakan oleh *web service* diantaranya adalah sebagai berikut:

- a) Modul Pendaftaran Pengguna
Modul ini digunakan oleh pengguna untuk mendaftar pada aplikasi Sistem Informasi Kelembagaan Petani.
- b) Modul Login

Modul ini digunakan oleh pengguna yang memiliki akun untuk masuk ke dalam sistem, dengan login pengguna dapat mencari kelembagaan petani melalui aplikasi.

- c) Modul Pencarian Kelompok Tani (Poktan)
Modul ini digunakan oleh pengguna untuk mencari letak sebaran kelompok tani berdasarkan kriteria tertentu.
- d) Modul Pencarian Gabungan Kelompok Tani (Gapoktan)
Modul ini digunakan oleh pengguna untuk mencari sebaran gabungan kelompok tani berdasarkan kriteria tertentu.
- e) Modul Pencarian Petani
Modul ini digunakan oleh pengguna untuk mencari data Petani yang menampilkan informasi berupa data pribadi Petani.
- f) Modul GIS
Modul ini digunakan untuk menampilkan informasi sebaran peta Poktan dan Gapoktan yang ada di wilayah Jawa Barat.
- g) Modul Manajemen Poktan
Modul ini digunakan oleh *admin* untuk melakukan proses manajemen data seperti menambah data, mengubah data, melihat data, mencari data dan menghapus data yang terkait dengan Poktan.
- h) Modul Manajemen Gapoktan
Modul ini digunakan oleh *admin* untuk melakukan proses manajemen data seperti menambah data, mengubah data, melihat data, mencari data dan menghapus data yang terkait dengan Gapoktan.

3.1.2 Analisis Kebutuhan Non-Fungsional

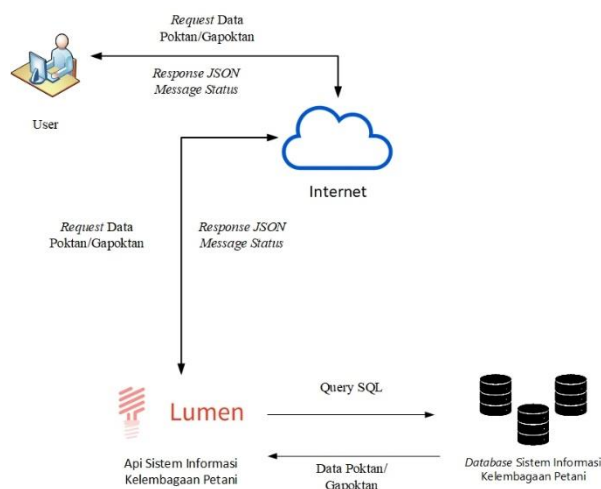
Analisis kebutuhan nonfungsional merupakan analisis yang digunakan untuk mengetahui bagian-bagian yang berinteraksi dengan sistem yang sedang berjalan. Berikut ini adalah kebutuhan nonfungsional sistem:

- a) Reliability *web service* dapat berjalan selama 24 jam/hari.
- b) Portability *web service* dapat diakses oleh pengguna melalui jaringan internet.
- c) Interoperability *web service* dapat digunakan oleh platform dan operating sistem yang berbeda-beda.

3.1.3 Analisis Arsitektur Sistem

Arsitektur sistem *web service* yang akan dibangun di aplikasi ini akan berjalan pada platform berbasis web. Akses yang ada dalam *web service* ini adalah *response* dan *request*,

komunikasi data yang ada dalam *web service* adalah komunikasi dua arah, di mana konsumen dapat melakukan *request* kepada *server* dengan parameter tertentu yang kemudian *request* akan sampai pada *server*, diolah dan disajikan dalam bentuk *response*. Dengan singkat terjadi hubungan *client - server* secara sederhana. Pada *web service* hubungan *client* dan *server* tidak secara langsung melainkan dijumpai oleh *file web service*. Dalam penelitian, format data yang digunakan adalah JSON. Sehingga aplikasi Sistem Informasi Kelembagaan Petani untuk mengakses *database* akan ditangani tidak secara langsung oleh *server* tetapi melalui perantara *web service*.

