

## **BAB 2**

### **LANDASAN TEORI**

#### **2.1 Badan Pusat Statistik**

Badan Pusat Statistik adalah Lembaga Pemerintah Non Kementerian yang bertanggungjawab langsung kepada Presiden. Sebelumnya, BPS merupakan Biro Pusat Statistik, yang dibentuk berdasarkan UU Nomor 6 Tahun 1960 tentang Sensus dan UU Nomor 7 Tahun 1960 tentang Statistik. Sebagai pengganti kedua UU tersebut ditetapkan UU Nomor 16 Tahun 1997 tentang Statistik. Berdasarkan UU ini yang ditindaklanjuti dengan peraturan perundangan di bawahnya, secara formal nama Biro Pusat Statistik diganti menjadi Badan Pusat Statistik [8].

Berdasarkan Undang-Undang yang telah disebutkan di atas, peranan yang harus dijalankan oleh BPS adalah sebagai berikut:

1. Menyediakan kebutuhan data bagi pemerintah dan masyarakat. Data ini didapatkan dari sensus atau survei yang dilakukan sendiri dan juga dari departemen atau lembaga pemerintahan lainnya sebagai data sekunder.
2. Membantu kegiatan statistik di departemen, lembaga pemerintah atau institusi lainnya, dalam membangun sistem perstatistikan Nasional.
3. Mengembangkan dan mempromosikan standar teknik dan metodologi statistik, dan menyediakan pelayanan pada bidang pendidikan dan pelatihan statistik.
4. Membangun kerjasama dengan institusi internasional dan negara lain untuk kepentingan perkembangan statistik Indonesia.

BPS mempunyai kantor di setiap Kabupaten/Kota yang dikepalai oleh seorang kepala BPS Kabupaten/Kota dan dibantu oleh 1(satu) orang Kepala Sub Bagian Tata Usaha, 5 orang kepala seksi yaitu Kepala Seksi Statistik Produksi, Kepala Seksi Statistik Sosial, Kepala Seksi Statistik Neraca dan Analisis Statistik, Kepala Seksi Distribusi serta Kepala Statistik Integrasi Pengolahan dan

Diseminasi Statistik beserta beberapa orang staf dan beberapa orang Koordinator Statistik Kecamatan (KSK).

BPS di setiap Kabupaten/Kota memiliki fungsi yang sama, setiap BPS Kabupaten/Kota dibawah oleh BPS Provinsi dan setiap BPS Provinsi dibawah oleh BPS Pusat. Tidak semua BPS melakukan survei harga produsen pedesaan salah satu yang melaksanakan kegiatan survei ini adalah BPS Kabupaten Subang. BPS Kabupaten Subang yang beralamat di Jalan Aipda KS Tubun No 12 Kelurahan Karanganyar Kecamatan Subang Kabupaten Subang.

### **2.1.1 Sejarah Badan Pusat Statistik**

#### **Masa Pemerintahan Hindia Belanda**

Pada bulan Februari tahun 1920, Kantor Statistik pertama kali didirikan oleh Direktur Pertanian, Kerajinan dan Perdagangan (*Directure Vand Landbow Nijeverheinden Handed*) dan bertempat di Bogor. Kantor ini disertai tugas untuk mengelola dan mempublikasikan data statistik.

Pada bulan Maret 1923, dibentuk suatu komisi untuk statistik yang anggotanya merupakan wakil dari tiap-tiap departemen. Komisi tersebut diberi tugas untuk merencanakan tindakan-tindakan yang mengarah sejauh mungkin untuk mencapai kesatuan dalam kegiatan di bidang statistik di Indonesia.

Pada tanggal 24 September 1924, nama lembaga tersebut diganti dengan nama *Central Kantor Voor de Statistik* (CKS) atau kantor statistik dan dipindahkan ke Jakarta bersama dengan ini beralih pula pekerjaan mekanisme Statistik Perdagangan yang semula dilakukan oleh kantor *Invelor Vitvoer en Accijnsen* (IVA) yang sekarang disebut kantor Bea dan Cukai.

#### **Masa Pemerintahan Jepang**

Pada bulan Juni 1944, pemerintah Jepang baru mengaktifkan kembali kegiatan Statistik yang utamanya diarahkan untuk memenuhi kebutuhan perang atau militer. Pada masa *Central Kantor Voor de Statistik* (CKS) diganti namanya menjadi *Shomubu Chosasitsu Gunseikanbu*.

### **Masa Kemerdekaan Republik Indonesia**

Setelah Proklamasi Kemerdekaan Indonesia RI tanggal 17 Agustus 1945, kegiatan statistik ditangani oleh lembaga atau instansi baru sesuai dengan suasana kemerdekaan yaitu KAPPURI (Kantor Penyelidik Perangkaan Umum Republik Indonesia) dipindahkan ke Yogyakarta sebagai sekuen dari perjanjian Linggarjati. Sementara itu, pemerintah Belanda (NICA) di Jakarta mengaktifkan kembali *Central Kantor Voor de Statistiek* (CKS). Berdasarkan surat edaran kementerian kemakmuran tanggal 12 Juni 1950 Nomor 219.C, KAPPURI (Kantor Penyelidik Perangkaan Umum Republik Indonesia) dan *Central Voor de Statistiek* (CKS) dilebur menjadi Kantor Pusat Statistik (KPS) dan berada dibawah dan bertanggung jawab menteri Kemakmuran.

Dengan surat Menteri Perekonomian tanggal 1 Maret 1952 Nomor p/44, Lembaga Kantor Pusat Statistik (KPS) berada dibawah dan bertanggung jawab menteri Perekonomian. Selanjutnya keputusan Menteri Perekonomian Tanggal 24 Desember 1953 Nomor 18.099/M, KPS dibagi menjadi dua bagian yaitu bagian *Research* yang disebut Afdeling A dan bagian penyelenggaraan tata usaha yang disebut Afdeling B dengan keputusan Presiden RI Nomor 131 1957, kemerdekaan perekonomian dipecah menjadi kementerian Perdagangan dan Kementerian Perindustrian. Untuk selanjutnya keputusan Presiden RI Nomor 172, terhitung tanggal 1 Juni 1957 Kantor Pusat Statistik (KPS) diubah menjadi Biro Pusat Statistik yang semula menjadi tanggung jawab dan wewenang berada di bawah perdana menteri.

### **Masa Orde Baru Sampai Sekarang**

Perencanaan dan evaluasi pembangunan maka untuk mendapatkan statistik yang handal, lengkap, tepat, akurat, dan terpercaya mulai diadakan pembenahan organisasi Biro Pusat Statistik. Dalam masa orde baru ini Badan Pusat Statistik telah mengalami 4(empat) kali perubahan struktur organisasi.

Tahun 1968 ditetapkan peraturan Pemerintah Nomor 16 Tahun 1968 yaitu tentang mengatur organisasi dan tata kerja di pusat dan di daerah. Tahun 1980 peraturan Pemerintah nomor 6 tahun 1980 tentang organisasi sebagai pengganti

peraturan Pemerintah Nomor 16 tahun 1968. Berdasarkan peraturan pemerintah Nomor 6 tahun 1980 di tiap propinsi terdapat perwakilan BPS.

Pada tanggal 17 Juni 1998 dengan keputusan Presiden Republik Indonesia Nomor 86 tahun 1998 ditetapkan Badan Pusat Statistik sekaligus mengatur tata kerja dan struktur organisasi BPS yang baru. Saat ini kantor BPS sudah terdapat di setiap Kabupaten/Kota [8].

### 2.1.2 Logo Badan Pusat Statistik

Gambar 2.1 berikut ini merupakan logo yang digunakan oleh Badan Pusat Statistik di seluruh Indonesia.



*Sumber gambar : [www.bps.go.id](http://www.bps.go.id) [8]*

#### **Gambar 2. 1 Logo Badan Pusat Statistik**

Logo Badan Pusat Statistik terdiri dari 3(tiga) warna yaitu biru, hijau, dan oranye yang mewakili 3 sensus besar yang dilakukan oleh BPS setiap 10 tahun sekali. Warna biru melambangkan Sensus Penduduk yang dilaksanakan 10 tahun sekali dengan tahun yang berakhiran angka 0. Warna hijau melambangkan Sensus Pertanian yang diadakan 10 tahun sekali dengan akhiran tahun angka 3. Warna oranye melambangkan Sensus Ekonomi yang diadakan 10 tahun sekali dengan akhiran angka 6.

### 2.1.3 Visi, Misi dan *Core Values* Badan Pusat Statistik

Setiap organisasi tentu punya visi dan misi yang jelas sehingga organisasi yang akan dibentuk mempunyai tujuan yang jelas. Fungsi dari visi dan misi

organisasi adalah untuk membuat jalannya organisasi tetap fokus dan terkendali. Pernyataan visi dan misi yang dibuat akan menjelaskan peran dari organisasi yang dibentuk. Berikut adalah Visi, Misi dan *Core Values* dari BPS:

1. Visi : Pelopor data statistik terpercaya untuk semua
2. Misi :
  - a. Menyediakan data statistik berkualitas melalui kegiatan statistik yang terintegrasi dan berstandar nasional maupun internasional.
  - b. Memperkuat sistem statistik nasional yang berkesinambungan melalui pembinaan dan koordinasi di bidang statistik.
  - c. Membangun insan statistik yang profesional, berintegritas, dan amanah untuk kemajuan perstatistikan.

