

**PENGEMBANGAN MODUL PELAPORAN HARGA KOMODITAS  
PERTANIAN PADA SISI PENGGUNA ADMIN DAN  
PEMERINTAH MENGGUNAKAN REST API**

**DIAN SETIANA**



**DEPARTEMEN ILMU KOMPUTER  
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
INSTITUT PERTANIAN BOGOR  
BOGOR  
2016**



## **PERNYATAAN MENGENAI SKRIPSI DAN SUMBER INFORMASI SERTA PELIMPAHAN HAK CIPTA \***

Dengan ini saya menyatakan bahwa skripsi berjudul Pengembangan Modul Pelaporan Harga Komoditas Pertanian pada Sisi Pengguna Admin dan Pemerintah Menggunakan REST API adalah benar karya saya dengan arahan dari komisi pembimbing dan belum diajukan dalam bentuk apa pun kepada perguruan tinggi mana pun. Sumber informasi yang berasal atau dikutip dari karya yang diterbitkan maupun tidak diterbitkan dari penulis lain telah disebutkan dalam teks dan dicantumkan dalam Daftar Pustaka di bagian akhir skripsi ini.

Dengan ini saya melimpahkan hak cipta dari karya tulis saya kepada Institut Pertanian Bogor.

Bogor, Agustus 2016

*Dian Setiana*  
NIM G64120010

## **ABSTRAK**

DIAN SETIANA. Pengembangan Modul Pelaporan Harga Komoditas Pertanian pada Sisi Pengguna Admin dan Pemerintah Menggunakan REST API. Dibimbing oleh DEAN APRIANA RAMADHAN.

Keterlambatan penyampaian informasi terkait hasil pertanian di suatu wilayah disebabkan oleh kurangnya komunikasi dua arah antara petani dengan pemerintah. Keterlambatan ini juga berimbas pada masyarakat yang kurang mendapatkan informasi terkait harga komoditas pertanian. Kehadiran tengkulak dalam distribusi komoditas pertanian mengakibatkan petani mengalami kerugian karena hasil panen dibayar dengan harga yang sangat rendah. Masyarakat pun sulit mencari harga murah karena harga yang ada di pasaran menjadi sangat mahal. Penelitian ini berupa pengembangan aplikasi pelaporan harga komoditas pertanian pada sisi pengguna admin dan pemerintah berbasis web menggunakan *representational state transfer* (REST) *application programming interface* (API) dengan metode *extreme programming*. Penelitian ini berhasil mengembangkan REST API dan aplikasi web yang memudahkan pemerintah dalam mengawasi dan mengontrol distribusi produk-produk hasil pertanian dari petani ke pasar, serta dapat mempercepat alur informasi harga komoditas pertanian dari pasar ke pemerintah.

Kata kunci: harga komoditas, pertanian, REST API

## **ABSTRACT**

DIAN SETIANA. Development of Reporting Module of Agricultural Commodity Price for Admin and Government User Sides Using REST API. Supervised by DEAN APRIANA RAMADHAN.

The delays in delivering agricultural information in a certain region are often caused by the lack of two-way communication between farmers and the government. These delays also affected other residents who are less informed about the price of agricultural commodities. The presence of middleman in the distribution of such commodities tend to cause losses for the farmers, as the yields were bought in a very low price. Consumers were difficult to find cheap commodities because the price on the market was always high. This research developed a web-based application for reporting agricultural commodities price on the admin and government user which utilizes *representational state transfer* (REST) *application programming interface* (API) with *extreme programming* methods. This research has successfully developed REST API and a web application that is able to make things easier for the government in terms of monitoring and controlling the distribution of agricultural products from farmers to the markets, and could accelerate the flow of information in agricultural commodity prices from market to government.

Keywords: agricultural, price commodities, REST API

**PENGEMBANGAN MODUL PELAPORAN HARGA KOMODITAS  
PERTANIAN PADA SISI PENGGUNA ADMIN DAN  
PEMERINTAH MENGGUNAKAN REST API**

Skripsi  
sebagai salah satu syarat untuk memperoleh gelar  
Sarjana Komputer  
pada  
Departemen Ilmu Komputer

**DEPARTEMEN ILMU KOMPUTER  
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
INSTITUT PERTANIAN BOGOR  
BOGOR  
2016**

- 1 Ir Meuthia Rachmaniah, MSc
- 2 Dr Yani Nurhadriyani, SSi MT

Judul Skripsi: Pengembangan Modul Pelaporan Harga Komoditas Pertanian pada  
Sisi Pengguna Admin dan Pemerintah Menggunakan REST API

Nama : Dian Setiana

NIM : G64120010

Disetujui oleh

Dean Apriana Ramadhan, SKomp MKom  
Pembimbing

Diketahui oleh

Dr Ir Agus Buono, MSi MKom  
Ketua Departemen

Tanggal Lulus:

## PRAKATA

Puji dan syukur penulis ucapkan kepada Allah *subhanahu wa ta'ala* atas segala karunia-Nya, sehingga penulis dapat menyelesaikan tugas akhir dengan judul Pengembangan Modul Pelaporan Harga Komoditas Pertanian pada Sisi Pengguna Admin dan Pemerintah Menggunakan REST API. Penelitian ini dilaksanakan mulai Maret sampai dengan Agustus 2016 bertempat di Departemen Ilmu Komputer

Pada kesempatan ini penulis ingin mengucapkan rasa terima kasih yang tidak terhingga dan rasa penghargaan yang setinggi-tingginya kepada :

- 1 Keluarga tercinta Ibu (Nunung Sukawati), Ayah (Asep Sutarman), Kakak (Erwan Riswanto dan Nenden Yulianti Sondari), terima kasih atas doa dan kasih sayang serta dorongan yang tak ternilai harganya baik berupa moril dan materil yang penulis rasakan dari awal perkuliahan sampai penulis menyelesaikan tugas akhir ini.
- 2 Bapak Dean Apriana Ramadhan, SKom MKom selaku dosen pembimbing yang selalu memberikan bimbingan dan saran dengan kesabarannya selama pengerjaan tugas akhir.
- 3 Ibu Ir Meuthia Rachmaniah, MSc dan Ibu Dr Yani Nurhadriyani, SSi MT selaku dosen penguji.
- 4 Teman-teman satu bimbingan, yaitu Moh Reza, Reza Pahlevi, Kadek Grihadevi, Indriyani, Dieva Setyatama, dan Shellafuri yang selalu memberikan kritik , saran, dukungan, dan semangatnya.
- 5 Teman-teman kosan, yaitu Rama, Rana, Ramdhan, Alanikika, Dirman, Nurdevi, Fachrizal, Yusuf, Irfan, Gaza, Juan, Hilmi, Iلمان dan Rahmat yang selalu memberikan doa, ilmu, informasi, motivasi, saran, dan semangatnya.
- 6 Teman-teman Vector 49 atas segala bantuan, kebersamaan, ilmu, informasi, dan semangat dalam penyelesaian tugas akhir ini.
- 7 Kepada semua pihak yang telah membantu dan memberikan dorongan baik langsung maupun tidak langsung kepada penulis dalam menyelesaikan tugas akhir ini.

Semoga karya ilmiah ini bermanfaat.

Bogor, Agustus 2016

*Dian Setiana*



## DAFTAR ISI

DAFTAR TABEL	vi
DAFTAR GAMBAR	vi
DAFTAR LAMPIRAN	vi
PENDAHULUAN	1
Latar Belakang	1
Perumusan Masalah	2
Tujuan Penelitian	2
Manfaat Penelitian	2
Ruang Lingkup Penelitian	3
TINJAUAN PUSTAKA	3
REST	3
<i>Roadmap</i> Penelitian	4
Tata Niaga Pertanian	4
METODE PENELITIAN	5
Data Penelitian	5
Perancangan Arsitektur Perangkat Lunak	6
Tahapan Penelitian	6
Peralatan Penelitian	7
HASIL DAN PEMBAHASAN	7
SIMPULAN DAN SARAN	22
Simpulan	22
Saran	22
DAFTAR PUSTAKA	22
RIWAYAT HIDUP	40

## DAFTAR TABEL

1	Data laporan harga komoditas	8
2	<i>User story</i> dari aktor pemerintah dan admin	9
3	Pengujian fungsi untuk aktor admin iterasi pertama	17
4	Pengujian fungsi untuk aktor admin dan pemerintah iterasi kedua	21

## DAFTAR GAMBAR

1	<i>Roadmap</i> penelitian	4
2	Struktur pemerintahan Kementerian Pertanian Republik Indonesia	5
3	Proses dari Model <i>Extreme Programming</i> (Pressman, 2010)	6
4	Arsitektur perangkat lunak	8
5	<i>Use case diagram</i> untuk aktor admin dan pemerintah	10
6	Pemodelan halaman memeriksa harga komoditas dan lokasi	11
7	Rancangan basis data Portal Harga	12
8	<i>Class diagram</i> Portal Harga	12
9	Potongan kode REST API membuat manajemen <i>user</i>	13
10	Format teks JSON ketika POST berhasil	13
11	Format teks JSON ketika POST gagal	13
12	Potongan kode REST API mengubah manajemen <i>user</i>	14
13	Format teks JSON ketika PUT gagal	14
14	Format teks JSON ketika PUT berhasil	14
15	Potongan kode REST API melihat manajemen <i>user</i>	15
16	Format teks JSON ketika GET berhasil	15
17	Format teks JSON ketika GET gagal	15
18	Potongan kode REST API menghapus manajemen <i>user</i>	16
19	Format teks JSON ketika DELETE berhasil	16
20	Format teks JSON ketika DELETE gagal	16
21	<i>Swimlane diagram</i> memeriksa harga komoditas dan lokasi	17
22	Implementasi halaman memeriksa harga komoditas dan lokasi	18
23	Implementasi halaman CRUD manajemen <i>user</i> pada <i>web browser</i>	18
24	Format teks JSON ketika GET berhasil	19
25	Potongan kode REST API GET komoditas	19
26	Format teks JSON ketika GET gagal	20
27	Potongan kode fungsi implementasi GET komoditas	20

## DAFTAR LAMPIRAN

1	Data masukan oleh pengguna	24
2	<i>Fully developed use case description</i>	25
3	Kamus data	32
4	<i>Swimlane diagram</i>	34
5	Implementasi antarmuka sistem	38

# PENDAHULUAN

## Latar Belakang

Indonesia dikenal sebagai negara agraris karena sebagian besar penduduk Indonesia mempunyai mata pencaharian di bidang pertanian. Pertanian mempunyai kontribusi penting baik terhadap perekonomian maupun terhadap kebutuhan pokok masyarakat, ditambah dengan semakin meningkatnya jumlah penduduk yang berarti bahwa kebutuhan akan pangan juga semakin meningkat. Menurut hasil riset yang dilakukan oleh Asosiasi Penyelenggara Jasa Internet Indonesia (APJII) pertumbuhan penggunaan internet di Indonesia terus meningkat. Pada tahun 2014, survey menunjukkan pengguna internet di Indonesia mencapai 88,1 juta jiwa dari seluruh penduduk Indonesia yang berjumlah 252,4 juta. Hal ini terlihat dari banyaknya jumlah penggunaan *smartphone* yang mudah digunakan dan dibawa kemana saja. Selain itu, pengguna dapat mengakses beragam informasi secara cepat dan *real time* melalui internet, karena teknologi internet berkembang semakin pesat dan daya aksesnya semakin luas sehingga mampu menyentuh setiap kalangan dimanapun berada.

Berdasarkan UU No. 18 Tahun 2012 tentang Pangan disebutkan bahwa kegiatan perencanaan, pelaksanaan, dan pengawasan dalam penyediaan, keterjangkauan, pemenuhan konsumsi pangan dan gizi, serta keamanan pangan dengan melibatkan peran serta antara pemerintah dan masyarakat yang terkoordinasi dan terpadu. Berdasarkan UU No. 18 Tahun 2012 pasal 55 tentang Pangan disebutkan bahwa pemerintah wajib melakukan stabilisasi pasokan dan harga pangan pokok di tingkat produsen dan konsumen.

Menurut Undang-undang Republik Indonesia nomor 19 tahun 2013 tentang perlindungan dan pemberdayaan petani disebutkan bahwa pemerintah dan pemerintah daerah sesuai kewenangannya bertanggung jawab atas perlindungan petani, dan juga pemerintah memiliki kewajiban menciptakan kondisi stabilisasi harga pangan. Meskipun pemerintah telah melakukan upaya perlindungan dan pemberdayaan terhadap petani, dalam praktiknya petani dihadapkan berbagai masalah seperti terjadinya perubahan iklim, sistem pasar yang tidak berpihak ke petani, kenaikan harga pasar yang tidak wajar, adanya permainan harga dari tengkulak, kesulitan pemerintah mengadakan operasi pasar, kesulitan konsumen mencari harga yang murah serta fluktuasi harga-harga komoditas pertanian. Hal ini karena kurang terciptanya komunikasi dua arah antara petani dengan pemerintah ataupun petani dengan konsumen yang mengakibatkan keterlambatan penyampaian informasi dari petani ke pemerintah terkait hasil pertanian mereka dan juga konsumen kurang mendapatkan informasi terkait harga komoditas pertanian di wilayah tertentu. Adanya permainan harga pasar dari tengkulak di tengah distribusi komoditas pertanian, membuat petani mengalami kerugian karena hasil panen dibayar dengan harga sangat rendah dan konsumen pun sulit mencari harga murah karena harga komoditas pertanian yang ada di pasaran sangat mahal.

Harga-harga produk pertanian di Indonesia sangat penting untuk diketahui karena sebagian besar produk pertanian merupakan bahan pangan pokok dan harga produk pertanian selalu berkembang di setiap daerah. Beberapa

alasan diperlukannya informasi perkembangan harga produk pertanian yaitu memantau perkembangan harga barang kebutuhan pokok masyarakat, menyusun kebijakan pengadaan dan penyaluran barang kebutuhan pokok masyarakat di Indonesia, menghindari terjadinya kenaikan atau penurunan harga diatas batas normal, dan mengurangi perbedaan harga yang tinggi antar daerah/wilayah di Indonesia (Kemendag 2011). Proses bisnis terkait pelaporan harga komoditas pertanian yang dilakukan oleh pemerintah untuk saat ini yaitu melalui penyuluh atau staf khusus yang terjun langsung ke pasar untuk memeriksa semua harga-harga komoditas pertanian yang kemudian melaporkannya ke pusat.

Penelitian-penelitian sebelumnya telah dilakukan oleh Husen (2014) yang melakukan pemetaan dan analisis distribusi harga produk pertanian menggunakan sistem informasi geografis dan pengembangan sistem informasi harga produk pertanian berbasis SMS oleh Astika *et al.* (2009).