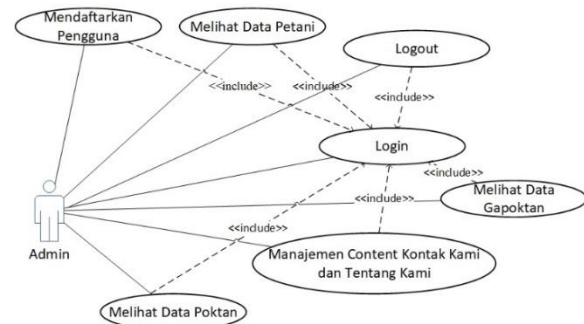


Gambar 2. Analisis Arsitektur Sistem

3.2 Perancangan

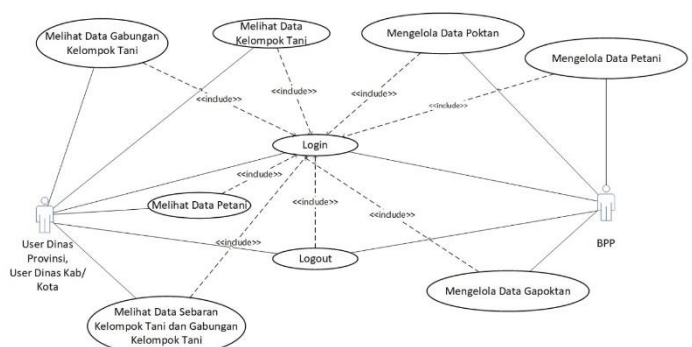
Use Case diagram dari *web service* Sistem Informasi Kelembagaan Petani ini menjelaskan hubungan antara pelaku (*actor*) dengan *use case* pada sistem. *Use case* atau diagram *use case* sendiri merupakan pemodelan untuk melakukan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat (Rosa A.S dan M.Shalahuddin, 2018). Pada perancangan *web service* dipenelitian ini terdapat 4 aktor yang akan berinteraksi dengan *service* diantaranya BPP, Dinas Provinsi, Dinas Kabupaten/Kota, dan *admin*. Untuk BPP, Dinas Provinsi, dan Dinas Kabupaten/Kota yang ingin menggunakan aplikasi harus didaftarkan terlebih dahulu oleh *admin* untuk nantinya mendapatkan *username* dan *password* agar nantinya bisa *login* ke dalam *dashboard*.

Sehingga nantinya BPP bisa melakukan manajemen data terkait dengan gabungan kelompok tani, kelompok tani, dan juga petani. Adapun untuk Dinas Provinsi dan Kabupaten/Kota hanya melihat data-data Gapoktan, Poktan dan Petani.



Gambar 3. Analisis *use case* BPP, Dinas Provinsi dan Dinas Kabupaten/Kota

Untuk aktor *admin* dapat mengelola data-data pengguna, melakukan, manajemen menu, melihat data Poktan, dan melihat data Gapoktan. Namun untuk dapat melakukan hal tersebut *admin* harus login terlebih dahulu.



Gambar 4. Analisis *use case* Diagram Admin

3.3 Implementasi

Sesuai analisis dan perancangan yang telah dilakukan, maka dibangun sebuah *web service* untuk transaksi data pada Sistem informasi Kelembagaan Petani guna untuk proses interoperabilitas antar aplikasi lainnya. *Web service* yang sudah dibuat akan diuji menggunakan perangkat lunak insomnia. terdapat 15 (lima Belas) *web service* yg dibangun dalam aplikasi ini yaitu sebagai berikut:

1. Registrasi Pengguna

Merupakan *web service* yang berfungsi untuk melakukan aksi dari pengguna yang

ingin mendaftar pada aplikasi Sistem Informasi Kelembagaan Petani. *Method* yang digunakan pada registrasi ini adalah *POST*. Berikut merupakan dokumentasi registrasi pengguna.

Tabel 1. Tabel Registrasi Pengguna

Registrasi Pengguna	
<i>URL</i>	http://192.128.0.0/register/
<i>Method</i>	<i>POST</i>
<i>URI Parameter</i>	<i>Username, password, nama, join_date, status, phone, mobile, last_login, id_dinas_prov, id_dinas_kabkot, kode_lembaga, id_bpp, forget_password, created_at, updated_at, Deleted_at</i>
<i>Succes Response</i>	{ "message": "register_success", "code": 201 }

2. Login Pengguna

Merupakan *web service* yang berfungsi untuk melakukan aksi dari pengguna yang ingin masuk kedalam aplikasi Sistem Informasi Kelembagaan Petani. *Method* yang digunakan pada login pengguna adalah *POST*. Berikut merupakan dokumentasi dari *login* pengguna.

