BAB II
TEORI PENUNJANG

2.1 Angkutan Kota (Angkot)

"Angkutan kota merupakan moda transportasi perkotaan yang merujuk pada kendaraan umum dengan rute yang sudah ditentukan dan jarak tempuh yang tidak terlalu jauh. Walaupun saat ini masyarakat dapat memiliki kendaraan pribadi sangat mudah, tetapi kendaraan angkutan kota masih diminati masyarakat sebagai moda trasportasi karena masyarakat hanya sebagai pengguna pasif, tarifnya yang relatif murah, mudah ditemukan, dan jumlah rute/trayek yang banyak sehingga memudahkan penggunanya sampai pada tujuan. Hal ini membuat para pelaku usaha melihat kendaraan angkutan kota masih sebagai peluang bisnis yang menjanjikan." [16]

2.2 *Location Based Services* (LBS)

"*Location-Based Service* (LBS) atau disebut dengan layanan berbasis lokasi adalah layanan informasi yang dapat diakses melalui mobile *device* dengan menggunakan mobile network, yang di lengkapi kemampuan untuk memanfaatkan lokasi dari mobile device tersebut. LBS memberikan kemungkinan komunikasi dan interaksi dua arah. Oleh karena itu pengguna memberitahu penyedia layanan unntuk mendapatkan informasi yang dibutuhkan, dengan referensi posisi pengguna tersebut. Layanan berbasis lokasi dapat di gambarkan sebagai suatu layanan yang berada pada pertemuan tiga teknologi yaitu: *Geographic Information System*, *Internet Service*, dan *Mobile Devices*. LBS *(Location Based Services*) merupakan salah satu layanan yang beraksi aktif terhadap perubahan entitas posisi sehingga mampu mendeteksi letak objek dan memberikan layanan sesuai dengan letak objek yang telah di ketahui tersebut. Menurut Safaat H (2012), dua unsur utama dalam LBS yaitu :

1. *Location Manager* (API *Maps*)

   Menyediakan perangkat bagi sumber atau *source* untuk LBS, *Aplication Programming Interface* (API) menyediakan fasilitas untuk menampilkan atau memanipulasi peta.

2. *Location Providers* (Api *Location*)

   Menyediakan teknologi pencarian lokasi yang di gunakan oleh perangkat. API *Location* berhubungan dengan data GPS dan data lokasi *real-time.* Data lokasi pengguna biasanya di dapatkan melalui jaringan telepon seluler ataupun menggunakan GPS." [3]

2.3 *Global Positioning System* (GPS)

2.3.1 *Definisi dan Konsep GPS*

Pada aplikasi ini, GPS merupakan komponen utama untuk memperoleh data. Karena dari GPS akan menghasilkan data berupa koordinat yang dikirim melalui satelit yang dapat menunjukkan lokasi kendaraan, dalam hal ini adalah mobil angkutan kota (Kendaraan Angkutan Kota).

Tanoe (2009), dalam bukunya yang berjudul *GPS Bagi Pemula, Dasar-Dasar Pemakaian Sehari-hari*, menjelaskan mengenai konsep dari GPS.

"GPS bukanlah nama alat, tetapi merupakan nama sebuah sistem navigasi global berbasis satelit (GNSS= *Global Navigation Satellite System*) yang dikembangkan oleh Departemen Pertahanan Amerika Serikat. Tetapi karena sistem ini adalah yang pertama kali serta satu-satunya di dunia yang berfungsi secara penuh saat ini dan dapat digunakan setiap saat oleh semua orang didunia secara gratis, maka nama GPS menjadi terkenal dan sering dipakai sebagai nama alat navigasi berbasis satelit. Sistem ini menggunakan kelompok satelit yang diberi nama NAVSTAR(*Navigational Satellite Timing and Ranging*)." [15]

Berdasarkan kutipan di atas, dapat disimpulkan bahwa seiring berjalannya waktu, GPS berkembang menjadi sebuah sistem yang dapat digunakan oleh semua orang dan dapat digunakan kapan pun, sehingga sistem GPS ini dapat diterapkan ke dalam sistem-sistem lain yang membutuhkan navigasi berbasis satelit. Lebih lanjut, Winardi (2006) dalam bukunya yang berjudul *Posisi Dengan GPS Untuk Survei Terumbu Karang* menjelaskan pengertian dari GPS.