Salah satu solusi adalah dengan mengembangkan aplikasi pelaporan harga komoditas pertanian berbasis web untuk mendapatkan data yang cepat, akurat, dan mengolahnya menjadi informasi yang bermanfaat sehingga memudahkan petani, konsumen dan pemerintah dalam pengambilan kebijakan, selain itu untuk memudahkan pengguna aplikasi web ini dapat diakses melalui komputer atau *smartphone*. Aplikasi web pelaporan harga komoditas pertanian merupakan salah satu upaya untuk mengatasi permasalahan diatas, juga sebagai media analisis pengambilan kebijakan dalam menentukan distribusi komoditas pertanian dengan melihat harga komoditas pertanian yang tersebar di daerah-daerah tertentu.

### **Perumusan Masalah**

Permasalahan yang diangkat dalam penelitian ini adalah

- 1 Bagaimana melakukan pemantauan harga dan perkembangan harga komoditas pertanian?
- 2 Bagaimana merancang aplikasi pelaporan harga komoditas pertanian berbasis web untuk pemantauan harga dan perkembangan harga?
- 3 Bagaimana pemerintah mendapatkan informasi data komoditas pertanian secara *real-time*?

### **Tujuan Penelitian**

Tujuan penelitian ini adalah:

- 1 Mengembangkan aplikasi berbasis web untuk melaporkan harga komoditas pertanian.
- 2 Membangun aplikasi web yang mempercepat penyampaian informasi harga komoditas pertanian dari pasar ke pemerintah.
- 3 Membangun aplikasi web yang dapat memberikan informasi data komoditas pertanian dari masyarakat dan petani di daerah tertentu kepada pemerintah.

### **Manfaat Penelitian**

Manfaat penelitian ini adalah:

- 1 Dari sisi admin
  - a Membantu dalam melakukan kontrol manajemen user, komoditas, dan pasar.

- 2 Dari sisi pemerintah
  - a Terciptanya komunikasi dua arah antara petani dengan pemerintah ataupun petani dengan konsumen sehingga informasi secara cepat diketahui oleh pemerintah, petani, dan konsumen.
  - b Membantu pemerintah dalam mengawasi distribusi produk pertanian.
  - c Menyajikan data untuk mendukung pemerintah dalam pengambilan kebijakan dan mengawasi distribusi pertanian.

### **Ruang Lingkup Penelitian**

Batasan penelitian ini adalah:

- 1 Komoditas pertanian yang digunakan terdiri dari beras, daging sapi, cabai (cabai merah besar, cabai merah keriting, cabai rawit merah, dan cabai rawit hijau), kedelai, dan bawang merah.
- 2 Pasar yang digunakan yaitu pasar di wilayah Jabodetabek (Pasar Induk Kramat Jati, Pasar Jatinegara, Pasar Sunter Podomoro, Pasar Grogol, Pasar Minggu, Pasar Senen, Pasar Kebon Kembang, Pasar Kemiri Muka, Pasar Induk Tanah Tinggi, dan Pasar Cikarang).
- 3 Mengembangkan modul untuk admin dan pemerintah.
- 4 Sistem yang akan dibuat untuk saat ini sebatas mengolah lalu lintas data dari *server* yang akan di *parsing* untuk di gunakan oleh *client* (web dan *mobile*), dengan tidak memperhatikan keamanan datanya sendiri dan diasumsikan semua datanya *valid*.

## **TINJAUAN PUSTAKA**

### **REST**

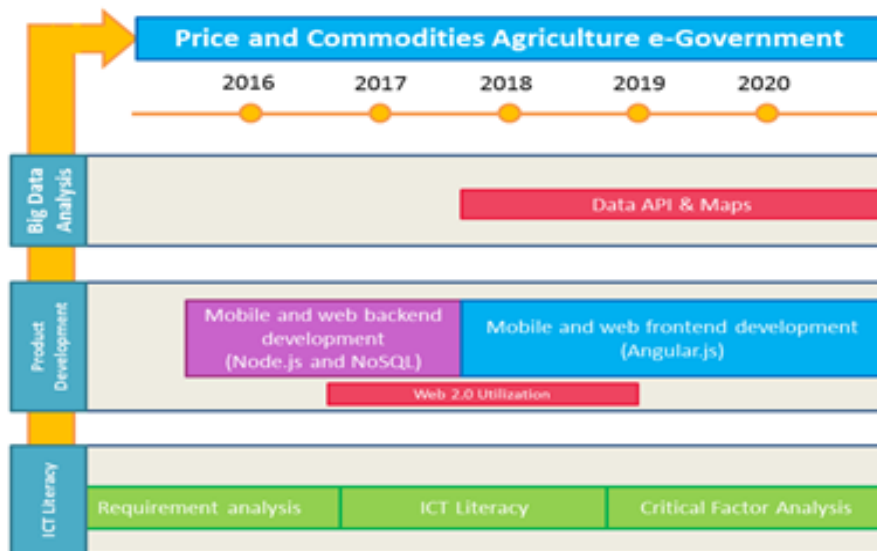
REST adalah protokol yang merupakan akronim dari *REpresentational State Transfer*, apabila dibandingkan dengan protokol-protokol lainnya seperti SOAP atau XML-RPC, REST lebih diartikan sebagai seperangkat prinsip dari pada sebagai protokol. REST adalah seperangkat ide tentang bagaimana data dapat ditransfer dengan lancar. REST mengambil keuntungan besar dari fitur-fitur yang ada pada HTTP (Richadrson dan Ruby 2007).

REST adalah gaya aksitektural yang memiliki aturan seperti antar muka yang seragam, apabila aturan tersebut diterapkan pada *web services* maka akan memaksimalkan kinerja *web services* terutama pada performa, skalabilitas, dan kemudahan modifikasi. REST menggunakan protokol HTTP yang bersifat *stateless*; perintah fungsi yang digunakan seperti GET, POST, PUT atau DELETE. Hasil yang dikirimkan dari server dalam bentuk format *Javascript Object Notation* (JSON) tanpa ada protokol pemaketan data, sehingga informasi yang diterima lebih mudah dibaca dan dikirim disisi *client*. Bentuk *web services* menggunakan REST *style* sangat cocok digunakan sebagai *backend* dari aplikasi berbasis mobile karena cara aksesnya yang mudah dan hasil data yang dikirimkan berformat JSON sehingga ukuran *file* menjadi lebih kecil (Kurniawan 2014).

Semua respon *Application Programming Interface* (API) menggunakan format JSON. Semua tanggapan API memanfaatkan format standar untuk menyediakan cara yang seragam sehingga bisa membedakan antara permintaan yang berhasil ditangani dan permintaan yang gagal ditangani API (Kowal *et al.* 2016).

### Roadmap Penelitian

*Roadmap* merupakan suatu proses perencanaan teknologi untuk mencapai, mengidentifikasi, membantu dan memenuhi serangkaian kebutuhan produk dalam kurun waktu tertentu. *Laboratorium Software Engineering and Information Science (SEInS) e-government research group* memiliki *roadmap* untuk mencapai *Price and Commodities Agriculture e-Government* dengan tiga tahapan perencanaan yaitu *ICT Literacy*, *Product Development*, dan *Big Data Analysis* dalam jangka waktu 2016 – 2020 yang dapat dilihat pada Gambar 1. Penelitian ini merupakan pelaksanaan dari *roadmap* pada tahapan *product development* pada web 2.0 *utilization* dengan mengembangkan *mobile and web backend*. Pada penelitian ini, pemanfaatan web 2.0 terdapat pada modul pelaporan harga komoditas pertanian dan pelaporan permintaan operasi pasar.

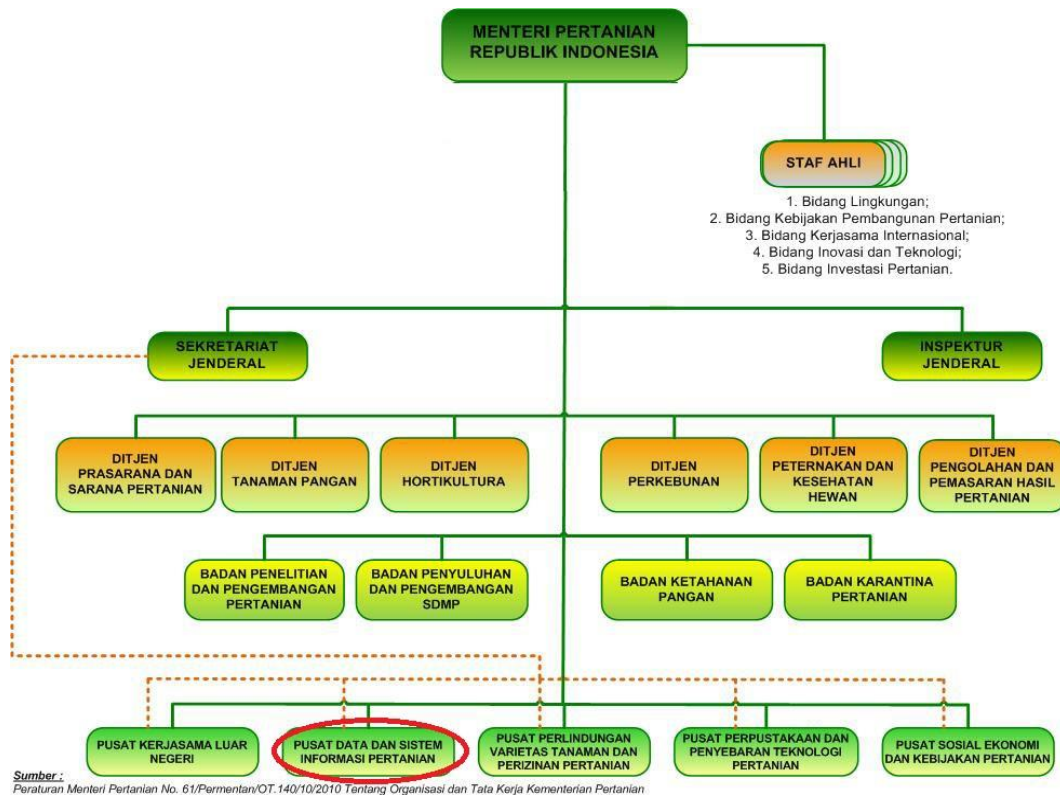


Gambar 1 *Roadmap* penelitian

### Tata Niaga Pertanian

Tata niaga adalah segala kegiatan dan usaha yang berhubungan dengan perpindahan hak milik dan fisik dari barang-barang hasil pertanian dan kebutuhan usaha pertanian dari tingkat produsen ke tingkat konsumen. Sebuah saluran tata niaga melaksanakan tugas memindahkan barang dari produsen ke konsumen. Hal ini mengatasi kesenjangan waktu, tempat, dan kepemilikan yang memisahkan barang atau jasa dari orang-orang yang membutuhkan atau menginginkannya. Adanya jarak antara produsen dengan konsumen maka proses penyaluran produk dari produsen ke konsumen melibatkan beberapa perantara mulai dari produsen sendiri, lembaga-lembaga perantara sampai ke konsumen akhir (Agustina 2008).

Di negara Indonesia terdapat beberapa kementerian yang bekerja sesuai dengan bidangnya masing-masing, salah satu kementerian yang menaungi bidang pertanian yaitu Kementerian Pertanian Republik Indonesia. Struktur Kementerian Pertanian Republik Indonesia dapat dilihat pada Gambar 2. Bagian yang menaungi sistem informasi dan pengelolaan data yaitu bagian Pusat Data dan Sistem Informasi Pertanian.



Gambar 2 Struktur pemerintahan Kementerian Pertanian Republik Indonesia

## METODE PENELITIAN

### Data Penelitian

Data yang digunakan dalam penelitian ini sebatas pencarian data yang ada di pasar di daerah Jabodetabek pada tahun 2016. Pengambilan data komoditas dilakukan dengan memilih jenis komoditas yang sering digunakan oleh masyarakat. Metode pengambilan data, tidak dilakukan turun langsung ke pasar namun dengan melihat dari situs web yang menyediakan data komoditas dan pasar milik pemerintah, salah satunya yaitu web <http://infopangan.jakarta.go.id>.

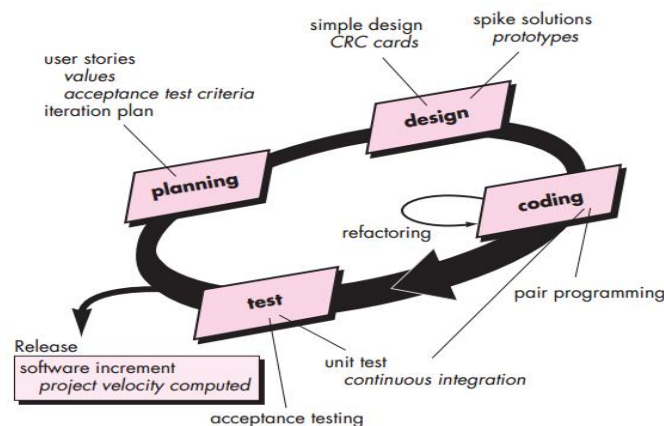
## Perancangan Arsitektur Perangkat Lunak

Arsitektur informasi pada perangkat lunak terdapat tiga bagian yaitu data, *client side*, dan *server side*. Pada bagian data terdiri dari MySQL, bagian *client side* terdapat aplikasi berbasis *website* dan *mobile android*. Perancangan aplikasi berbasis *website* menggunakan PHP *frameworks* Code Igniter 3.0.3. Pada bagian *server side* terdapat *Google Maps* API dan REST API dengan menggunakan Node.js.

### Tahapan Penelitian

Penelitian ini akan dilakukan dengan menggunakan metode *Extreme Programming* (XP) yang termasuk salah satu model proses dari *Agile Software Development*. Model ini memungkinkan proses yang toleransi terhadap perubahan kebutuhan sehingga perubahan dapat cepat ditanggapi.

Tahapan pengembangan sistem dalam Model XP menurut Pressman (2010) dapat dilihat pada Gambar 3. Tahapan yang akan dilakukan selama penelitian adalah:



Gambar 3 Proses dari Model *Extreme Programming* (Pressman, 2010)

#### 1 Perencanaan (*Planning*)

Tahap perencanaan dimulai dengan mendapatkan gambaran fitur dan fungsi dari sistem yang akan dibangun. Tahapan ini ditentukan pelayanan yang diprioritaskan untuk diselesaikan terlebih dahulu melalui indeks prioritas dan dibagi menjadi modul-modul yang lebih kecil. Modul yang dikembangkan adalah modul untuk aktor admin dan pemerintah. Tahap pertama pembuatan aplikasi ini diawali dengan merancang arsitektur informasi dilanjutkan penyusunan *user story*, *use case* dan *fully developed use case description* untuk mengetahui rancangan aplikasi yang akan dibangun. *User story* ini akan memuat alur penggunaan sistem oleh pengguna, sedangkan *fully developed use case description* akan memuat skenario, deskripsi, aktor, aktivitas dari *use case*.

#### 2 Desain (*Design*)

Tahap desain pada model proses XP menjadi panduan dalam membangun perangkat lunak yang didasari dari perencanaan pada tahap sebelumnya. Proses desain seperti perancangan antarmuka aplikasi, basis data, dan *Class*



*Responsibility Collaborator* (CRC). Proses desain terjadi sebelum dan sesudah aktivitas *coding* berlangsung, yang berarti aktivitas desain terjadi secara terus-menerus selama proses pengembangan aplikasi berlangsung.