Tabel 2. Login Pengguna

Login	
<i>URL</i>	http://192.128.0.0/auth/
<i>Method</i>	<i>POST</i>
<i>URI Parameter</i>	<i>Username, password</i>
<i>Succes Response</i>	{ "message": "login_success", "status": true, "code": 200, }

3. GET Petani

Get Petani merupakan *web service* yang berfungsi untuk mencari data-data Petani pada aplikasi Sistem Informasi Kelembagaan Petani. *Method* yang digunakan adalah *GET*. Berikut ini merupakan dokumentasi dari *GET* Petani.

Tabel 3. Dokumentasi Service GET Petani

GET Petani	
<i>URL</i>	http://192.128.0.0/poktan_anggota/Find
<i>Method</i>	<i>GET</i>
<i>URI Parameter</i>	<i>id_anggota, id_poktan, nama_anggota, jenis_kelamin, kode_komoditas, volume, subsektor, alamat, no_ktp, alamat_ktp,</i>

	<i>status_anggota, tempat_lahir, tgl_lahir, bln_lahir, thn_lahir, nama_ibu_kandung, id_prov, id_kab, id_kec, id_desa, lahan_dimiliki, nama_poktan, no_hp</i>
<i>Succes Response</i>	"status": 200, "message": "success", "data": { "current_page": 1

4. GET Poktan

GET Poktan merupakan *web service* yang berfungsi untuk mencari data-data Poktan pada aplikasi Sistem Informasi Kelembagaan Petani. *Method* yang digunakan adalah *GET*. Berikut merupakan dokumentasi dari *service GET* data Poktan.

Tabel 4. Dokumentasi Service GET Poktan

GET Poktan	
<i>URL</i>	http://192.128.0.0/api/poktan/
<i>Method</i>	<i>GET</i>
<i>URI Parameter</i>	<i>Id_poktan, id_provinsi, id_kab, id_kec, id_desa, noreg, id_gapoktan, id_bpp, nama_poktan, sk_pengukuhan, ketua_poktan, sekertaris_poktan, bedahara_poktan, alamat, simluh_tahun_bentuk, simluh_noreg, status, latitude, longitude</i>
<i>Succes Response</i>	{ "status": 200, "message": "success", "data": { "current_page": 1, }

5. FIND Poktan

FIND Poktan merupakan *web service* yang berfungsi untuk mencari data-data Poktan. Berdasarkan dengan kriteria tertentu *Method* yang digunakan adalah *GET*. Berikut merupakan dokumentasi dari *FIND* data Poktan.

Tabel 5. FIND Poktan

FIND Poktan	
<i>URL</i>	http://192.128.0.0/poktan/Find
<i>Method</i>	<i>GET</i>
<i>URI Parameter</i>	<i>Id_poktan, id_provinsi, id_kab, id_kec, id_desa, noreg, id_gapoktan, id_bpp, nama_poktan, sk_pengukuhan, ketua_poktan, sekertaris_poktan, bedahara_poktan, alamat, simluh_tahun_bentuk, simluh_noreg, status, latitude, longitude</i>

Succes Response	{ "status": 200, "message": "success", }
-----------------	---

6. GET Data Gapoktan

GET Gapoktan merupakan *web service* yang berfungsi untuk mencari data-data Gapoktan. *Method* yang digunakan adalah GET. Berikut merupakan dokumentasi GET data Gapoktan.

Tabel 6. GET Data Gapoktan

GET Gapoktan	
URL	http://192.128.0.0/gapoktan /
Method	POST
URI Parameter	id_gapoktan, noreg, id_prov, id_kec, id_bpp, id_bpp, id_desa, id_kab, nama_gapoktan, ketua_gapoktan, sk_pengukuhan, waktu_pengukuhan, tahun_pengukuhan, alamat, is_aktif, id_usaha, latitude, longitude
Succes Response	{ "status": 200, "message": "success", }

7. FIND Gapoktan

FIND Gapoktan merupakan *web service* yang berfungsi untuk mencari data-data Gapoktan berdasarkan kriteria tertentu. *Method* yang digunakan adalah GET. Berikut merupakan dokumentasi Find data Gapoktan.

