

# Repository Analysis and Design for Food Department in Salatiga

Salomo Aditya Fallo

Sistem Informasi, Fakultas Teknologi  
Informasi

Universitas Kristen Satya Wacana  
Jl. Notohamidjojo No.1 Salatiga

salomoadityafallo@gmail.com

Ariya Dwika Cahyono

Komputerisasi Akuntansi, Fakultas  
Teknologi Informasi

Universitas Kristen Satya Wacana  
Jl. Notohamidjojo No.1 Salatiga

ariyadc@uksw.edu

Melkior N. N. Sitokdana

Sistem Informasi, Fakultas Teknologi  
Informasi

Universitas Kristen Satya Wacana  
Jl. Notohamidjojo No.1 Salatiga

melkior.sitokdana@uksw.edu

## ABSTRACT

*Government is one of the fields that is always involved with data. Regarding the amount of data, a data storage is needed to collect all data which can be accessed by the government. A repository is a place where several databases or files are placed to be accessed through a network or a data repository is a location that is directly accessible to the user. Food Department in Salatiga has duties and functions in formulating and implementing the policies of the food sector, evaluating and reporting the food and implementation of the administrative services and others. Regarding its duties and functions, it needs a repository to store data from the food department in Salatiga and others department for the smooth development in various fields in Salatiga.*

*This repository involves RESTful web service which contributes in securing and accessing data from various sources, and also JSON (JavaScript Object Notation) which is a lightweight data exchange format, easy to read and write, easy to translated and generated by a computer. The repositories are created for the Food Department in Salatiga which contains a regional data, data on content type, food data, year data and account data.*

*A data warehouse is a database that stores current and past data from various operational systems and other sources (external sources) for management and analysis reporting purposes of decision making. The scope of the repository includes subject orientation, integration, nonvolatility, time variance and exploring data.*

## Keywords

*Repository, Data Warehouse, RESTful, JSON*

## 1. PENDAHULUAN

Sistem Informasi saat ini sangat dibutuhkan dalam berbagai aspek kehidupan, salah satunya organisasi Pemerintahan yang menerapkan sistem informasi untuk meningkatkan kualitas layanan yang lebih efektif dan efisien. Salah satu manfaat dari sistem informasi adalah penerapan sistem repositori dilingkungan pemerintahan. Repositori merupakan suatu tempat dimana beberapa pangkalan data atau file ditempatkan untuk diakses melalui sebuah jaringan atau sebuah repositori data merupakan suatu lokasi yang secara langsung dapat diakses oleh pengguna [1]. Data warehouse adalah suatu sistem yang dapat membantu dalam mengarsipkan, mengumpulkan, mengelola dan menganalisa data yang sudah tersimpan didalam data warehouse atau history. Data warehouse juga dapat membantu dalam pengambilan keputusan suatu organisasi.

Dinas Pangan Kota Salatiga mempunyai tugas membantu Walikota melaksanakan urusan pemerintahan yang menjadi kewenangan daerah bidang pangan serta tugas pembantuan yang diberikan kepada daerah. Adapun fungsi yang diselenggarakan yaitu perumusan kebijakan bidang pangan, pelaksanaan kebijakan bidang pangan, pelaksanaan evaluasi dan pelaporan bidang pangan, pelaksanaan administrasi Dinas, pelaksanaan fungsi lain yang diberikan oleh Walikota terkait dengan tugas dan fungsinya (Peraturan Walikota Salatiga nomor 41 tahun 2016).

Permasalahan yang dialami Dinas Pangan Kota Salatiga adalah data Dinas Pertanian dan data Dinas Pangan yang dikumpulkan secara manual dan belum adanya suatu tempat penyimpanan data tersebut. Data yang sudah dikumpulkan dan disimpan dalam database yang berbeda. Pada Dinas Pangan Kota Salatiga, repositori dimanfaatkan sebagai tempat pengumpulan data series. Sistem informasi yang akan dibangun bertujuan untuk memudahkan kinerja dari organisasi pemerintahan dan dapat menyimpan serta menyediakan data bagi pengguna repositori. Data yang sudah dikumpulkan di database operasional akan diolah secara terpusat di data warehouse sehingga data yang sudah diolah data warehouse akan diambil sebagai pengambilan keputusan. Penelitian ini diharapkan agar repositori yang dirancang dapat mencakup semua aktivitas penyimpanan dan penyediaan data untuk kepentingan masing-masing yang membutuhkan data dari Dinas Pangan Kota Salatiga. Dengan kata lain data yang sudah disimpan dapat dicari, agar dapat menyediakan data yang dibutuhkan.

## 2. TINJAUAN PUSTAKA

### 2.1 Penelitian Terdahulu

Penelitian tentang perancangan repositori bukan hal yang baru, sudah dilakukan penelitian terdahulu, sehingga beberapa penelitian dijadikan sebagai referensi awal penelitian ini.

Penelitian yang berjudul "Rancang Bangun Data Repositori Biro Penelitian Publikasi dan Pengabdian Masyarakat Universitas Kristen Satya Wacana", dengan permasalahan yang dihadapi yaitu BP3M membutuhkan media yang relevan dan mudah diakses oleh semua kalangan untuk mempublikasi hasil dari penelitian-penelitian yang telah dilakukan maupun yang sedang dilakukan. Berdasarkan kebutuhan tersebut maka dibuat aplikasi data repositori untuk Biro Penelitian, Publikasi, dan Pengabdian Masyarakat Universitas Kristen Satya Wacana berbasis web dalam mengoleksi dan mengelola dokumen-dokumen penelitian secara lebih baik dan terkomputerisasi. Penelitian ini menggunakan metode *system development life cycle (SDLC)*. Metode SDLC sangat membantu dalam membuat *software*

aplikasi yang sesuai dengan harapan konsumen, dapat berjalan efektif dan efisien dalam infrastruktur teknologi informasi sekarang dan pada masa yang akan datang [2].

