

## **BAB 2**

### **TINJAUAN PUSTAKA**

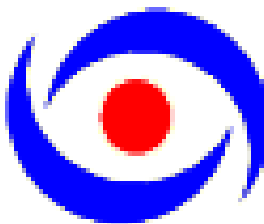
#### **2.1. Profil Perusahaan**

PT. Dinus Cipta Mandiri adalah sebuah perusahaan nasional yang bergerak dibidang distribusi makanan yang berdiri sejak tahun 2000 dan berkantor pusat di Jakarta. Berawal dari menjadi distributor selai merk Budy Jam yang di produksi PT. Bersama Cipta Mandiri (Dinus Group) kemudian berkembang dengan mengimport marshmallow, Candy Toys, dan coklat dari luar negeri dengan merk ChomChomp melalui PT. Catur Global Sukses (Dinus Group).

Saat ini perusahaan telah memiliki kantor cabang di Bandung, Semarang dan Surabaya sebagai jalur disribusi utama dan di dukung oleh provider-provider yang bekerjasama untuk wilayah diluar pulau Jawa. Dengan jangkauan area distribusi yang menyebar luas ke seluruh wilayah, maka produk-produknya dapat mudah diperoleh di supermarket-supermarket dan toko-toko swalayan (modern trade) di kota-kota besar dan kecil serta tersedia disemua jaringan mini market Alfamart, Alfamidi, Indomaret diseluruh Indonesia [1].

#### **2.1.1. Logo Perusahaan**

Dalam perusahaan PT. Dinus Cipta Mandiri terdapat logo perusahaan tersendiri, yang dapat dilihat pada Gambar 2.1 Logo Perusahaan.



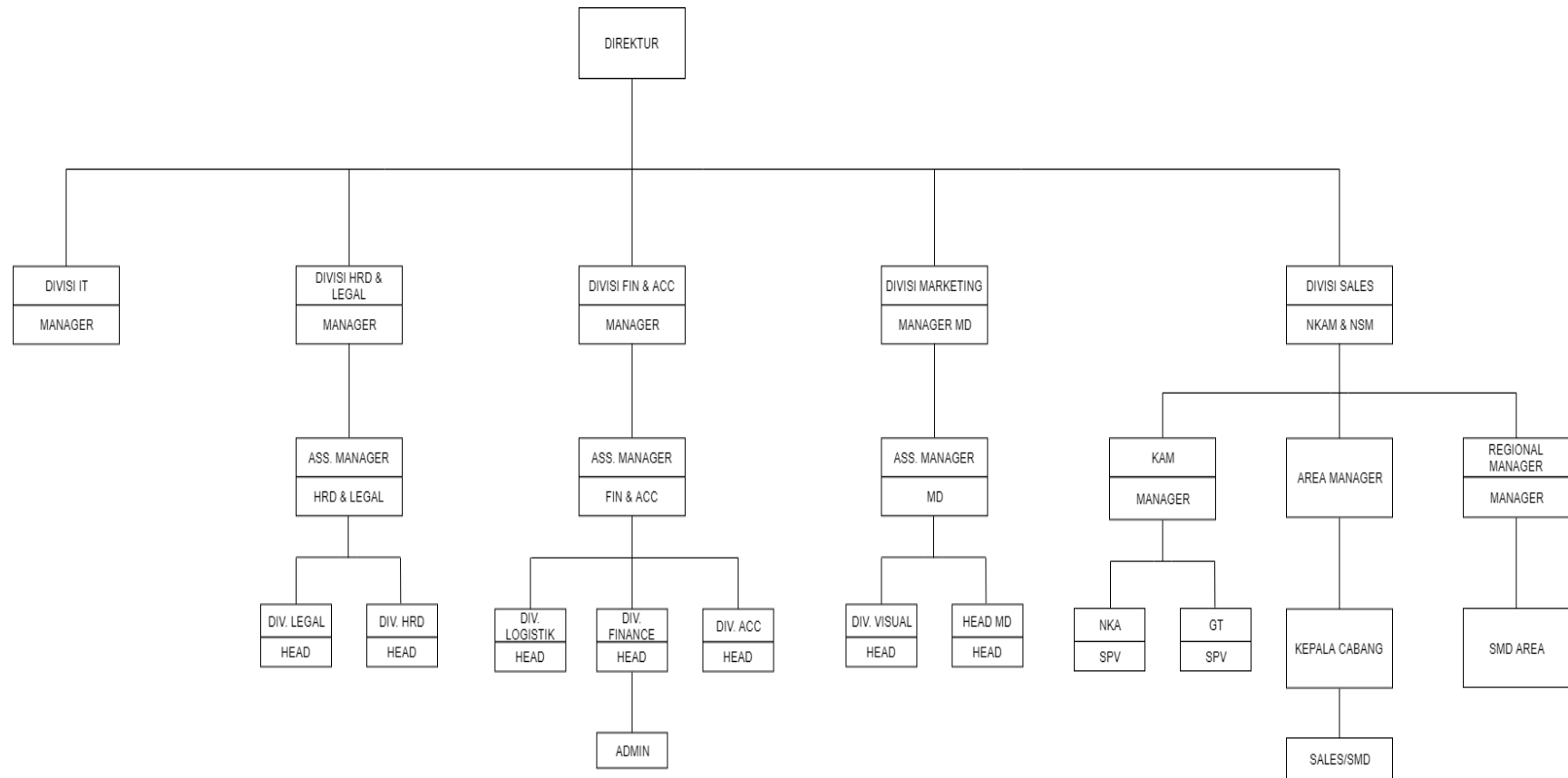
**Gambar 2.1 Logo Perusahaan**

### **2.1.2. Struktur Organisasi**

Suatu instansi dapat berjalan dengan baik sesuai dengan tujuannya jika ditunjang dengan suatu struktur organisasi dan anggaran yang tersusun dengan baik pula. Dengan adanya struktur organisasi yang teratur maka setiap fungsi organisasi dapat berjalan sesuai dengan tugas dan perannya masing-masing. Dari struktur organisasi tersebut maka dengan sendirinya setiap pegawai dapat mengetahui kepada siapa harus mempertanggung jawabkan hasil pekerjaannya. Kata yang tepat dalam pembagian tugas yaitu "*The Right Place*" (menempatkan pegawai pada pekerjaan/posisi yang tepat).

Struktur organisasi tersebut perlu dilengkapi dengan suatu gambaran uraian pekerjaan dan tanggung jawab yang harus dilaksanakan oleh setiap pemegang tanggung jawab sehingga dapat menjadi efektif dan efisien. Tanpa dilengkapi dengan suatu uraian tugas, struktur organisasi tidak dapat melaksanakan tugas dan memenuhi tanggung jawabnya.

Struktur Organisasi pada PT. Dinus Cipta Mandiri terdapat pada Gambar 2.2 Struktur Organisasi.