Tabel 7. FIND gapoktan

FIND Gapoktan	
URL	http://192.128.0.0/gapoktan/Find
Method	GET
URI Parameter	id_gapoktan, noreg, id_prov, id_kec, id_bpp, id_bpp, id_desa, id_kab, nama_gapoktan, ketua_gapoktan, sk_pengukuhan, waktu_pengukuhan, tahun_pengukuhan, alamat, is_aktif, id_usaha, latitude, longitude
Succes Response	{ "status": 200, "message": "success", }

8. INSERT Poktan

Merupakan *web service* yang berfungsi untuk melakukan aksi yang dilakukan oleh *admin* ketika ingin melakukan manajemen data Poktan dalam hal ini menambahkan data-data Poktan. *Method* yang digunakan pada insert

poktan adalah POST. Berikut ini adalah dokumentasi dari INSERT Poktan.

Tabel 8. INSERT Poktan

INSERT Poktan	
URL	http://192.128.0.0/poktan/Insert/
Method	POST
URI Parameter	Id_poktan, id_provinsi, id_kab, id_kec, id_desa, noreg, id_gapoktan, id_bpp, nama_poktan, sk_pengukuhan, ketua_poktan, sekretaris_poktan, bedahara_poktan, alamat, simluh_tahun_bentuk, simluh_noreg, status, latitude, longitude
Succes Response	{ "status": 200, "message": "Berhasil Insert Data", }

9. INSERT Gapoktan

Merupakan *web service* yang berfungsi untuk melakukan aksi yang dilakukan oleh *admin* ketika ingin melakukan manajemen data Gapoktan dalam hal ini menambahkan data-data Gapoktan. *Method* yang digunakan dalam insert poktan adalah POST. Berikut ini adalah dokumentasi dari INSERT Gapoktan.

Tabel 9. INSERT Gapoktan

INSERT Gapoktan	
URL	http://192.128.0.0/gapoktan/Insert/
Method	POST
URI Parameter	id_gapoktan, noreg, id_prov, id_kec, id_bpp, id_bpp, id_desa, id_kab, nama_gapoktan, ketua_gapoktan, sk_pengukuhan, waktu_pengukuhan, tahun_pengukuhan, alamat, is_aktif, id_usaha, latitude, longitude
Succes Response	{ "status": 200, "message": "Berhasil Insert Data", }

10. EDIT Poktan

Merupakan *web service* yang berfungsi untuk melakukan aksi yang dilakukan oleh *admin* ketika ingin melakukan manajemen data Poktan dalam hal ini mengubah data-data Poktan. *Method* yang digunakan dalam edit poktan adalah POST. Berikut ini adalah dokumentasi dari EDIT Poktan.

Tabel 10. EDIT Poktan

EDIT Poktan	
-------------	--

URL	http://192.128.0.0/poktan/update/896529
Method	POST
URI Parameter	Id_poktan, id_provinsi, id_kab, id_kec, id_desa, noreg, id_gapoktan, id_bpp, nama_poktan, sk_pengukuhan, ketua_poktan,sekertaris_poktan, bedahara_poktan, alamat, simluh_tahun_bentuk, simluh_noreg, status, latitude, longitude
Succes Response	{ "status": 200, "message": "Berhasil Update Data", }

11. EDIT Gapoktan

Merupakan *web service* yang berfungsi untuk melakukan aksi yang dilakukan oleh *admin* ketika ingin melakukan manajemen data Gapoktan dalam hal ini mengubah data-data Gapoktan. *Method* yang digunakan dalam edit poktan adalah *POST*. Berikut ini adalah dokumentasi dari *EDIT* Gapoktan.