3. *Core Values* (Nilai-nilai Inti)

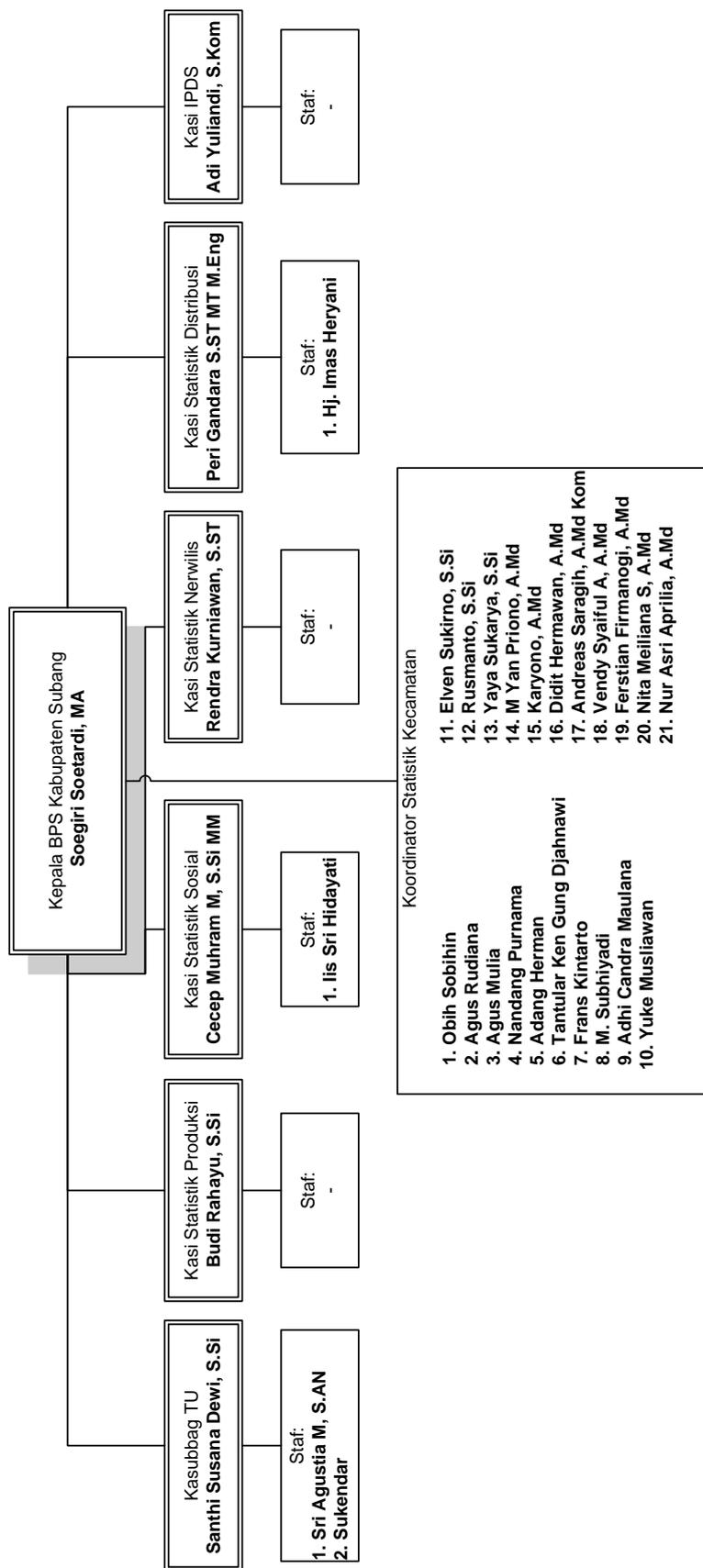
*Core Values* (nilai-nilai inti) BPS merupakan pondasi yang kokoh untuk membangun jati diri dan penuntunan perilaku setiap insan BPS dalam melaksanakan tugas. Nilai-nilai BPS terdiri dari:

- a. Profesional
  1. Kompeten yaitu mempunyai keahlian dalam bidang tugas yang diemban.
  2. Efektif yaitu memberikan hasil maksimal.
  3. Efisien yaitu mengerjakan setiap tugas secara produktif, dengan sumber daya minimal.
  4. Inovatif yaitu selalu melakukan pembaruan dan/atau penyempurnaan melalui proses pembelajaran diri secara terus menerus.
  5. Sistemik yaitu meyakini bahwa setiap pekerjaan mempunyai tata urutan proses pekerjaan yang satu menjadi bagian tidak terpisahkan dari pekerjaan yang lain.
- b. Integritas
  1. Dedikasi yaitu memiliki pengabdian yang tinggi terhadap profesi yang diemban dan institusi.

2. Disiplin yaitu melaksanakan pekerjaan sesuai dengan ketentuan yang telah ditetapkan.
  3. Konsistensi yaitu satunya kata dengan perbuatan.
  4. Terbuka yaitu menghargai ide, saran, pendapat, masukan, dan kritik dari berbagai pihak.
  5. Akuntabel yaitu bertanggung jawab dan setiap langkahnya terukur.
- c. Amanah
1. Terpercaya yaitu melaksanakan pekerjaan sesuai dengan ketentuan, yaitu tidak hanya didasarkan pada logika tetapi juga sekaligus menyentuh dimensi mental spiritual.
  2. Jujur yaitu melaksanakan semua pekerjaan dengan tidak menyimpang dari prinsip moralitas.
  3. Tulus yaitu melaksanakan tugas tanpa pamrih, menghindari konflik kepentingan (pribadi, kelompok, dan golongan), serta mendedikasikan semua tugas untuk perlindungan kehidupan manusia, sebagai amal ibadah atau perbuatan untuk Tuhan Yang Maha Esa.
  4. Adil yaitu menempatkan sesuatu secara berkeadilan dan memberikan haknya [8].

#### **2.1.4 Struktur Organisasi Badan Pusat Statistik Kabupaten Subang**

Pada Gambar 2.2 merupakan Struktur Organisasi di Badan Pusat Kabupaten Subang:



**Gambar 2. 2 Struktur Organisasi BPS Kabupaten Subang**

Setiap BPS Kabupaten/Kota dikepalai oleh seorang Kepala BPS Kabupaten/Kota, dimana membawahi 5 orang Kepala Seksi (Kasi), 1 orang Kepala Subbagian Tata Usaha, dan 21 Koordinator Statistik Kecamatan (KSK) yang bertanggung jawab langsung kepada kepala. Kasi dan Kasubbag membawahi beberapa staf, untuk di Kabupaten Subang sendiri ada beberapa Kasi yang tidak memiliki staf dikarenakan beberapa staf berpindah satuan kerja menjadi KSK. Untuk KSK sendiri setiap orang nya ditugaskan untuk memegang 1 Kecamatan sebagai wilayah tugasnya, tetapi untuk Kabupaten Subang sendiri karena keterbatasan pegawai sehingga 9 KSK harus merangkap memegang kecamatan.

Salah satu survei yang diadakan oleh BPS adalah survei harga pedesaan hortikultura yang masing-masing terdapat sampel di Kecamatan terpilih sesuai dengan sampel dari BPS Pusat. Survei harga ini sebagai salah satu faktor penunjang untuk kesejahteraan karena data yang dihasilkan kontinyu.

## 2.2 Survei

Menurut Masri Singarimbun dalam bukunya yang berjudul Metode Penelitian Survei, pengertian survei pada umumnya dibatasi pada penelitian yang datanya dikumpulkan dari sampel atau populasi untuk mewakili seluruh populasi. Dengan demikian, penelitian survei adalah penelitian yang mengambil sampel dari satu populasi dan menggunakan kuesioner sebagai alat pengumpulan data yang pokok.

Sedangkan menurut Mohammad Musa dalam bukunya yang berjudul Metodologi Penelitian, survei memiliki arti pengamatan/penyelidikan yang kritis untuk mendapatkan keterangan yang jelas dan baik terhadap suatu persoalan di dalam suatu daerah tertentu. Tujuan dari survei adalah untuk mendapatkan gambaran yang mewakili suatu daerah dengan benar. Suatu survei tidak akan meneliti semua individu dalam sebuah populasi, namun hasil yang diharapkan harus dapat menggambarkan sifat dari populasi yang bersangkutan. Karena itu, metode pengambilan contoh (*sampling method*) di dalam suatu survei memegang peranan yang sangat penting. Metode pengambilan contoh (*sampling method*) yang tidak benar akan merusak hasil survei.

Survei merupakan suatu metode untuk menentukan hubungan-hubungan antar variabel serta membuat generalisasi untuk suatu populasi yang dipelajari. Survei mampu mengerjakan hal tersebut karena prosedur pengumpulan data yang dipergunakan telah dibuat seragam dan telah distandarisasikan. Individu-individu yang dipilih dalam contoh (*sample*) dihadapkan pada sejumlah pertanyaan yang telah ditetapkan. Jawaban dari pertanyaan diklasifikasikan secara sistematis, sehingga dapat dibuat perbandingan-perbandingan kuantitatif.

Teknik-teknik yang telah distandarisasikan tersebut menimbulkan kelemahan-kelemahan. Metode tersebut menghadapkan individu-individu yang diteliti pada pertanyaan-pertanyaan yang dinormalisasikan dan jawaban-jawaban yang diperoleh diklasifikasikan ke dalam beberapa tipe yang sederhana, tanpa memandang perbedaan kualitas dari jawaban-jawaban tersebut. Berikut merupakan beberapa faktor yang mempengaruhi kualitas jawaban seseorang yang tidak dapat dicakup oleh prosedur dalam survei yang dijalankan:

1. Tafsiran penjawab terhadap pertanyaan yang diajukan. Untuk pertanyaan yang sama, orang-orang yang dimintai keterangan mungkin memiliki tafsiran yang berbeda-beda.
2. Kesudian penjawab untuk membantu penelitian yang dilaksanakan. Penjawab akan berusaha memberikan keterangan-keterangan yang sebaik mungkin, dalam arti keterangan tersebut sesuai dengan keadaan yang sebenarnya, apabila si penjawab secara jujur mau membantu penelitian.
3. Keadaan penjawab tatkala survei diadakan. Pada waktu diadakan survei, penjawab dapat berada dalam keadaan senang, gembira, sedih, jengkel, marah, dan lain-lain. Tiap-tiap keadaan tersebut akan memberikan pengaruh yang berbeda terhadap suatu pertanyaan yang diajukan.
4. Perhatian penjawab terhadap persoalan yang dikemukakan. Seseorang yang menggemari seni, misalnya, akan menaruh perhatian besar apabila pertanyaan yang diajukan berkaitan dengan seni. Besar kecilnya perhatian penjawab terhadap persoalan yang dikemukakan mungkin dipengaruhi oleh

perasaannya, apakah ia memiliki kepentingan terhadap soal tersebut atau tidak.

5. Ingatan dan taksiran penjawab, karena banyak pertanyaan yang harus dijawab berdasarkan ingatan dan taksiran, contohnya pertanyaan kepada petani mengenai pendapatan yang diperoleh selama jangka waktu 1 tahun.

Sebagaimana telah diuraikan di atas, survei adalah metode penelitian yang bertujuan untuk mencapai generalisasi dengan jalan membuat perbandingan kuantitatif dari data yang dikumpulkan. Metode ini tidak dapat digunakan untuk menjawab persoalan-persoalan dimana perbandingan kuantitatif itu tidak terdapat karena tekanan diberikan kepada perbandingan kuantitatif.