**Gambar 2.2 Struktur Organisasi**

## **2.2. Pembangunan**

Pembangunan secara umum pada hakekatnya adalah proses perubahan yang terus menerus untuk menuju keadaan yang lebih baik berdasarkan norma-norma tertentu. Mengenai pengertian pembangunan, para ahli memberikan definisi yang bermacam-macam seperti halnya perencanaan. Istilah pembangunan bisa saja diartikan berbeda oleh satu orang dengan orang lain, daerah yang satu dengan daerah lainnya, Negara satu dengan Negara lain. Namun secara umum ada suatu kesepakatan bahwa pembangunan merupakan proses untuk melakukan perubahan [2].

Pengertian tentang pembangunan sebagai “Suatu usaha atau rangkaian usaha pertumbuhan dan perubahan yang berencana dan dilakukan secara sadar oleh suatu bangsa, negara dan pemerintah, menuju modernitas dalam rangka pembinaan bangsa (nation building) [3]. “Suatu proses perubahan ke arah yang lebih baik melalui upaya yang dilakukan secara terencana” [4].

Sedangkan dalam pengertian ekonomi murni, pembangunan adalah suatu usaha proses yang menyebabkan pendapatan perkapita masyarakat meningkat dalam jangka Panjang. Dengan demikian, proses pembangunan terjadi di semua aspek kehidupan masyarakat, ekonomi, sosial, budaya, politik, yang berlangsung pada level makro (nasional) dan mikro. Makna penting dari pembangunan adalah adanya kemajuan/perbaikan (progress), pertumbuhan dan diversifikasi [5].

Sebagaimana dikemukakan oleh para ahli di atas, pembangunan adalah semua proses perubahan yang dilakukan melalui upaya-upaya secara sadar dan terencana [6].

## **2.3. Aplikasi**

Aplikasi adalah penggunaan atau penerapan suatu konsep yang menjadi suatu pokok pembahasan. Aplikasi dapat diartikan juga sebagai program komputer yang dibuat untuk menolong manusia dalam melaksanakan tugas tertentu [7].

Aplikasi software yang dirancang untuk suatu tugas khusus dapat dibedakan menjadi dua jenis, yaitu:

Aplikasi software spesialis, program dengan dokumentasi tergabung yang dirancang untuk menjalankan tugas tertentu.

Aplikasi software paket, suatu program dengan dokumentasi tergabung yang dirancang untuk jenis masalah tertentu.

#### **2.4. Absensi**

Absensi merupakan daftar hadir sekumpulan orang dari suatu kelompok orang banyak yang tergabung dalam sebuah instansi secara resmi yang mempunyai peraturan-peraturan, ketentuan-ketentuan, serta batasan-batasan, dan orang-orang yang terlibat di dalamnya terikat oleh peraturan tersebut. Jika sekelompok orang tersebut melanggarnya maka akan dikenakan sanksi sebagai hukuman dari pelanggaran yang dilakukan oleh orang tersebut sesuai dengan peraturan dan ketentuan yang diterapkan oleh instansi tersebut. Setiap instansi mempunyai peraturan dan ketetapan yang berbeda sesuai perjanjian diantara sekelompok orang tersebut pada saat mereka akan bergabung dalam sebuah instansi tertentu.

Absensi didefinisikan sebagai berikut: “absensi adalah bukti kehadiran pekerja ditempat kerja. Pekerja wajib untuk melaksanakan sendiri 10 pencatatan waktu hadirnya pada mesin pencatat waktu dan tanda tangan baik pada saat masuk bekerja dan pada saat meninggalkan tempat pekerjaan” [8].

#### **2.5. WIFI**

*WiFi* adalah satu standar Wireless Networking tanpa kabel, hanya dengan komponen yang sesuai dapat terkoneksi ke jaringan [9]. Teknologi Wi-Fi adalah bagian atau daerah atau wilayah yang terkoneksi jaringan internet tanpa kabel. Wi-Fi (Wireless Fidelity) adalah istilah populer untuk jaringan wireless (tanpa kabel) dengan frekuensi tinggi. Wireless Network adalah jaringan tanpa kabel yang menggunakan media penghantar gelombang radio atau infrared [10]. Saat ini sudah semakin banyak outlet atau lokasi tertentu yang menyediakan layanan wireless network. Sehingga pengguna dapat dengan mudah melakukan akses internet tanpa kabel. Frekuensi yang digunakan pada radio untuk jaringan komputer biasanya menggunakan frekuensi tinggi yaitu 2,4 GHz dan 5,8 Ghz. Sedangkan penggunaan infrared umumnya hanya terbatas untuk jenis jaringan

yang hanya melibatkan dua buah komputer saja atau disebut *point to point*. Hal ini menyebabkan infrared tidak sepopuler gelombang radio.

Komponen utama teknologi jaringan *Wifi*:

1. *Acces Point*
2. *Wireless LAN Device*
3. *Mobile/Desktop PC*
4. *Ethernet LAN*

Wi-Fi merupakan merek dagang wireless LAN yang diperkenalkan dan distandarisasi oleh Wi-Fi Alliance. Teknologi Wi-Fi dirancang berdasarkan spesifikasi IEEE 802.11. Saat ini ada empat variasi dari 802.11, yaitu:

1. 802.11a
2. 802.11b
3. 802.11g
4. 802.11n

Di banyak bagian dunia, frekuensi yang digunakan oleh Wi-Fi, pengguna tidak diperlukan mendapatkan izin dari pengatur lokal, misalnya Komisi Komunikasi Federal di Amerika Serikat [11].

Seperti telah kita ketahui bahwa jaringan wireless memanfaatkan frekuensi radio untuk mengirimkan data satu host ke host lainnya dan diakses secara bebas. Karena kebebasan itu pulalah yang menyebabkan sistem jaringan wireless memiliki sifat dasar yang tidak aman. Karena itu, muncul beberapa solusi yang bertujuan untuk mengamankan jaringan wireless dalam menjaga informasi. Sistem yang mampu menjaga informasi adalah sistem pengamanan yang memiliki fitur berikut adalah sebagai berikut [12]:

1. *Confidentiality* yaitu system dapat menjamin dan menjaga kerahasiaan dari informasi yang optimal.
2. *Integrity* yaitu system dapat menjamin ketersediaan informasi secara optimal.
3. *Availability* yaitu system dapat menjamin ketersediaan informasi secara optimal.

Seperti halnya Ethernet-LAN (jaringan dengan kabel), jaringan Wi-Fi juga dikonfigurasi ke dalam tiga jenis tipe jaringan Wi-Fi adalah sebagai berikut [13]:

1. Jaringan *Peer to Peer Ad Hoc Wireless LAN*

Komputer dapat saling berhubungan berdasarkan nama SSID (Service Set Identifier). SSID adalah identitas komputer yang memiliki komponen nirkabel.

2. Jaringan *Service Based/ Wireless Infrastructur*

Sistem infrastruktur membutuhkan sebuah komponen khusus yang berfungsi sebagai Access Point.

3. Topologi *Repeater*

Topologi ini biasa digunakan untuk menjangkau wilayah yang tidak dapat dijangkau oleh jaringan kabel, dan wilayah tersebut tidak termasuk ke dalam coverage area sebuah access point.