Penelitian berjudul “Repositori Institusi Berbasis OMEKA (Studi kasus di *Medical Knowledge Center-FK UPNVJ*)” dengan permasalahan yang dihadapi yaitu belum menerapkan sistem informasi yang handal dalam melayani kebutuhan pemustaka akan bahan pustaka dan mengenai kendala serta tantangan dari proses pengembangan. Berdasarkan masalah tersebut maka dibutuhkan suatu *software* yang dapat mendukung terlahirnya layanan repositori institusi secara *online* agar proses deposit dan publikasi dapat berjalan dengan baik. Penelitian ini menggunakan *software* OMEKA untuk memudahkan proses operasional seperti instalasi, kustomisasi, *entry data processing*, sarana penelusuran koleksi repositori, *backup data* dan *transfer data*. OMEKA dibuat untuk memudahkan pengguna agar lebih fokus dari segi konten maupun deskripsinya sehingga tidak terlalu dipusingkan dengan bahasa pemrograman. Ini menjadi sebuah bukti bahwa OMEKA dapat diandalkan untuk diterapkan pada sistem repositori institusi [3].

Penelitian berjudul “Penerapan *WEB Semantik* Untuk Aplikasi Pencarian pada Repositori Koleksi Penelitian, Studi Kasus : Program Studi Sistem Informasi STMIK Mikroskil Medan” dengan permasalahan yang dihadapi sulitnya mencari dan melakukan perbandingan terhadap koleksi penelitian yang pernah dikerjakan sebelumnya ataupun pencarian terhadap koleksi penelitian terbaru. Dilihat dari permasalahan tersebut, maka repositori koleksi penelitian dapat dikembangkan dengan menggunakan struktur *ontology* dan jaringan *semantic* dari teknologi *web semantik* [4].

Keunggulan penelitian ini dibandingkan dengan penelitian tentang “Rancang Bangun Data Repositori Biro Penelitian Publikasi dan Pengabdian Masyarakat Universitas Kristen Satya Wacana”, “Repositori Institusi Berbasis OMEKA (Studi kasus di *Medical Knowledge Center-FK UPNVJ*)”, dan “Penerapan *WEB Semantik* Untuk Aplikasi Pencarian pada Repositori Koleksi Penelitian, Studi Kasus : Program Studi Sistem Informasi STMIK Mikroskil Medan” adalah repositori menerapkan data *warehouse* sehingga data yang didalam *database* operasional diolah secara terpusat. Dalam penerapan repositori di Dinas Pangan Kota Salatiga, penulis memanfaatkan MQTT, JSON dan *RESTful*. MQTT membantu *extract* data, transformasi data dan *loading* data dan akan masuk dalam Data *Warehouse*. Penerapan format data JSON memiliki pertukaran data mudah untuk dibaca dan ditulis, serta mudah diterjemahkan. Sedangkan penerapan *RESTful* membantu untuk mengakses data dengan berbagai format data yang diperlukan oleh Dinas Pangan Kota Salatiga sehingga dalam penelitian ini memiliki suatu keunggulan tersendiri.

## 2.2 Dasar Teori

Terdapat macam-macam pandangan para ahli mengenai pengertian sistem informasi. Stair dan Reynolds 2012 (dalam Henriyadi & Mulyati. 2014) berpendapat bahwa Sistem informasi adalah kumpulan dari beberapa elemen ataupun komponen yang saling berkaitan yang menghimpun, memanipulasi, menyimpan dan menyebarkan data atau informasi serta menyediakan mekanisme umpan balik agar sesuai dengan tujuan [1]. Sistem Informasi adalah suatu sistem dalam organisasi yang mengumpulkan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu laporan yang diperlukan [5]. Cakupan dari sistem informasi antara lain (1) *input* yaitu kegiatan untuk mengumpulkan data melalui

pengumpulan data mentah; (2) proses merupakan kegiatan manipulasi data melalui konversi, mengubah, menghitung, membandingkan data atau mengambil tindakan alternatif. Selain itu, dalam kegiatan proses, data akan disimpan untuk digunakan di masa depan. Pengolahan data menjadi informasi yang berguna adalah inti dari suatu sistem informasi; (3) *output* adalah kegiatan dimana informasi yang berguna disebarluaskan kepada para pemangku kepentingan. Informasi yang disampaikan dapat berupa beberapa format sesuai dengan kebutuhan pengguna; dan (4) umpan balik adalah aktivitas terakhir dalam suatu sistem informasi. Umpan balik adalah suatu mekanisme yang bertujuan untuk meningkatkan kualitas *output* sesuai dengan tujuan yang sudah ditetapkan. Tergantung dari hasil analisis, umpan balik yang menggunakan informasi dari *output* tersebut akan diteruskan ke bagian proses ataupun bagian *input* [1].