### 3 Pengkodean (*Coding*)

Setelah menyelesaikan perencanaan dan desain untuk aplikasi secara keseluruhan maka tahapan selanjutnya adalah melakukan pengkodean (*coding*). Model XP juga lebih mengedepankan *pair programming* yang mana melibatkan lebih dari satu orang untuk melakukan pengkodean, yang kemudian digabungkan menjadi satu sistem yang utuh. Pengkodean (*coding*) menggunakan bahasa pemrograman PHP dengan *framework Code Igniter* (CI) versi 3.03, Node.js, Javascript, dan AJAX. Jika terjadi perubahan pada fungsi atau desain maka dilakukan *refactoring*.

### 4 Pengujian (*Testing*)

Setelah semua modul telah dikumpulkan dalam sebuah sistem, barulah dilakukan pengujian pada bagian terkecil dari sebuah fungsi (*unit test*) dan pengujian penerimaan (*acceptance test*). Pada tahapan pengujian ini aplikasi langsung diuji sebanyak dua kali oleh tim internal dengan menggunakan metode *Black Box Testing*. Selain itu pengujian dilakukan dengan menggunakan aplikasi Postman untuk menguji REST API. Setelah pengujian berhasil dilakukan selanjutnya melakukan *release* terhadap aplikasi web yang telah dibuat.

## Peralatan Penelitian

Penelitian ini dilakukan dengan menggunakan perangkat keras dan perangkat lunak sebagai berikut :

Perangkat keras berupa komputer personal dengan spesifikasi sebagai berikut :

- 1 Processor Intel Core i5-2430M
- 2 RAM 6 GB
- 3 700 GB HDD

Perangkat lunak :

- 1 Sistem operasi Windows 10
- 2 Bahasa pemrograman HTML, AJAX, Node.js, Javascript, dan PHP
- 3 Sublime Text sebagai *text editor*
- 4 Web server: Apache2
- 5 DBMS : MySQL
- 6 *Browser* : Google Chrome
- 7 Balsamiq digunakan untuk membuat *mock-up*
- 8 Postman digunakan untuk menguji REST API

## HASIL DAN PEMBAHASAN

### Data Penelitian

Data yang digunakan dalam penelitian ini adalah data yang bersumber dari pengguna aplikasi yang melaporkan jenis komoditas, harga, lokasi, *request* operasi pasar yang dikirim masyarakat ke pemerintah. Data laporan harga komoditas dapat

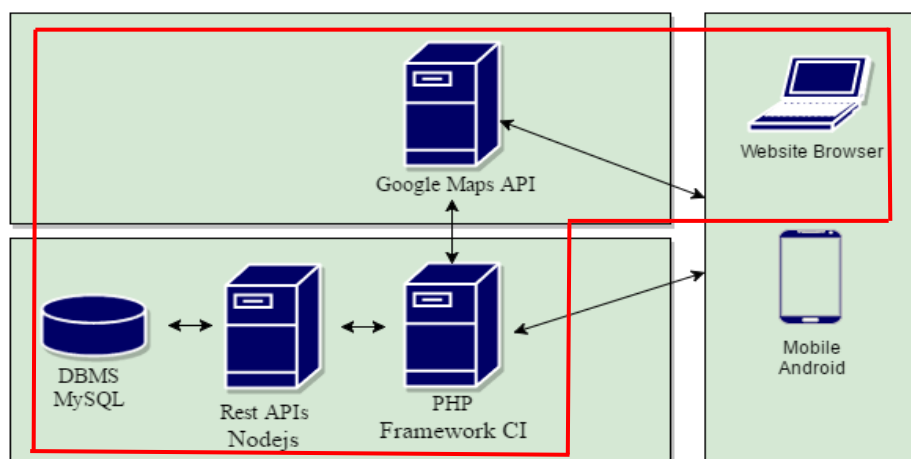
dilihat pada Tabel 1. Pada data laporan harga komoditas terdapat *field* komoditas, harga, tanggal, lokasi, *latitude*, dan *longitude*. *Field* komoditas adalah nama jenis komoditas yang dimasukkan pengguna. *Field* harga adalah data harga komoditas dalam satuan rupiah. *Field* tanggal adalah informasi waktu dimasukkannya data komoditas oleh pengguna. *Field* lokasi, *latitude*, dan *longitude* adalah keterangan lokasi data komoditas di masukkan. Data laporan harga komoditas lainnya dapat dilihat pada Lampiran 1.

Tabel 1 Data laporan harga komoditas

Komoditas	Harga	Tanggal	Lokasi	<i>Latitude</i>	<i>Longitude</i>
Cabai merah keriting	32000	06-07-2016	East Bekasi, Bekasi City, West Java	-6.2362538	107.0244170

### Arsitektur Perangkat Lunak

Arsitektur perangkat lunak yang digunakan pada penelitian ini dapat dilihat pada Gambar 4. Pada arsitekur perangkat lunak ini terdapat tiga bagian yaitu bagian data, *server side*, dan *client side*. Pada bagian data terdapat MySQL yang digunakan sebagai basis data. Pada bagian *server side* terdapat REST API dengan menggunakan bahasa pemrograman Node.js. REST API digunakan sebagai perantara *request* dari *client* ke *server*. Bahasa pemrograman yang digunakan yaitu PHP dengan *framework* Code Igniter (CI) versi 3.03. Google Maps API digunakan untuk menampilkan peta dan menandai lokasi pelaporan dari masyarakat dan petani dengan mengambil *longitude* dan *latitude*. Pada bagian *client* terdapat *website browser* dan *mobile android*. Modul pada penelitian ini terdiri dari empat modul yaitu modul admin, masyarakat, petani, dan pemerintah. Penelitian ini dilakukan secara tim, pengembangan *mobile android* dikerjakan oleh Dieva, modul masyarakat dikerjakan oleh Kadek, dan modul petani dikerjakan oleh Shellafuri. Pada penelitian ini berfokus pada pengembangan aplikasi *website* untuk modul admin dan pemerintah.



Gambar 4 Arsitektur perangkat lunak

## Tahapan Pengembangan Sistem

Aplikasi pelaporan harga komoditas pertanian berbasis web dikembangkan dengan menggunakan Model *Extreme Programming* menurut Pressman (2010). Pembuatan aplikasi web ini mengalami dua kali iterasi *Extreme Programming* (XP). Metode ini terdiri atas empat tahap, berikut adalah tahapan-tahapan pengembangan aplikasi web pelaporan harga komoditas pertanian.

### 1. Iterasi Pertama

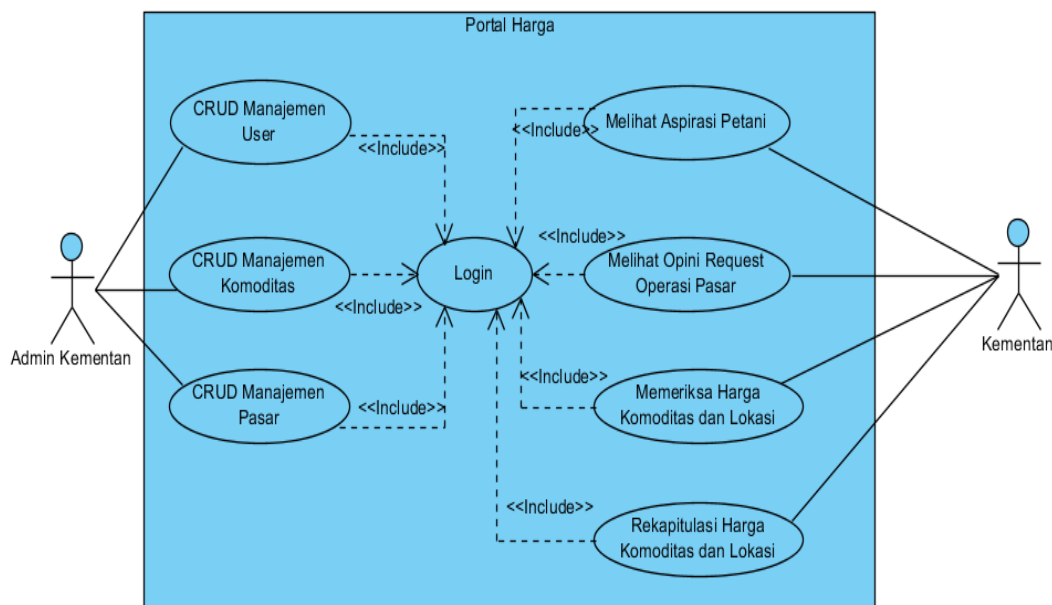
#### 1.1 Tahap Perencanaan (XP Planning)

Tahap pertama perencanaan dimulai dengan pembuatan *user story*, *use case diagram*, dan *fully developed use case description*. *User story* dari aktor pemerintah dan admin dapat dilihat pada Tabel 2. Setelah dibuat *user story* dari masing-masing *task* selanjutnya membuat *use case diagram* dari modul admin dan pemerintah yang dapat dilihat pada Gambar 5. *Use Case* ini bertujuan untuk mengetahui rancangan aplikasi yang akan dibangun.

Tabel 2 *User story* dari aktor pemerintah dan admin

Aktor	Task	User Story	Prioritas
Pemerintah	Memeriksa harga komoditas dan lokasi	Pemerintah melihat harga-harga komoditas pertanian yang telah dilaporkan oleh masyarakat dan petani. Pemerintah melakukan pemantauan harga dengan cara melihat titik lokasi di <i>map</i> .	1
	Melihat opini <i>request</i> operasi pasar	Pemerintah mendapatkan notifikasi opini <i>request</i> operasi pasar dari masyarakat. Aspirasi-aspirasi yang terkumpul segera direspon dan diproses oleh pemerintah untuk ditindak lanjuti.	2
	Melihat aspirasi petani	Pemerintah akan mendapatkan aspirasi bisa berupa saran dan kritikan dari petani. Aspirasi-aspirasi yang terkumpul segera direspon dan diproses oleh pemerintah untuk ditindak lanjuti.	3
	Rekapitulasi harga komoditas dan lokasi	Pemerintah bisa mendapatkan rekapitulasi harga-harga komoditas pertanian dan lokasi secara keseluruhan. Data rekapitulasi bisa didownload dalam format Microsoft Excel, PDF, dan CSV.	4

Aktor	Task	User Story	Prioritas
Admin	CRUD manajemen user	Admin memiliki hak akses untuk melakukan manajemen <i>user</i> seperti menambah, melihat, mengubah dan menghapus <i>user</i> .	1
	CRUD manajemen komoditas	Admin memiliki hak akses untuk melakukan manajemen komoditas seperti menambah, melihat, mengubah dan menghapus komoditas.	2
	CRUD manajemen pasar	Admin memiliki hak akses untuk melakukan manajemen pasar seperti menambah, melihat, mengubah dan menghapus pasar.	3



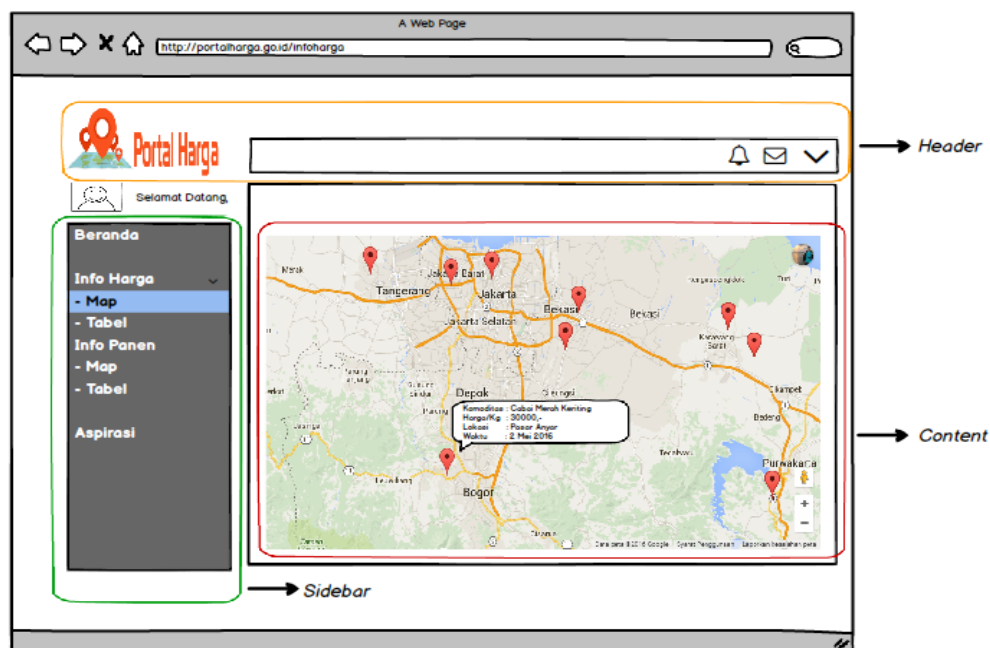
Gambar 5 *Use case diagram* untuk aktor admin dan pemerintah

Admin Kementrian Pertanian (Kementan) merupakan aktor yang memiliki kewenangan untuk melakukan manajemen *user*, pasar, dan komoditas seperti menambah, melihat, mengubah dan menghapus. Pemerintah dalam hal ini Kementan merupakan aktor yang memiliki kewenangan untuk berkomunikasi dengan masyarakat dan petani melalui aspirasi petani dan juga opini *request* operasi pasar, selain itu pemerintah bisa memeriksa harga komoditas, lokasi, dan melihat rekapitulasi harga komoditas dan lokasi. Data yang diperlukan oleh modul pemerintah yaitu dengan menggunakan data dari modul eksternal petani dan masyarakat. Setelah pembuatan *use case diagram* barulah dibuat *fully developed use case description* yang dapat dilihat pada Lampiran 2.

## 1.2 Tahap Desain (XP Design)

Tahap desain dimulai dengan pembuatan rancangan antarmuka aplikasi berdasarkan *use case* dan skenario yang telah dibuat pada tahap perencanaan. Perancangan antarmuka aplikasi web ini terdiri dari tiga bagian yaitu *header*, *sidebar*, dan *content*. Bagian *header* berisi logo dan *icon* pemberitahuan. Bagian *sidebar* berisi menu navigasi. Bagian *content* berisi fungsi-fungsi utama pada setiap halaman.