Keunggulan dan kelemahan Teknologi Jaringan *Wifi* adalah sebagai berikut [14]:

Keunggulan:

1. Biaya pemeliharaan murah
2. Infrastruktur berdimensi kecil
3. Pembangunannya cepat
4. Mudah dan murah untuk direlokasi
5. Mendukung portabilitas

Kelemahan:

1. Biaya perawatan mahal
2. *Delay* yang sangat besar
3. Kesulitan karena masalah propagasi radio
4. Mudah untuk terinterferensi
5. Kapasitas jaringan kecil
6. Keamanan/kerahasiaan data kurang terjamin

Manfaat yang dapat kita peroleh apabila menggunakan Jaringan Wi-Fi adalah efisiensi kerja meningkat dan memungkinkan kita untuk memakai satu

printer yang terhubung dengan jaringan secara bersama-sama dalam area jaringan [15]. Selain itu manfaat Jaringan Nirkabel (Jaringan Wi-Fi) adalah dimana orang-orang diseluruh dunia memahami manfaat konektivitas jaringan untuk memeriksa e-mail, menjelajah internet, dan mengakses aplikasi korporat [15]. Kemajuan produk selanjutnya yang mencakup alat penghubung nirkabel memungkinkan orang-orang tersebut untuk tidak terikat serta dapat memetik manfaat atas mobilitas dan fleksibilitas. Hasilnya adalah efisiensi, akurasi, dan realibilitas yang lebih tinggi. Beberapa alasan memilih jaringan wireless dibandingkan jaringan kabel antara lain [16]:

1. Jaringan *wireless* lebih bersifat *mobile*, kita dapat mengakses *resource* dari manapun dan dapat dilakukan secara berpindah-pindah, terhindar dari masalah-masalah yang ditimbulkan oleh kabel.
2. Perangkat wireless saat ini sudah relatif murah dan cepat, sehingga bisa mengimbangi atau menyaingi kemampuan teknologi kabel.

Selain beberapa keuntungan yang diperoleh dengan menggunakan teknologi wireless, tentunya teknologi wireless masih memiliki beberapa kekurangan yang perlu diperhatikan ketika mengimplementasikan teknologi wireless tersebut. Beberapa permasalahan yang muncul ketika kita menggunakan teknologi wireless antara lain:

1. Tingkat keadaan (*reability*) dari teknologi *wireless* saat ini masih belum sebaik teknologi kabel.
2. Pengiriman data melalui jaringan wireless menggunakan media radio frekuensi, media tersebut dapat diakses secara bebas oleh siapapun. Akibatnya data yang dikirim melalui media tersebut kurang begitu aman. Beberapa solusi pun muncul untuk melakukan pengamanan terhadap data yang dikirimkan melalui media radio frekuensi. Banyaknya solusi pengamanan yang muncul pun membuat para pengguna bingung memilih solusi yang paling optimal.

Walaupun masih memiliki beberapa kekurangan, teknologi *wireless* ini sudah banyak diimplementasikan dalam kehidupan sehari-hari, antara lain:



1. Penggunaan teknologi Bluetooth, biasa digunakan untuk komunikasi *wireless* jarak dekat (*personal area network*).
2. Implementasi *wireless* di rumah atau kantor, dengan teknologi *wireless LAN* (WLAN) akan sangat memudahkan untuk membangun jaringan LAN di rumah ataupun di kantor. Dengan adanya teknologi *wireless LAN* ini pun memunculkan banyak hotspot (sebuah wilayah terbatas yang dilayani oleh satu/beberapa *access point*), sehingga kita dapat mengakses resource secara bebas, mudah, dan *mobile*, Hotspot banyak di area *public cafe*, hotel, *mall*, kampus.
3. Teknologi *wireless* pun banyak digunakan untuk menghubungkan antara dua jaringan yang terdapat di antara dua gedung. Bagi para ISP, teknologi *wireless* ini sangat menguntungkan. Mereka dapat menghubungkan jaringan yang dimiliki oleh pelanggannya dengan jaringan ISP melalui jaringan *wireless*. Selain itu, ditinjau dari sudut biaya relatif murah jika dibandingkan dengan menggunakan kabel.

## 2.6. GPS

*Global Positioning System* (GPS) adalah sistem satelit navigasi dan penentuan posisi yang dimiliki dan dikelola oleh Amerika Serikat. Sistem ini didesain untuk memberikan posisi dan kecepatan tiga-dimensi serta informasi mengenai waktu, secara kontinyu diseluruh dunia tanpa bergantung waktu dan cuaca, bagi banyak orang secara simultan [17].

Cara kerja GPS secara logic ada 5 langkah [18]:

1. Memakai perhitungan "*Triangulation*" dari satelit.
2. Untuk perhitungan "*Triangulation*", GPS mengukur jarak menggunakan *travel time* sinyal radio.
3. Untuk mengukur *travel time*, GPS memerlukan akurasi waktu yang tinggi.
4. Untuk perhitungan jarak, kita harus tahu dengan pasti posisi satelit dan ketinggian pada orbitnya.
5. Terakhir harus mengoreksi delay sinyal waktu perjalanan di atmosfer sampai diterima *receiver*.

## 2.7. Android

Android adalah sistem operasi mobile yang berbasis open source linux kernel yang awalnya dibuat oleh Android Inc. Android termasuk sistem operasi yang dirancang untuk digunakan secara optimal dalam lingkungan mobile yang fleksibel [19].

Android adalah sistem operasi open source berbasis Linux. Pada awalnya, android hanya untuk ponsel, tapi sekarang dapat digunakan pada tablet, TV, komputer, dan stereo mobile [20].

### 2.7.1. Arsitektur Android

Secara umum arsitektur android terdiri dari application, application framework, libraries, android runtime dan linux kernel. Berikut ini penjelasan dari arsitektur android [21]:

#### 1. *Aplication*

*Application* adalah layer dimana kita berhubungan dengan aplikasi saja, dimana biasanya kita mengunduh aplikasi kemudian kita melakukan instalasi dan menjalankan aplikasi tersebut. Di layer ini terdapat aplikasi inti termasuk email, SMS, Kalender, peta, browser, kontak, dan lain-lain. Semua aplikasi ditulis menggunakan bahasa pemrograman Java.

#### 2. *Aplication Framework*

*Applications framework* adalah layer dimana pembuat aplikasi melakukan pengembangan/pembuatan aplikasi yang akan dijalankan di sistem operasi android, karena pada layer inilah aplikasi dapat dirancang dan dibuat, seperti content-providers yang berupa sms dan panggilan telepon.

Komponen-komponen yang termasuk dalam applications framework adalah sebagai berikut:

- a. *View System*
- b. *Content Provider*
- c. *Telephone Manager*
- d. *Location Manager*
- e. *Resource Manger*
- f. *Notification Manager*

g. *Activity Manager*

### 3. *Libraries*

*Libraries* adalah layer dimana fitur-fitur android berada, biasanya para pengembang aplikasi mengakses *libraries* untuk menjalankan aplikasinya. Layer ini meliputi berbagai library C/C++ inti seperti Libc dan SSL, serta:

- a. *Libraries* media untuk pemutaran media audio dan video.
- b. *Libraries* untuk manajemen tampilan.
- c. *Libraries* Graphics mencakup SGL dan OpenGL untuk grafis 2D dan 3D.
- d. *Libraries* SQLite untuk dukungan basis data.
- e. *Libraries* SSL dan WebKit terintegrasi dengan *web browser* dan *security*.
- f. *Libraries LiveWebcore* mencakup modern *web browser* dengan *engine embedded web view*.
- g. *Libraries* 3D yang mencakup implementasi OpenGL ES 1.0 API's.