Repositori adalah sebuah sarana dimana dokumen, informasi atau data dikumpulkan, disimpan dan digunakan. Dimana data tersebut adalah *output* dari sebuah organisasi. Repositori juga dapat memberikan akses pada konten digital [2]. Menurut Clifford, repositori institusi adalah perubahan manajemen teknologi, dan migrasi konten digital dari satu set teknologi ke depan sebagai bagian dari komitmen organisasi untuk menyediakan layanan repositori [14]. Sedangkan menurut Pendit, istilah repositori institusi atau simpan kelembagaan merujuk ke sebuah kegiatan menghimpun dan melestarikan koleksi digital yang merupakan hasil karya intelektual dari sebuah komunitas tertentu [15].

Data *warehouse* adalah basis data yang menyimpan data sekarang dan data masa lalu yang berasal dari berbagai sistem operasional dan sumber yang lain (sumber eksternal) yang menjadi perhatian penting bagi manajemen dalam organisasi dan ditujukan untuk keperluan analisis dan pelaporan manajemen dalam rangka pengambilan keputusan [6] [7]. Data *warehouse* digunakan untuk mendukung pengambilan keputusan, bukan untuk melaksanakan pemrosesan transaksi. Filipe dkk menambahkan bahwa data *warehouse* dapat diuraikan sebagai data disimpan dalam repositori terpusat (*star, snowflake schemas*) untuk menyediakan integrasi bagi perusahaan. Sebuah repositori utama yang memiliki keuntungan untuk mengelola dan mengendalikan data [8]. Data *warehouse*, di sisi lain dapat menampung baik saat memulai transaksi sampai akhir transaksi. Hal ini memungkinkan implementasi data *warehouse* dengan kemampuan untuk mengelompokkan data yang lebih banyak terkait bagaimana data diproduksi pada tingkat yang jauh lebih rinci dan terperinci daripada biasanya yang tersedia di *database* operasional [7].

## 3. METODOLOGI PENELITIAN

Pengembangan sistem ini menggunakan metode *Prototype*. Metode ini cocok digunakan untuk mengembangkan sebuah perangkat yang akan dikembangkan atau diperbaiki kembali sesuai dengan kebutuhan pengguna. Pengembangan sistem menggunakan metode *Prototype* terdiri dari tiga tahap yaitu; *Listen to Customer*, *Build/Revise Mock-Up* dan *Customer Test-drives Mock-Up*. Tahap-tahap tersebut dalam jelaskan sebagai berikut:



**Gambar 1. Model Prototyping**

Pada tahapan *Listen to Customer* dilakukan identifikasi kebutuhan dengan metode observasi, dokumentasi dan wawancara yang bersumber dari sumber-sumber yang relevan dalam hal ini adalah Dinas Pangan Kota Salatiga, diantaranya: Melakukan wawancara dengan Kepala Dinas Pangan Kota Salatiga, Ir. Husnanani, MM. selaku yang membutuhkan aplikasi yang dapat menyimpan semua data dari Dinas Pangan Kota Salatiga; melakukan observasi di Dinas Pangan Kota Salatiga yang membutuhkan repositori.

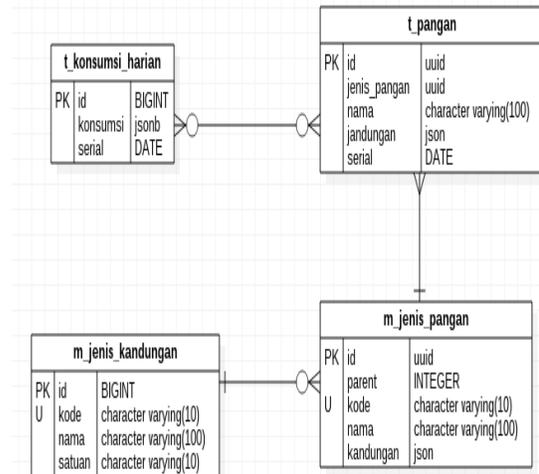
Pada bagian *Build/Revise Mock-Up* ini dilakukan dua tahap yaitu: Perancangan Sistem dan Pembangunan Sistem (*coding*) yaitu : Perancangan Sistem menggunakan *Unified Modeling Language* (UML) yang terdiri dari *Class Diagram*; Pembangunan Sistem (*Coding*) adalah pembuatan Repositori menggunakan *database Postgres*.

Pada tahapan *Customer Test-drives Mock-Up* hanya melakukan pengujian terhadap sistem menggunakan metode *Black Box*, yang berfokus pada spesifikasi fungsional dari perangkat lunak. *Black-Box Testing* lebih pada melakukan pengecekan terhadap fungsi yang tidak benar atau tidak ada, kesalahan antarmuka (*interface errors*), kesalahan pada struktur data dan akses basis data, kesalahan performansi (*performance errors*), kesalahan inisialisasi dan terminasi. Penelitian ini tidak sampai pada tahap implementasi sehingga pada pengujian sistem tidak dimasukkan dalam pembahasan ini.

### 3.1 Perancangan Sistem

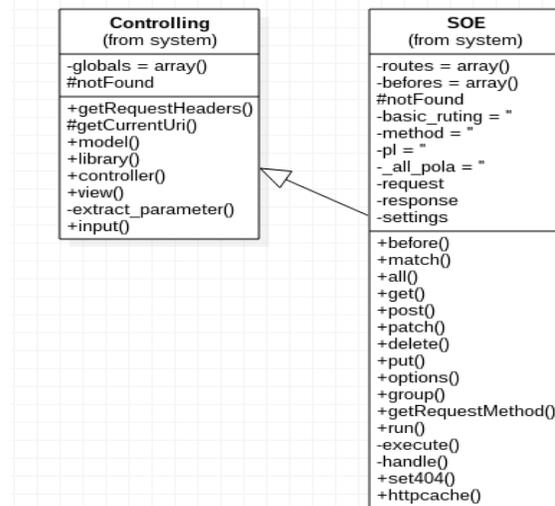
#### 3.1.1 Class Diagram

*Class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem [13]. *Class diagram* sistem ini dapat ditunjukkan pada gambar 2 dan gambar 3.