"GPS adalah sistem untuk menentukan letak di permukaan bumi dengan bantuan penyelarasan (*synchronization*) sinyal satelit. Sistem ini menggunakan 24 satelit yang mengirimkan sinyal gelombang mikro ke Bumi. Sinyal ini diterima oleh alat penerima di permukaan, dan digunakan untuk menentukan letak, kecepatan, arah, dan waktu." [6]

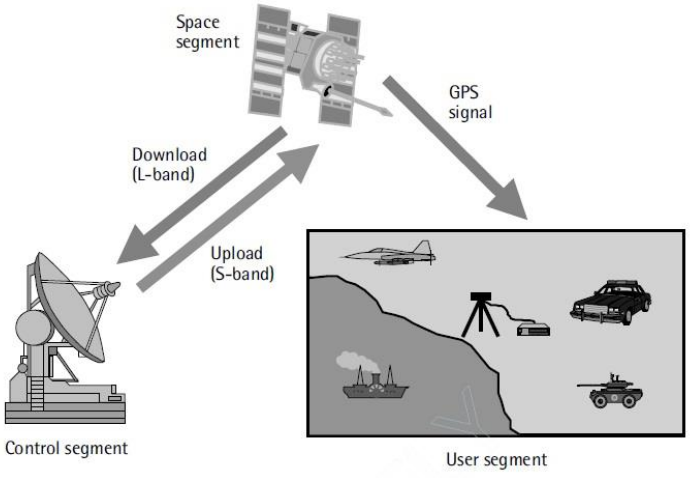
Dari kedua penjelasan tersebut dapat disimpulkan bahwa GPS merupakan sebuah sistem, dan bukanlah sebuah alat yang digunakan untuk mencari lokasi dan navigasi berbasis satelit. Sistem ini menggunakan 24 satelit yang merupakan bagian dari kelompok NAVSTAR.

Dalam teori kedua yang dikemukakan oleh Winardi (2006) juga sedikit menyinggung mengenai konsep dasar cara kerja dari sistem GPS, yaitu pertama

satelit mengirimkan sinyal gelombang mikro ke bumi, kemudian diterima oleh alat penerima, dan digunakan untuk menentukan letak, kecepatan, arah, dan waktu. Konsep cara kerja dari GPS ini akan dibahas lebih dalam pada point selanjutnya.

2.3.2 *Cara Kerja Global Positioning System (GPS)*

Bagian yang paling penting dalam sistem navigasi GPS adalah beberapa satelit yang berada di orbit bumi. Satelit GPS saat ini berjumlah 24 unit yang semuanya dapat memancarkan sinyal ke bumi yang lalu dapat ditangkap oleh alat penerima sinyal tersebut atau GPS Tracker. Dalam laman *website www.gps.gov* disebutkan bahwa selain satelit terdapat 2 sistem lain yang saling berhubungan, sehingga jadilah 3 bagian penting dalam sistem GPS. Ketiga bagian tersebut terdiri dari: GPS *Control Segment* (Bagian Kontrol), GPS *Space Segment* (bagian angkasa), dan GPS *User Segment* (bagian pengguna). Cara kerja GPS secara umum dapat digambarkan dalam gambar di bawah ini.



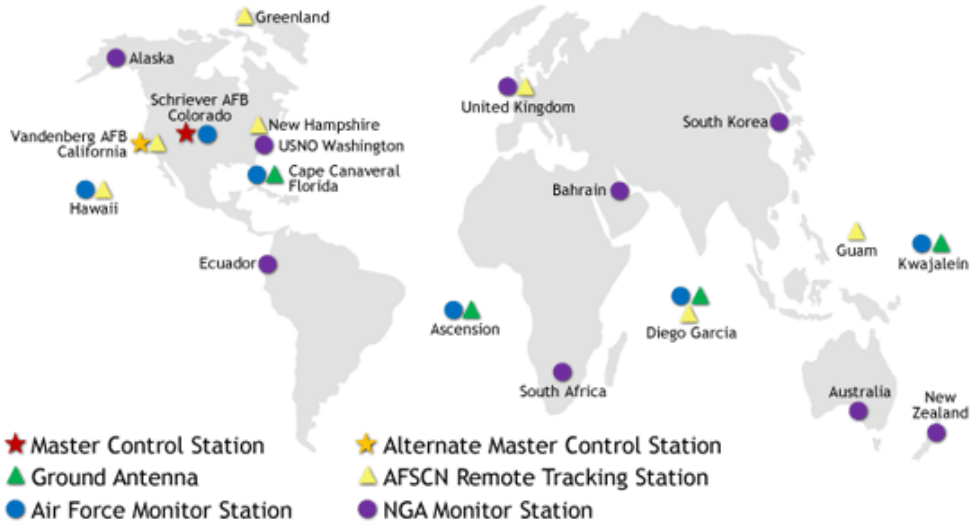
Gambar 2.1 Cara Kerja GPS (http://www.gps.gov/systems/gps/)

