

# Sistem Informasi Pemasaran Hasil Pertanian Polinela Berbasis *JSON*

Dwirgo Sahlinal<sup>1</sup>, Rima Maulini<sup>2</sup>, Dewi Kania Widyawati<sup>3</sup>

<sup>1</sup> Program Studi Management Informatika Politeknik Negeri Lampung

<sup>2</sup> Program Studi Management Informatika Politeknik Negeri Lampung

<sup>3</sup> Program Studi Management Informatika Politeknik Negeri Lampung

<sup>1</sup> *dwirgo\_sahlinal@polinela.ac.id*

<sup>2</sup> *rima\_maulini@polinela.ac.id*

<sup>3</sup> *dewi\_mi@polinela.ac.id*

## Abstract

*Agricultural information system aims to build a marketing system based on JSON information system technology (Java Script Object Notation). Specifically the research aims: (1) build a database using JSON; (2) establish marketing information system of agricultural products; (3) provide information on the implementation of identification, inventory monitoring of agricultural product marketing issues. The research begins with literature review, planning and preparation of system model, model verification, field trial, evaluation and development and implementation result. Output targets to be generated: (1) basic data of agricultural product marketing processing; (2) built a marketing information analysis system with JSON-based information technology.*

**Keywords:** *Information System; Marketing; Agricultural Product; JSON.*

## Abstrak

Sistem informasi hasil pertanian bertujuan membangun sistem pemasaran berbasis teknologi sistem informasi *JSON* (Java Script Object Notation). Secara khusus penelitian bertujuan: (1) membangun database dengan menggunakan *JSON*; (2) membangun sistem informasi pemasaran hasil pertanian; (3) memberikan informasi pelaksanaan identifikasi, inventarisasi pemantauan permasalahan pemasaran hasil pertanian. Penelitian dimulai dengan kajian literatur, perencanaan dan penyusunan model sistem, verifikasi model, ujicoba lapangan, evaluasi dan pengembangan serta hasil implementasi. Target luaran yang akan dihasilkan: (1) diperoleh data dasar pengolahan pemasaran hasil pertanian; (2) dibangun sistem informasi analisa pemasaran dengan teknologi informasi berbasis *JSON*.

**Kata Kunci :** Sistem Informasi; Pemasaran; Hasil Pertanian; *JSON*.

## 1. PENDAHULUAN

Pentingnya informasi pasar yang akurat dan up to date bagi para pelaku pasar itulah, penelitian yang berjudul Sistem Informasi Pemasaran Hasil Pertanian POLINELA Berbasis *JSON* (Java Script Object Notation). Sistem ini akan dikembangkan sebagai sebuah sistem yang terpadu, yang mengintegrasikan informasi pasar komoditas hasil pertanian di Polinela. Penelitian ini juga diharapkan akan menghasilkan sebuah sistem yang mampu memperkuat jaringan informasi pasar komoditas hasil pertanian, menyediakan informasi secara cepat, tepat sasaran, akurat dan dapat dipertanggungjawabkan, baik untuk petani, pedagang, pembeli maupun pemerintah atau pihak-pihak lain yang terkait dengan kebijaksanaan pasar komoditas hasil-hasil pertanian.

Sistem informasi pemasaran jika didefinisikan dalam arti luas adalah kegiatan perseorangan dan organisasi yang memudahkan dan mempercepat bubungan pertukaran yang memuaskan dalam lingkungan yang dinamis melalui penciptaan pendistribusian promosi dan penentuan harga barang, jasa dan gagasan. Sistem informasi pemasaran

selalu digunakan oleh bagian pemasaran dalam sebuah perusahaan untuk memasarkan produk-produk perusahaan tersebut. Sistem informasi ini merupakan gabungan dari keputusan yang berkaitan dengan 4P yaitu:

- Produk (*product*): produk apa yang dibeli pelanggan untuk memuaskan kebutuhannya.
- Promosi (*Promotion*): Meningkatkan atau mendorong penjualan.
- Tempat (*Place*): Cara mendistribusikan produk secara fisik kepada pelanggan melalui saluran distribusi.
- Harga (*Price*): Terdiri dari semua element yang berhubungan dengan apa yang dibayar oleh pelanggan.