**Gambar 2. Class Diagram**

Gambar 2 menunjukkan terdapat 4 tabel yang dibuat pada *database Postgres* untuk menampung data yang akan ditambahkan dari *database* operasional Dinas Pangan dan *database* operasional Dinas Pertanian, juga relasi antar tabel-tabel pada *database* yang dibuat.



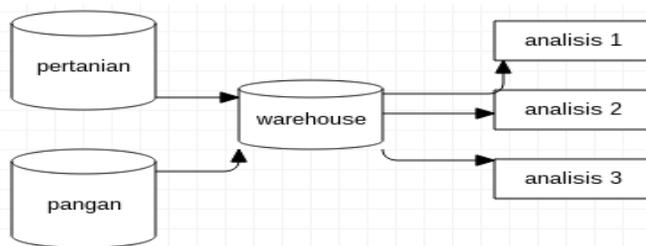
**Gambar 3. Class Diagram**

## 4. PEMBAHASAN

### 4.1 Analisis Database dan Data Warehouse Repositori

Perancangan data repositori menghasilkan konsep yang digunakan oleh Dinas Pangan Kota Salatiga. Uraian hasil yang dijelaskan sebagai berikut, konsep repositori yang dihasilkan (gambar 4), dibangun dengan menerapkan data *warehouse* untuk menampung dan mengoleksi setiap data operasional. Data operasional yang dimaksud adalah data yang dikumpulkan dari *database* operasional Dinas Pertanian dan *database* operasional Dinas Pangan. Dari setiap data yang dikoleksi dapat digunakan untuk mendukung pengambilan keputusan, bukan untuk melaksanakan pemrosesan transaksi [6]. Cakupan data yang dikoleksi yaitu data *series*. Data *series* yang dimaksudkan adalah satu set kumpulan nilai yang memiliki karakteristik yang sama, dan diberi nama tertentu, data *series* ini juga akan dihitung dalam kurun waktu satu tahun kerja. Sebagai contoh, seri data untuk konsumsi harian dan untuk jumlah konsumsi yang sudah dikonsumsi. Tiap nilai dari seri data ini biasanya memiliki persinggungan dimana kedua nilai bertemu di satu atau beberapa kategori. Sebagai contoh, baik konsumsi harian maupun jumlah konsumsi yang sudah dikonsumsi masyarakat memiliki pencatatan nilai untuk bulan Januari s/d Desember.

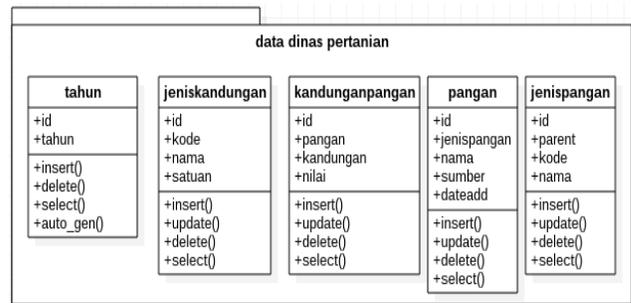
Mengacu pada gambar 4, maka cakupan dari penerapan repositori antara lain: *subject orientation*, *integration*, *nonvolatility*, *time variance* dan *data explore*.



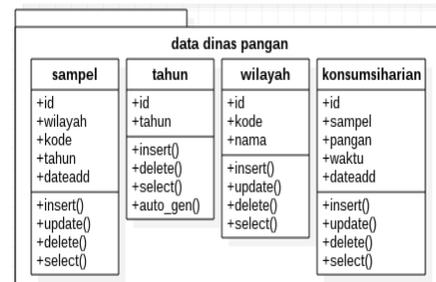
Gambar 4. Arsitektur Repositori

#### 4.1.1 Subject Orientation

*Subject orientation* adalah data yang didapatkan dari masing-masing Dinas terkait yaitu Dinas Pertanian dan Dinas Pangan Kota Salatiga (*database* operasional). Data yang diperoleh dari Dinas Pertanian antara lain (gambar 5): data tahun, jeniskandungan, kandungan pangan, pangan dan jenispangan. Sedangkan data yang diperoleh dari Dinas Pangan antara lain (gambar 6): sampel, tahun, wilayah dan konsumsi. Dari data tersebut maka ditransformasikan ke dalam *data warehouse* (gambar 7) sebagai data rangkuman dari kedua *database* operasional yang digunakan. Secara *default*, fungsi manipulasi data pada masing-masing *database* operasional antara lain *insert*, *delete*, *update* dan *select* bersifat *public* atau dapat dimanipulasi oleh pihak-pihak yang menggunakannya. Berbeda dengan data pada data *warehouse*, fungsi *insert* bersifat *private* dan *select* bersifat *public*, dengan kata lain bahwa data pada data *warehouse* tidak bisa bersifat transaksional. Hal ini dimaksudkan agar data yang ada pada data *warehouse* hanya bisa dikelola oleh sistem. Perbedaan tersebut diilustrasikan pada gambar 8.