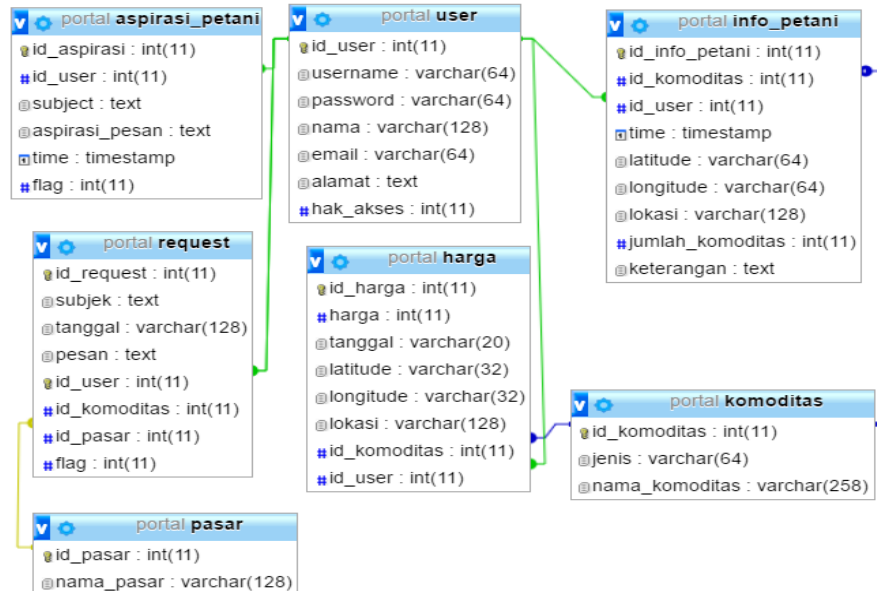
Gambar 6 merupakan pemodelan antarmuka halaman dari fungsi memeriksa harga komoditas dan lokasi. Bagian *header* berisi logo dan *icon* pemberitahuan. Bagian *sidebar* berisi menu navigasi terdiri dari beranda, menu info harga, info panen, dan aspirasi. Bagian *content* menampilkan *map* berupa titik-titik lokasi hasil data yang di *input* oleh masyarakat. Titik-titik lokasi pada *map* menampilkan detail informasi berupa lokasi, nama, dan harga komoditas. Selain menampilkan *map* berupa titik-titik lokasi hasil data yang di *input* oleh masyarakat, juga bisa menampilkan tabel. Petani dapat melaporkan terkait harga panen dan juga masyarakat dapat melaporkan harga komoditas pertanian di daerahnya. Hal ini bertujuan untuk menyajikan data secara cepat dan membantu pemerintah dalam pengambilan suatu kebijakan.



Gambar 6 Pemodelan halaman memeriksa harga komoditas dan lokasi

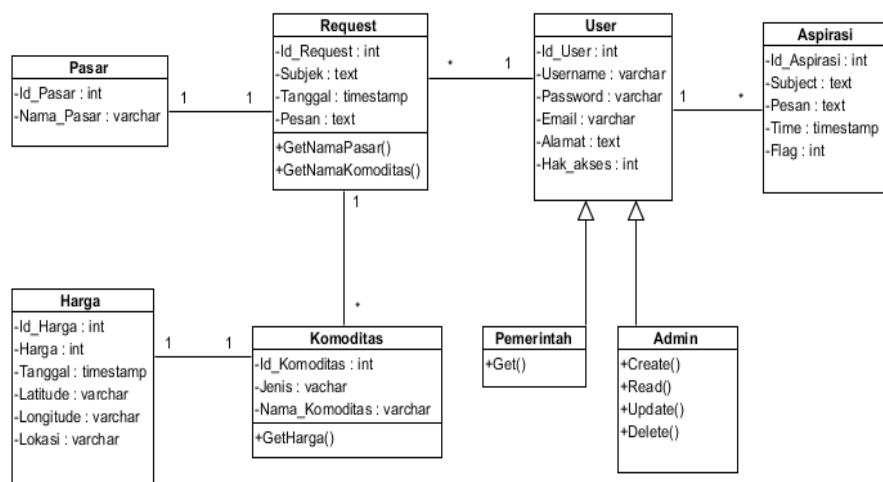
Pada tahap ini dilakukan perancangan basis data yang dapat dilihat pada Gambar 7. Basis data dirancang dengan tujuh entitas, yaitu entitas *user*, pasar, harga, komoditas, aspirasi\_petani, info\_petani, *request*. Entitas *user* memiliki hubungan dengan entitas aspirasi\_petani, info\_petani yang hubungannya *user* petani bisa menginformasikan hasil panen di daerahnya dan mengirimkan aspirasi ke pemerintah.

Entitas *request*, memiliki hubungan dengan entitas *user* dan pasar yang hubungannya *user* masyarakat bisa melakukan *request* operasi pasar ke pemerintah dan memberikan info terkait harga-harga komoditas pertanian di daerahnya. Entitas komoditas memiliki hubungan dengan entitas harga dan *info\_petani* yang hubungannya memberikan informasi terkait harga dan jenis komoditas. Kamus data dari *database* ini dapat dilihat pada Lampiran 3.



Gambar 7 Rancangan basis data Portal Harga

Pada tahap ini juga dilakukan pembuatan *class diagram* yang bertujuan untuk menggambarkan objek yang ada di dalam sistem. Pada penelitian ini terdapat delapan *class* yang terdiri dari *class* pasar, *request*, *user*, aspirasi, harga, komoditas, pemerintah, dan admin yang digunakan untuk merepresentasikan objek yang terkait dengan sistem. *Class* pemerintah dan admin merupakan generalisasi dari *class* user dan setiap *class* memiliki nilai atribut dan *method*. Seluruh *class* yang dibuat dapat dilihat secara lengkap pada Gambar 8.



Gambar 8 Class diagram Portal Harga

### 1.3 Tahap Pengkodean (XP Coding)

Tahap pengkodean dipelajari terlebih dahulu pembuatan REST API dan *method* yang ada pada Node.js seperti POST, GET, PUT dan DELETE. Hal ini dilakukan untuk membuat API untuk kemudian digunakan oleh sisi *client* yaitu *website* dan *mobile*. Terdapat beberapa fungsi yang dibuat, diantaranya adalah fungsi CRUD (*Create, Read, Update, Delete*) manajemen *user*, pasar, dan komoditas. Fungsi REST API yang dibuat yaitu POST, GET, PUT, dan DELETE. Gambar 9 merupakan potongan kode REST dari fungsi *Create* manajemen *user*. Potongan kode REST ini menggunakan *method* POST yang berfungsi untuk memasukkan data ke dalam tabel dengan *field* yang dimasukkan oleh admin.

```

77 router.post("/users", function(req,res){
78   var query="INSERT INTO ??(?,?,?, ?, ?, ?) VALUES
      (?,?, ?, ?, ?, ?) ";
79   var table=
      ["user","username","password","nama","email","alamat","hak_a
      kses",req.body.username,md5(req.body.password),req.body.nama
      ,req.body.email,req.body.alamat,req.body.hak_akses];
80   query=mysql.format(query,table);
81   connection.query(query,function(err,rows){
82     if(err) {
83       res.json({"Error" : true, "Message" : "Error
      executing MySQL query"});
84     } else {
85       res.json({"Error" : false, "Message" : "User Adedd !" });
86     }
87   });

```

Gambar 9 Potongan kode REST API membuat manajemen *user*

Fungsi REST API pada Gambar 9 dapat dijalankan dengan mengakses alamat pada URL [http://localhost:3000/api\\_portalharga/users](http://localhost:3000/api_portalharga/users) dengan *method* POST. Jika fungsi diatas gagal karena MySQL tidak terkoneksi maka terdapat *error* dan akan muncul keluaran berupa format teks JSON seperti pada Gambar 10. Jika prosesnya berhasil maka data yang dimasukkan berhasil tersimpan ke tabel *user* dan menghasilkan keluaran teks JSON seperti pada Gambar 11.

```

1  {
2    "Error": true,
3    "Message": "Error executing MySQL query"
4  }

```

Gambar 10 Format teks JSON ketika POST gagal

```

1  {
2    "Error" : false,
3    "Message" : "User Adedd !"
4  }

```

Gambar 11 Format teks JSON ketika POST berhasil

```

89 router.put("/users",function(req,res) {
90     var query = "UPDATE ?? SET ?? = ?, ?? = ?, ?? = ?, ?? =
        ?, ?? = ? WHERE ?? = ?";
91     var table =
["user","nama",req.body.nama,"username",req.body.username,"
email",req.body.email,"alamat",req.body.alamat,"hak_akses",
req.body.hak_akses,"id_user",req.body.id_user];
92     query = mysql.format(query,table);
93     connection.query(query,function(err,rows) {
94         if(err) {
95             res.json({"Error" : true, "Message" : "Error
        executing MySQL query"});
96         }else{
97             res.json({"Error" : false, "Message" :
        "Success", "Users update" : rows});
98         }
99     });
100 });

```

Gambar 12 Potongan kode REST API mengubah manajemen *user*

Gambar 12 merupakan potongan kode REST dari fungsi mengubah manajemen *user*. Potongan kode REST ini menggunakan *method* PUT yang berfungsi untuk merubah data pada tabel *user* yang dirubah oleh admin. Jika admin akan melakukan perubahan pada tabel *user*, dapat dijalankan dengan mengakses alamat pada URL [http://localhost:3000/api\\_portalharga/users](http://localhost:3000/api_portalharga/users) dengan *method* PUT maka tabel *user* akan berubah. Format keluaran teks JSON pada fungsi mengubah manajemen *user* dapat dilihat pada Gambar 13 dan Gambar 14.

```

1 {
2     "Error": true,
3     "Message": "Error executing MySQL query"
4 }

```

Gambar 13 Format teks JSON ketika PUT gagal

```

1 {
2     "Error" : false,
3     "Message" : "User Updated !"
4 }

```

Gambar 14 Format teks JSON ketika PUT berhasil

Gambar 15 merupakan potongan kode REST dari fungsi melihat manajemen *user*. Potongan kode REST ini menggunakan *method* GET yang berfungsi untuk mengambil data pada tabel *user*. Fungsi REST API tersebut dapat dijalankan dengan mengakses alamat pada URL [http://localhost:3000/api\\_portalharga/users](http://localhost:3000/api_portalharga/users) dengan *method* GET. Jika fungsi REST API tersebut berhasil maka akan muncul keluaran berupa format JSON seperti pada Gambar 16.



```

101 router.get("/users",function(req,res){
102     var query = "SELECT * FROM ??";
103     var table = ["user"];
104     query = mysql.format(query,table);
105     connection.query(query,function(err,rows){
106         if(err) {
107             res.json({"Error" : true, "Message" :
108                 "Error executing MySQL query"});
109         } else {
110             res.json({"Error" : false, "Message" :
111                 "Success", "Users" : rows});
112         }
113     });
114 }

```

Gambar 15 Potongan kode REST API melihat manajemen *user*

Objek pada format teks JSON berada di dalam kurung siku dan antar objeknya dipisahkan dengan kurung kurawal. *Array* JSON pada Gambar 16 adalah data *user* yang memiliki beberapa objek dengan tujuh *field* yaitu *id\_user*, *username*, *password*, *nama*, *email*, *alamat*, dan *hak\_akses*. *Field id\_user* yaitu urutan identitas unik bagi pengguna ketika registrasi, *field username* dan *password* yaitu identitas untuk melakukan login. *Field nama*, *email*, dan *alamat* yaitu identitas penjabar bagi pengguna, *field hak\_akses* yaitu kewenangan yang diberikan bagi pengguna ketika login. Jika fungsi tersebut gagal karena MySQL tidak terkoneksi maka terdapat *error* dan akan muncul keluaran berupa format teks JSON seperti pada Gambar 17.

```

1  {
2    "Error": false,
3    "Message": "Success",
4    "Users": [
5      {
6        "id_user": 1,
7        "username": "pemerintah",
8        "password": "d784ea4b4a2ca7825ebf6d6e802d3",
9        "nama": "Pemerintah",
10       "email": "Pemerintah@gmail.com",
11       "alamat": "Bogor",
12       "hak_akses": 1
13     },
14   ]
15 }

```

Gambar 16 Format teks JSON ketika GET berhasil

```

1  {
2    "Error": true,
3    "Message": "Error executing MySQL query"
4  }

```

Gambar 17 Format teks JSON ketika GET gagal

Gambar 18 merupakan potongan kode REST dari fungsi menghapus manajemen *user*. Potongan kode REST ini menggunakan *method* DELETE yang berfungsi untuk menghapus data pada tabel *user* oleh admin. Jika admin akan menghapus data pada tabel *user*, dapat dijalankan dengan mengakses alamat pada URL `http://localhost:3000/api_portalharga/users/:id_user` dengan *method* DELETE maka data pada tabel *user* akan terhapus. Format keluaran teks JSON pada fungsi menghapus manajemen *user* dapat dilihat pada gambar 19 dan Gambar 20. *Pair Programming* pada iterasi ini dilakukan dengan pertemuan rutin dengan tim internal bagian pengembangan *website* dan melakukan pengkodean dalam satu komputer yang sama.

```

113 router.delete("/users/:id_user",function(req,res){
114     var query = "DELETE from ?? WHERE ??=?";
115     var table = ["user","id_user",req.params.id_user];
116     query = mysql.format(query,table);
117     connection.query(query,function(err,rows){
118         if(err) {
119             res.json({"Error" : true, "Message" : "Error
120                 executing MySQL query"});
121         } else {
122             res.json({"Error" : false, "Message" :
123                 "Deleted the user with id_user
124                 "+req.params.id_user});
125         }
126     });
127 });

```

Gambar 18 Potongan kode REST API menghapus manajemen *user*

```

1 {
2   "Error": false,
3   "Message": "Deleted the user with id_user"
4 }

```

Gambar 19 Format teks JSON ketika DELETE berhasil

```

1 {
2   "Error": true,
3   "Message": "Error executing MySQL query"
4 }

```

Gambar 20 Format teks JSON ketika DELETE gagal

#### 1.4 Tahap Pengujian (XP Testing)

Tahap pengujian yang dilakukan dalam pembuatan aplikasi web pelaporan harga komoditas pertanian ini dilakukan oleh internal tim dengan menguji setiap fungsi yang dibuat, yaitu CRUD manajemen *user*, manajemen pasar, dan manajemen komoditas yang dapat dilihat pada Tabel 3. Hal ini bertujuan untuk mengetahui fungsi - fungsi yang terdapat pada sistem apakah berjalan sesuai dengan yang diharapkan. Selain itu pengujian dilakukan dengan menggunakan aplikasi Postman untuk memeriksa apakah pengkodean REST API berjalan dengan baik.