Java Script Object Notation (*JSON*) merupakan salah satu format pertukaran data yang memiliki format penulisan yang sederhana dan berkonsekuensi pada ukuran *file* yang dihasilkan. Ukuran *file* yang dihasilkan oleh *JSON* sangat kecil, sehingga dapat diakses dengan waktu yang relatif cepat. Dalam kaitannya sebagai penghubung komunikasi aplikasi client dengan aplikasi *server*, *web service* menggunakan suatu format serialisasi data untuk mengirimkan data. Sebelum data dikirimkan, baik dari *client* menuju *server* atau sebaliknya, harus diubah dalam format data tertentu dahulu sesuai dengan *web service* yang digunakan. Jenis format serialisasi data yang digunakan dalam *web service* diantaranya *XML* dan *JSON*.

Salah satu faktor yang mempengaruhi kemampuan format data adalah seberapa cepat *request* itu diproses, sehingga dibutuhkan *request* dengan format pertukaran data yang efisien dan cepat dalam proses parsing tersebut. *XML* merupakan format pertukaran data yang umum digunakan dalam aplikasi dengan berbasis *web*. Namun hal ini bukanlah cara yang terbaik. *XML* memiliki sintaks yang kompleks dan harus diproses menjadi Document Object Model. Format pertukaran data selain *XML* yaitu *JSON* yang merupakan bagian dari *Java Script* sehingga parsing dilakukan oleh *Java Script* tersebut dan memiliki sintaks yang lebih sederhana dari pada *XML*. Secara spesifikasi *JSON* lebih baik dari pada *XML*.

## 2. KERANGKA TEORI

Menurut Ahmad, *JSON (Java Script Object Notation)* merupakan format pertukaran data berdasarkan notasi *Java Script*. Dari beberapa sumber mengatakan bahwa *JSON* lebih baik dari *XML* sebagai format pertukaran data. Ahmad mencoba untuk membuat uji untuk membandingkan kedua format data tersebut pada *ASP.NET Web Service*. Parameter pembandingan yang digunakan yaitu besar *stream* yang ditransmisikan dan *response time*. Untuk menganalisis kinerja keduanya digunakan *tool* buatan Nikhil Kotari yaitu *Web Development Helper*. Pertama dibuat sebuah *object Employee* dengan data *EmpId*, *Name*, *Sex* dan *Title*. *Object* ini yang akan diserialisasi ke *JSON* dan *XML* sebagai *return value Web Service*. Dari hasil uji tersebut, format *XML* memerlukan 226 byte sedangkan *JSON* hanya 132 byte. Ini jauh lebih kecil (kurang dari setengah) dibandingkan *XML*. Dari sisi *response time*, pada *invoke* yang pertama *XML* membutuhkan waktu 0:0:3430 sedangkan *JSON* 0:0:3280 dan *invoke* berikutnya untuk *XML* 0:0:0150 sedangkan *JSON* 0:0:0000. Dari data ini membuktikan bahwa *JSON* memang lebih ringan dibandingkan *XML* baik dari sisi ukuran *stream* yang ditransmisikan lebih kecil maupun waktu proses di server juga lebih singkat.

Menurut Xiaopeng, Objek data berbasis *XML* pada transmisi jaringan memiliki pembacaan yang baik dan *scalability*. Tetapi, *browser* dan pendekatan *codec* membatasi penerapan *XML*. *JSON* adalah objek yang memiliki format yang kompak. Programmer dapat menggunakan sintaks berorientasi obyek untuk mem-parsing objek data. Pada aplikasi dimasa depan, *XML* akan sepenuhnya diganti dengan *JSON* pada kriteria tertentu.

Majid Khosravi, melakukan penelitian dengan mengkoneksikan *device mobile android* ke *web service REST* untuk mengukur *delay time* yang terjadi pada *XML* dan *JSON* dari segi jumlah objek data, ukuran data, *loading time*, *parsing time* dengan menyediakan operasi *select*, *update*, *insert* dan *delete*. Dalam penelitian tersebut, *JSON*

mempunyai *respon time* yang lebih cepat jika dibandingkan dengan *XML* ketika jumlah objek data kecil (dibawah 1300).