### **2.3 Hortikultura**

Hortikultura berasal dari kata *hortus* (garden atau kebun) dan *colere* (to cultivate atau budidaya). Secara harfiah istilah Hortikultura diartikan sebagai usaha membudidayakan tanaman buah-buahan, sayuran dan tanaman. Hortikultura merupakan suatu cabang dari ilmu pertanian yang mempelajari budidaya buah-buahan, sayuran dan tanaman hias. Sedangkan dalam GBHN 1993-1998 selain buah-buahan, sayuran dan tanaman hias, yang termasuk dalam kelompok hortikultura adalah tanaman obat-obatan. Ditinjau dari fungsinya tanaman hortikultura dapat memenuhi kebutuhan jasmani sebagai sumber vitamin, mineral dan protein (dari buah dan sayur), serta memenuhi kebutuhan rohani karena dapat memberikan rasa tenteram, ketenangan hidup dan estetika (dari tanaman hias/bunga). Peranan hortikultura adalah [9] :

1. Memperbaiki gizi masyarakat
2. Memperbesar devisa negara
3. Memperluas kesempatan kerja
4. Meningkatkan pendapatan petani
5. Pemenuhan kebutuhan keindahan dan kelestarian lingkungan.

Namun dalam membahas masalah hortikultura perlu diperhatikan pula mengenai sifat khas dari hasil hortikultura, yaitu :

1. Tidak dapat disimpan lama
2. Perlu tempat lapang (voluminous),
3. Mudah rusak (perishable) dalam pengangkutan
4. Melimpah/meruah pada suatu musim dan langka pada musim yang lain,
5. Fluktuasi harganya tajam [9].

#### **2.4 Survei Harga Pedesaan Hortikultura**

Pengumpulan data survei harga produsen pedesaan dilakukan setiap tanggal 15 tiap bulan dengan wawancara langsung kepada responden yaitu dengan menanyakan harga transaksi barang/jasa pertanian antara tanggal 1 sampai dengan 15 bulan bersangkutan (periode pencacahan), dengan menggunakan kuesioner/daftar HD. Satu kecamatan bisa terpilih menjadi sampel pada lebih dari satu jenis HD, bila kecamatan tersebut mempunyai potensi pertanian lebih dari satu subsektor. Namun, kecamatan yang mempunyai potensi pertanian pada beberapa subsektor, belum tentu menjadi kecamatan terpilih lebih dari satu jenis HD (bila ada kecamatan lain yang lebih potensi di Kabupaten tersebut).

Daftar HD digunakan untuk mencatat harga-harga komoditas subsektor tertentu di satu kecamatan terpilih. Misalnya, suatu kecamatan terpilih menjadi sampel HD, berarti kecamatan tersebut mempunyai potensi pertanian Subsektor Tanaman Hortikultura. Pelaksana mencatat harga-harga berbagai jenis komoditas hortikultura yang banyak dihasilkan oleh petani di kecamatan tersebut. Harga yang dicatat di sini adalah harga transaksi antara petani dengan pedagang pengumpul. Dengan demikian, baik petani (sebagai penjual) maupun pedagang pengumpul (sebagai pembeli) bisa menjadi responden untuk mencacah harga produsen yang diterima petani.

Responden Survei Harga Produsen Pedesaan (sektor pertanian) adalah petani yang menghasilkan komoditi pertanian. Selain petani, sebagai responden untuk mendapatkan data harga biaya produksi yang dikeluarkan oleh petani di kecamatan sampel, datanya ditanyakan antara lain kepada buruh tani, pedagang/petani yang menjual bibit, pedagang yang menjual pupuk dan obat-obatan, pedagang yang menjual alat-alat pertanian, usaha jasa persewaan lahan,

usaha jasa persewaan alat-alat pertanian, usaha jasa angkutan, dan lain-lain yang berkaitan dengan proses produksi pertanian.

Hal-hal yang perlu diperhatikan dalam pengisian daftar kuesioner HD adalah sebagai berikut:

1. Daftar isian hendaknya diisi dengan teliti, jelas, lengkap dan benar.
2. Jika terjadi satuan setempat yang berbeda dengan satuan standar, harus dikonversikan dahulu ke satuan standar.
3. Jagalah kualitas dan kuantitas komoditas pada responden yang sama. Hal ini sangat penting agar perubahan harga baik naik atau turunnya tidak bias karena pengaruh penggantian responden dipengaruhi oleh hal-hal tersebut.
4. Bila ada penggantian jenis barang yang hilang di pasaran, maka harap diberikan keterangan di blok catatan dan jangan lupa untuk menanyakan harga bulan sebelumnya dari jenis barang pengganti yang hilang tersebut dan harganya dicantumkan di kolom (6).
5. Usahakan setiap jenis barang yang telah dicantumkan pada daftar isian dan ada transaksi panen pada kecamatan yang bersangkutan dapat terisi seluruhnya.
6. Periksalah daftar isian sekali lagi dengan membandingkan harga dua bulan yang berurutan.
7. Jika ada perubahan harga yang mencolok harap diberikan catatan di kolom keterangan dengan menyebutkan sebab-sebabnya secara lengkap dan jelas [1].

## **2.5 Aplikasi**

Pengertian aplikasi menurut para ahli adalah sebagai berikut Pengertian aplikasi menurut para ahli adalah sebagai berikut:

1. Menurut Jogiyanto adalah penggunaan dalam suatu komputer, instruksi (instruction) atau pernyataan (statement) yang disusun sedemikian rupa sehingga komputer dapat memproses input menjadi output.
2. Menurut Kamus Kamus Besar Bahasa Indonesia adalah penerapan dari rancang sistem untuk mengolah data yang menggunakan aturan atau

ketentuan bahasa pemrograman tertentu. Aplikasi adalah suatu program komputer yang dibuat untuk mengerjakan dan melaksanakan tugas khusus dari pengguna.

3. Menurut Rachmad Hakim S, adalah perangkat lunak yang digunakan untuk tujuan tertentu, seperti mengolah dokumen, mengatur Windows &, permainan (game), dan sebagainya.
4. Menurut Harip Santoso, adalah suatu kelompok file (form, class, rePort) yang bertujuan untuk melakukan aktivitas tertentu yang saling terkait, misalnya aplikasi payroll, aplikasi fixed asset [9].

## 2.6 JSON

JSON (JavaScript Object Notation) adalah format pertukaran data yang ringan, mudah dibaca dan ditulis oleh manusia, serta mudah diterjemahkan dan dibuat (generate) oleh komputer. Format ini dibuat berdasarkan bagian dari Bahasa Pemrograman JavaScript, Standar ECMA-262 Edisi ke-3 – Desember 1999. JSON merupakan format teks yang tidak bergantung pada bahasa pemrograman apapun karena menggunakan gaya bahasa yang umum digunakan oleh programmer keluarga C termasuk C, C++, C#, Java, JavaScript, Perl, Python dll. Oleh karena sifat-sifat tersebut, menjadikan JSON ideal sebagai bahasa pertukaran-data.

Penulis menggunakan metode JSON dalam pengiriman data yang dilakukan, karena JSON memiliki beberapa kelebihan - kelebihan dibandingkan XML, kelebihan – kelebihan tersebut adalah:

1. Format Penulisan Untuk merepresentasikan sebuah struktur data yang rumit dan berbentuk hirarkis penulisan JSON relatif lebih terstruktur dan mudah.
2. Ukuran-ukuran karakter yang dibutuhkan JSON lebih kecil dibandingkan XML untuk data yang sama. Hal ini tentu berpengaruh pula pada kecepatan pertukaran data, walaupun tidak signifikan untuk data yang kecil, namun cukup berarti jika koneksi yang digunakan relatif lambat untuk mengakses aplikasi web kaya fitur yang memanfaatkan pertukaran data. Di sini JSON lebih unggul dibandingkan XML, kecuali jika data dikompresi terlebih dahulu

sebelum dikirimkan, perbedaan JSON dan XML yang telah dikompresi tidaklah signifikan.

3. Browser Parsing Proses parsing merupakan proses pengenalan token atau bagian-bagian kecil dalam rangkaian dokumen XML/JSON. Contohnya, terdapat data text dalam format JSON. Data tersebut harus di-parsing terlebih dahulu sebelum dapat diakses dan dimanipulasi. Browser parsing berarti proses parsing yang terjadi pada sisi client/browser. Melakukan browser parsing pada JSON lebih sederhana dibandingkan pada XML, JSON menggunakan function JavaScript `eval()` untuk melakukan parsing. Sementara dokumen XML di-parsing oleh `XMLHttpRequest`. Rata-rata survei menobatkan JSON sebagai pemenang jika diadu kecepatan parsingnya.

JSON terbuat dari dua struktur:

1. Kumpulan pasangan nama/nilai. Pada beberapa bahasa, hal ini dinyatakan sebagai objek (*object*), rekaman (*record*), struktur (*struct*), kamus (*dictionary*), tabel hash (hash table), daftar berkunci (*keyed list*), atau *associative array*.
2. Daftar nilai terurutkan (*an ordered list of values*). Pada kebanyakan bahasa, hal ini dinyatakan sebagai larik (*array*), vektor (*vector*), daftar (*list*), atau urutan (*sequence*).

Struktur-struktur data ini disebut sebagai struktur data universal. Pada dasarnya, semua bahasa pemrograman moderen mendukung struktur data ini dalam bentuk yang sama maupun berlainan. Hal ini pantas disebut demikian karena format data mudah dipertukarkan dengan bahasa-bahasa pemrograman yang juga berdasarkan pada struktur data ini [10].

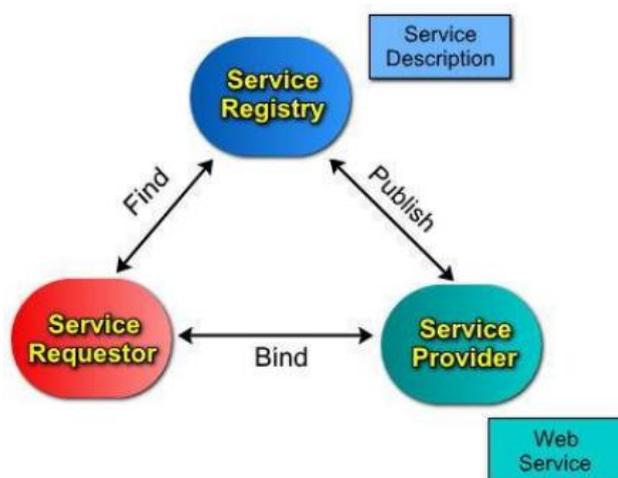
## 2.7 Web Service

Web service adalah salah satu bentuk sistem perangkat lunak yang didesain untuk mendukung interaksi mesin-ke-mesin melalui jaringan. Web service memiliki interface yang dideskripsikan dalam format yang dapat dibaca oleh mesin. Sistem-sistem lainnya berinteraksi dengan web service menggunakan pesan SOAP yang umumnya dikirim melalui HTTP dalam bentuk XML.