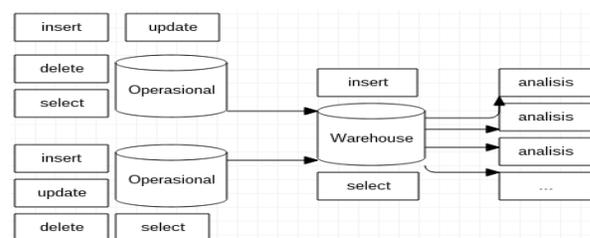
Gambar 5. Data operasional Dinas Pertanian.



Gambar 6. Data operasional Dinas Pangan.



Gambar 7. Data Warehouse.



Gambar 8. Subject Oriented.

#### 4.1.2 Integration

Untuk melakukan proses integrasi *database*, maka teknologi yang digunakan adalah MQTT. Protokol MQTT (*Message Queue Telemetry Transport*) adalah protokol pesan ringan (*lightweight*) berbasis *publish-subscribe* digunakan diatas protokol TCP/IP. Protokol ini mempunyai ukuran paket data *bandwidth* yang kecil (minimal 2 *gigabyte*) dengan konsumsi daya kecil. [9] [10]. MQTT berguna untuk memudahkan *extract* data, transformasi data dan *loading* data dan akan masuk dalam *Data Warehouse*.

```
<?php
// mendefinisikan koneksi ke alamat mqtt dari client
define('BROKER', 'localhost');
define('PORT', 1883);
define('CLIENT_ID',"pubclient_"+getmypid());
// membangun connection mqtt ke server operation
$client= new Mosquitto\Client(CLIENT_ID);
$client->onConnect('connect');
$client->onDisconnect('disconnect');
$client->onSubscribe('subscribe');
$client->onMessage('message');
$client->connect(BROKER, PORT, 60);
$client->subscribe('#', 1); // Subscribe to all messages

// looping ke mqtt
$client->loopForever();
// melakukan koneksi mqtt
function connect($r) {
echo "Received response code {$r}\n";
//berlangganan pesan dari server operational
function subscribe() {
echo "Subscribed to a topic\n";
// mengambil pesan dari koneksi mqtt
function message($message) {
printf("Got a message on topic %s with payload:\n%s\n",
$message->topic, $message->payload);
}
// memutuskan koneksi mqtt
function disconnect() {
echo "Disconnected cleanly\n"; }
```

Script 1. MQTT

Pada protokol MQTT (script 1), Data yang dikirim biasanya berupa data *binary*, *text* bahkan XML ataupun JSON dan protokol ini memakai model *publish/subscribe* daripada model *client-server*. Sistem umum MQTT membutuhkan dua komponen perangkat lunak utama yaitu: MQTT *Client* yang terletak di sisi *client* yang berfungsi untuk membangun koneksi dengan server operasional. MQTT *Broker* yang berfungsi untuk menangani *publish* dan *subscribe* data. Keuntungan dari sistem *publish/subscribe* adalah antara sumber pengirim data (*publisher* /

*server* operasional) dan penerima data (*client*) tidak saling mengetahui karena ada *broker* diantara kedua komponen tersebut atau *space decoupling* dan yang lebih penting lagi yaitu adanya *time decoupling* (waktu koneksi) dimana *publisher* dan *client* tidak perlu terkoneksi secara bersamaan, misalnya *client* bisa saja *disconnect* setelah melakukan *subscribe* ke *broker* dan beberapa saat kemudian *client connect* kembali ke *broker* dan *client* tetap akan menerima data yang terpending sebelumnya proses ini dikenal dengan *mode offline*.

Setiap data yang bersumber dari *database* operasional akan dirangkumkan ke dalam format data JSON yang kemudian akan disimpan ke dalam data *warehouse*. Pemilihan format data JSON karena format data JSON sangat sederhana dan memiliki ukuran yang lebih kecil dibandingkan dengan format data yang lain [11].

#### 4.1.3 Nonvolatility

Selama data ada di *database* operasional, data dapat diakses dan dimanipulasi secara teratur. Namun ketika data dimasukkan ke dalam *data warehouse* data tersebut dapat diakses namun tidak dapat dimanipulasi. Data yang dimasukkan akan tersimpan dalam format JSON. Ketika terjadi perubahan data yang masuk dari *database* operasional, data di *warehouse* akan *ter-update* namun catatan data yang masuk pada waktu sebelumnya akan tetap tersimpan. Dengan demikian sejarah dari data yang disimpan di *warehouse* akan dapat dengan mudah ditemukan. Untuk membedakan data yang diakses ditentukan pada *time variancy* dalam hal ini kolom *serial* pada *t\_konsumsi\_harian* pada gambar 12.

#### 4.1.4 Time Variancy

Struktur kunci data operasional dapat atau tidak mengandung beberapa elemen waktu, seperti tahun, bulan, hari, dan sebagainya. Struktur kunci dari data *warehouse* selalu mengandung beberapa elemen waktu. Penyisipan elemen waktu dapat menggunakan banyak bentuk, seperti *timestamp* pada setiap *record*, penanda waktu untuk keseluruhan *database*, dan sebagainya. Hal ini berfungsi untuk memudahkan pengelompokan data berdasarkan waktu saat data diambil dari *database* operasional. Pengambilan data dari *database* operasional biasanya terjadi secara berkala. Saat data baru diambil dari *database* operasional, dan disimpan ke *data warehouse*, maka secara otomatis data yang sudah ada sebelumnya di *data warehouse* akan dianggap usang atau dinonaktifkan. Sebagai contoh pada *t\_konsumsi\_harian* (gambar 12), terdapat salah satu kolom yang berfungsi sebagai penanda atau *time variancy* yaitu kolom *serial* dengan tipe data *date*.