Nurzhan Nurseitov, melakukan studi kasus membandingkan perbedaan antara dua format pertukaran data, yaitu *JSON* dan *XML*. Dalam menganalisa hasil digunakan pengamatan kualitatif dan menganalisa lebih lanjut tiap-tiap hasil pengukuran yang signifikan diuji dengan menggunakan uji statistik, yaitu t-test. Hasilnya menunjukkan bahwa *JSON* lebih cepat dan menggunakan sumber daya yang lebih sedikit daripada *XML*. *JSON* dan *XML* mempunyai keunikan tersendiri, tetapi pentingnya kinerja dan pemanfaatan sumber daya harus dipahami ketika membuat keputusan pemilihan antara format pertukaran data.

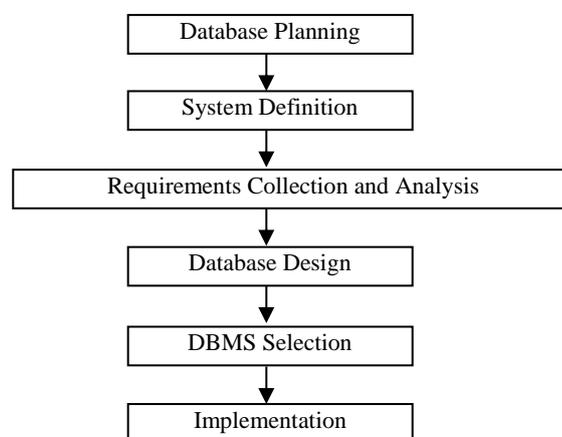
### 3. METODOLOGI

#### 3.1 Rancangan Penelitian

Rancangan penelitian yang digunakan dalam penelitian ini adalah rancangan penelitian kualitatif. Metode penelitian ini terdiri dari beberapa tahapan, yaitu tahapan pengembangan sistem dan pengumpulan data atau informasi. Sedangkan dalam pengumpulan data adalah dengan metode wawancara dan observasi langsung. Bahan atau data yang digunakan dalam penelitian ini adalah sebagai berikut : Data spasial berupa data hasil pertanian di Polinela, data non spasial yang digunakan adalah data primer (survei lapangan) dan data sekunder dari literatur mengenai data hasil pertanian, foto objek sebagai dokumentasi.

#### 3.2 Tahapan Penelitian

Penelitian ini dikhususkan untuk menyimpan data pada *database*. Berdasarkan Conolly dan Begg (2015) tahapan *database system development lifecycle* memiliki 13 tahapan yaitu *database planning, system definition, requirements collection and analysis, database design, DBMS selection, application design, prototyping (optional), implementation, data conversion and loading, testing, dan operational maintenance*. Pada penelitian ini akan mengadopsi enam tahapan awal yaitu *database planning, system definition, requirement collection and analysis, database design, DBMS selection, dan implementation*. Tahapan penelitian dapat dilihat di Gambar 1.



Gambar 1. *Database System Development Lifecycle* (mengadopsi dari Connolly dan Begg 2015)

### 1. Database planning

Pengumpulan tujuan umum dari pengembangan *database* dan fungsi-fungsi apa saja yang perlu dipenuhi *database* untuk memenuhi kebutuhan.

### 2. System definition

Identifikasi cakupan dan batasan sistem *database*, serta mendefinisikan kebutuhan *database* dari berbagai sudut pandang pengguna.

### 3. Requirement collection and analysis

Proses pengumpulan dan analisis informasi mengenai organisasi untuk mendukung sistem *database*, kemudian menggunakan informasi tersebut untuk mengidentifikasi kebutuhan *database* baru. Kebutuhan untuk tiap pengguna terdiri dari deskripsi data yang digunakan/dihasilkan serta detail bagaimana suatu data digunakan/dihasilkan

### 4. Database design

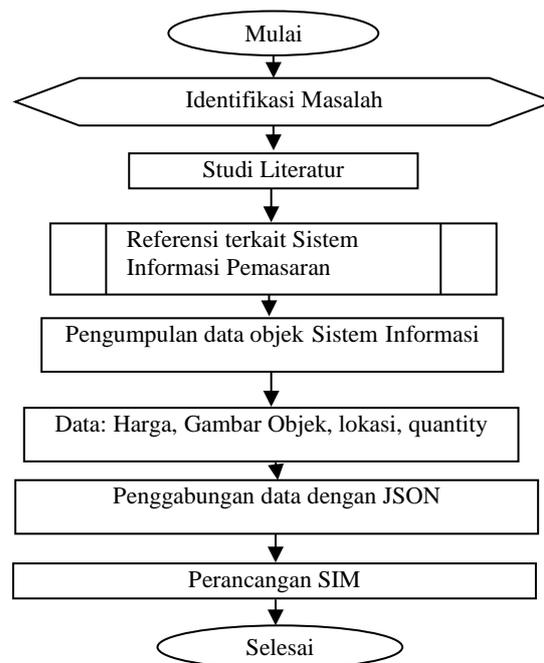
Tidak seperti SQL dimana skema *table* harus ditentukan sebelum insert data, pada *JSON* tidak memiliki struktur dokumen, sehingga fleksibel dalam memetakan dokumen ke dalam entitas maupun objek. *Design* dari model data mengikuti penggunaan data pada sistem.