### 4. *Android Runtime*

Layer ini merupakan layer yang memungkinkan aplikasi android dapat dijalankan dimana dalam prosesnya menggunakan implementasi Linux. Dalvik Virtual Machine (DVM) merupakan mesin yang membentuk dasar kerangka aplikasi android. Di dalam android runtime dibagi menjadi dua bagian, yaitu:

- a. *Core Libraries*: aplikasi android dibangun dalam bahasa java, sementara dalvik sebagai virtual mesinnya, sehingga diperlukan sebuah *libraries* yang berfungsi untuk menerjemahkan bahasa Java/C yang ditangani oleh *core libraries*.
- b. *Dalvik Virtual Machine*: virtual mesin berbasis register yang dioptimalkan untuk menjalankan fungsi-fungsi secara efisien, dimana merupakan pengembangan yang mampu membuat linux kernel untuk melakukan *threading* dan manajemen tingkat rendah.

### 5. *Linux Kernel*

Linux Kernel adalah layer inti dari sistem operasi android. Layer ini berisi *file-file* sistem yang mengatur sistem *processing*, *memory*, *resource*, *driver*, dan sistem-sistem operasi android yang lainnya.

## 2.8. 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 [22].

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 sebagiannya.
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 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[23].

## **2.9. MAC Address**

*MAC address (Media Access Control Address)* adalah sebuah alamat jaringan yang diimplementasikan pada lapisan data-link dalam tujuh lapisan model OSI, yang merepresentasikan sebuah node tertentu dalam jaringan. Dalam sebuah jaringan berbasis Ethernet, MAC address merupakan alamat yang unik yang memiliki panjang 48-bit (6 byte) yang mengidentifikasi sebuah komputer, interface dalam sebuah router, atau node lainnya dalam jaringan. *MAC Address* juga sering disebut sebagai *Ethernet address*, *physical address*, atau *hardware address* [24].

## 2.10. MAC Address Filtering

MAC Address Filtering merupakan metode filtering untuk membatasi hak akses dari MAC Address yang bersangkutan. Hampir setiap wireless access point maupun router difasilitasi dengan keamanan MAC Filtering. MAC filters ini juga merupakan metode sistem keamanan yang baik dalam WLAN, karena peka terhadap jenis gangguan seperti pencurian pc card dalam MAC filter dari suatu *access point sniffing* terhadap WLAN.

Fitur *MAC Address Filter* ini berfungsi untuk membantu anda untuk mencegah pengguna asing (tidak diinginkan) yang berniat untuk mengakses masuk ke jaringan router nirkabel anda. Dengan menerapkan fitur ini, maka hanya perangkat nirkabel yang memiliki alamat MAC yang telah terdaftar (ditetapkan) saja yang dapat memperoleh akses ke router nirkabel.

Wireless LAN dapat memfilter berdasarkan *MAC address* dari station/client, hampir semua *access point* mempunyai kemampuan untuk memfilter berdasarkan *MAC address*. Administrator jaringan dapat mengompilasi, mendistribusikan, dan memelihara daftar *MAC address* yang diizinkan.

Dan memprogram masing-masing *access point*, jika sebuah *PC card* atau client lain dengan sebuah *MAC address* yang tidak terdaftar mencoba untuk mengakses wifi, kemampuan *MAC address filtering* tidak akan mengizinkan client berhubungan dengan *access point* [25].

## 2.11. 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 [26].

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 sistem 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 (sistem perangkat lunak, sistem informasi, atau sistem lainnya). Pendekatan berorientasi objek akan memandang sistem 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 sistem. Saat mengabstraksikan dan memodelkan objek, data dan proses-proses yang dimiliki 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 sistem yang dibangun dengan berdasarkan metode berorientasi objek adalah sebuah sistem yang komponennya dibungkus (dienkapsulasi) menjadi kelompok data dan fungsi. Setiap komponen dalam sistem 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.

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.

### 3. Metode (*method*)

Operasi atau metode atau method pada sebuah kelas hamper 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 privet 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.

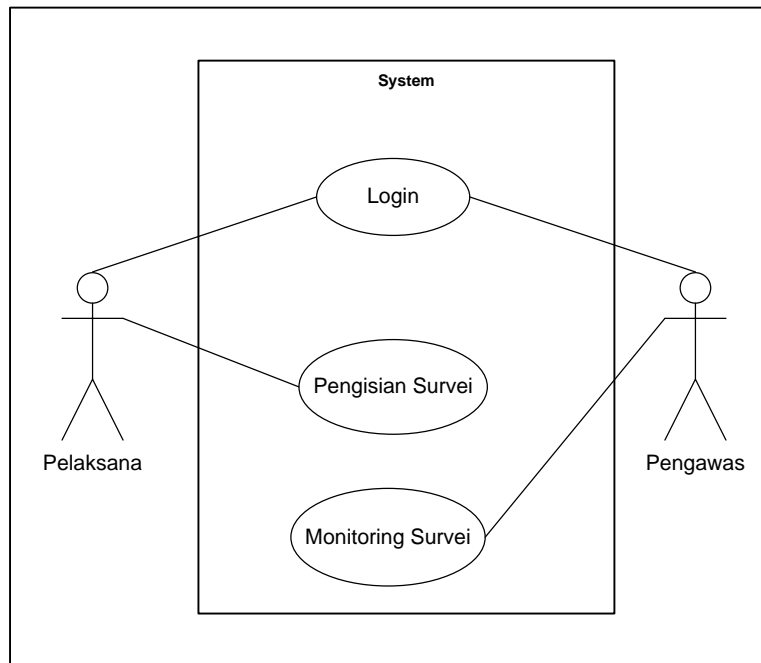
## 2.12. UML (Unified Modeling Language)

UML adalah keluarga notasi grafis yang didukung oleh meta-model tunggal yang membantu pendeskripsian dan desain sistem perangkat lunak, khususnya sistem yang dibangun menggunakan pemrograman berorientasi objek.

UML dikeluarkan oleh OMG (*Object Management Group, Inc*) yaitu organisasi internasional yang dibentuk pada 1989, terdiri dari perusahaan sistem informasi, *software developer* dan para user sistem komputer [27].