Definisi diatas diberikan oleh World Wide Web Consortium(W3C) yang merupakan badan yang menciptakan dan mengembangkan standar web service. Tetapi secara umum, web service tidak terbatas hanya pada standar SOAP saja. Salah satu pustaka yang mengulas lengkap tentang web service menyebutkan definisi yang lebih umum: web service adalah aplikasi yang diakses melalui internet menggunakan protokol standar internet dan menggunakan XML sebagai format pesannya [11].

### 2.7.1 Arsitektur Web Service

Secara umum, arsitektur web service dapat dilihat pada Gambar 2.3



Sumber gambar : Jurnal Komputer (2007) [11]

**Gambar 2. 3 Arsitektur Web Service**

Pada Gambar 2.3, ada tiga komponen yang membuat web service berjalan. Ketiga komponen itu adalah:

1. Service provider, merupakan pemilik Web service yang berfungsi menyediakan kumpulan operasi dari Web service.
2. Service requestor, merupakan aplikasi yang bertindak sebagai klien dari Web service yang mencari dan memulai interaksi terhadap layanan yang disediakan.
3. Service registry, merupakan tempat dimana Service provider mempublikasikan layanannya. Pada arsitektur Web service, Service registry

bersifat optional. Teknologi web service memungkinkan kita dapat menghubungkan berbagai jenis software yang memiliki platform dan sistem operasi yang berbeda [11].

## 2.8 Android

Menurut Nazarudin Safaat H, “Android adalah sebuah sistem operasi untuk perangkat mobile berbasis linux yang mencakup sistem operasi, *middleware*, dan aplikasi”. Android adalah sistem operasi untuk telepon seluler yang berbasis linux. Android menyediakan *platform* terbuka bagi para pengembang untuk membuat aplikasi mereka sendiri. Pada awalnya dikembangkan oleh Android Inc, sebuah perusahaan pendatang baru yang membuat perangkat lunak untuk ponsel yang kemudian dibeli oleh Google Inc. untuk pengembangannya, dibentuklah *Open Handset Alliance* (OHA), konsorsium dari 34 perusahaan perangkat keras, perangkat lunak, dan telekomunikasi termasuk Google, HTC, Intel, Motorola Qualcomm, T-Mobile, dan Nvidia [9].

Android dipuji sebagai “*platform mobile* pertama yang lengkap, terbuka dan bebas” [10].

### 1. Lengkap (*Complete Platform*)

Para desainer dapat melakukan pendekatan yang komprehensif ketika mereka sedang mengembangkan platform Android. Android merupakan sistem operasi yang aman dan banyak menyediakan tools dalam membangun software dan memungkinkan untuk peluang pengembangan aplikasi.

### 2. Terbuka (*Open Source Platform*)

*Platform* Android disediakan melalui lisensi *open source*. Pengembang dapat dengan bebas untuk mengembangkan aplikasi. Android sendiri menggunakan Linux Kernel 2.6.

### 3. Free (*Free Platform*)

Android adalah *platform/aplikasi* yang bebas untuk *develop*. Tidak ada lisensi atau biaya royalti untuk dikembangkan pada *platform* Android. Tidak ada biaya keanggotaan diperlukan. Tidak diperlukan biaya pengujian. Tidak ada

kontrak yang diperlukan. Aplikasi untuk android dapat didistribusikan dan diperdagangkan dalam bentuk apapun [10].

### 2.8.1 Sejarah Android

Pada saat perilis perdana Android, 5 November 2007, Android bersama *Open Handset Alliance* menyatakan mendukung pengembangan open source pada perangkat mobile. Di lain pihak, Google merilis kode-kode Android di bawah lisensi Apache, sebuah lisensi perangkat lunak dan open platform perangkat seluler [10].

Di dunia ini terdapat dua jenis distributor sistem operasi Android. Pertama yang mendapat dukungan penuh dari Google atau *Google Mail Service* (GMS) dan kedua adalah yang benar-benar bebas distribusinya tanpa dukungan langsung Google atau dikenal sebagai *Open Handset Distribution* (OHD) [10].

Sekitar September 2007 Google mengenalkan Nexus One, salah satu jenis *smartphone* yang menggunakan Android sebagai sistem operasinya. *Telepon seluler* ini diproduksi oleh HTC corporation dan tersedia di pasaran pada 5 Januari 2010. Pada 9 Desember 2008, diumumkan anggota baru yang Communications, diproduksi oleh Asustek Computer Inc, Garmin Ltd, Softbank, Sony Ericsson, Toshiba Corp, dan Vodafone Group Plc. Seiring pembentukan Open Handset Alliance, OHA mengumumkan produk perdana mereka Android, perangkat mobile yang merupakan modifikasi kernel Linux 2.6. Sejak Android dirilis telah dilakukan berbagai pembaruan berupa perbaikan bug dan penambahan fitur baru [10].

Pada masa saat ini sebagian besar vendor-vendor *smartphone* sudah memproduksi *smartphone* berbasis Android, vendor-vendor itu antara lain HTC, Motorola, Samsung, LG, HKC, Huawei, Archos, Webstation Camangi, Dell, Nexus, SciPhone, WayteQ, Sony Ericsson, LG, Acer, Philips, T-Mobile, Nexian, IMO, Asus dan masih banyak lagi vendor *smartphone* di dunia yang memproduksi android. Hal ini, karena android itu adalah sistem operasi yang open source sehingga bebas didistribusikan dan dipakai oleh vendor manapun [10].

Tidak hanya menjadi sistem operasi di *smartphone*, saat ini Android menjadi pesaing utama dari Apple pada sistem operasi Tablet PC. Pesatnya pertumbuhan Android selain faktor yang disebutkan diatas adalah karena Android itu sendiri adalah platform yang sangat lengkap baik itu sistem operasinya, Aplikasi dan Tool pengembangan, Market aplikasi Android serta dukungan yang sangat tinggi dari komunitas *Open Source* di dunia, sehingga Android terus berkembang pesat baik dari segi teknologi maupun dari segi jumlah *device* yang ada di dunia [10].

### 2.8.2 A-GPS

Kemampuan sebuah *smartphone* dalam mengetahui posisi di muka bumi menjadi semakin penting. Bahkan saat ini sudah merambah ke penggunaannya yang sangat umum. Sedemikian umumnya, hingga semua *smartphone* sudah wajib memiliki kemampuan untuk mengetahui posisi penggunanya. Teknologi yang digunakan untuk mengetahui posisi suatu perangkat yaitu GPS (*Global Positioning System*).

GPS singkatan dari *Global Positioning System* (Sistem Pencari Posisi Global), adalah suatu jaringan satelit yang secara terus menerus memancarkan sinyal radio dengan frekuensi yang sangat rendah. Alat penerima GPS secara pasif menerima sinyal ini, dengan syarat bahwa pandangan ke langit tidak boleh terhalang, sehingga biasanya alat ini hanya bekerja di ruang terbuka. Satelit GPS bekerja pada referensi waktu yang sangat teliti dan memancarkan data yang menunjukkan lokasi dan waktu pada saat itu. Operasi dari seluruh satelit GPS yang ada disinkronisasi sehingga memancarkan sinyal yang sama. Alat penerima GPS akan bekerja jika ia menerima sinyal dari sedikitnya 4 buah satelit GPS, sehingga posisinya dalam tiga dimensi bisa dihitung. Pada saat ini sedikitnya ada 24 satelit GPS yang beroperasi setiap waktu dan dilengkapi dengan beberapa cadangan. Satelit tersebut dioperasikan oleh Departemen Pertahanan Amerika Serikat, mengorbit selama 12 jam (dua orbit per hari) pada ketinggian sekitar 11.500 mil dan bergerak dengan kecepatan 2000 mil per jam. Ada stasiun penerima di bumi yang menghitung lintasan orbit setiap satelit dengan teliti [11].

## 2.9 Android SDK

Android SDK adalah tools API (*Application Programming Interface*) yang diperlukan untuk mulai mengembangkan aplikasi pada platform Android menggunakan bahasa pemrograman Java. Android merupakan subset perangkat lunak untuk ponsel yang meliputi sistem operasi, *middleware* dan aplikasi kunci yang di release oleh Google. Saat ini disediakan Android SDK (*Software Development Kit*) sebagai alat bantu dan API untuk mulai mengembangkan aplikasi pada platform aplikasi-netral, Android memberikan kesempatan untuk membuat Aplikasi yang kita butuhkan yang bukan merupakan aplikasi bawaan Handphone/*Smartphone*. Beberapa fitur-fitur Android yang paling penting adalah [10]:

1. *Framework* Aplikasi yang mendukung pengganti komponen dan *reusable*.
2. Mesin Virtual Dalvik dioptimalkan untuk mobile.
3. *Integrated browser* berdasarkan *engine open source webkit*.
4. Grafis yang dioptimalkan dan didukung oleh libraries grafis 2D, grafis 3D berdasarkan spesifikasi opengi ES 1.0 (Opsional akselerasi hardware).
5. SQL untuk menyimpan data.
6. Media Support yang mendukung audio, video, dan gambar (MPEG4, H.264, MP3, AAC, AMR, JPG, PNG, GIF), GSM Telephone (Tergantung *hardware*).
7. Bluetooth, EDGE, 3G, dan Wifi (tergantung *hardware*).
8. Kamera, GPS, kompas, dan *accelerometer* (tergantung *hardware*).
9. Lingkungan Development yang lengkap dan kaya termasuk perangkat emulator, tools untuk debugging, profil dan kinerja memori, dan plugin untuk IDE Eclipse.

Setiap kali Google merilis versi baru dari Android, sebuah SDK yang sesuai juga dirilis. Untuk dapat menulis program dengan fitur terbaru, pengembang harus men *download* dan menginstal SDK versi masing-masing untuk ponsel tertentu. Agar sistem alat bantu dapat dijalankan pada semua versi, maka Android SDK dirasa perlu digunakan.