### 5. DBMS selection

Memilih *database* yang sesuai dengan kebutuhan dari sistem

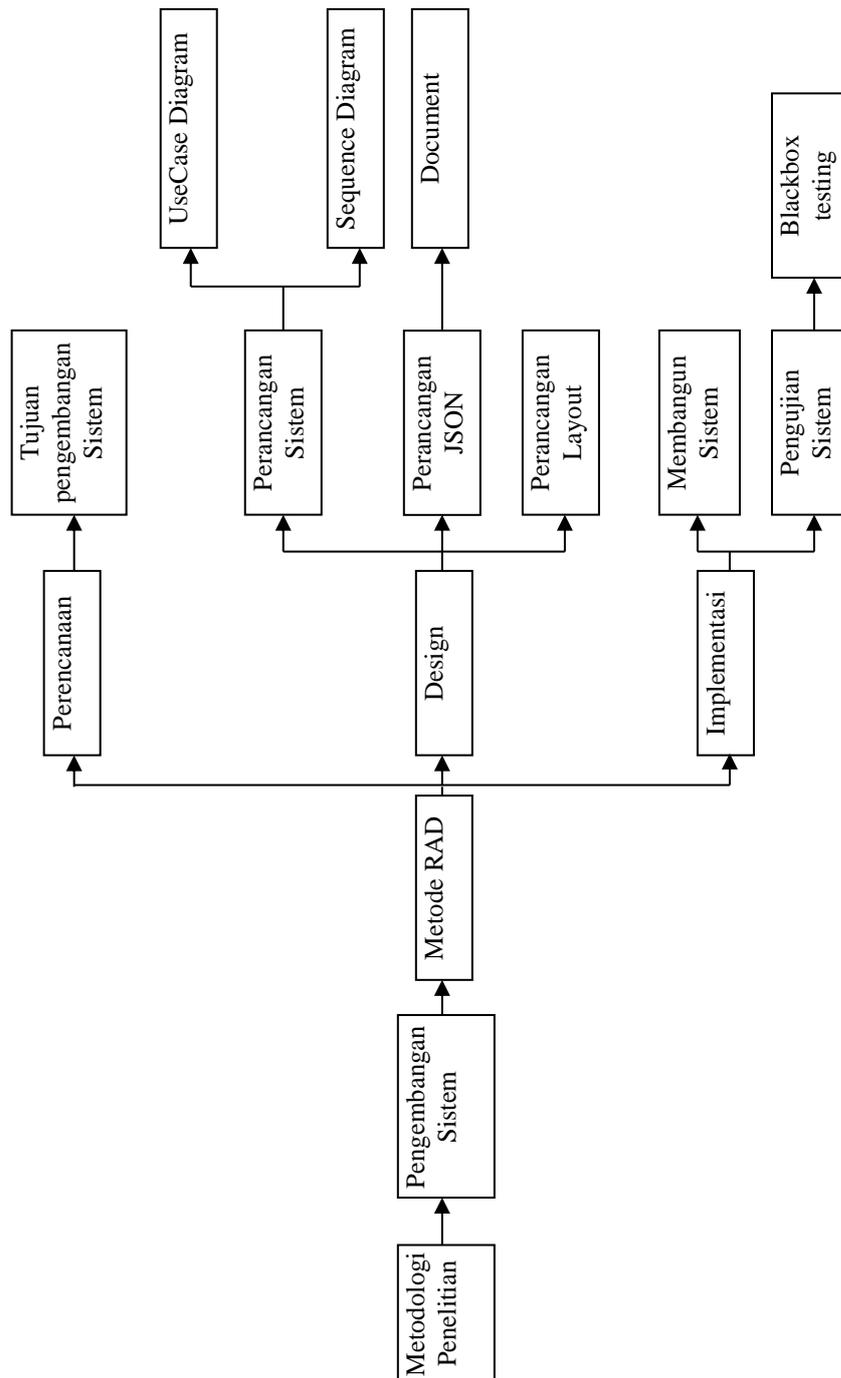
### 6. Implementation

Proses implementasi *database* ke organisasi yang membutuhkan *database*.