Tabel 3 Pengujian fungsi untuk aktor admin iterasi pertama

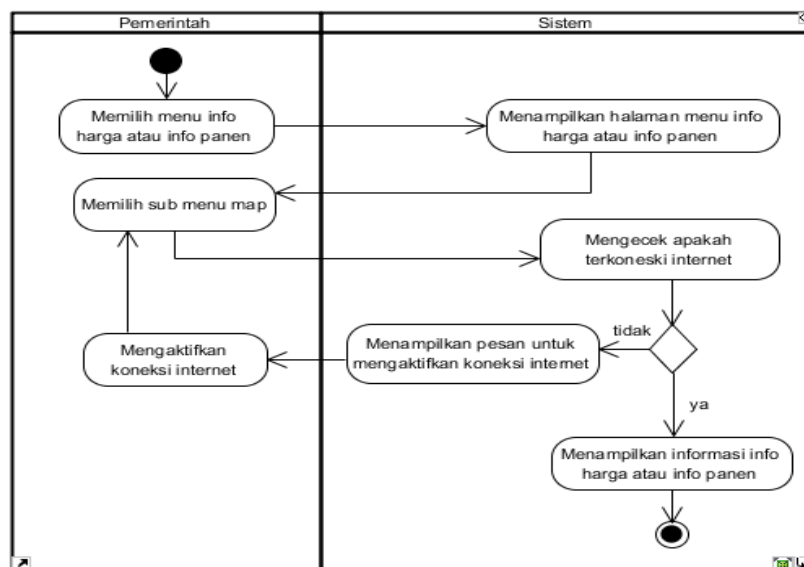
Actor	Fungsi	Status Pengujian
Admin	<i>GET user</i>	Berhasil
	<i>POST user</i>	Berhasil
	<i>DELETE user</i>	Berhasil
	<i>PUT user</i>	Berhasil
	<i>GET komoditas</i>	Berhasil
	<i>POST komoditas</i>	Berhasil
	<i>DELETE komoditas</i>	Berhasil
	<i>PUT komoditas</i>	Berhasil
	<i>GET pasar</i>	Berhasil
	<i>POST pasar</i>	Berhasil
	<i>DELETE pasar</i>	Berhasil
	<i>PUT pasar</i>	Berhasil
	CRUD manajemen user	Berhasil
	CRUD manajemen komoditas	Berhasil
	CRUD manajemen pasar	Berhasil

## 2. Iterasi Kedua

### 2.1 Tahap Perencanaan (XP Planning)

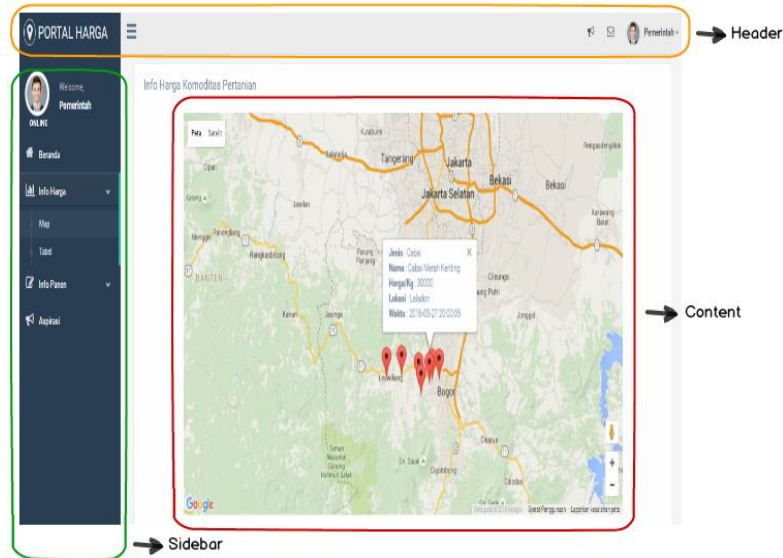
Tahap perencanaan iterasi kedua dilakukan pemeriksaan dari *use case diagram*, basis data, *user story*, dan *fully developed use case description*. Pada tahap perencanaan ini juga di buat *swimlane diagram* yang berfungsi menggambarkan interaksi dari beberapa bagian yang terlibat dalam suatu fungsi.

Gambar 21 merupakan salah satu *swimlane diagram* dari fungsi memeriksa harga komoditas dan lokasi. Bagian yang terlibat dalam proses ini yaitu aktor pemerintah dan sistem. *Swimlane diagram* dari fungsi lainnya dapat dilihat pada Lampiran 4.

Gambar 21 *Swimlane diagram* memeriksa harga komoditas dan lokasi

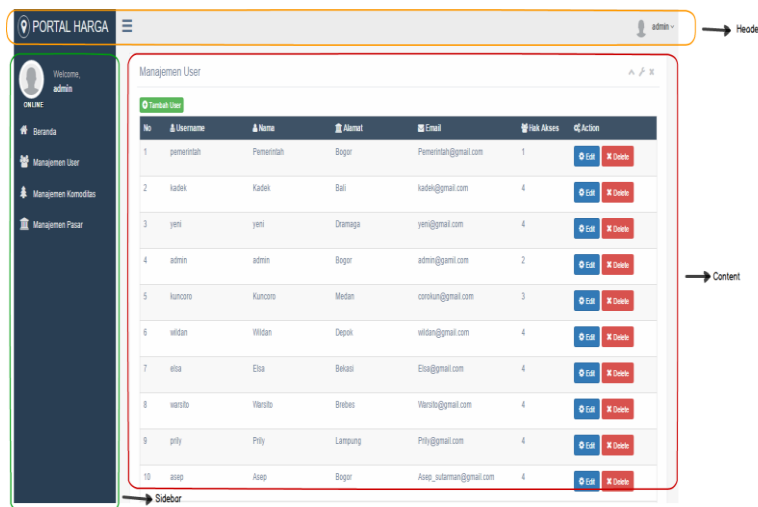
## 2.2 Tahap Desain (XP Design)

Tahap desain iterasi kedua dilakukan pemodelan halaman menu pemerintah dan admin dari fungsi utama langsung pada tampilan *web browser*. Gambar 22 merupakan halaman memeriksa harga komoditas dan lokasi pada *web browser*. Bagian *header* berisi logo dan *icon* pemberitahuan. Bagian *sidebar* berisi menu navigasi terdiri dari beranda, menu info harga, info panen, dan aspirasi. Bagian *content* berisi fungsi memeriksa harga komoditas dan lokasi, dengan melihat titik - titik lokasi pada *map* menampilkan detail informasi berupa jenis, lokasi, nama, waktu, dan harga komoditas.



Gambar 22 Implementasi halaman memeriksa harga komoditas dan lokasi pada *web browser*

Gambar 23 merupakan halaman CRUD manajemen *user* pada *web browser*. Bagian *header* berisi logo. Bagian *sidebar* berisi menu navigasi terdiri dari beranda, menu manajemen *user*, manajemen komoditas, dan manajemen pasar. Bagian *content* berisi fungsi CRUD untuk melakukan manajemen *user*.



Gambar 23 Implementasi halaman CRUD manajemen *user* pada *web browser*

### 2.3 Tahap Pengkodean (XP Coding)

Tahap pengkodean iterasi kedua dilakukan pembuatan fungsi REST API untuk aktor admin dan pemerintah. Pengkodean REST API fungsi GET komoditas dapat dilihat pada Gambar 24. *Method* yang digunakan pada Gambar 24 adalah GET. Proses GET ini akan mengambil data sesuai dengan *query* diminta ke *server*. Jika proses GET berhasil, maka *server* akan memberikan respons dengan menampilkan data berupa format teks JSON seperti pada Gambar 25 yang nantinya akan digunakan oleh sisi *client* yaitu *website* dan *mobile*.

```

125 router.get("/komoditas",function(req,res){
126     var query = "SELECT * from ??, ?? where
        komoditas.id_komoditas=harga.id_komoditas";
127     var table = ["komoditas", "harga"];
128     query=mysql.format(query,table);
129     connection.query(query,function(err,data){
130         if(err) {
131             res.json({"Error" : true, "Message" : "Error
        executing MySQL query"});
132         }else {
133             res.json({data});
134         }
135     });
136 }
```

Gambar 24 Potongan kode REST API GET komoditas

```

1  {"data": [
2    {
3      "id_komoditas": 2,
4      "jenis": "Cabai",
5      "nama_komoditas": "Cabai Merah Keriting ",
6      "id_harga": 2,
7      "harga": 30000,
8      "tanggal":"2016-05-27 19:30:51",
9      "latitude":"-6.5860408",
10     "longitude":"106.7201375",
11     "lokasi": "Laladon",
12     "id_user": 2
13   },
```

Gambar 25 Format teks JSON ketika GET berhasil

Array JSON pada Gambar 25 adalah data komoditas yang memiliki satu objek dengan sepuluh *field* yaitu *id\_komoditas*, *jenis*, *nama\_komoditas*, *id\_harga*, *harga*, *tanggal*, *latitude*, *longitude*, *lokasi*, dan *id\_user*. *Field* *id\_komoditas*, *jenis*, *nama\_komoditas* yaitu identitas dari komoditas. *Field* *id\_harga*, *harga* yaitu identitas harga pada komoditas. *Field* *tanggal* digunakan sebagai keterangan kapan laporan itu dibuat oleh masyarakat.

*Field latitude, longitude* akan digunakan untuk memetakan *map* pada fungsi memeriksa komoditas dan lokasi. *Field* lokasi yaitu tempat dimana laporan itu di buat oleh masyarakat. *Field id\_user* yaitu identitas dari pembuat laporan. Jika fungsi tersebut gagal karena MySQL tidak terkoneksi maka terdapat *error* dan akan muncul keluaran berupa format teks JSON seperti pada Gambar 26.

```

1  {
2  "Error": true,
3  "Message": "Error executing MySQL query"
4  }

```

Gambar 26 Format teks JSON ketika GET gagal

Pada iterasi ini juga dilakukan implementasi ke *website* dari semua fungsi REST API menggunakan bahasa pemrograman PHP *framework Code Igniter* (CI) versi 3.03 dan *Asynchronous JavaScript and XML* (AJAX). Implementasi potongan kode pada Gambar 27 berfungsi untuk memanggil URL dari REST API dan melakukan *parsing* dari format teks JSON yang kemudian akan ditampilkan dalam bentuk web. Implementasi dari antarmuka web aplikasi dapat dilihat pada Lampiran 5. Selain itu ada penambahan fungsi *token authentication* yang bertujuan agar data lebih terjaga kerahasiaannya. *Pair Programming* pada iterasi ini dilakukan dengan pertemuan rutin dengan tim internal bagian pengembangan *website* dan melakukan pengkodean dalam satu komputer yang sama.

```

137 function komoditas() {
138     $.ajax({
139         type: 'GET',
140         url:
141             "http://localhost:3000/api_portalharga/get_komoditas
142             /?token=portalharga",
143         success: function(msg) {
144             $("table tbody#body2 tr").remove();
145             $("#tab").show();
146             $(msg.komoditas).each(function(key, value) {
147                 key++;
148                 $("table
149                     #body2").append("<tr><td>"+key+"<td>"+value.jen
150                     is+"</td><td>"+value.nama_komoditas+"</td></td>
151                     <td><button class='btn btn-primary btn-sm'
152                     onclick='get_edit_komoditas(\"+value.id_komodita
153                     s+\")'><i class='glyphicon glyphicon-cog'></i>
154                     Edit</button><button class='btn btn-danger btn-
155                     sm'
156                     onclick='hapus_komoditas(\"+value.id_komoditas+
157                     \")'><i class='glyphicon glyphicon-remove'></i>
158                     Delete</button></td></tr>");
159             });
160         }
161     });
162 }

```

Gambar 27 Potongan kode fungsi implementasi GET komoditas

## 2.4 Tahap Pengujian (XP Testing)

Tahap pengujian pada iterasi kedua menggunakan metode *Black Box Testing*. Proses pengujian dilakukan oleh internal tim dengan menguji setiap fungsi yang dibuat, yaitu fungsi setiap REST API dengan *software* Postman, memeriksa harga komoditas dan lokasi, *request* operasi pasar, rekapitulasi, dan aspirasi yang dapat dilihat pada Tabel 4. Setelah pengujian REST API berhasil tahap selanjutnya adalah pengujian terhadap implementasi REST API pada aplikasi web. Hasil pengujian pada semua fungsi dapat berjalan dengan baik.

Tabel 4 Pengujian fungsi untuk aktor admin dan pemerintah iterasi kedua

<i>Actor</i>	Fungsi	Status Pengujian
Admin	<i>GET user</i>	Berhasil
	<i>POST user</i>	Berhasil
	<i>DELETE user</i>	Berhasil
	<i>PUT user</i>	Berhasil
	<i>GET</i> komoditas	Berhasil
	<i>POST</i> komoditas	Berhasil
	<i>DELETE</i> komoditas	Berhasil
	<i>PUT</i> komoditas	Berhasil
	<i>GET</i> pasar	Berhasil
	<i>POST</i> pasar	Berhasil
	<i>DELETE</i> pasar	Berhasil
	<i>PUT</i> pasar	Berhasil
	CRUD manajemen user	Berhasil
	CRUD manajemen komoditas	Berhasil
	CRUD manajemen pasar	Berhasil
Pemerintah	<i>GET</i> komoditas	Berhasil
	<i>GET</i> info harga	Berhasil
	<i>GET</i> info panen	Berhasil
	<i>GET</i> aspirasi petani	Berhasil
	<i>GET request</i> operasi pasar	Berhasil
	<i>GET</i> notifikasi <i>request</i>	Berhasil
	<i>GET</i> notifikasi aspirasi	Berhasil
	Melihat aspirasi petani	Berhasil
	Melihat opini <i>request</i> operasi pasar	Berhasil
	Memeriksa harga komoditas dan lokasi	Berhasil
	Rekapitulasi harga komoditas dan lokasi	Berhasil

## SIMPULAN DAN SARAN

### Simpulan

Penelitian ini telah berhasil mengembangkan REST API dan aplikasi web portal harga untuk melaporkan harga komoditas pertanian pada sisi pengguna admin dan pemerintah. Aplikasi web ini dapat digunakan untuk memberikan informasi data komoditas pertanian dari masyarakat dan petani di daerah tertentu kepada pemerintah, serta dapat mempercepat penyampaian informasi harga komoditas pertanian dari pasar ke pemerintah.

### Saran

Aplikasi web portal harga dapat dikembangkan dengan menambahkan fitur *reward* yang merupakan fitur bagi petani dan masyarakat untuk mendapatkan hadiah, sehingga mereka tertarik untuk menginformasikan terkait info harga atau panen yang ada di daerahnya. Selain itu membuat sistem keamanan data di *server* menggunakan *token OAuth* agar lebih terjaga keamanannya.