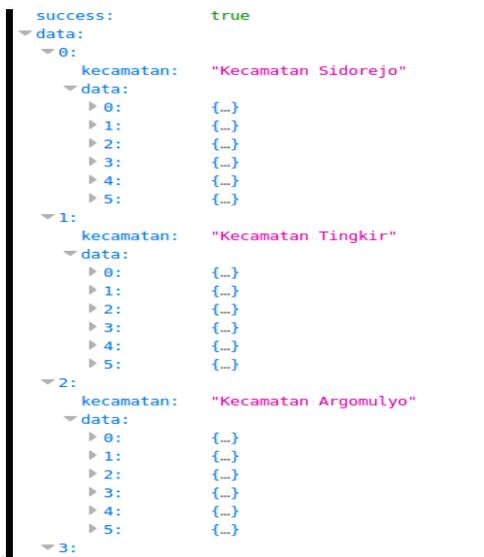
#### 4.1.5 Data Explore

Data dari *data warehouse* dapat diakses penggunaannya melalui *webservice*. Pemanfaatan *webservice* sendiri bertujuan untuk pengamanan data dan pengaksesan data dari berbagai sumber atau pengguna. Pengguna data *warehouse* tidak hanya Dinas Pangan Salatiga melainkan juga instansi-instansi lain. Karena kebutuhan akan format data dari tiap instansi berbeda, maka *Webservice RESTful* hadir untuk menjawab hal ini. *RESTful* sangat berkontribusi dalam pemecahan masalah pengaksesan data dari beberapa pengguna. Ketika suatu instansi membutuhkan data dengan format tertentu dari data *warehouse* yang diakses melalui *RESTful*, maka *RESTful* akan menyediakannya sesuai kebutuhan karena *RESTful* sudah dilengkapi dengan berbagai format data. Sebagai contoh ketika Dinas Kesehatan membutuhkan data analisis pemenuhan gizi masyarakat dalam format PDF dari data *warehouse* maka *RESTful* akan menyediakan data berbasis PDF

seperti yang dibutuhkan. Sama halnya jika Dinas Pertanian ingin mengakses data dalam format Excel, maka *RESTful* menyediakan data sesuai dengan format yang dibutuhkan.

#### 4.1.6 JSON

JSON (*JavaScript Object Notation*) adalah format pertukaran data yang ringan, mudah dibaca dan ditulis, serta mudah diterjemahkan dan dibuat (*generate*) oleh komputer. Format ini dibuat berdasarkan bagian dari Bahasa Pemrograman *JavaScript*, Standar ECMA-262 Edisi ke-3 - Desember 1999. JSON adalah format teks yang tidak bergantung pada bahasa pemrograman apapun sehingga dapat digunakan dengan berbagai bahasa pemrograman. Oleh karena sifat-sifat tersebut, menjadikan JSON sangat ideal sebagai bahasa pertukaran data [12]. Salah satu kelebihan yang menjadi tawaran bagi penggunaan *typedata* JSON adalah fleksibilitas. Hal ini dikarenakan konsep penerapan *key/value*. Dalam *typedata* JSON, sebuah data sangat mungkin untuk digabungkan atau dikombinasikan dengan data yang lain. Dalam kasus tersebut, data *warehouse* sering dikelompokkan secara periodik dimana penambahan data periodik ke dalam sebuah *typedata* JSON bisa dilakukan dengan kombinasi *key* adalah periodik dan *value* adalah satuan waktu. Tampilan dibawah ini dibangun dengan menggunakan *database postgres*.

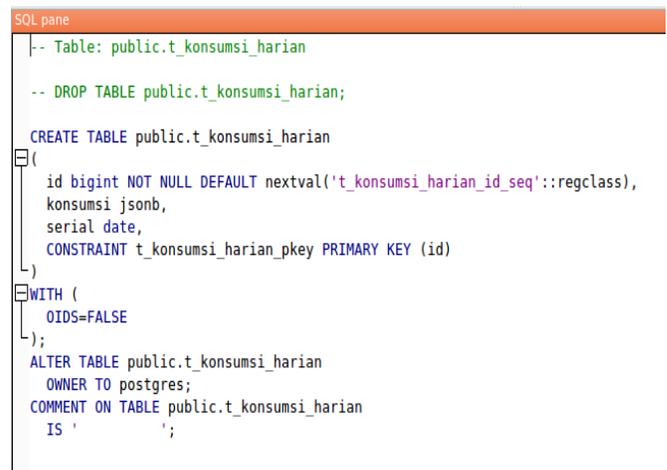


Gambar 9. Isi data kecamatan dengan tipe data JSON



Gambar 10. Isi data kelurahan dengan tipe data JSON

Pada gambar 9 menjelaskan terdapat nama kecamatan dan beberapa data kelurahan. Terdapat lebih dari satu kelurahan dalam satu kecamatan. Dari struktur data kelurahan (gambar 10), terdapat beberapa data konsumsi yang diambil per kelurahan. Data konsumsi merupakan himpunan hasil rekapan data sampel rumah tangga yang dikumpulkan pada *database* operasional. Data konsumsi mencakup konsumsi pangan, kalori, energi, protein, lemak, dll.