Tabel 11. EDIT Poktan

Edit Gapoktan	
URL	http://192.128.0.0/gapoktan/update/77049
Method	POST
URI Parameter	id_gapoktan, noreg, id_prov, id_kec, id_bpp, id_bpp, id_desa, id_kab, nama_gapoktan, ketua_gapoktan, sk_pengukuhan, waktu_pengukuhan, tahun_pengukuhan, alamat, is_aktif, id_usaha, latitude, longitude
Succes Response	{ "status": 200, "message": "Berhasil Update Data", }

12. DELETE Poktan

Merupakan *web service* yang berfungsi untuk melakukan aksi yang dilakukan oleh *admin* dalam melakukan manajemen Poktan dalam hal ini *admin* Ketika ingin melakukan

aksi menghapus Poktan. *Method* yang digunakan adalah *DELETE*. Berikut ini adalah dokumentasi dari *Service DELETE* Poktan.

Tabel 12. DELETE Poktan

DELETE Poktan	
URL	http://192.128.0.0/poktan/Delete/896527
Method	DELETE
URI Parameter	Id_poktan
Succes Response	{ "status": 200, "message": "Berhasil Delete Data", }

13. DELETE Gapoktan

Merupakan *web service* yang berfungsi untuk menangani aksi yang dilakukan oleh *admin* dalam melakukan manajemen Gapoktan dalam hal ini *admin* dalam melakukan aksi menghapus Gapoktan. *Method* yang digunakan adalah *DELETE*. Berikut ini adalah dokumentasi *DELETE* Gapoktan.

Tabel 13. DELETE Poktan

DELETE Poktan	
URL	http://192.128.0.0/gapoktan/Delete/77047
Method	DELETE
URI Parameter	Id_gapoktan
Succes Response	{ "status": 200, "message": "Berhasil Delete Data", }

14. GET Rekap Rekap Anggota Poktan

GET rekap anggota Poktan merupakan *web service* yang berfungsi untuk menampilkan rekap sebaran anggota yang dimiliki oleh masing-masing Poktan. *Method* yang digunakan adalah *GET*. Berikut ini adalah dokumentasi dari *service GET* rekap anggota Poktan.

Tabel 14. GET Rekap Anggota Poktan

GET Rekap Anggota Poktan	
URL	http://192.128.0.0/api/rekap_anggota_poktan/

<i>Method</i>	<i>GET</i>
<i>URI Parameter</i>	<i>Id_poktan, id_prov, id_kab, id_kec, id_desa, ketua_poktan, nama_poktan, alamat, nama_kabupaten, nama_kecamatan, nama_desa, nama_komoditas, luas_lahan, latitude, longitude, deleted_at, jumlah_anggota</i>
<i>Succes Response</i>	{ "status": 200, "message": "success", }

15. GET Rekap Anggota Gapoktan

GET rekap anggota Gapoktan merupakan *web service* yang berfungsi untuk menampilkan rekap sebaran anggota yang dimiliki oleh masing-masing Gapoktan. *Method* yang digunakan adalah *GET*. Berikut ini adalah *source code* dari *service Get* rekap anggota Gapoktan.

Tabel 15. GET Rekap Anggota Gapoktan

GET Rekap Anggota Gapoktan	
<i>URL</i>	http://192.128.0.0/api/rekap_anggota_gapoktan/
<i>Method</i>	<i>GET</i>
<i>URI Parameter</i>	<i>Id_gapoktan, nama_kabupaten, nama_kecamatan, nama_desa, nama_gapoktan, alamat, ketua_gapoktan, deleted_at, latitude, longitude, deleted-at, jumlah_anggota</i>
<i>Succes Response</i>	{ "status": 200, "message": "success", }

3.4 Pengujian

Blackbox testing adalah pengujian yang dilakukan untuk mengamati hasil eksekusi melalui data uji serta menyelidiki fungsional asal software. Blackbox testing pada *web service* Sistem Informasi Kelembagaan Petani ditunjukkan pada Tabel 15.

Tabel 15. Hasil Pengujian

No	URI API Pengujian	Keterangan	Status
1.	http://192.128.0.0/register/	Pendaftaran BPP, Dinas Prov, Dan Kab/Kota	Berhasil
2.	http://192.128.0.0/auth/	Login User	Berhasil