## DAFTAR PUSTAKA

- Agustina L. 2008. Analisis tata niaga dan keterpaduan pasar kubis (Studi kasus Desa Cimenyan, Kecamatan Cimenyan, Kabupaten Bandung , Provinsi Jawa Barat) [skripsi]. Bogor (ID): Institut Pertanian Bogor.
- [APJII] Asosiasi Penyelenggara Jasa Internet Indonesia. 2014. Profil Pengguna Internet Indonesia [internet]. [diunduh 2015 Des 20]. Tersedia pada: <http://www.slideshare.net/internetsehat/profil-pengguna-internet-indonesia-2014-riset-oleh-apjii-dan-puskakom-ui>
- Astika IW, Solahudin M, Maradona RF, Nugraha MPA. 2009. *Sistem Informasi Harga Produk Pertanian Berbasis SMS* [internet]. Di dalam: Prosiding Seminar Nasional Himpunan Informatika Pertanian Indonesia; 2009 Agustus 6-7; Bogor, Indonesia. Bogor (ID): [penerbit tidak diketahui]. hlm. 1-7.
- Husen. 2014. Pemetaan dan analisis distribusi harga produk-produk pertanian menggunakan sistem informasi geografis [skripsi]. Bogor (ID): Institut Pertanian Bogor.
- [Kemendag] Kementrian Perdagangan. 2011. Peraturan Menteri Perdagangan No. 46 Tahun 2011 tentang Petunjuk Teknis Pelaksanaan Kegiatan Dekonsentrasi Bidang Perdagangan Tahun Anggaran 2012. Jakarta (ID): Kemendag
- Kowal BM, Schreier TR, Dauer JT, Helikar T. 2016. Programmatic access to logical models in the cell collective modeling environment via a REST API. *Biosystems*. 139:12-16. doi:10.1016/j.biosystems.2015.11.005.
- Kurniawan E. 2014. *Implementasi REST Web Service untuk Sales Order dan Sales Tracking Berbasis Mobile*. Jurnal EKSIS, Volume 07 No.01 Mei 2014 (1-12).



- [Sekneg] Sekretariat Negara Pemerintah Republik Indonesia. 2012. Undang - Undang Republik Indonesia Nomor 18 Tahun 2012 tentang Pangan. Jakarta (ID) : Sekretariat Negara
- [Sekneg] Sekretariat Negara Pemerintah Republik Indonesia. 2013. Undang - Undang Republik Indonesia Nomor 19 Tahun 2013 tentang Perlindungan dan Pemberdayaan Petani. Jakarta (ID) : Sekretariat Negara
- Pressman RS. 2010. *Software Engineering : A Practitioner's Approach*. Ed ke-7. Boston(US) : Mc Graw Hill.
- Richardson L, Ruby S. 2007. *RESTfull Web Service*. United State of America: O'Reilly Media, Inc.

## Lampiran 1 Data masukan oleh pengguna

## a Data laporan harga komoditas

Komoditas	Harga	Tanggal	Lokasi	<i>Latitude</i>	<i>Longitude</i>
Beras	12000	09-06-2016	Dramaga, Bogor, West Java	-6.5842592	106.7310479
Cabai merah keriting	32000	06-07-2016	East Bekasi, Bekasi City, West Java	-6.2362538	107.0244170
Daging merah	130000	10-07-2016	Pondok Labu, Cilandak, South Jakarta City	-6.3076465	106.797199
Kedelai	12000	16-07-2016	Lebak Bulus, Cilandak, South Jakarta City	-6.3031123	106.7794934
Cabai rawit hijau	28000	19-07-2016	Pela Mampang, Mampang Prpt. South Jakarta City	-6.2488888	106.817837

b Data *request* operasi pasar

Subjek	Tanggal	Komoditas	Pasar	Pesan
Harga kedelai naik	09-06-2016	Kedelai	Pasar Kebon Kembang	Harga kedelai yang semula 8500 naik menjadi 12000 hari ini. Mohon ditangani. Terima kasih.
Daging sapi makin mahal	08-07-2016	Daging sapi	Pasar Minggu	Pak, daging sapi di Pasar Minggu sekarang sudah mencapai 150000 per kilonya.
Bawang merah naik lagi	10-07-2016	Bawang merah	Pasar Jatinegara	Bawang merah di Pasar Jatinegara hari ini sudah mencapai 45000 dari yang semula 38000 per kilonya.

Lampiran 2 *Fully developed use case description*

a *Fully developed use case description* memeriksa harga komoditas dan lokasi

<i>Use case name:</i>	Memeriksa harga komoditas dan lokasi						
<i>Scenario:</i>	Pemerintah ingin melihat data harga komoditas dan lokasi						
<i>Triggering Event:</i>	Pemerintah memilih menu info harga						
<i>Brief Description:</i>	Melihat data-data harga komoditas dan lokasi yang sebelumnya telah masukkan oleh masyarakat						
<i>Actors:</i>	Pemerintah						
<i>Related Use Case:</i>	Login						
<i>Stakeholders:</i>	Pemerintah						
<i>Preconditions:</i>	Data harga komoditas dan lokasi berhasil di masukkan oleh masyarakat						
<i>Postconditions:</i>	Pemerintah berhasil melihat harga komoditas dan lokasi						
<i>Flow of Activities:</i>	<table border="1"> <thead> <tr> <th><i>Actor</i></th><th><i>System</i></th></tr> </thead> <tbody> <tr> <td>1 Memilih menu info harga</td><td></td></tr> <tr> <td>2 Memilih sub menu <i>map</i>, tabel</td><td>2.1 Menampilkan halaman yang berisi <i>map</i> dan tabel info harga komoditas</td></tr> </tbody> </table>	<i>Actor</i>	<i>System</i>	1 Memilih menu info harga		2 Memilih sub menu <i>map</i> , tabel	2.1 Menampilkan halaman yang berisi <i>map</i> dan tabel info harga komoditas
<i>Actor</i>	<i>System</i>						
1 Memilih menu info harga							
2 Memilih sub menu <i>map</i> , tabel	2.1 Menampilkan halaman yang berisi <i>map</i> dan tabel info harga komoditas						
<i>Exceptions Conditions :</i>	- Server telah diaktifkan						

*b Fully developed use case description* melihat opini *request* operasi pasar

<i>Use case Name:</i>	Melihat opini <i>request</i> operasi pasar						
<i>Scenario:</i>	Pemerintah ingin melihat opini <i>request</i> operasi pasar						
<i>Triggering Event:</i>	Pemerintah memilih menu aspirasi						
<i>Brief Description:</i>	Melihat opini <i>request</i> operasi pasar yang telah dimasukkan oleh masyarakat						
<i>Actors:</i>	Pemerintah						
<i>Related Use Case:</i>	Login						
<i>Stakeholders:</i>	Pemerintah						
<i>Preconditions:</i>	Data <i>request</i> operasi pasar berhasil di masukkan oleh masyarakat						
<i>Postconditions:</i>	Pemerintah berhasil melihat <i>request</i> operasi pasar						
<i>Flow of Activities:</i>	<table border="1"> <thead> <tr> <th><i>Actor</i></th><th><i>System</i></th></tr> </thead> <tbody> <tr> <td>1 Memilih menu aspirasi</td><td></td></tr> <tr> <td>2 Memilih tab menu <i>request</i> operasi pasar</td><td>2.1 Menampilkan informasi opini <i>request</i> operasi pasar</td></tr> </tbody> </table>	<i>Actor</i>	<i>System</i>	1 Memilih menu aspirasi		2 Memilih tab menu <i>request</i> operasi pasar	2.1 Menampilkan informasi opini <i>request</i> operasi pasar
<i>Actor</i>	<i>System</i>						
1 Memilih menu aspirasi							
2 Memilih tab menu <i>request</i> operasi pasar	2.1 Menampilkan informasi opini <i>request</i> operasi pasar						
<i>Exceptions Conditions :</i>	<ul style="list-style-type: none"> <li>- Server telah diaktifkan</li> <li>- Jika masyarakat belum ada yang melaporkan opini <i>request</i> operasi pasar maka tidak ada data yang ditampilkan.</li> </ul>						

*c Fully developed use case description* melihat aspirasi petani

<i>Use case Name:</i>	Melihat aspirasi petani						
<i>Scenario:</i>	Pemerintah ingin melihat aspirasi petani						
<i>Triggering Event:</i>	Pemerintah memilih menu aspirasi						
<i>Brief Description:</i>	Melihat aspirasi yang telah dimasukkan oleh petani						
<i>Actors:</i>	Pemerintah						
<i>Related Use Case:</i>	Login						
<i>Stakeholders:</i>	Pemerintah						
<i>Preconditions:</i>	Data aspirasi berhasil di dimasukkan oleh petani						
<i>Postconditions:</i>	Pemerintah berhasil melihat aspirasi petani						
<i>Flow of Activities:</i>	<table border="1"> <thead> <tr> <th><i>Actor</i></th><th><i>System</i></th></tr> </thead> <tbody> <tr> <td>1 Memilih menu aspirasi</td><td></td></tr> <tr> <td>2 Memilih tab menu aspirasi</td><td>2.1 Menampilkan informasi aspirasi petani</td></tr> </tbody> </table>	<i>Actor</i>	<i>System</i>	1 Memilih menu aspirasi		2 Memilih tab menu aspirasi	2.1 Menampilkan informasi aspirasi petani
<i>Actor</i>	<i>System</i>						
1 Memilih menu aspirasi							
2 Memilih tab menu aspirasi	2.1 Menampilkan informasi aspirasi petani						
<i>Exceptions Conditions :</i>	<ul style="list-style-type: none"> <li>- Server telah diaktifkan</li> <li>- Jika petani belum ada yang memberikan aspirasi maka tidak ada data yang ditampilkan.</li> </ul>						

*d Fully developed use case description* rekapitulasi harga komoditas dan lokasi

<i>Use case Name:</i>	Rekapitulasi harga komoditas dan lokasi								
<i>Scenario:</i>	Pemerintah ingin melakukan rekapitulasi harga komoditas dan lokasi								
<i>Triggering Event:</i>	Pemerintah memilih menu info harga atau info panen								
<i>Brief Description:</i>	Melakukan rekapitulasi harga komoditas pertanian dan lokasi secara keseluruhan								
<i>Actors:</i>	Pemerintah								
<i>Related Use Case:</i>	Login								
<i>Stakeholders:</i>	Pemerintah								
<i>Preconditions:</i>	Data harga komoditas pertanian dan lokasi berhasil di dimasukkan oleh masyarakat dan petani								
<i>Postconditions:</i>	Pemerintah berhasil melihat hasil rekapitulasi info harga komoditas dan lokasi								
<i>Flow of Activities:</i>	<table border="1"> <thead> <tr> <th><i>Actor</i></th><th><i>System</i></th></tr> </thead> <tbody> <tr> <td>1 Memilih menu info harga atau info panen</td><td></td></tr> <tr> <td>2 Memilih sub menu tabel</td><td>2.1 Menampilkan informasi harga info harga atau info panen dalam bentuk tabel</td></tr> <tr> <td>3 Memilih format rekapitulasi yang digunakan</td><td>3.1 Menampilkan hasil rekapitulasi</td></tr> </tbody> </table>	<i>Actor</i>	<i>System</i>	1 Memilih menu info harga atau info panen		2 Memilih sub menu tabel	2.1 Menampilkan informasi harga info harga atau info panen dalam bentuk tabel	3 Memilih format rekapitulasi yang digunakan	3.1 Menampilkan hasil rekapitulasi
<i>Actor</i>	<i>System</i>								
1 Memilih menu info harga atau info panen									
2 Memilih sub menu tabel	2.1 Menampilkan informasi harga info harga atau info panen dalam bentuk tabel								
3 Memilih format rekapitulasi yang digunakan	3.1 Menampilkan hasil rekapitulasi								
<i>Exceptions Conditions :</i>	- Server telah diaktifkan								

*e Fully developed use case description CRUD manajemen user*

<i>Use case Name:</i>	CRUD manajemen <i>user</i>
<i>Scenario:</i>	Menambah, mengubah, dan menghapus <i>user</i>
<i>Triggering Event:</i>	Menambah <i>user</i> : Terdapat <i>user</i> baru Mengubah <i>user</i> : Terdapat kesalahan pada saat <i>input user</i> Menghapus <i>user</i> : <i>User</i> perlu dihapus
<i>Brief Description:</i>	Manajemen <i>user</i> akan dilakukan oleh admin. <i>User</i> yang ditambahkan apabila terdapat <i>user</i> baru. <i>User</i> yang sudah terdaftar dapat diubah dan dihapus apabila terdapat kesalahan dan perlu untuk dirubah atau dihapus.
<i>Actors:</i>	Admin
<i>Related Use Case:</i>	Login
<i>Stakeholders:</i>	Admin
<i>Preconditions:</i>	Menambah <i>user</i> : <i>User</i> belum tercatat Mengubah <i>user</i> : Terdapat kesalahan pada saat <i>input user</i> Menghapus <i>user</i> : <i>User</i> perlu dihapus
<i>Postconditions:</i>	Menambah <i>user</i> : <i>User</i> berhasil ditambahkan Mengubah <i>user</i> : <i>User</i> berhasil diubah Menghapus <i>user</i> : <i>User</i> berhasil dihapus
<i>Flow of Activities:</i>	Lihat lampiran 4d
<i>Exceptions Conditions :</i>	<ul style="list-style-type: none"> <li>- Jika admin klik <i>cancel</i> pada saat menambah <i>user</i>, maka data <i>user</i> tidak disimpan ke basis data dan kembali ke halaman manajemen <i>user</i></li> <li>- Jika admin klik <i>cancel</i> pada saat mengubah <i>user</i>, maka data <i>user</i> tidak dirubah ke basis data dan kembali ke halaman manajemen <i>user</i></li> <li>- Jika admin klik <i>cancel</i> pada saat menghapus <i>user</i>, maka data <i>user</i> tidak disimpan ke basis data dan kembali ke halaman manajemen <i>user</i></li> </ul>

*f Fully developed use case description* CRUD manajemen pasar

<i>Use case Name:</i>	CRUD manajemen pasar
<i>Scenario:</i>	Menambah, mengubah, dan menghapus pasar
<i>Triggering Event:</i>	Menambah pasar : Terdapat pasar baru Mengubah pasar : Terdapat kesalahan pada saat <i>input</i> pasar Menghapus pasar : Pasar perlu dihapus
<i>Brief Description:</i>	Manajemen pasar akan dilakukan oleh admin. Pasar yang ditambahkan apabila terdapat pasar baru. Pasar yang sudah terdaftar dapat diubah dan dihapus apabila terdapat kesalahan dan perlu untuk dirubah atau dihapus.
<i>Actors:</i>	Admin
<i>Related Use Case:</i>	Login
<i>Stakeholders:</i>	Admin
<i>Preconditions:</i>	Menambah pasar : Pasar belum tercatat Mengubah pasar : Terdapat kesalahan pada saat <i>input</i> pasar Menghapus pasar : Pasar perlu dihapus
<i>Postconditions:</i>	Menambah pasar : Pasar berhasil ditambahkan Mengubah pasar : Pasar berhasil diubah Menghapus pasar : Pasar berhasil dihapus
<i>Flow of Activities:</i>	Lihat lampiran 4e
<i>Exceptions Conditions :</i>	<ul style="list-style-type: none"> <li>- Jika admin klik <i>cancel</i> pada saat menambah pasar, maka data pasar tidak disimpan ke basis data dan kembali ke halaman manajemen pasar</li> <li>- Jika admin klik <i>cancel</i> pada saat mengubah pasar, maka data pasar tidak dirubah ke basis data dan kembali ke halaman manajemen pasar</li> <li>- Jika admin klik <i>cancel</i> pada saat menghapus pasar, maka data pasar tidak disimpan ke basis data dan kembali ke halaman manajemen pasar</li> </ul>