**Gambar 2.** Diagram Alir Penelitian

### 3.3 Metode Penelitian



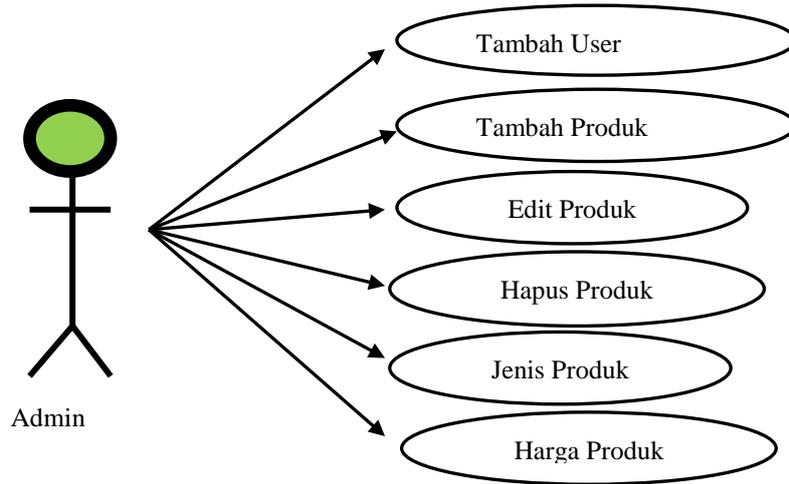
Gambar 3. Diagram Alir Pengolahan Data

#### 3.3.1 Perancangan Sistem

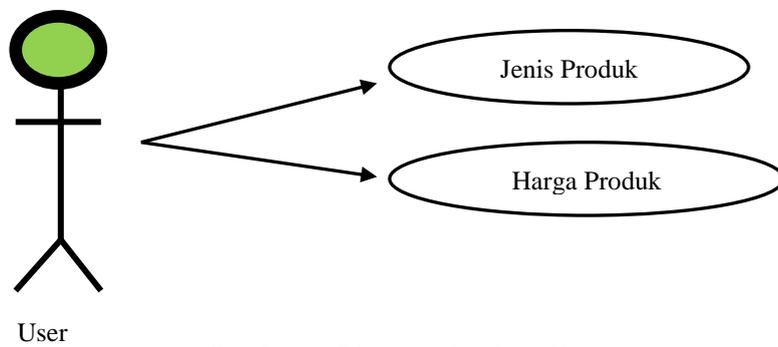
##### a. Use Case Diagram

Sistem yang akan dibangun memiliki dua aktor yang akan terinteraksi langsung dengan sistem. Kedua aktor tersebut adalah *Admin* dan *User*. Aktor *Admin* bertugas untuk menambah *user*, menambah produk, mengedit

produk, menghapus produk, melihat jenis produk dan harga produk pada sistem. Sedangkan *User* dapat melihat jenis produk dan harga produk. Gambar 4 dan 5 menunjukkan diagram *use case* untuk *admin* dan *user*.