## 2.13. Use Case Diagram

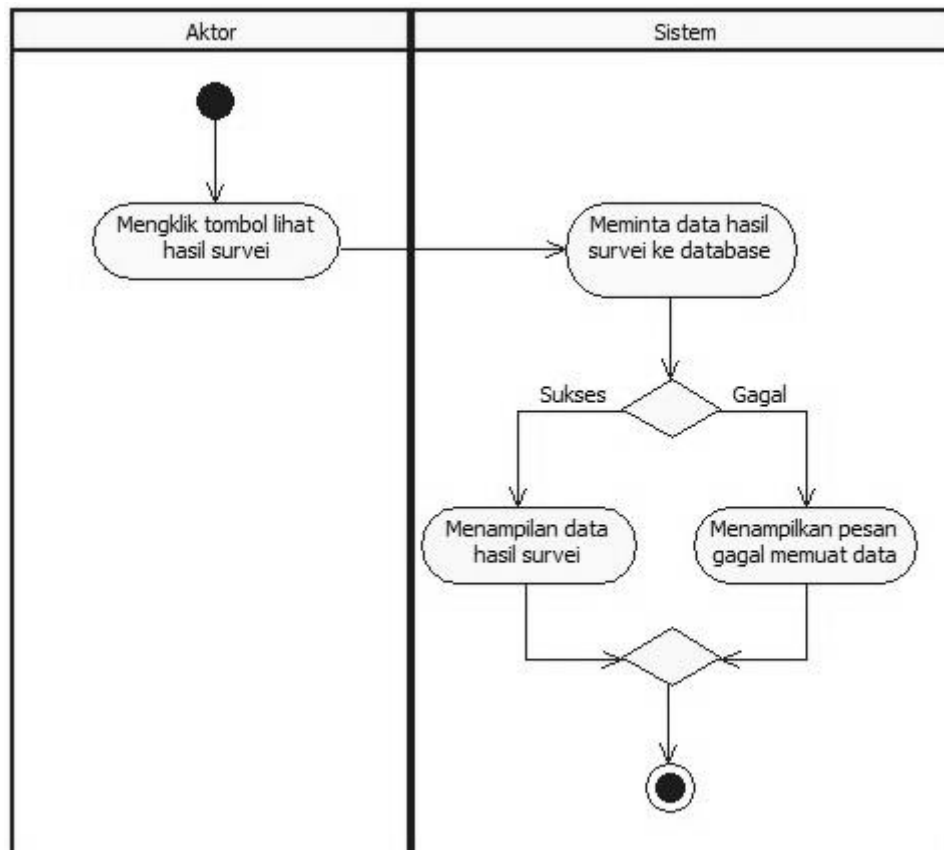
*Use case diagram* merupakan sebuah gambaran fungsionalitas sebuah sistem. Sebuah *use case* merepresentasikan interaksi antara aktor dengan sistem. *Use case* sangat menentukan karakteristik sistem yang sedang dibuat. Seorang/sebuah aktor adalah sebuah entitas manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan-pekerjaan tertentu [28]. Contoh *Use Case Diagram* dapat dilihat pada Gambar 2.3 Contoh *Use Case Diagram*.



**Gambar 2.3 Contoh Use Case Diagram**

#### 2.14. 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 [28]. Contoh *Activity Diagram* dapat dilihat pada Gambar 2.4 Contoh *Activity Diagram*.



**Gambar 2.4 Contoh Activity Diagram**

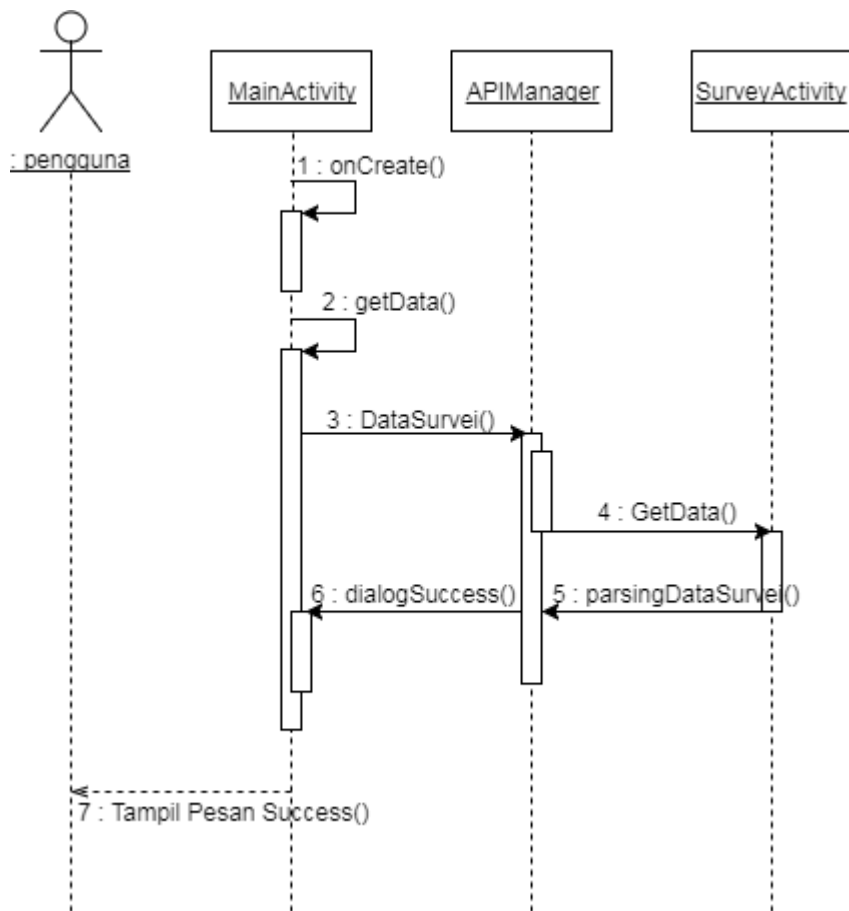
### 2.15. Class Diagram

*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* [28].

### 2.16. 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* dapat di lihat pada Gambar 2.5 Contoh *Sequence Diagram*.



**Gambar 2.5** Contoh *Sequence Diagram*

## 2.17. Apache

Apache merupakan web server yang dapat di jalankan di berbagai sistem operasi yang berbeda seperti UNIX, BSD, Linux, Windows dan Novell Netware yang bertugas untuk melayani fasilitas web dengan menggunakan *protocol* HTTP.

Apache sendiri bersifat *open source*, yang artinya dapat digunakan secara bebas oleh semua pengguna. Apache dikenal memiliki fitur-fitur canggih seperti penanganan pesan kesalahan yang dapat di konfigurasi serta otentikasi berbasis data dan memiliki tampilan GUI yang memungkinkan penangan server menjadi mudah.

*Web Server* Apache akan bekerja saat pengguna mengetikkan *request* melalui protokol `http://` untuk membuat suatu halaman. Apache akan menjawab permintaan yang diketikkan pengguna dan kemudian menampilkan halaman yang diminta. Dalam hal ini Apache punya peranan penting untuk menjalankan sebuah *request* halaman bahasa pemrograman.

## 2.18. XAMPP

*Web server* merupakan sebuah bentuk server yang khusus digunakan untuk menyimpan halaman *website* atau *homepage*. Apache merupakan turunan dari *web server* yang dikeluarkan oleh NSCA yaitu NSCA HTTPd sekitar tahun 1995. Pada dasarnya, Apache adalah “APatCHy” (*patch*) dan pengganti dari NCSA HTTPd. Apache web server merupakan tulang punggung permintaan dari *client* yang menggunakan *browser*, seperti Netscape Navigator, Internet Explorer, Mozilla, lynx dan lain-lain. *Web Server* dalam berkomunikasi dengan kliennya menggunakan protokol HTTP (*Hyper Text Transfer Protocol*). Apache berada di bawah GNU, *General Public License* yang bersifat *free* sehingga Apache dapat di *download* gratis pada alamat `http://www.apache.org`. Adapun pertimbangan dalam memilih Apache adalah:

- Apache termasuk dalam kategori *free software* (software gratis).