## 2.10 Android Studio

Android Studio adalah sebuah IDE untuk Android Development yang diperkenalkan google pada acara Google I/O 2013. Android Studio merupakan pengembangan dari Eclipse IDE, dan dibuat berdasarkan IDE Java populer, yaitu IntelliJ IDEA. Android Studio merupakan IDE resmi untuk pengembangan aplikasi Android.

Sebagai pengembangan dari Eclipse, Android Studio mempunyai banyak fitur-fitur baru dibandingkan dengan Eclipse IDE. Berbeda dengan Eclipse yang menggunakan Ant, Android Studio menggunakan Gradle sebagai build environment. Fitur-fitur lainnya adalah sebagai berikut :

1. Menggunakan Gradle-based build system yang fleksibel.
2. Bisa mem-build multiple APK .
3. Template support untuk Google Services dan berbagai macam tipe perangkat.
4. Layout editor yang lebih bagus.
5. Built-in support untuk Google Cloud Platform, sehingga mudah untuk integrasi dengan Google Cloud Messaging dan App Engine.
6. Import library langsung dari Maven repository

Jika dibandingkan dengan Android Studio memang dari sisi build lebih baik dibandingkan Eclipse, karena Android Studio menggunakan Gradle. Pada Android Studio, kita tidak perlu lagi dipusingkan dengan *dependencies package* berbeda dengan Eclipse. Satu hal tambahan lagi yang membuat Android Studio unggul adalah dukungan layout xml editor secara visual yang jauh lebih baik daripada Eclipse. Walaupun begitu, Android Studio saat ini masih dalam tahap beta dan belum mempunyai dukungan untuk NDK/Native Development Kit.

## 2.11 Java

Java adalah sebuah bahasa pemrograman berorientasi objek yang dikembangkan oleh Sun Microsystems. Bahasa pemrograman Java adalah bahasa yang digunakan untuk menghasilkan aplikasi-aplikasi Java. Pada umumnya, bahasa pemrograman hanya mendefinisikan sintaks dan perilaku bahasa [12].

Pada saat program Java dikompilasi, ia akan dikonversi ke bentuk *bytecode*, yang merupakan bahasa mesin yang portable. Selanjutnya *bytecode* tersebut dijalankan di *Java Virtual Machine* (JVM). Meskipun JVM dapat diimplementasikan langsung di perangkat keras, namun biasanya diimplementasikan dalam bentuk program perangkat lunak yang mengemulsi mesin (komputer) dan digunakan untuk menginterpretasi *bytecode* [12].

Guna mencakup lingkungan-lingkungan aplikasi yang berbeda, Sun mendefinisikan 3 (tiga) edisi Java, yaitu [12]:

1. J2ME (*Java 2 Micro Edition*)

Edisi ini ditujukan bagi lingkungan dengan sumber daya terbatas, seperti *smartcard*, ponsel dan PDA.

2. J2SE (*Java 2 Standard Edition*)

3. Edisi ini ditujukan bagi lingkungan *workstation*, seperti pembuatan aplikasi-aplikasi desktop.

4. J2EE (*Java 2 Enterprise Edition*)

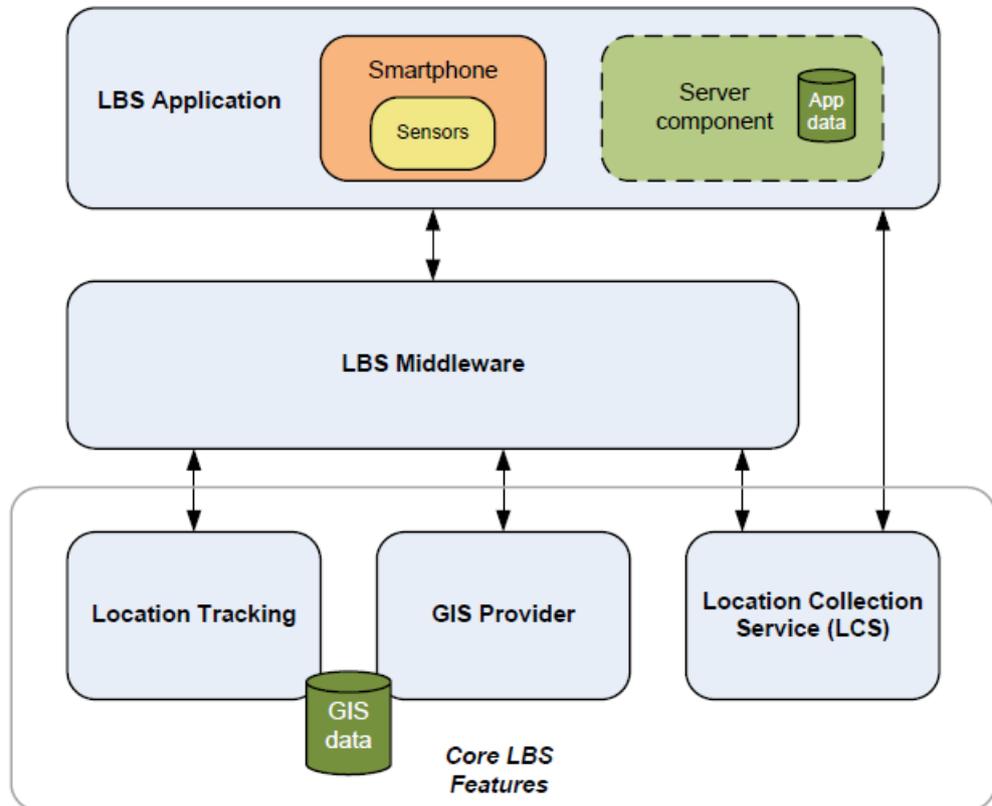
Edisi ini ditujukan bagi lingkungan internet atau aplikasi terdistribusi dalam skala besar.

Pada sistem alat bantu survei harga produsen pedesaan, smartphone android dijadikan media dalam menjalankan sistem sehingga bahasa pemrograman yang sesuai yaitu bahasa pemrograman java.

## **2.12 Location Based Service (LBS)**

Ardiansyah menyatakan “LBS merupakan Layanan IP nirkabel yang menggunakan informasi geografis untuk memberikan layanan informasi lokasi kepada pengguna. Beberapa layanan aplikasi yang memberikan petunjuk posisi/lokasi piranti mobile berada”. LBS memberikan layanan personalisasi kepada pengguna perangkat bergerak (*mobile device*) yang disesuaikan dengan lokasi mereka saat ini. LBS adalah layanan informasi yang dapat diakses melalui mobile device dengan menggunakan mobile network, yang dilengkapi kemampuan untuk memanfaatkan lokasi dari mobile device tersebut. LBS memberikan kemungkinan komunikasi dan interaksi dua arah. Oleh karena itu

pengguna memberitahu penyedia layanan untuk mendapatkan informasi yang dia butuhkan, dengan referensi posisi pengguna tersebut [13].



Sumber gambar : Amit Kushuwa (2011) [13]

#### Gambar 2. 4 Arsitektur LBS

Sistem yang diusulkan akan memberikan fasilitas kepada pengguna ketika pengguna baru datang ke tempat manapun melalui aplikasi. Aplikasi peringatan kedekatan sistem yang diusulkan akan memberikan pemberitahuan kepada pengguna kedatangan tujuan yang ditentukan saat bepergian dalam bentuk alarm. Aplikasi lain dari sistem yang diusulkan adalah jalur terpendek yang akan memberikan layanan kepada pengguna dengan menambang jarak terpendek antara sumber dan tujuan. Aplikasi lain yang bernama pencari teman dari sistem yang diusulkan akan memberikan pemberitahuan kepada pengguna tentang teman-temannya di dekatnya. Jika dua teman berada di lokasi yang sama, maka yang pertama datang yang dianggap terdekat [14].

Pencarian lengkap dari semua jalur yang mungkin akan menemukan rute terpendek, tetapi secara komputasi sulit untuk semua set kecil dari lokasi. Untuk masalah yang lebih besar, teknik pengoptimalkan diperlukan untuk mencari ruang solusi dan mencari solusi yang mendekati optimal. Secara matematis, masalah perjalanan dapat direpresentasikan sebagai grafik, dimana lokasi adalah node dan ujung (atau busur) mewakili rute langsung antara node. Berat masing-masing ujung adalah jarak antar simpul. Tujuannya adalah menemukan jalur dengan jumlah bobot terpendek. Google juga menyediakan cara untuk memecahkan TSP secara sederhana dari lokasi dunia nyata tanpa mengunduh *OR-Tools*. API Google memiliki kunci Arah sehingga dapat menyelesaikan TSP lokasi dunia nyata dengan API, Google juga menyediakan lokasi di URL dan akan mendapatkan tanggapan kembali sebagai JSON. API Google dapat digunakan secara gratis untuk pengembangan atau untuk penggunaan komersial.

### **2.13 Google Maps API (*Application Programming Interface*)**

API merupakan software interface yang terdiri atas kumpulan instruksi yang disimpan dalam bentuk library dan menjelaskan bagaimana agar suatu software dapat berinteraksi dengan software lain. Penjelasan ini dapat dicontohkan dengan analogi apabila akan dibangun suatu rumah. Dengan menyewa kontraktor yang dapat menangani bagian yang berbeda, pemilik rumah dapat memberikan tugas yang perlu dilakukan oleh kontraktor tanpa harus mengetahui bagaimana cara kontraktor menyelesaikan pekerjaan tersebut. Dari analogi tersebut, rumah merupakan software yang akan dibuat, dan kontraktor merupakan API yang mengerjakan bagian tertentu dari software tersebut tanpa harus diketahui bagaimana prosedur dalam melakukan pekerjaan tersebut.

Google Map adalah sebuah jasa peta global virtual gratis dan online yang disediakan oleh Google dan dapat ditemukan di <http://maps.google.com>. Google Map memberikan layanan untuk menunjukkan jalan-jalan yang ada di seluruh dunia. Bentuknya seperti peta yang umum, tetapi Google Map hanya menunjukkan jalan-jalan saja, dan tidak ada nama gunung, sungai, ataupun batas-batas daerah yang biasa ditemukan di peta umum. Walaupun begitu, peta yang ada

di Google Map ini sangat lengkap. Untuk kota-kota besar, Google Map juga menampilkan nama-nama jalan dan gedung. Selain dalam bentuk peta, Google Map juga bisa menampilkan dalam bentuk foto satelit [19].