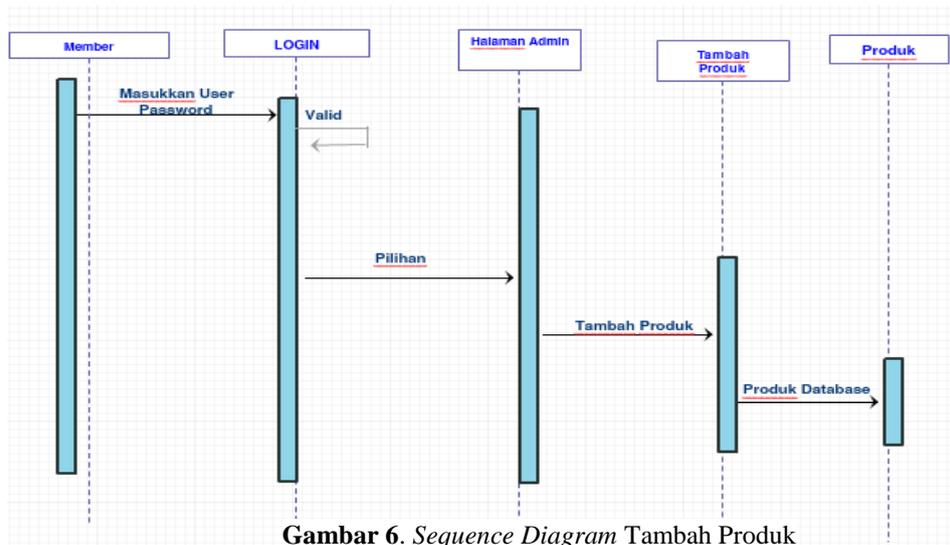


Gambar 4. Diagram *Use Case Admin*

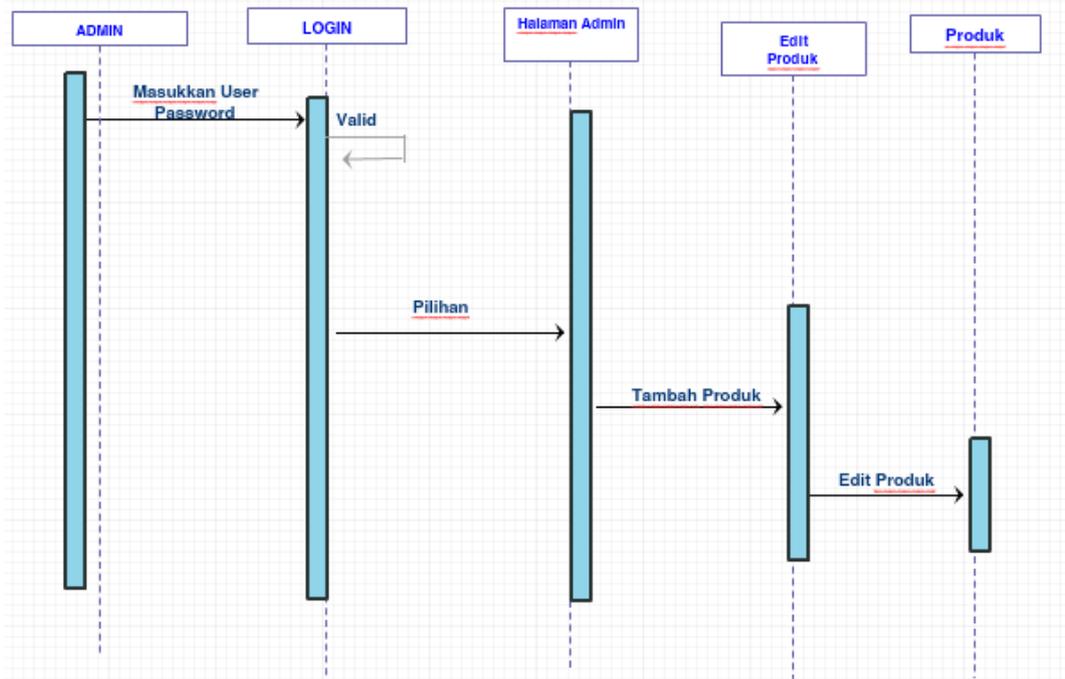


Gambar 5. Diagram *Use Case User*

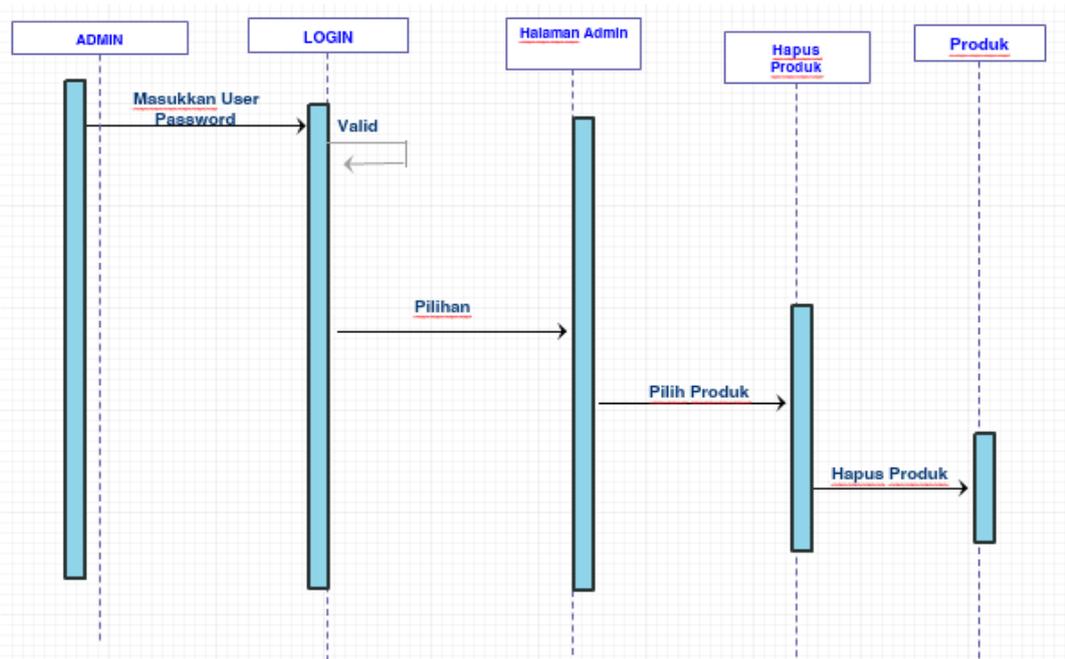
*b. Sequence Diagram*



Gambar 6. *Sequence Diagram Tambah Produk*



Gambar 7. Sequence Diagram Edit Produk



Gambar 8. Sequence Diagram Hapus Produk

### 3.3.2 Perancangan JSON

Data JSON contoh tentang hasil pertanian *source code* adalah:

```

{"records":
[
  {"kode_produk":"M001","nama_produk":"Wortel","harga":"12000",
  "ketersediaan":"Terjual","gambar_produk":"wortel2.jpg","id":0
  },
  {"kode_produk":"M002","nama_produk":"Selada","harga":"7000",
  "ketersediaan":"Tersedia","gambar_produk":"selada2.jpg","id":1
  },
  {"kode_produk":"M003","nama_produk":"Bayam","harga":"9000",
  
```

```

    "ketersediaan":"Tersedia","gambar_produk":"bayam1.jpg","id":2
  },
  {"kode_produk":"M004","nama_produk":"Alpukat","harga":"8000",
  "ketersediaan":"tersedia","gambar_produk":"alpukat1.jpg",
  "tambah":"Submit Query"
}
]
}

```