Gambar 11. Struktur tabel sesudah dibuat JSON.

id	konsumsi	serial
[PK] bigint	jsonb	date
1	{ "kecamatan": "Kecamatan Sidorejo", "kelurahan": { "nama": "Kelurahan Blotongan", "sampel": [ { "hari": 1, "jumlah": 5, "sampel": 1101, "konsumi": "2018-03-01" } ] } }	
2	{ "kecamatan": "Kecamatan Tingkir", "kelurahan": { "nama": "Kelurahan Sidorejo Kidul", "sampel": [ { "hari": 1, "jumlah": 5, "sampel": 2101, "konsumi": "2018-03-01" } ] } }	
3	{ "kecamatan": "Kecamatan Argomulyo", "kelurahan": { "nama": "Kelurahan Ledak", "sampel": [ { "hari": 1, "jumlah": 2, "sampel": 3101, "konsumi": "2018-03-01" } ] } }	
4	{ "kecamatan": "Kecamatan Sidomukti", "kelurahan": { "nama": "Kelurahan Mangunsari", "sampel": [ { "hari": 1, "jumlah": 4, "sampel": 4101, "konsumi": "2018-03-01" } ] } }	

Gambar 12. Contoh isi tabel konsumsi harian sesudah dibuat JSON.

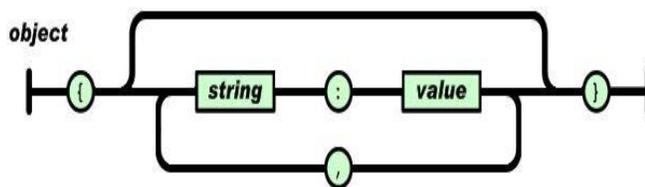
Dari hasil percobaan yang dilakukan, didapatkan pengurangan penggunaan tabel pada tabel konsumsi kecamatan, yang seharusnya memiliki beberapa tabel kelurahan dibuat menjadi satu baris data dalam satu tabel kecamatan. Sehingga format JSON membantu melakukan perintah *join* beberapa tabel sehingga proses *join* dapat dikurangi dan tidak banyak memakan memori pada *database* yang digunakan.

Pada gambar 12, sampel yang diambil dari tiap kecamatan merupakan data yang dikumpulkan dari *database* operasional Dinas Pangan dan Dinas Pertanian. Data tersebut merupakan data periodik yang diambil pertahun. Data yang diambil, kemudian dikombinasikan ke dalam *typedata* JSON. Dimana data tersebut adalah kombinasi dari data kecamatan, kelurahan dan data konsumsi harian. Dari kombinasi-kombinasi tersebut, dikombinasikan juga dengan data periodik.

```
{
  "kecamatan": "Sidorejo", "periode": "01-04-2018",
  "kelurahan": [
    {
      "nama": "kelurahan xxx",
      "sampel": []
    }, {
      "nama": "kelurahan aaaa",
      "sampel": []
    }
  ]
}
```

Script 2: Data Sample.

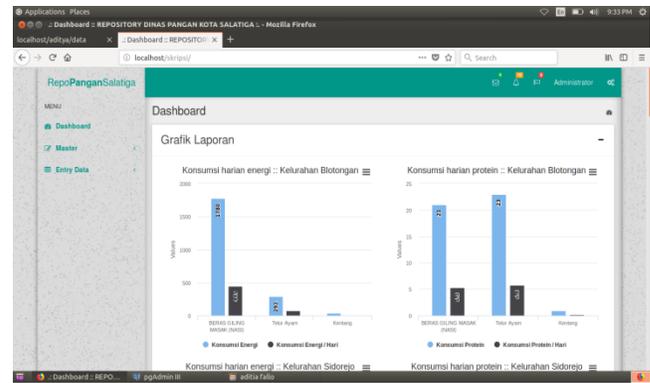
Script 2 dapat diartikan sebagai berikut, masing-masing data adalah data kecamatan, periode dan kelurahan. Data kelurahan memiliki *multiple* data di mana data kelurahan menampung lebih dari satu kelurahan. Setiap kelurahan juga memiliki data yang *multiple*. Data *multiple* tersebut akan dikelompokkan dengan menggunakan tanda "[]".



Gambar 13. Penambahan Key dan Value untuk pengambilan sampel.

#### 4.2 Perancangan

Perancangan ini akan menghasilkan *prototype* dari tiap perancangan diagram yang sudah dibangun.



Gambar 14. Tampilan dashboard Web Repositori Dinas Pangan.

Gambar 14 menunjukkan tampilan *dashboard* repositori Dinas Pangan yang berisi grafik konsumsi harian berupa konsumsi energi dan protein per hari di empat kecamatan dan beberapa kelurahan. Kemudian pada baris *menu* terdapat *menu master* yang berisi tampilan Lihat Data Wilayah, Lihat Data Jenis Kandungan, Lihat Data Jenis Pangan, Lihat Data Jenis Tahun, dan Lihat Data Akun. Sedangkan pada *menu entry* data terdapat tampilan Lihat Data Kandungan Pangan, Data Pangan, Data Sampel, dan Data Konsumsi Harian.

Standar konsumsi energi dan protein yang sudah ditentukan oleh Dinas Pangan Kota Salatiga akan dibandingkan dengan konsumsi energi dan protein dari masyarakat Kota Salatiga. Gambar 14 menunjukkan perbandingan antara standar konsumsi energi dan protein dari Dinas Pangan Kota Salatiga dengan standar konsumsi energi dan protein masyarakat Kota Salatiga untuk tiap kelurahan. Gambar tersebut menunjukkan rendahnya konsumsi energi dan protein masyarakat Kota Salatiga jika dibandingkan dengan standar yang ditetapkan oleh Dinas Pangan Kota Salatiga. Tentunya hal ini akan berubah tergantung *input* data ke dalam *database* operasional.