Berikut ini adalah fasilitas dari Google Map :

1. Legenda-legenda (*legend*) adalah keterangan tentang obyek-obyek yang ada di peta, seperti warna hijau adalah hutan, garis merah adalah jalan, simbol buku adalah universitas, dan sebagainya.
2. Zoom in / out Peta di layar dapat diperbesar dengan zoom in dan diperkecil dengan zoom out.
3. Pan dengan fasilitas pan peta dapat digeser-geser untuk melihat daerah yang dikehendaki.
4. Kemampuan untuk mengubah tampilan dari peta jalan ke tampilan satelit maupun hybrid view dan akan mengkombinasikan tampilan satelit dengan nama jalan dan bangunan.
5. Search Bar yang berguna mencari suatu lokasi secara cepat
6. Mencari arah jalan dengan mudah dan secara otomatis akan ditampilkan rute perjalanannya.
7. Bisa memilih tampilan full screen ketika membuka peta jalan.
8. Bisa menyimpan lokasi-lokasi favorit anda dan menandainya sehingga bisa kembali menemukannya dengan mudah.
9. Bisa dengan mudah mengambil koordinat dari sebuah lokasi yang pilih.

Google maps API adalah fungsi-fungsi pemrograman yang disediakan oleh Google maps agar Google maps bisa diintegrasikan ke dalam Web atau aplikasi. Google Maps API merupakan aplikasi interface yang dapat diakses lewat *javascript* agar Google Map dapat ditampilkan pada sebuah halaman web. Pada Google Maps JavaScript API versi 2, untuk dapat mengakses Google Map pada sebuah halaman web diperlukan Google API Key. API Key adalah sederetan kode sebagai izin untuk menampilkan Google Map pada sebuah halaman web. Namun untuk versi 3 tidak memerlukan API Key, tetapi pihak google menganjurkan menggunakan API Key untuk mempermudah mengontrol Google Maps API. API

Key bersifat tunggal, hanya berlaku untuk sebuah URL. Salah satu syarat untuk mendapatkan API Key adalah mempunyai akun google/Gmail untuk meng-generate domain atau URL web pada link <https://code.google.com/apis/console/>. API Key bersifat gratis sampai batas 25.000 pengunjung per hari. Jika melebihi 25.000 pengunjung per hari maka diperlukan membeli kuota tambahan. [19]

#### 2.14 Clarifai

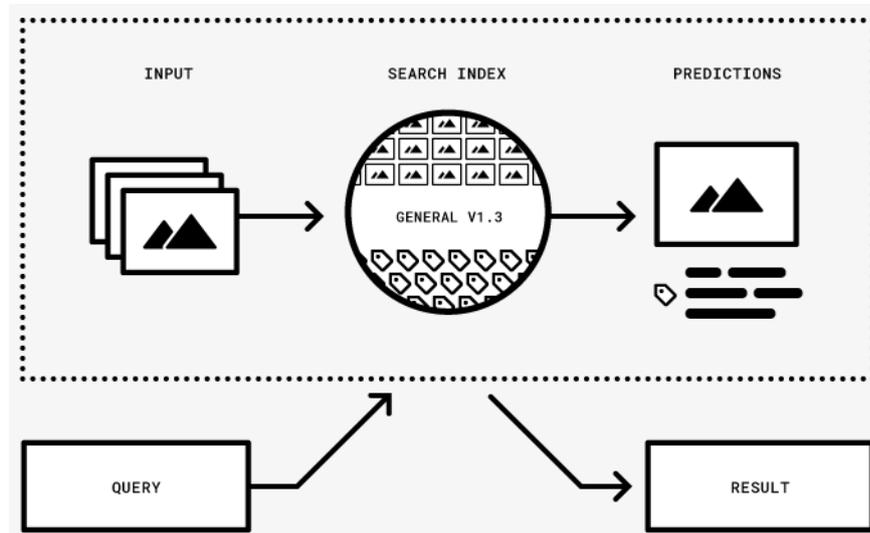
Saat ini, penggunaan alat eksternal atau *remote* yang menyediakan beberapa fungsi atau informasi. Sistem informasi yang mengolah seperti *grid and cloud computing* yang memungkinkan mengakses berbagai layanan dan kemampuan baru yaitu sistem Clarifai. Sistem ini menyediakan informasi teknologi untuk menganalisis gambar dan mengidentifikasi anotasi deskriptif yang ada keterkaitannya. Clarifai menawarkan *Application Programming Interface* (API) yang memperoleh 20 anotasi paling deskriptif dari gambar yang dikirimkan [5].

Teknologi Clarifai bergantung pada penggunaan *Convolutional Neural Networks* (CNN) untuk memproses gambar. CNN didefinisikan sebagai model pembelajaran mesin hierarkis yang mempelajari gambar yang kompleks representasi dari volume besar data beranotasi. Mereka banyak menggunakan lapisan transformasi dasar yang menghasilkan representasi yang sangat canggih. Pendekatan Clarifai pertama kali diusulkan ke ImageNet pada tahun 2013 dimana sistem menghasilkan 5 hasil teratas [5].

Clarifai bekerja melalui analisis gambar untuk menghasilkan deskripsi dari gambar tersebut. Sistem ini juga akan memberikan kemungkinan terbesar dari gambar yang dimaksud. API Clarifai dapat diakses melalui website dengan menggunakan gambar sebagai inputannya, CNN akan menganalisa dengan memberikan kemungkinan beberapa jawaban [5].

Kasus umum untuk menggunakan Clarifai adalah untuk mendapatkan konsep yang diprediksi dalam sebuah gambar dan kemudian menggunakan konsep tersebut untuk mencari daya. API Pencarian memungkinkan Anda mengirim gambar (url atau byte) ke layanan dan meminta mereka diindeks oleh konsep

model umum dan representasi visualnya. Setelah diindeks, dapat mencari gambar berdasarkan konsep atau dengan gambar. Cara kerja Clarifai dapat dilihat pada Gambar 2.5 [20]:



Sumber gambar : [www.clarifai.com/developer/guide](http://www.clarifai.com/developer/guide) [20]

**Gambar 2. 5 Ilustrasi Cara Kerja Clarifai**

### 2.15 Pemrograman Berorientasi Objek

Metodologi berorientasi objek adalah suatu strategi pembangunan perangkat lunak yang mengorganisasikan perangkat lunak sebagai kumpulan objek yang berisi data dan operasi yang diberlakukan terhadapnya. Metodologi berorientasi objek merupakan suatu cara bagaimana system perangkat lunak dibangun melalui pendekatan objek secara sistematis. Metode berorientasi objek didasarkan pada penerapan prinsip-prinsip pengelolaan kompleksitas. Metode berorientasi objek meliputi rangkaian aktivitas analisis berorientasi objek, perancangan objek, pemrograman berorientasi objek, dan pengujian berorientasi objek [15].

Keuntungan menggunakan metodologi berorientasi objek adalah sebagai berikut:

1. Meningkatkan produktivitas

Karena kelas dan objek yang ditemukan dalam suatu masalah masih dapat dipakai ulang untuk masalah lainnya yang melibatkan objek tersebut (reusable).

2. Kecepatan pengembangan

Karena system yang dibangun dengan baik dan benar pada saat analisis dan perancangan akan menyebabkan berkurangnya kesalahan pada saat pengkodean

3. Kemudahan pemeliharaan.

Karena dengan model objek, pola-pola yang cenderung tetap dan stabil dapat dipisahkan dan pola-pola yang mungkin sering berubah-ubah.

4. Adanya konsistensi

Karena sifat pewarisan dan penggunaan notasi yang sama pada saat analisis, perancangan maupun pengkodean.

5. Meningkatkan kualitas perangkat lunak

Karena pendekatan pengembangan lebih dekat dengan dunia nyata dan adanya konsistensi pada saat pengembangannya, perangkat lunak yang dihasilkan akan mampu memenuhi kebutuhan pemakai serta mempunyai sedikit kesalahan.

Pendekatan berorientasi objek merupakan suatu teknik atau cara pendekatan dalam melihat permasalahan dan system (system perangkat lunak, system informasi, atau system lainnya). Pendekatan berorientasi objek akan memandang system yang akan dikembangkan sebagai suatu kumpulan objek yang berkorespondensi dengan objek-objek dunia nyata.

Ada banyak cara untuk mengabstraksikan dan memodelkan objek-objek tersebut, mulai dari abstraksi objek, kelas, hubungan antarkelas sampai abstraksi system. Saat mengabstraksikan dan memodelkan objek, data dan proses-proses yang dipunyai oleh objek akan dienkapsulasi (dibungkus) menjadi satu kesatuan.

Dalam rekayasa perangkat lunak, konsep pendekatan berorientasi objek dapat diterapkan pada tahap analisis, perancangan, pemrograman, dan pengujian

perangkat lunak. Ada berbagai teknik yang dapat digunakan pada masing-masing tahap tersebut, dengan aturan dan alat bantu pemodelan tertentu.

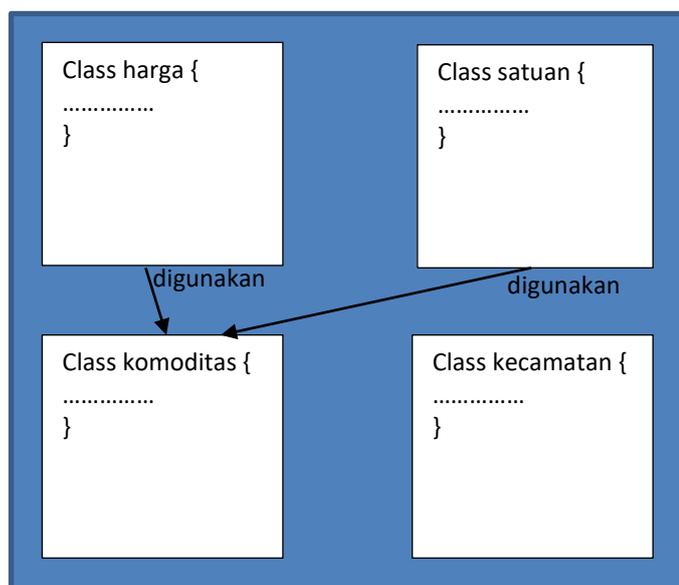
Sistem berorientasi objek merupakan sebuah system yang dibangun dengan berdasarkan metode berorientasi objek adalah sebuah system yang komponennya dibungkus (dienkapsulasi) menjadi kelompok data dan fungsi. Setiap komponen dalam system tersebut dapat mewarisi atribut dan sifat dan komponen lainnya. Dan dapat berinteraksi satu sama lain.