- Instalasi apache sangat mudah.

- Mampu beroperasi pada banyak *platform* sistem operasi, seperti Linux, Windows dan lain-lain.

Apache Web Server merupakan *web server* yang bersifat *open source* dan mempunyai *performance* yang sangat bagus, fleksibel dan mendukung berbagai macam *platform* sistem operasi seperti Windows NT/9x, UNIX, Netware 5x, OS/2 dan berbagai macam sistem operasi lainnya. Apache sangat cepat sekali mengeluarkan *update* terbarunya, sehingga mengurangi munculnya *bugs* dan kelemahan program.

## 2.19. Hyper Text Markup Language (HTML)

HTML adalah bahasa standar penulisan dokumen web. Semua informasi yang akan diletakkan diweb menggunakan format penulisan HTML. *File* HTML adalah *file* teks yang dilengkapi simbol-simbol untuk keperluan *display*. Simbol-

simbol tadi disebut *tag*. HTML kependekan dari *Hyper Text Markup Language*. Dokumen HTML adalah *file* teks murni yang dapat dibuat dengan *editor* teks sembarang. Dokumen ini dikenal sebagai *webpage*. Dokumen HTML merupakan dokumen yang disajikan dalam *browser web surfer*. Dokumen ini umumnya berisi informasi ataupun *interface* aplikasi didalam internet.

Dokumen HTML disusun oleh elemen-elemen. Elemen merupakan istilah bagi komponen-komponen dasar pembentuk dokumen pembentuk HTML. Beberapa contoh HTML adalah: *head*, *body*, *table*, *paragraph*, dan *list*. Untuk menandai berbagai elemen dalam suatu dokumen HTML, dapat menggunakan *tag*. *Tag* HTML terdiri atas sebuah kurung sudut kiri (<, Tanda lebih kecil), sebuah nama *tag*, dan sebuah kurung sudut kanan (>, tanda lebih besar). *Tag* umumnya berpasangan (misalnya <H1> dengan </H1>, *tag* yang berpasangan selalu diawali dengan karakter garis miring (/). *Tag-tag* yang pertama menunjukkan *tag* awal yang berarti awal elemen, dan yang kedua menunjukkan tag akhir, berarti akhir elemen.

Elemen yang dibutuhkan untuk membuat suatu dokumen HTML dinyatakan dengan *tag*<html>,<head>, dan <body> berikut *tag-tag* pasangannya. Setiap dokumen terdiri atas *tag head* dan *body*. Elemen *head* berisi informasi tentang dokumen tersebut, dan elemen *body* berisi tentang teks yang sebenarnya yang tersusun dari link, grafik, paragraf, dan elemen lainnya.

## **2.20. Cascading Style Sheet (CSS)**

*Cascading Style Sheet* (CSS) adalah suatu bahasa *stylesheet* yang digunakan untuk mengatur tampilan suatu dokumen yang ditulis dalam bahasa *markup*. Penggunaan yang paling umum dari CSS adalah untuk memformat halaman web yang ditulis dengan HTML dan XHTML. Walaupun demikian, bahasanya sendiri dapat dipergunakan untuk semua jenis dokumen XML termasuk SVG dan XUL. Spesifikasi CSS diatur oleh *World Wide Web Consortium* (W3C).

## **2.21. Hypertext Preprocessor (PHP)**

Menurut *Firdaus* (2007:18) PHP (*Hypertext Preprocessor*) merupakan bahasa yang hanya dapat berjalan pada server yang hasilnya dapat ditampilkan pada klien. Dalam mengeksekusi kode PHP pada sisi server (disebut *server side*)



berbeda dengan mesin maya Java yang mengeksekusi program pada sisi klien (*client side*). Proses eksekusi kode PHP yang disisipkan pada halaman HTML.

PHP merupakan bahasa standar yang digunakan dalam dunia *website*. PHP adalah bahasa pemrograman yang berbentuk *script* yang diletakan didalam server web. Sekitar tahun 1994, Rasmus Lerdorf telah meletakan bersama *Perl script* untuk membuat siapa yang telah melihat resumenya terkesan. Kemudian sedikit demi sedikit *user* mulai menyukai *script* ini.

### **2.22. CodeIgniter**

*CodeIgniter* merupakan aplikasi *open source* yang berupa *framework* (kerangka kerja) PHP dengan model MVC (*Model, View, Controller*) untuk membangun website dinamis dengan menggunakan PHP. CodeIgniter memudahkan pengembang web untuk membuat aplikasi web dengan cepat mudah dibandingkan dengan membuatnya dari awal. CodeIgniter dirilis pertama kali pada 28 Februari 2006.

### **2.23. Android Studio**

Android studio merupakan *integrated development environment* (IDE) atau dalam artian lain adalah sebuah lingkungan pengembangan terintegrasi resmi yang memang dirancang khusus untuk pengembangan sistem operasi Google Android. Aplikasi yang satu ini dibangun di atas sebuah perangkat lunak yang dinamakan IntelliJ IDEA milik JetBrains. Bisa juga dibilang bahwa Android Studio merupakan pengganti dari Eclipse Android Development Tools atau ADT sebagai IDE utama dalam pengembangan aplikasi android yang asli [29].

### **2.24. 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[30]:

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.

## **2.25. Java**

Java adalah bahasa pemrograman berorientasi objek yang serbaguna. Bahasa ini menyediakan sejumlah perluasan yang mendukung pengembangan aplikasi GUI (antarmuka kepada pemakai yang berbentuk grafis). Bahasa pemrograman java dapat berjalan pada sembarang platform yang memiliki interpreter dan lingkungan runtime. Walaupun java adalah bahasa penerjemah, program java harus dikompilasi terlebih dulu. Kompiler java (*javac*) mengubah kode sumber program menjadi bytecode yang dapat dieksekusi dalam lingkungan runtime java.

Java merupakan bahasa pemrograman yang bersifat lintas platform yang dimana bahasa ini dapat dipakai untuk menyusun program pada berbagai sistem operasi [31].

Java adalah bahasa pemrograman berorientasi objek yang serbaguna yang menyediakan sejumlah perluasan yang mendukung pengembangan aplikasi GUI (Graphical User Interface), pengembangan aplikasi client/server dan aplikasi internet [32].

Java adalah bahasa pemrograman berorientasi objek yang dibuat oleh Sun Microsystems yang merevolusi pemrograman aplikasi untuk world wide web dan perusahaan intranet dan extranet. Java mirip dengan bahasa C++ dan bahasa pemrograman Objective C. Tetapi bahasa java jauh lebih sederhana, lebih aman dan multiplatform [33].