*Records* adalah adalah *array JSON*. *Array* terdiri dari beberapa *object*. Dalam penelitian ini, menggunakan 4 *object records*. Masing masing *object record* ini mempunyai 6 *field*.

Data *JSON* contoh tentang *username* dan *password source code* adalah:

```

[
  {"id":0,"user":"admin","password":"admin","nama lengkap":"dwirgo",
  "alamat":"kaliawi","jenis kelamin":"laki-laki"
},
  {"id":1,"user":"","password":""},{id":2,"user":"inal","password":"inal"
},
  {"user":"malani","password":"malani","nama lengkap":null,
  "alamat":"a","jenis kelamin":"perempuan","level":"konsumen"
}
]

```

*Array JSON* terdiri dari beberapa *object*. Dalam penelitian ini, menggunakan 3 *object*. Masing masing *object* ini mempunyai 7 *field*.

### 3.3.3 Pengujian Sistem

Pengujian sistem dilakukan dengan dua metode yaitu metode *black box*. Metode *black box* bertujuan menguji kesesuaian hasil pembuatan sistem terhadap analisis kebutuhan yang telah dibuat sebelumnya. Hasil pengujian dengan metode *BlackBox* pada *Admin* ditunjukkan pada Tabel 1. Tabel tersebut menunjukkan kesesuaian antara hasil pembuatan sistem dan analisis kebutuhan sistem. Sedangkan pada Tabel 2 menunjukkan hasil pengujian terhadap *User*.