Berikut ini adalah beberapa konsep dasar yang harus dipahami tentang metodologi berorientasi objek:

1. Kelas (*class*)

Kelas adalah kumpulan objek-objek dengan karakteristik yang sama. Kelas merupakan definisi static dan himpunan objek yang sama yang mungkin lahir atau diciptakan dan kelas tersebut. Sebuah kelas akan mempunyai sifat (atribut). Kelakuan (operasi/metode), hubungan (*relationship*) dan arti. Suatu kelas dapat diturunkan dan kelas yang lain, dimana atribut dan kelas semula dapat diwariskan ke kelas yang baru.

Secara teknis, kelas adalah sebuah struktur tertentu dalam pembuatan perangkat lunak. Kelas merupakan bentuk struktur pada kode program yang menggunakan metodologi berorientasi objek. Ilustrasi dari sebuah kelas dapat dilihat pada Gambar 2.6:



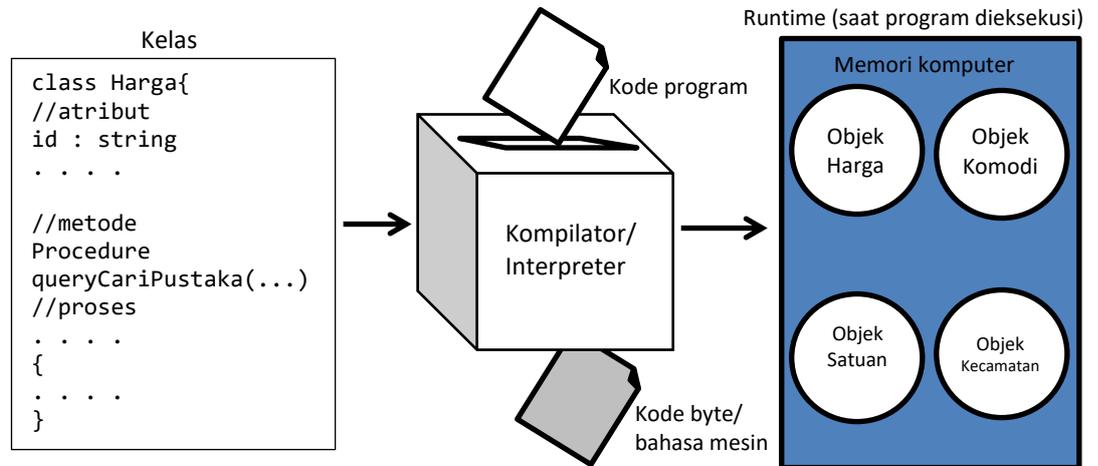
**Gambar 2. 6 Ilustrasi Kelas**

Kelas secara fisik adalah berkas atau file yang berisi kode program, dimana kode program merupakan semua hal yang terkait dengan nama kelas.

## 2. Objek (*object*)

Objek adalah abstraksi dan sesuatu yang mewakili dunia nyata seperti benda, manusia, satuan organisasi, tempat, kejadian, struktur, status, atau hal - hal lain yang bersifat abstrak. Objek merupakan suatu entitas yang mampu menyimpan informasi (status) dan mempunyai operasi (kelakuan) yang dapat diterapkan atau dapat berpengaruh pada status objeknya. Objek mempunyai siklus hidup yaitu diciptakan, dimanipulasi, dan dihancurkan.

Secara teknis, sebuah kelas saat program dieksekusi maka akan di buat sebuah objek. Objek dilihat dari segi teknis adalah elemen pada saat runtime yang akan diciptakan, dimanipulasi, dan dihancurkan saat eksekusi sehingga sebuah objek hanya ada saat sebuah objek hanya ada saat sebuah program dieksekusi, jika masih dalam bentuk kode, disebut sebagai kelas jadi pada saat runtime (saat sebuah program dieksekusi), yang kita punya adalah objek, di dalam teks program yang kita lihat hanyalah kelas.



**Gambar 2. 7 Ilustrasi Kelas dan Objek**

### 3. Metode (*method*)

Operasi atau metode atau method pada sebuah kelas hampir sama dengan fungsi atau prosedur pada metodologi struktural. Sebuah kelas boleh memiliki lebih dari satu metode atau operasi. Metode atau operasi yang berfungsi untuk memanipulasi objek itu sendiri. Operasi atau metode merupakan fungsi atau transformasi yang dapat dilakukan terhadap objek atau dilakukan oleh objek.

Metode atau operasi dapat berasal dari event, aktivitas atau aksi keadaan, fungsi, atau kelakuan dunia nyata. Atau kelakuan dunia nyata. Contoh metode atau operasi misalnya read, write, move copy, dan sebagainya. Kelas sebaiknya memiliki metode get dan set untuk setiap atribut agar konsep enkapsulasi tetap terjaga. Metode get digunakan untuk memberikan akses kelas ini dalam mengakses atribut, dan set adalah metode yang digunakan untuk mengisi atribut, agar kelas ini tidak mengakses atribut secara langsung.

#### a. Atribut (*attribute*)

Atribut dari sebuah kelas adalah variabel global yang dimiliki sebuah kelas. Atribut dapat berupa nilai atau elemen-elemen data yang dimiliki oleh objek dalam kelas objek. Atribut dipunyai secara individual oleh sebuah objek, misalnya berat, jenis, nama, dan sebagainya. Atribut sebaiknya bersifat privat untuk menjaga konsep enkapsulasi.

#### b. Abstraksi (*abstraction*)

Prinsip untuk merepresentasikan dunia nyata yang kompleks menjadi satu bentuk model yang sederhana dengan mengabaikan aspek-aspek lain yang tidak sesuai dengan permasalahan.

c. Enkapsulasi (*encapsulation*)

Pembungkusan atribut data dan layanan (operasi-operasi) yang dipunyai objek untuk menyembunyikan implementasi dan objek sehingga objek lain tidak mengetahui cara kerjanya.

d. Pewarisan (*inheritance*)

Mekanisme yang memungkinkan satu objek mewarisi sebagian atau seluruh definisi dan objek lain sebagai bagian dan dirinya.

e. Antarmuka (*interface*)

Antarmuka atau interface sangat mirip dengan kelas, tapi tanpa atribut kelas dan memiliki metode yang dideklarasikan tanpa isi. Deklarasi metode pada sebuah interface dapat diimplementasikan oleh kelas lain. Sebuah kelas dapat mengimplementasikan lebih dari satu antarmuka dimana kelas ini akan mendeklarasikan metode pada antarmuka yang dibutuhkan oleh kelas itu sekaligus mendefinisikan isinya pada kode program kelas itu. Metode pada antarmuka yang diimplementasikan pada suatu kelas harus sama persis dengan yang ada pada antarmuka. Antarmuka atau *interface* biasanya digunakan agar kelas yang lain tidak mengakses langsung ke suatu kelas, mengakses antarmuka.

f. *Reusability*

Pemanfaatan kembali objek yang sudah didefinisikan untuk suatu permasalahan pada permasalahan lainnya yang melibatkan objek tersebut. Misalkan dalam sebuah aplikasi peminjaman buku diperlukan kelas anggota, maka ketika membuat aplikasi penyewaan VCD, kelas anggota ini bisa digunakan kembali dengan sedikit perubahan untuk aplikasi penyewaan VCD tanpa harus membuat dari awal kembali.

g. Generalisasi dan Spesialisasi

Menunjukkan hubungan antara kelas dan objek umum dengan kelas dan objek yang khusus. Misalnya kelas yang lebih umum (generalisasi) adalah

kendaraan darat dan kelas khususnya (spesialisasi) adalah mobil, motor, dan kereta.

h. Komunikasi Antar Objek

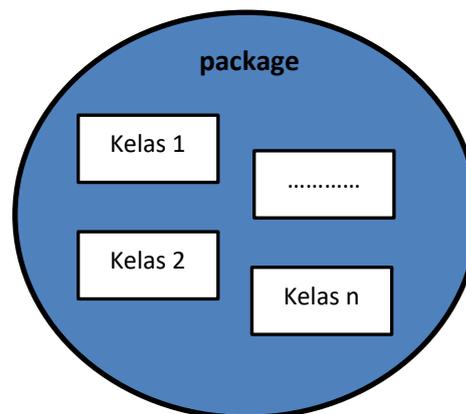
Komunikasi antar-objek dilakukan lewat pesan (message) yang dikirim dan satu objek ke objek lainnya.

i. Polimorfisme (*polymorphism*)

Kemauan suatu objek untuk digunakan di banyak tujuan yang berbeda dengan nama yang sama sehingga menghemat baris program.

j. *Package*

*Package* adalah sebuah container atau kemasan yang dapat di gunakan untuk mengelompokkan kelas-kelas sehingga memungkinkan beberapa kelas yang bernama sama disimpan dalam package yang berbeda. Ilustrasi dari sebuah package dapat dilihat pada Gambar 2.8



Sumber gambar: Buku *Rekayasa Perangkat Lunak* oleh Rosa A.S dan M.

Shalahuddin (2013) [15]

**Gambar 2. 8 Package**

## 2.16 UML (*Unified Modelling Language*)

*Unified Modeling Language* (UML) adalah suatu alat untuk memvisualisasikan dan mendokumentasikan hasil analisa dan desain yang berisi sintak dalam memodelkan sistem secara visual. Juga merupakan satu kumpulan konvensi pemodelan yang digunakan untuk menentukan atau menggambarkan sebuah sistem *software* yang terkait dengan objek [16].