#### **2.26. My SQL**

MySQL atau yang biasa dibaca “mai-se-kuel” adalah sebuah program pembuat *database* yang bersifat *open source*, yang artinya siapa saja boleh menggunakannya. MySQL sebenarnya produk yang berjalan pada *platform* linux, karena sifatnya yang *open source*. MySQL dapat dijalankan pada semua *platform* baik windows maupun linux, MySQL juga merupakan program pengakses *database* yang bersifat jaringan sehingga dapat digunakan untuk aplikasi *multi user* (banyak pengguna).

Kelebihan dari MySQL adalah dengan menggunakan bahasa *query* standar yang dimiliki SQL (*Structure Query Language*). SQL adalah suatu Bahasa permintaan yang terstruktur yang telah distandarkan untuk semua program pengakses *database* seperti Oracle, Postgre SQL, SQL Server, dan lain-lain.

#### **2.27. Pengumpulan Data**

Untuk memperoleh data-data yang peneliti perlukan dan dianggap relevan dengan masalah yang peneliti teliti. Teknik pengumpulan data merupakan langkah yang paling strategis dalam penelitian, karena tujuan utama dari penelitian adalah mendapatkan data [34]. Tanpa mengetahui teknik pengumpulan data, maka peneliti tidak akan mendapatkan data yang memenuhi standar data yang

ditetapkan. Pengumpulan data adalah prosedur yang sistematis dan standar untuk memperoleh data yang diperlukan[34].

Dari penjelasan tersebut maka dalam penelitian ini menggunakan teknik pengumpulan data yang digunakan dalam penelitian ini adalah sebagai berikut:

1. Observasi

Observasi adalah teknik pengumpulan data dengan observasi digunakan bila, penelitian berkenaan dengan perilaku manusia, proses kerja, gejala-gejala alam dan bila responden yang diamati tidak terlalu besar. Data yang akan terkumpul dari observasi berupa informasi keaktifan belajar siswa yang di dapat melalui pengamatan peneliti ketika melihat secara langsung bagaimana proses belajar mengajar pelajaran ekonomi berlangsung di sekolah[34].

2. Kuesioner

Kuesioner merupakan teknik pengumpulan data yang dilakukan dengan cara memberi seperangkat pertanyaan dan pernyataan tertulis kepada responden untuk dijawabnya. Skala yang digunakan dalam penelitian ini adalah skala likert. Skala likert adalah skala yang digunakan untuk mengukur sikap, pendapat, dan persepsi seorang atau sekelompok orang tentang fenomena social[34].

3. Studi Literatur

Studi literatur merupakan teknik pengumpulan data dengan mengadakan studi penelaah terhadap buku-buku, literaturliteratur, catatan-catatan, dan laporan-laporan yang ada hubungannya dengan masalah yang dipecahkan. Teknik ini digunakan untuk memperoleh dasar-dasar dan pendapat secara tertulis yang dilakukan dengan cara mempelajari berbagai literatur yang berhubungan dengan masalah yang diteliti. Hal ini juga dilakukan untuk mendapatkan data sekunder yang akan digunakan sebagai landasan perbandingan antara teori dengan prakteknya di lapangan. Data sekunder melalui metode ini diperoleh dengan browsing di internet, membaca berbagai literatur, hasil kajian dari peneliti terdahulu, catatan perkuliahan, serta sumber-sumber lain yang relevan[34].

## **2.28. Pengujian Sistem**

Pengujian sistem merupakan proses mengeksekusi sistem perangkat lunak untuk menentukan apakah sistem perangkat lunak tersebut cocok dengan spesifikasi sistem dan berjalan sesuai dengan lingkungan yang diinginkan [34].

### **2.28.1. Pengujian *Black-Box***

Pengujian *black-box* berfokus pada persyaratan fungsional perangkat lunak. Dengan demikian, pengujian *black-box* memungkinkan perancang perangkat lunak mendapatkan serangkaian kondisi input yang sepenuhnya menggunakan semua persyaratan fungsional untuk program. Pengujian *black-box* berusaha menemukan kesalahan dalam kategori sebagai berikut:

1. Fungsi-fungsi yang tidak benar atau hilang
2. Kesalahan interface
3. Kesalahan dalam struktur data atau akses database *eksternal*
4. Kesalahan kinerja
5. Inisialisasi dan kesalahan terminasi

Pengujian *black-box* cenderung diaplikasikan selama tahap akhir pengujian, karena pengujian *black-box* memperhatikan struktur kontrol, maka perhatian berfokus pada domain informasi.

### **2.28.2. Pengujian Beta**

Pengujian beta juga dikenal sebagai pengujian pengguna berlangsung di lokasi pengguna akhir oleh pengguna akhir untuk memvalidasi kegunaan, fungsi, kompatibilitas, dan uji reliabilitas dari software yang dibuat. Aktifitas pengujian beta menambah nilai siklus hidup pengembangan produk karena memungkinkan kesempatan user untuk memberikan masukan ke dalam desain, fungsi, dan kegunaan dari produk. Masukan ini tidak hanya penting untuk keberhasilan produk tetapi juga investasi ke produk masa depan Hal ini juga dikenal sebagai uji lapangan. Ini terjadi di lokasi user. Ini mengirimkan sistem untuk pengguna yang menginstal dan menggunakannya di bawah kondisi kerja dunia nyata.

Tes beta merupakan tahap kedua dari pengujian perangkat lunak di mana pengguna mencoba produk. Tahap pertama meliputi pengujian unit, pengujian

komponen, dan pengujian sistem. Pengujian beta dapat dianggap “pengujian pra-rilis artinya sebelum produk tersebut dilempar ke pasaran maka harus dipastikan dari sisi pelanggan bahwa perangkat lunak tersebut terbebas dari cacat atau kegagalan.

Tujuan dari pengujian beta adalah untuk menempatkan aplikasi di tangan pengguna yang sebenarnya yang berada di luar tim teknik untuk menemukan setiap kekurangan atau masalah dari perspektif pengguna akhir.

## 2.29. State Of Art

Pada *State of Art* ini, diambil beberapa contoh penelitian terlebih dahulu sebagai panduan ataupun contoh untuk penelitian yang dilakukan yang nantinya akan menjadi acuan dan perbandingan dalam melakukan penelitian ini.