3.	http://192.128.0.0/poktan_anggota/Find	Pencarian Petani Berdasarkan Kriteria Tertentu	Berhasil
4.	http://192.128.0.0/poktan /	Pencarian Poktan	Berhasil
5.	http://192.128.0.0/poktan/Find	Pencarian Poktan Berdasarkan Kriteria Tertentu	Berhasil
6.	http://192.128.0.0/gapoktan /	Pencarian Gapoktan	Berhasil
7.	http://192.128.0.0/gapoktan/Find	Pencarian Gapoktan Berdasarkan Kriteria Tertentu	Berhasil
8.	http://192.128.0.0/poktan/Insert/	Menambahka n Poktan	Berhasil
9.	http://192.128.0.0/poktan/update/896529	Mengubah Data Poktan	Berhasil
10.	http://192.128.0.0/poktan/Delete/896527	Menghapus Data Poktan	Berhasil
11.	http://192.128.0.0/gapoktan/Insert/	Menambahka n Gapoktan	Berhasil
12.	http://192.128.0.0/gapoktan/update/77049	Mengubah Gapoktan	Berhasil
13.	http://192.128.0.0/gapoktan/Delete/77047	Menghapus Gapoktan	Berhasil
14.	http://192.128.0.0/api/rekap_anggota_poktan/	GIS Poktan	Berhasil
15.	http://192.128.0.0/api/rekap_anggota_gapoktan/	GIS Gapoktan	Berhasil

4 Kesimpulan

Kesimpulan dari hasil penelitian yang sudah dilakukan, telah dibangun *web service* untuk Aplikasi Sistem Informasi Kelembagaan Petani dengan metode REST. Didapati metode REST berhasil diterapkan pada *web service* Sistem Informasi Kelembagaan Petani dengan menggunakan Lumen *micro-framework* sebagai teknologi untuk mengembangkan *web service*. Sehingga komunikasi dan transaksi data pada aplikasi Sistem Informasi Kelembagaan Petani berbasis *web* berhasil dilakukan dan berjalan dengan baik sesuai dengan hasil pengujian yang dilakukan. Dalam hasil pengujian, kecepatan

Lumen saat melakukan *request* dari hasil pengujian pada penelitian ini tidak mutlak, kecepatan *request* tergantung dari beberapa parameter, diantaranya jumlah pengguna jaringan internet saat melakukan *request* terhadap API Sistem Informasi Kelembagaan Petani, jumlah *response* dari API Sistem Informasi Kelembagaan Petani, *hardware* dari *device* yang melakukan *request* terhadap API Sistem Informasi Kelembagaan Petani.

References

- Arifien, F., dan Riastuti, M. (2019). Model Interoperabilitas Web Service Feeder PDDIKTI Menggunakan Enterprise Javabeans (EJB) dan REST-API. 3.
- Asiyah, L. N., Sulistyanto, M. P. T., dan Aziz, A. (2020). Penerapan Restful Web Service Untuk Optimalisasi Kecepatan Akses Pada Aplikasi Berbasis Android. JOINTECS (Journal of Information Technology and Computer Science), 5(2), 129. <https://doi.org/10.31328/jointecs.v5i2.1260>
- BAPPENAS. (2015). Rencana pembangunan jangka menengah bidang pangan dan pertanian 2015-2019. 1-419.
- Firdaus, A., Widodo, S., Sutrisman, A., Nasution, S. G. F., dan Mardiana, R. (2019). Rancang Bangun Sistem Informasi Perpustakaan Menggunakan Web Service Pada Jurusan Teknik Komputer Polstri. Jurnal Informatika, 5(2407-1730), 83.
- Insomnia. (n.d.). Insomnia. Retrieved April 13, 2021, from <https://insomnia.rest/>
- Komputer, J. S. (2021). Web Service Untuk Transaksi Data Pada Aplikasi Fasilitas Keuangan Dengan Metode. 5(September), 852-865.
- Rofiq, F. N. (2019). Implementasi RESTful Web Service untuk Sistem Penghitungan Suara Secara Cepat pada Pilkada Implementation of Restful Web Servicefor Quick Count System of MayorElection. Eksplora Informatika, 6(2), 159-168.
- Rosa A.S, dan M.Shalahuddin. (2018). Rekayasa Perangkat Lunak (Terstruktur dan Berorientasi Objek) (Edisi Revi). Informatika Bandung.
- Safitri, R. K., dan Putro, H. P. (2021). Implementasi REST API untuk Komunikasi Antara ReactJS dan NodeJS (Studi Kasus : Modul Manajemen User Solusi247). Automata, 2(1), 0-4. <https://journal.uui.ac.id/AUTOMATA/article/view/17381>.