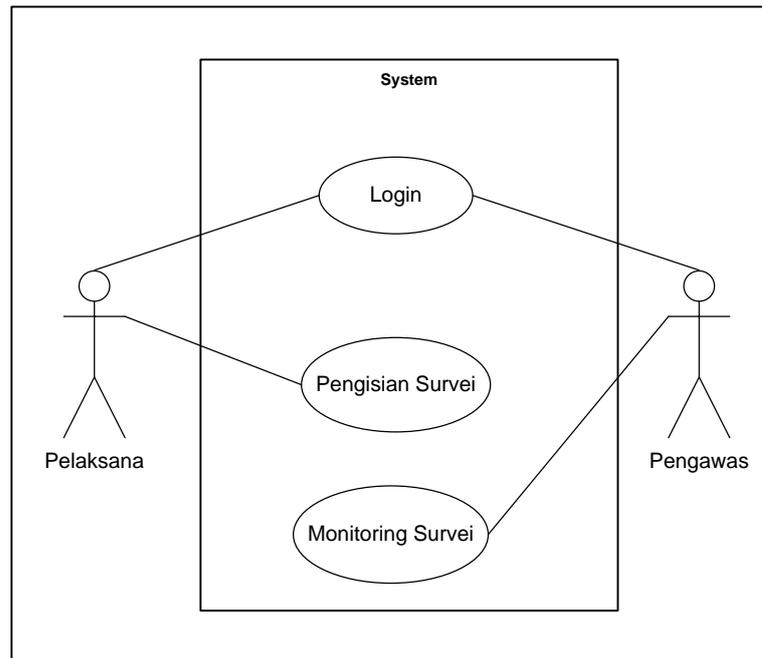
Sejarah UML sendiri terbagi dalam dua fase sebelum dan sesudah munculnya UML. Dalam fase sebelum, UML sebenarnya sudah mulai diperkenalkan sejak tahun 1990, namun notasi yang dikembangkan oleh para ahli analisis dan desain berbeda-beda, sehingga dapat dikatakan belum memiliki standarisasi.

Saat ini sebagian besar para perancang sistem informasi dalam menggambarkan informasi dengan memanfaatkan UML diagram dengan tujuan utama untuk membantu tim proyek berkomunikasi, mengeksplorasi potensi desain, dan memvalidasi rancangan arsitektur perangkat lunak atau pembuat program. Secara filosofi UML, diilhami oleh konsep yang telah ada yaitu konsep pemodelan *Object Oriented* karena konsep ini menganalogikan sistem seperti kehidupan nyata yang didominasi oleh objek dan digambarkan atau dinotasikan dalam simbol-simbol yang cukup spesifik [16].

UML memiliki beberapa diagram antara lain : *use case diagram, class diagram, statechart diagram, activity diagram, sequence diagram, collaboration diagram, component diagram, deployment diagram*. Berikut ini penjelasan untuk beberapa diagram yang akan digunakan dalam penelitian ini [17].

### **2.16.1 Use Case Diagram**

*Use case diagram* merupakan sebuah gambaran fungsionalitas sebuah sistem. Sebuah *use case* merepresentasikan interaksi antara aktor dengan sistem. *Use case* sangat menentukan karakteristik sistem yang sedang dibuat. Seorang/sebuah aktor adalah sebuah entitas manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan-pekerjaan tertentu [17]. Contoh Use Case Diagram pada Gambar 2.9:



**Gambar 2. 9 Contoh Use Case Diagram**

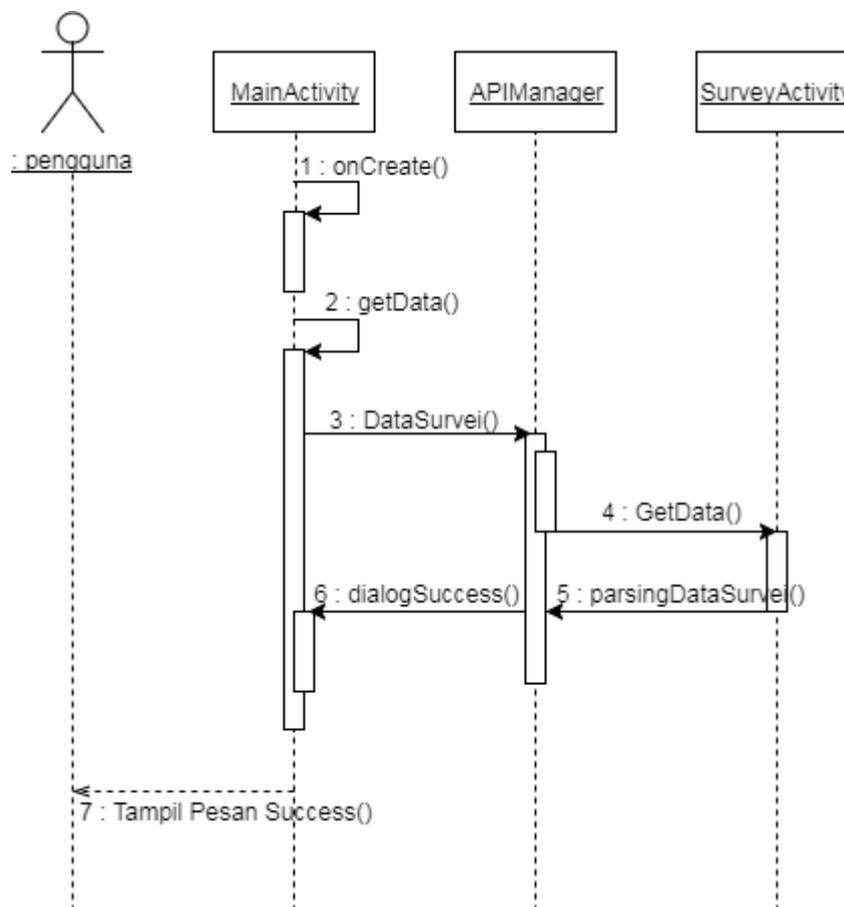
### 2.16.2 Class Diagram

*Class* merupakan inti dari pengembangan dan desain berorientasi objek. *Class* menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metoda/fungsi). *Class diagram* menggambarkan struktur dan deskripsi *class*, *package* dan objek yang berhubungan satu sama lain seperti *containment*, asosiasi, dan lain-lain. Berikut adalah simbol-simbol yang ada pada *class diagram* [17]. Contoh Class Diagram pada Gambar

### 2.16.3 Sequence Diagram

*Sequence diagram* menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, *display*, dan sebagainya) berupa *message* yang digambarkan terhadap waktu. *Sequence diagram* terdiri antar dimensi vertikal (waktu) dan dimensi horizontal (objek-objek yang terkait). *Sequence diagram* biasa digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai respon dari sebuah *event* untuk menghasilkan *output* tertentu. Diawali dari apa yang men-*trigger* aktivitas tersebut, proses dan

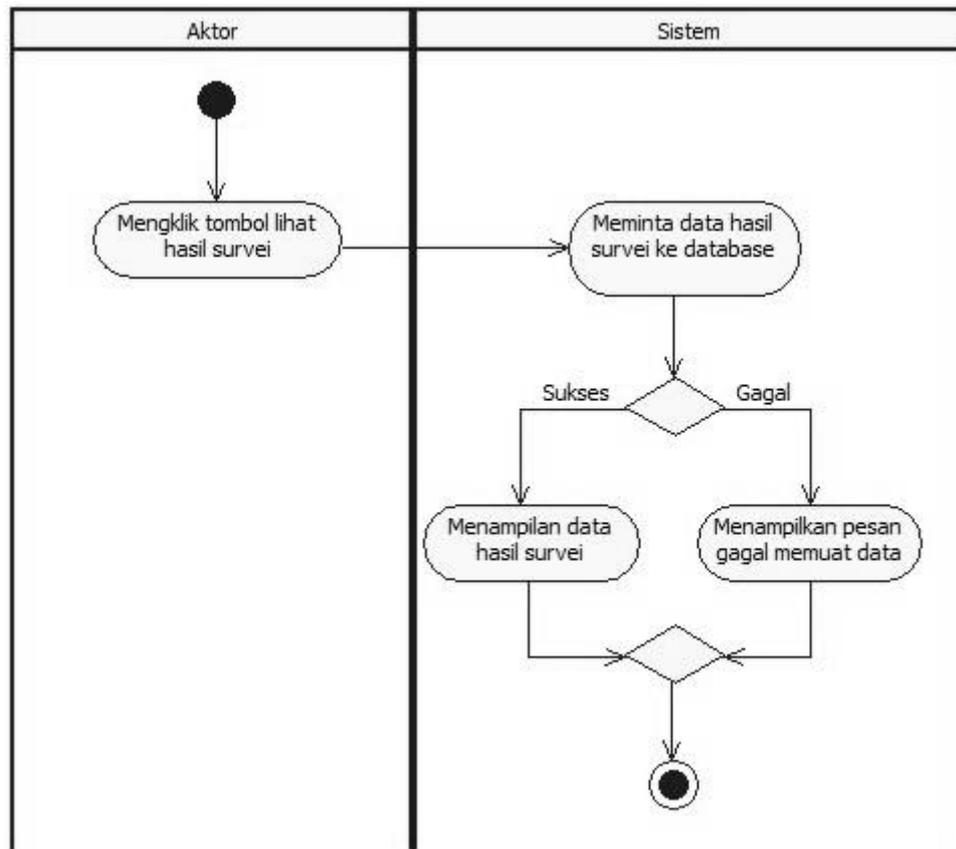
perubahan apa saja yang terjadi secara internal dan *output* apa yang dihasilkan [17]. Contoh Sequence Diagram pada Gambar 2.10



**Gambar 2. 10 Contoh Sequence Diagram**

#### 2.16.4 Activity Diagram

*Activity Diagram* merupakan state diagram khusus, dimana sebagian besar state adalah *action* dan sebagian besar transisi di-*trigger* oleh selesainya state sebelum (*internal processing*). Oleh karena itu *activity diagram* tidak menggambarkan *behavior internal* sebuah sistem (dan interaksi antar subsistem) secara eksak, tetapi lebih menggambarkan proses-proses dan jalur-jalur aktivitas dari level atas secara umum. Sebuah aktivitas menggambarkan proses yang berjalan, sementara *use case* menggambarkan bagaimana aktor menggunakan sistem untuk melakukan aktivitas [17]. Contoh Activity Diagram pada Gambar 2.11:



**Gambar 2. 11 Contoh Activity Diagram**

### 2.17 Data Flow Diagram

Rancangan model logika (*logical model*) dari sistem informasi lebih menjelaskan kepada user bagaimana nantinya fungsi-fungsi di sistem informasi secara logika akan bekerja. Model logika dapat digambar dengan menggunakan diagram arus data (*data flow diagram*). DFD menggambarkan arus data dari suatu sistem informasi, baik sistem lama maupun sistem baru secara logika tanpa mempertimbangkan lingkungan fisik dimana data tersebut berada [18].