Berikut adalah *state of art* pertama yang dapat dilihat pada Tabel 2.1 *State Of Art 1*:

**Tabel 2.1 State Of Art 1**

Judul Penelitian	Aplikasi Absensi Menggunakan Metode Lock GPS Dengan Android di PT.PLN (Persero) APP Malang Basecamp Mojokerto
Penulis	Ronny Makhfuddin Akbar & Nanu Prabowo
Dipublikasikan	Majalah Techno Agustus 2015 ISSN: 2081-9210
Hasil Penelitian	Penelitian ini dilakukan di PT.PLN (Persero) APP Malang <i>Basecamp</i> Mojokerto yang bergerak dibidang jasa penyedia jaringan transmisi listrik. Sistem absensi yang berjalan sekarang yang menggunakan sistem absensi sidik jari. Dalam melaksanakan kewajiban pencatatan kehadiran, pegawai dituntut untuk menghampiri alat absensi yang diletakan di lobi perusahaan. Jika ada jadwal pekerjaan di suatu gardu induk (sebutan untuk gardu listrik tegangan tinggi), pegawai harus datang ke kantor terlebih dahulu untuk melakukan absensi. Namun apabila lokasi gardu induk dari rumah lebih dekat, maka kebanyakan pegawai akan langsung menuju lokasi pekerjaan dari pada ke kantor terlebih dahulu. Karena dirasa kurang efektif, pada penelitian ini dibuat aplikasi agar dapat melakukan absensi tanpa harus mendatangi mesin absensi dengan memanfaatkan ponsel android yang berbekal fitur GPS dan WIFI sebagai alat untuk melakukan absensi kehadiran langsung dari meja atau Gardu Induk lokasi pekerjaan.
Persamaan	Persamaannya yaitu terdapat tujuan yang sama, membuat sebuah sistem atau aplikasi absensi memanfaatkan WIFI.

Perbedaan	Perbedaanya yaitu pada penelitian sebelumnya membahas tentang pembangunan aplikasi menggunakan WIFI dengan menggunakan metode <i>lock</i> GPS.
-----------	--

Berikut adalah *state of art* kedua yang dapat dilihat pada Tabel 2.2 *State Of Art 2*:

**Tabel 2.2 State Of Art 2**

Judul Penelitian	Pemanfaatan <i>Mac Adres</i> Hotspot Dalam Pengembangan Sistem Absensi GPS Dalam Rangka Meningkatkan Keakuratan Posisi Pengguna
Penulis	Fransiskus Adikira
Dipublikasikan	Jurnal SISFO: Inspirasi Profesional Sistem Informasi Vol 5 No 4
Hasil Penelitian	Kendala terjadi bagi perusahaan yang mempunyai tempat kerja pada gedung bertingkat, jika menggunakan absensi hanya berbasis GPS, maka GPS tidak akan dapat memberikan lokasi tepatnya dimana karyawan tersebut berada. Melanjutkan penelitian sebelumnya mengenai perancangan sistem absensi berbasis GPS (Adikara, 2013), maka akan dilakukan pengembangan sistem tersebut dengan pemanfaat teknologi lainnya sebagai penanda letak dari karyawan tersebut di dalam gedung bertingkat, yaitu dengan menggunakan fasilitas <i>WIFI Router</i> (Hotspot) untuk kemudian mengirimkan <i>MAC Adres</i> dari <i>WIFI Router</i> (Hotspot) tersebut. Permasalahan yang akan di selesaikan pada penelitian ini adalah bagaimana mengembangkan sistem absensi berbasis GPS pada android versi 4.x dengan menambahkan fungsi pendeteksian <i>MAC Address</i> dari <i>WIFI Router</i> (Hotspot) yang terhubung agar dapat meningkatkan keakuratan posisi karyawan atau penggunanya.
Persamaan	Persamaannya yaitu menggunakan <i>Mac Address</i> dari <i>wifi</i> untuk proses absensi.
Perbedaan	Perbedaanya yaitu penelitian sebelumnya hanya melakukan absen ketika di berada di kantor sedangkan penelitian ini akan melakukan absen pada saat dinas luar (pusat atau cabang).

Berikut adalah *state of art* ketiga yang dapat dilihat pada Tabel 2.3 *State Of Art 3*:

**Tabel 2.3 State Of Art 3**

Judul Penelitian	Perancangan Sistem Absensi Online Menggunakan Android Guna Mempercepat Proses Kehadiran Karyawan Pada PT. Sintech Berkah Abadi
Penulis	Al Husain, Abdul Haqi Aji Prastian, dan Andre Ramadhan
Dipublikasikan	Technomedia Journal (TMJ) Vol.2 No.1 Edisi Agustus 2017 E-ISSN: 2528-6544 P-ISSN: 2620-3383

Hasil Penelitian	Penelitian ini dilakukan di PT. Sintech Berkah Abadi. Sistem absensi yang berjalan saat ini belum begitu optimal dan efektif dalam proses absensi, sehingga dibuat lah solusi untuk mengatasinya dengan cara memanfaatkan <i>wireless</i> berbasis android.
Persamaan	Persamaannya adalah menggunakan <i>wireless</i> dalam membangun aplikasi berbasis android ini.
Perbedaan	Perbedaannya yaitu penelitian sebelumnya hanya memanfaatkan teknologi <i>wireless</i> sedangkan penelitian ini akan menggunakan teknologi GPS.

Berikut adalah *state of art* keempat yang dapat dilihat pada Tabel 2.4 State Of Art 4:

**Tabel 2.4 State Of Art 4**

Judul Penelitian	Rancang Bangun Aplikasi Absensi Mahasiswa pada <i>Platform</i> Android
Penulis	A.A. Raka Onny Diar Danur C, I Nyoman Piarsa, dan A.A. Kompiang Oka Sudana
Dipublikasikan	Merpati Vol. 4 No. 1 April 2016 ISSN:2252-3006
Hasil Penelitian	Hasil penelitian ini yaitu proses absensi pada penerapan di jurusan teknologi informasi saat ini memiliki kendala dalam proses pengelolaan. Berkas absensi yang dimiliki oleh masing-masing matakuliah, terkadang bisa terdapat kerusakan dan bahkan mudah hilang. Hal tersebut dikarenakan masih digunakannya metode absensi yang diterapkan dengan tanda tangan pribadi pada sebuah lampiran kertas. Lembaran kertas yang menjadi alat absensi digunakan selama masa perkuliahan atau satu semester berlangsung. Maka munculah dasar pemikiran pembuatan aplikasi yang dapat mengatasi permasalahan tersebut. Aplikasi ini diterapkan pada <i>Platform</i> android yang dirasa mampu untuk mengatasi permasalahan yang ada.
Persamaan	Persamaannya yaitu terdapat tujuan untuk membangun sebuah aplikasi absensi berbasis android.
Perbedaan	Perbedaannya yaitu penelitian yang akan dilakukan memanfaatkan teknologi <i>wifi</i> dan GPS.

Berikut adalah *state of art* kelima yang dapat dilihat pada Tabel 2.5 State Of Art 5:

**Tabel 2.5 State Of Art 5**

Judul Penelitian	Mengamankan Wireless Dengan Menggunakan Two Factor, Password dan Mac Address Filtering
Penulis	Didi Susianto dan Iis Yulianti



Dipublikasikan	Expert-Jurnal Manajemen Sistem Informasi Dan Teknologi, Vol 5, No.2. ISSN: 2088-5555
Hasil Penelitian	Hasil penelitian ini yaitu membahas keamanan pada jaringan <i>wireless</i> dengan menggunakan <i>password</i> dan <i>filtering mac address</i> , <i>mac address filtering</i> merupakan metode <i>filtering</i> untuk membatasi hak akses dari <i>mac address</i> yang bersangkutan.
Persamaan	Persamaannya yaitu menggunakan keamanan jaringan <i>mac filtering</i>
Perbedaan	Perbedaannya yaitu penelitian yang akan dilakukan memanfaatkan juga teknologi GPS.