**Tabel 1.** Hasil Pengujian dengan metode *Black Box* pada *Admin*

Kelas Uji	Skenario Uji	Hasil yang diharapkan	Kesimpulan
<b>Tampilan Login pada admin</b>	Tekan tombol login setelah memasukan <i>username</i> dan <i>password</i>	Tampil Menu Utama Yaitu Pertanian dan Daftar <i>User</i>	Berhasil
	Tekan tombol registrasi	Tampil isian registrasi	Berhasil
<b>Menu Utama</b>	Tekan tombol Pertanian	Tampil Jenis Tanaman Pertanian	Berhasil
	Tekan tombol produk   lihat produk	Tampil lihat produk	Berhasil
	Tekan tombol produk   tambah produk	Tampil tambah produk	Berhasil
	Tekan tombol hapus produk	Tampil hapus produk	Berhasil
<b>Logout</b>	Takan tombol <i>logout</i>	Tampil ke <i>username</i> dan <i>password</i>	Berhasil

Tabel 2. Hasil Pengujian dengan metode *Black Box* pada *User*

Kelas Uji	Skenario Uji	Hasil yang diharapkan	Kesimpulan
Tampilan pada user	<i>Login</i> Tekan tombol login setelah memasukkan <i>username</i> dan <i>password</i>	Tampil Menu Utama Yaitu Pertanian dan Daftar <i>User</i>	Berhasil
	Menu Utama	Tekan tombol registrasi Tekan tombol Pertanian Tekan tombol produk   lihat produk	Tampil isian registrasi Tampil Jenis Tanaman Pertanian Tampil lihat produk
<i>Logout</i>	Takan tombol <i>logout</i>	Tampil ke <i>username</i> dan <i>password</i>	Berhasil

**4. HASIL DAN PEMBAHASAN**

Hasil penelitian yang dilakukan dijabarkan berdasarkan tahapan penelitian yang dikemukakan pada metodologi. Berikut merupakan beberapa tampilan antarmuka aplikasi *web*.

Pada sub menu lihat produk dan tambah produk akan ditampilkan contoh produk pertanian seperti gambar berikut:



Gambar 9. Tampilan Lihat Produk Admin

Hasil penambahan produk disimpan dengan menggunakan *JSON*.

**a. Tampilan menu produk dan sub menu lihat produk user**



Gambar 10. Tampilan menu user

Pada menu produk | lihat produk hanya menampilkan lihat produk pada *user* dengan *field* tabel kode produk, nama produk, harga, ketersediaan, dan gambar produk sebagai berikut

kode produk	nama produk	harga	ketersediaan	gambar
M001	Wortel	12000	Terjual	
M002	Selada	7000	Tersedia	
M003	Bayam	9000	Tersedia	
M004	Alpukat	8000	tersedia	

**Gambar 11.** Tampilan menu lihat produk pada user

Pada penelitian ini untuk melakukan proses transaksi penjualan hasil pertanian belum dilaksanakan dengan menggunakan *JSON*.

## 5. KESIMPULAN

Dari pembahasan yang diuraikan maka penyusun menarik kesimpulan sebagai berikut. Sistem Informasi Pemasaran Hasil Pertanian dengan *JSON* ini berhasil dibuat dan dapat dijadikan acuan mengenai harga tanaman pertanian dan perkebunan dengan mudah dan tepat dikarenakan proses aplikasi ini berjalan secara real time tanpa menggunakan database. Akan tetapi, masih akan dilakukan penelitian lanjut untuk melakukan proses transaksi penjualan hasil pertanian dengan menggunakan *JSON*.

## DAFTAR PUSTAKA

- Ahmad Masykur. 2008. *JSON vs XML*. <http://www.masykur.web.id/post/JSON-vs-XML.aspx>. diakses tanggal April 2015
- Hunlock, Patrick. 2007. *Mastering JSON (JavaScript Object Notation)*.  
<http://www.scribd.com/doc/15009816/Mastering-in-JSON-JavaScript-Object-Notation->.
- Hasibuan, Z.A., 2002, *“Electronic Government for Good Governance”*, Jurnal Sistem Informasi dan Manajemen Teknologi Informasi, Vol. 1, Nomor 1
- Khosravi Majid.2012. *“XML vs JSON parsing in Android”*. <http://www.majidkhosravi.com/xml-vs-json-android/>.
- Topcu *et al.* *“Web 2.0 for E-Science Environments”*. Community Grids Laboratory, Indiana University, 501 North Morton Street, Suite 224 Bloomington, IN 47404