Dari gambar diatas, dapat dijelaskan bahwa GPS bekerja dengan mentransmisikan sinyal dari satelit ke perangkat GPS (ke perangkat *smartphone* yang dilengkapi GPS *receiver*). GPS bekerja optimal pada ruangan terbuka untuk memperoleh posisi seakurat mungkin. Informasi GPS ditransmisikan oleh beberapa satelit (minimal tiga satelit) sehingga GPS *receiver* mampu

mengkalkulasi dan menampilkan seakurat mungkin posisi, kecepatan dan informasi waktu kepada pengguna GPS. Untuk lebih jelasnya, penjelasan mengenai cara kerja GPS akan dibahas per point sebagai berikut. [12]

2.3.2.1 GPS *Control Segment*

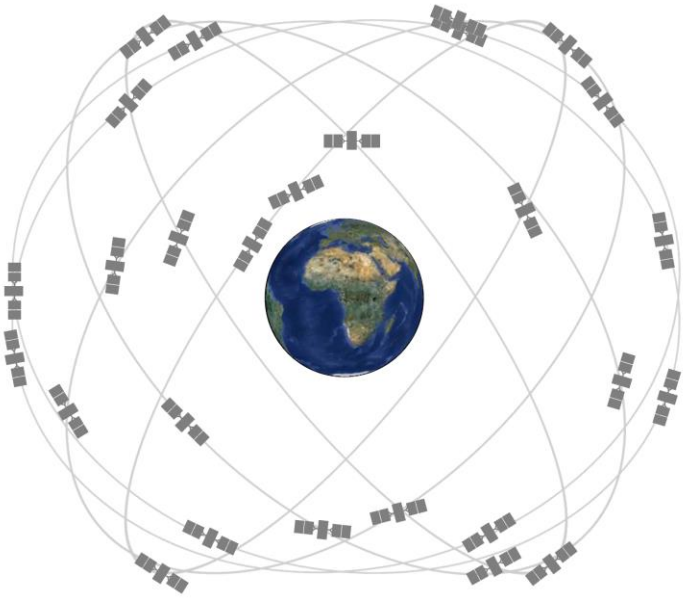
Penjelasan mengenai GPS *control segment* yang diperoleh dari *website www.gps.gov* terdiri dari *global network of ground facilities* yang melacak satelit GPS, memantau transmisi mereka, melakukan analisis, dan mengirim perintah dan data ke konstelasi. Saat ini operasional pada *control segment* termasuk stasiun kontrol induk, sebuah stasiun master kontrol alternatif, 11 komando dan kontrol antena, dan 15 situs monitoring. Lokasi fasilitas ini ditunjukkan pada peta di bawah ini."[6]



Gambar 2.2 Peta Lokasi Global Network Of Ground Facilities (http://www.gps.gov/systems/gps/)

2.3.2.2 GPS *Space Segment*

Pada *website www.gps.gov* juga dijelaskan mengenai GPS *space segment* yang terdiri dari konstelasi satelit yang mentransmisikan sinyal radio ke pengguna. Amerika Serikat berkomitmen untuk menjaga ketersediaan minimal operasional 24 satelit GPS, 95% dari waktu. Untuk memastikan komitmen ini, Angkatan Udara telah menerbangkan 31 satelit GPS operasional selama beberapa tahun terakhir."[12]



Gambar 2-II.3 GPS Satellite (http://www.gps.gov/systems/gps/)

2.3.2.3 GPS *User Segment*

GPS *user segment* terdiri dari peralatan penerima gps, yang menerima sinyal dari satelit gps dan menggunakan informasi yang dikirimkan untuk menghitung posisi tiga dimensi pengguna dan waktu.[12] Dalam aplikasi ini, peralatan penerima GPS adalah smartphone yang memiliki sensor GPS untuk menerima koordinat yang dikirim oleh satelit.