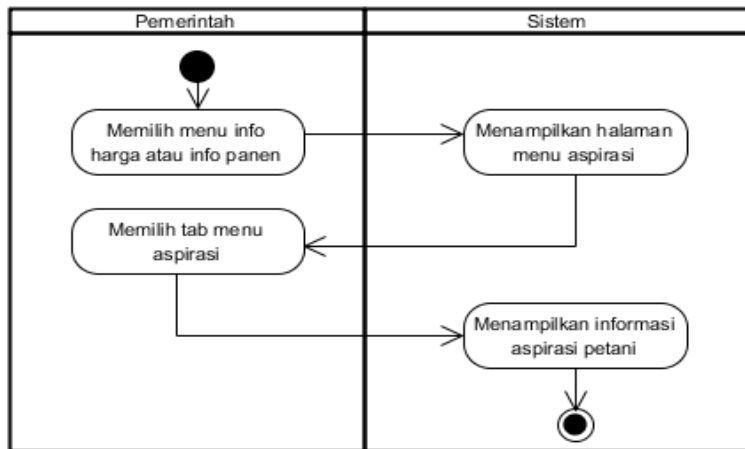
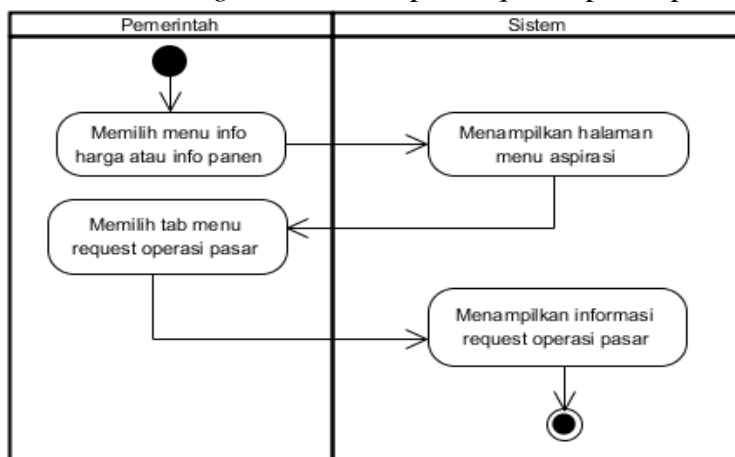
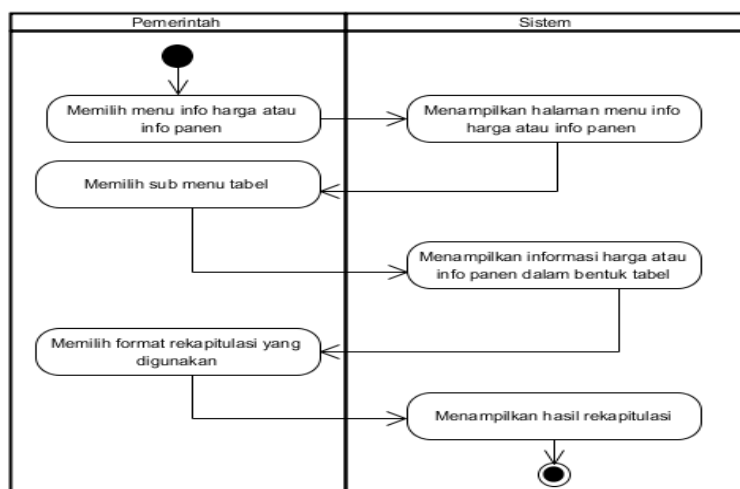
*g Fully developed use case description CRUD manajemen komoditas*

<i>Use case Name:</i>	CRUD manajemen komoditas
<i>Scenario:</i>	Menambah, mengubah, dan menghapus komoditas
<i>Triggering Event:</i>	Menambah komoditas : Terdapat komoditas baru Mengubah komoditas : Terdapat kesalahan pada saat <i>input</i> komoditas Menghapus komoditas : Komoditas perlu dihapus
<i>Brief Description:</i>	Manajemen komoditas akan dilakukan oleh admin. Komoditas yang ditambahkan apabila terdapat komoditas baru. Komoditas yang sudah terdaftar dapat diubah dan dihapus apabila terdapat kesalahan dan perlu untuk dirubah atau dihapus.
<i>Actors:</i>	Admin
<i>Related Use Case:</i>	Login
<i>Stakeholders:</i>	Admin
<i>Preconditions:</i>	Menambah komoditas : Komoditas belum tercatat Mengubah komoditas : Terdapat kesalahan pada saat <i>input</i> komoditas Menghapus komoditas: Komoditas perlu dihapus
<i>Postconditions:</i>	Menambah komoditas : Komoditas berhasil ditambahkan Mengubah komoditas : Komoditas berhasil diubah Menghapus komoditas : Komoditas berhasil dihapus
<i>Flow of Activities:</i>	Lihat lampiran 4f
<i>Exceptions Conditions :</i>	<ul style="list-style-type: none"> <li>- Jika admin klik <i>cancel</i> pada saat menambah komoditas, maka data komoditas tidak disimpan ke basis data dan kembali ke halaman manajemen komoditas</li> <li>- Jika admin klik <i>cancel</i> pada saat mengubah komoditas, maka data komoditas tidak dirubah ke basis data dan kembali ke halaman manajemen komoditas</li> <li>- Jika admin klik <i>cancel</i> pada saat menghapus komoditas maka data komoditas tidak disimpan ke basis data dan kembali ke halaman manajemen komoditas</li> </ul>

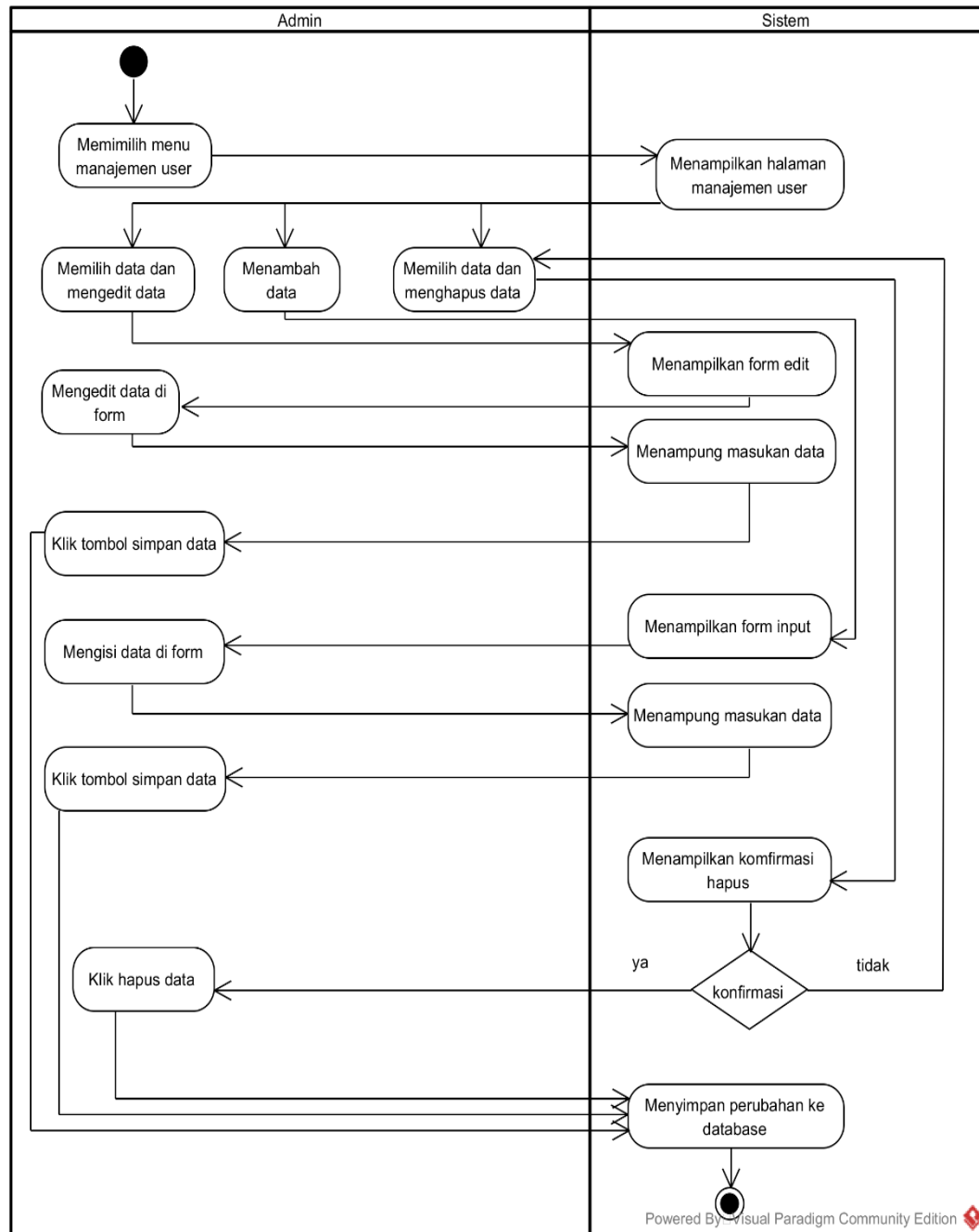
## Lampiran 3 Kamus data

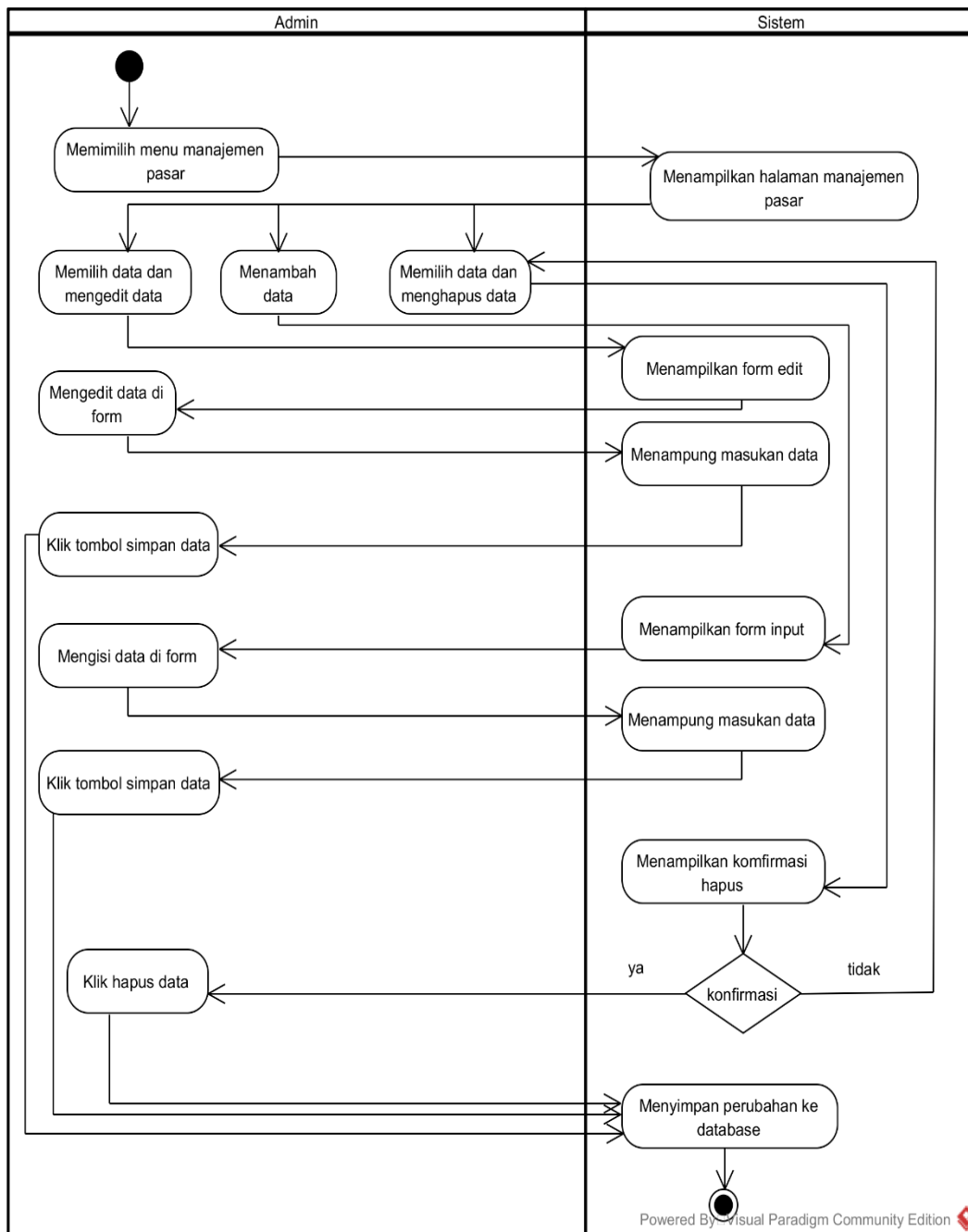
No	Nama tabel	Nama atribut	Tipe data	Keterangan
1	<i>user</i>	<i>id_user</i> <i>username</i> <i>password</i> <i>nama</i> <i>email</i> <i>alamat</i> <i>hak_akses</i>	Integer(11) Varchar(64) Varchar(64) Varchar(128) Varchar(64) Text Integer(11)	Tabel untuk menyimpan data <i>user</i> Contoh data: 1, wildan, wildan12, Wildan Ramdani, wildan_r@gmail.com, 4
2	<i>harga</i>	<i>id_harga</i> <i>harga</i> <i>tanggal</i> <i>latitude</i> <i>longitude</i> <i>lokasi</i> <i>id_komoditas</i> <i>id_user</i>	Integer(11) Integer(11) Varchar(20) Varchar(32) Varchar(32) Varchar(128) Integer(11) Integer(11)	Tabel untuk menyimpan data harga komoditas Contoh data :1, 5000, 2016-05-27 19:30:51, - 6.5860408, 106.7201375, Dramaga, 1, 1
3	<i>pasar</i>	<i>id_pasar</i> <i>nama_pasar</i>	Integer(11) Varchar(128)	Tabel untuk menyimpan data pasar Contoh data : 1, Pasar Induk Kramat Jati
4	<i>komoditas</i>	<i>id_komoditas</i> <i>jenis</i> <i>nama_komoditas</i>	Integer(11) Varchar(64) Varchar(256)	Tabel untuk menyimpan data komoditas Contoh data : 1, Beras, Beras
5	<i>info_petani</i>	<i>id_info_petani</i> <i>id_komoditas</i> <i>id_user</i> <i>time</i> <i>latitude</i> <i>longitude</i> <i>lokasi</i> <i>jumlah_komoditas</i> <i>keterangan</i>	Integer(11) Integer(11) Integer(11) Timestamp Varchar(256) Varchar(256) Varchar(256) Integer(11) Text	Tabel untuk menyimpan data harga info harga yang dilaporkan petani Contoh data : 1, 6, 1, 2016-05-11 17:23:16, - 6.5662957, 106.56859,
6	<i>aspirasi_petani</i>	<i>id_aspirasi</i> <i>id_user</i> <i>subject</i> <i>aspirasi_pesan</i> <i>time</i> <i>flag</i>	Integer(11) Integer(11) Text Text Timestamp Integer(11)	Lewuisadeng, 200, Komoditas Langka Tabel untuk menyimpan aspirasi dari petani