Dinas Pangan Kota Salatiga sebelum menggunakan aplikasi membutuhkan waktu satu bulan untuk proses *input*, olah, analisa dan *output* data, dengan sumber daya manusia minimal tiga orang dalam proses tersebut. Sedangkan dengan aplikasi hanya membutuhkan waktu satu minggu dan sumber daya manusia maksimal satu orang dalam proses *input*, olah, analisa dan *output* data. Sehingga data yang dihasilkan aplikasi *web* juga sudah berupa grafik dengan begitu proses pengambilan keputusan lebih mudah.

## 5. KESIMPULAN

Berdasarkan penelitian dan pembahasan dapat diambil kesimpulan sebagai berikut: (1) Bahwa Repositori Dinas Pangan Kota Salatiga dapat memudahkan dalam membantu penyimpanan data. (2) Tersedianya informasi dari diagram batang ini dapat membantu menganalisa atau pengambilan keputusan sesuai dengan kebutuhan dari instansi yang memerlukan sehingga permasalahan yang dialami dapat dengan mudah ditangani.

Saran pengembangan yang dapat diberikan untuk penelitian lebih lanjut adalah penelitian ini sebagai simulasi Repositori Dinas Pangan Kota Salatiga, untuk penerapan bisa dimasukkan dalam *web* Kota Salatiga agar data konsumsi harian di Kota Salatiga dapat diakses dengan mudah oleh instansi atau organisasi yang membutuhkannya.

## DAFTAR PUSTAKA

- [1] Henriyadi and R. Mulyati, 2014, 'Usability Testing Sistem Informasi: Studi Kasus Pada Aplikasi Repositori Publikasi Badan Penelitian Dan Pengembangan Pertanian Usability Testing For Information System: A Case Study Of IAARD Publication Repository Information System', *J. Perpust. Pertan.*, vol. 23, no. 2, pp. 54–63.
- [2] P. F. Tanaem, A. R. Tanaamah, and F. S. Papilaya, 2013, 'Rancang Bangun Data Repositori Biro Penelitian Publikasi Dan Pengabdian Masyarakat Universitas Kristen Satya Wacana Program Studi Sistem Informasi Fakultas Teknologi Informasi Universitas Kristen Satya Wacana Salatiga Juni', pp. 27–36.
- [3] D. F. Saputra, 2015, 'Repositori Institusi Berbasis Omeka (Studi Kasus di Medical Knowledge Center-FK UPNVI)', *Khazanah al-Hikmah J. Ilmu Perpustakaan, Informasi, dan Kearsipan*, vol. 3, no. 1, pp. 11–19.
- [4] Gunawan and F. Halim, 2014, 'Penerapan Web Semantik Untuk Aplikasi Pencarian Pada Repositori Koleksi Penelitian, Studi Kasus: Program Studi Sistem Informasi STMIK Mikroskil Medan', *JSM STMIK Mikroskil*, vol. 15 No.1, no. 1, pp. 51–60.
- [5] Jogiyanto. 2003. *Konsep Dasar Sistem Informasi*. Yogyakarta: Andi Offset.
- [6] R. A. Nugroho, J. Tambotoh, and T. J. Hoetama, 2008, 'Aplikasi Data Warehouse untuk Analisis Penjualan Mobil Berbasis Multidimensional Modeling (MDM) dan Star Schema Design (Studi Kasus PT. Asco Automotive)', *J. Teknol. Informasi-Aiti*, vol. 5, no. 2, pp. 101–200.
- [7] A. M. Langer, 2008, 'Analysis and Design of Information Systems Third Edition', New York.
- [8] A. M. Langer, 2008, 'Analysis and Design of Information Systems Third Edition', New York.
- [9] A. M. Langer, 2008, 'Analysis and Design of Information Systems Third Edition', New York.
- [10] J. B. L. Filipe, S. Hammoudi, and M. Piattini, 2005, '*Enterprise Information Systems V*', Kluwer Academic Publishers.
- [11] G. Y. Saputra, A. D. Afrizal, F. K. R. Mahfud, F. A. Pribadi, and F. J. Pamungkas, 2017, 'Penerapan Protokol Mqtt Pada Teknologi Wan ( Studi Kasus Sistem Parkir Univeristas Brawijaya )', *J. Inform. Mulawarman*, vol. 12, no. 2, pp. 69–75.
- [12] E. G. Davis, A. Calveras, and I. Demirkol, 2013, 'Improving packet delivery performance of publish/subscribe protocols in wireless sensor networks', *Sensors (Switzerland)*, vol. 13, no. 1, pp. 648–680.
- [13] P. F. Tanaem, D. Manongga, and A. Iriani, 2016, 'RESTFul Web Service Untuk Sistem Pencatatan Transaksi Studi Kasus PT. XYZ', *J. Tek. Inform. dan Sist. Inf.*, vol. 2.
- [14] Jackson W, JSON Quick Syntax Reference, 2016, 'The Expert's Voice in Web Development', Lompoc, California, USA:Apress.
- [15] A. S., Rosa dan Shalahuddin, M. 2013. *Rekayasa Perangkat Lunak Terstruktur Dan Berorientasi Objek*. Informatika. Bandung.
- [16] Lynch, C, 2003, 'Institutional Repositories: Essential infrastructure for scholarship in the digital age', ARL Bimonthly Report, No. 226.
- [17] Pendit, Putu Laxman, 2008, '*Perpustakaan Digital A sampai Z*', Jakarta: Karya Ciptaaksa.