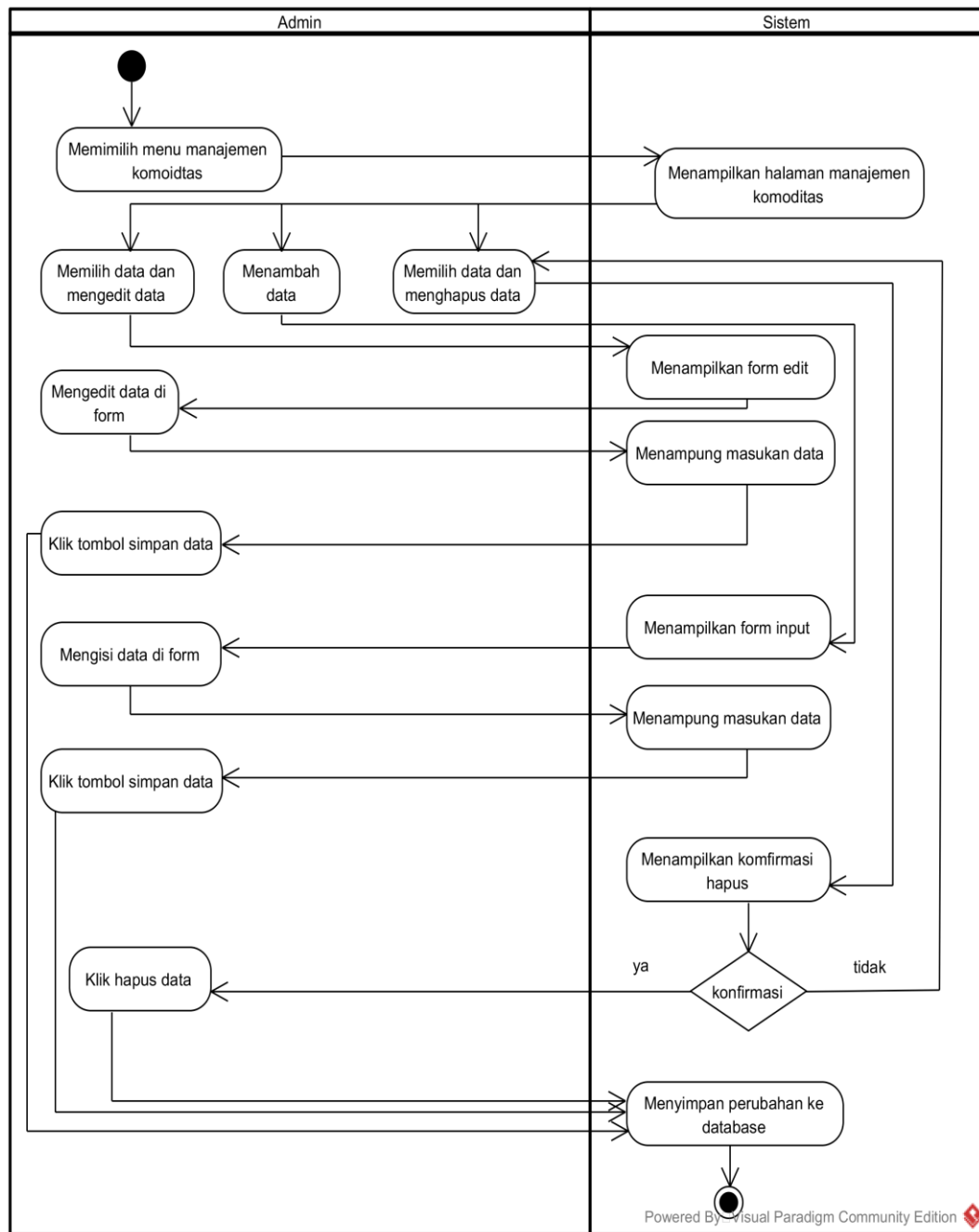
No	Nama tabel	Nama atribut	Tipe data	Keterangan
7	<i>request</i>	<i>id_request</i>	Integer(11)	Contoh data : 1,
		<i>subjek</i>	Text	4, Kritik dan
		<i>tanggal</i>	Varchar(128)	Saran, Pak hasil
		<i>pesan</i>	Text	panen cabe merah
		<i>id_user</i>	Integer(11)	di daerah sini
		<i>id_komoditas</i>	Integer(11)	sangat melimpah,
		<i>id_pasar</i>	Integer(11)	2016-05-30
		<i>flag</i>	Integer(11)	16:01:33, 1
				Tabel untuk
				menyimpan
				laporan operasi
				pasar dari
				masyarakat
				Contoh data : 2,
				Cabai mahal,
				2016-06-08, Pak
				bawang merah di
				pasar sunter
				podomoro sedang
				naik tolong segera
				kesini pak, 10, 8,
				3, 1

Lampiran 4 *Swimlane diagram*a *Swimlane diagram* melihat aspirasi petanib *Swimlane diagram* melihat opini request operasi pasarc *Swimlane diagram* rekapitulasi harga komoditas dan lokasi

d *Swimlane diagram* CRUD manajemen user

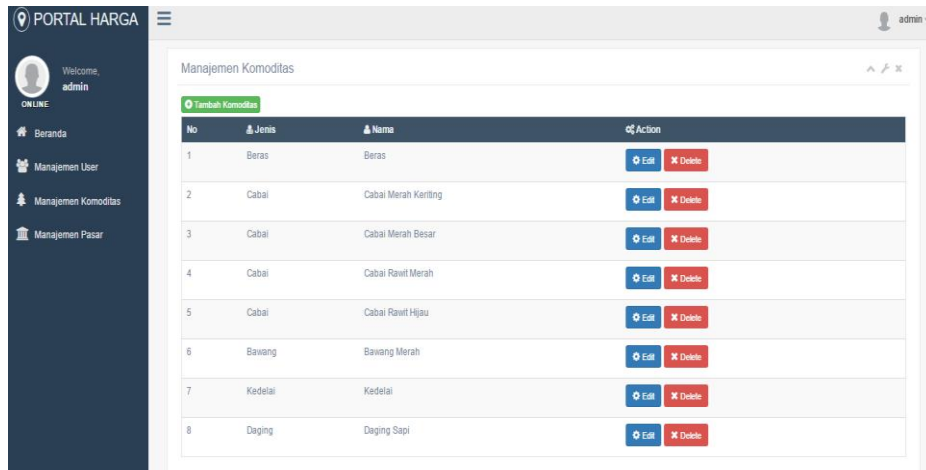


e *Swimlane diagram* CRUD manajemen pasar

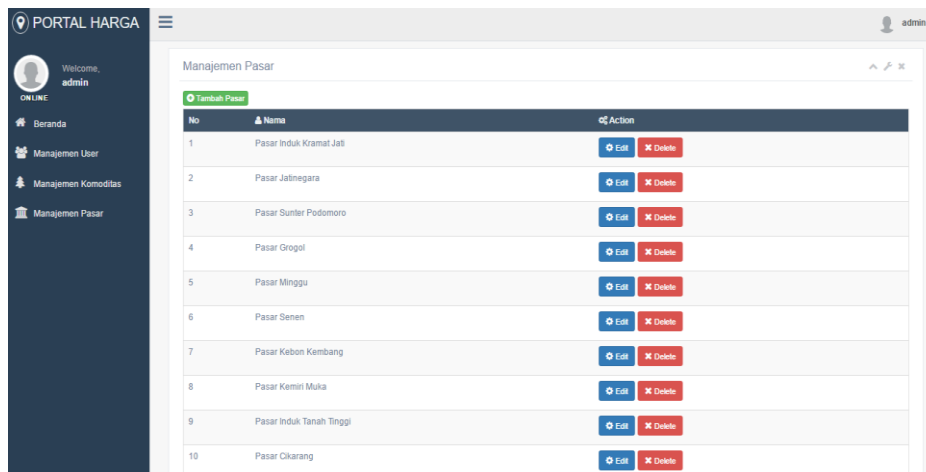
f *Swimlane diagram manajemen komoditas*

## Lampiran 5 Implementasi antarmuka sistem

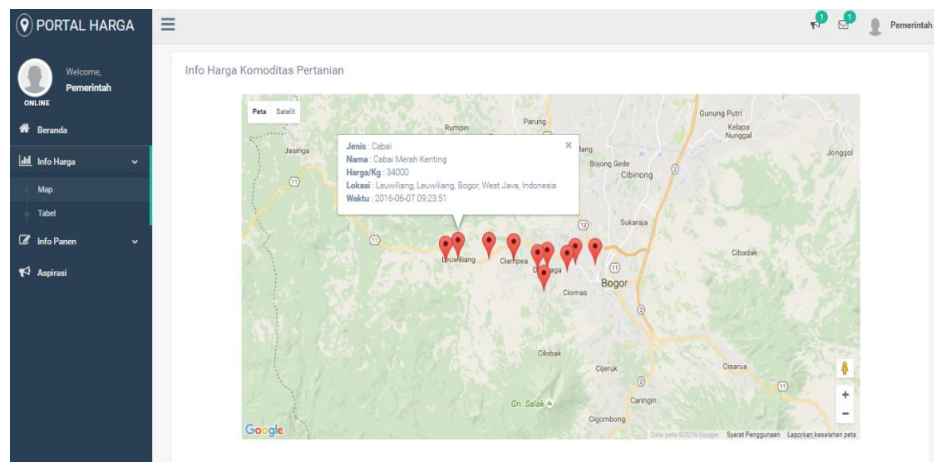
### a Implementasi halaman CRUD manajemen komoditas



### b Implementasi halaman CRUD manajemen pasar



### c Implementasi halaman memeriksa harga komoditas dan lokasi





#### d Implementasi halaman melihat aspirasi

**PORTAL HARGA**

Welcome, Pemerintah

ONLINE

Beranda

Info Harga

Info Panen

Aspirasi

Aspirasi Request Operasi Pasar

**yeni**

Pak hasil panen cabe merah di daerah sini sangat melimpah, sedangkan harga di pasaran sangat murah, petani disini kesulitan untuk menjualnya pak. Terima kasih

Dramaga | 2016-05-30T09:01:33.000Z

**Asep**

Pak, kalau bisa 1 minggu sekali berikan pelatihan terkait budidaya buah naga kepada petani di daerah Bogor, parung. Terima Kasih

Bogor | 2016-05-30T09:01:37.000Z

**Nunung**

Soal mempertahankan lahan pertanian di daerah Cianjur Pak, saat ini sudah digodok untuk menjadi perda. Sementara persoalan pupuk sudah kita programkan, bagaimana pendistribusiannya agar tidak terjadi kelangkaan

Cianjur | 2016-05-30T09:01:33.000Z

#### e Implementasi halaman melihat opini *request* operasi pasar

**PORTAL HARGA**

Welcome, Pemerintah

ONLINE

Beranda

Info Harga

Info Panen

Aspirasi

Aspirasi Request Operasi Pasar

**Asep**

Harga kedelai naik

Kedelai | Pasar Kebon Kembang

Pak disini harga kedelai udah naik jadi 12500. Kami jadi susah jualan tempe. Tolong ditangani ya Pak.

Bogor | 2016-06-10

**Prily**

Cabai mahal

Cabai Rawit Merah | Pasar Grogol

Cabai rawit yang semula 30000 sekarang menjadi 40000 disini. Tolong tambahkan stok cabai lagi ya pak. Terima kasih.

Lampung | 2016-06-08

**Warsito**

Daging sapi makin mahal

Daging Sapi | Pasar Jatinegara

Pak sekarang daging sapi sudah 150000 per kilonya, padahal belum lebaran. Bagaimana ya pak? Tolong ditangani.

Brebes | 2016-06-10

#### f Implementasi halaman rekapitulasi harga komoditas dan lokasi

**PORTAL HARGA**

Welcome, Pemerintah

ONLINE

Beranda

Info Harga

Info Panen

Aspirasi

Info Harga Komoditas Pertanian

CSV Excel PDF Print

Search:

No	Tanggal	Komoditas	Harga (Kg)	Lokasi
1	2016-05-27 19:30:51	Beras	5000	Dramaga
2	2016-06-09 19:30:20	Beras	15000	Jalan Bungur Besar No. 169, Bungur, Senen, Bungur, Jakarta Pusat, Kota Jakarta Pusat, Daerah Khusus Ibukota Jakarta, Indonesia
3	2016-05-27 20:00:05	Cabai Merah Keriting	30000	Laladon
4	2016-06-09 17:30:51	Cabai Merah Keriting	35000	Jl. Matraman Raya, RT5/RW 6, Ball Mester, Jatinegara, Kota Jakarta Timur, Daerah Khusus Ibukota Jakarta, Indonesia
5	2016-06-10 11:30:51	Cabai Merah Keriting	24000	Pulo Gadung, East Jakarta City, Special Capital Region of Jakarta, Indonesia
6	2016-06-09 20:30:51	Cabai Merah Keriting	35000	Dramaga, Dramaga, Bogor, West Java, Indonesia
7	2016-06-07 09:23:51	Cabai Merah Keriting	34000	Leuwiliang, Leuwiliang, Bogor, West Java, Indonesia
8	2016-05-27 19:32:44	Cabai Merah Besar	50000	Leuwiliang
9	2016-06-10 10:30:51	Cabai Merah Besar	30000	Pondok Labu, Cilandak, South Jakarta City, Special Capital Region of Jakarta, Indonesia
10	2016-06-09 12:30:51	Cabai Merah Besar	35000	Ciampea, Ciampea, Bogor, West Java, Indonesia

Showing 1 to 10 of 19 entries

Previous 1 2 Next

## **RIWAYAT HIDUP**

Penulis dilahirkan di Sumedang Provinsi Jawa Barat pada tanggal 02 Desember 1994. Penulis adalah anak ketiga dari tiga bersaudara, anak dari pasangan Asep Sutarman dan Nunung Sukawati. Penulis menempuh pendidikan Sekolah Menengah Atas di SMA Negeri 3 Sumedang pada tahun 2009 hingga 2012. Kemudian penulis melanjutkan pendidikan perguruan tinggi di Institut Pertanian Bogor, Fakultas Matematika dan Ilmu Pengetahuan Alam, Departemen Ilmu Komputer dari tahun 2012 hingga 2016.

Selama mengikuti perkuliahan, penulis pernah aktif menjadi pengurus Himpunan Mahasiswa Ilmu Komputer selama satu tahun masa jabatan. Penulis menjabat sebagai staff divisi HRD periode 2013/2014. Penulis melaksanakan Praktik Kerja Lapangan (PKL) di PT. Bank Rakyat Indonesia (BRI), Divisi Teknologi dan Sistem Informasi (TSI), Jakarta.