2.3.3 *GPS Tracking*

Kegunaan GPS selain untuk navigasi, militer dan informasi geografis juga dapat digunakan untuk pelacakan kendaraan atau sering disebut dengan GPS *tracking*.

"GPS tracking adalah suatu metode untuk mencari dimana sesuatu berada dengan tepat. GPS *tracking* dapat ditempatkan di dalam kendaraan, di ponsel, perangkat GPS khusus, atau unit portabel. GPS bekerja dengan memberikan informasi lokasi dengan tepat. GPS *tracking* dapat melacak pergerakan kendaraan atau orang. Misalnya, sistem GPS *tracking* dapat digunakan oleh perusahaan untuk memantau rute dan kemajuan sebuah truk pengiriman, dan oleh orang tua untuk memeriksa lokasi anak mereka, atau bahkan untuk memantau aset bernilai tinggi ketika transit."[10]

Dalam aplikasi ini, GPS *tracking* berperan untuk mencari posisi atau letak mobil angkutan kota dalam keadaan *real time* yang kemudian ditampilkan pada peta digital *Google Maps* yang telah disinkronkan dalam aplikasi ini. Kemudian *receiver* mengirimkan koordinat ke *cloud database*.

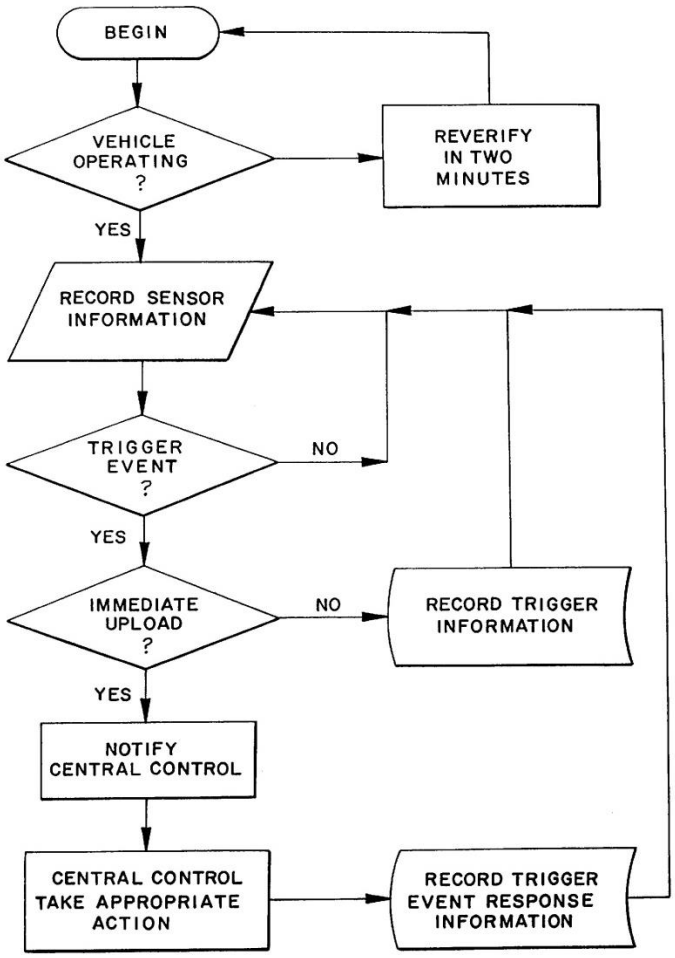
Selain dapat mencari posisi kendaraan secara *real-time* dan dapat ditampilkan pada *Google Maps*, GPS *Tracking* juga memiliki fitur lain, yaitu fitur *geofence*. Dalam aplikasi ini fitur geofence berfungsi untuk membuat suatu kondisi agar output yang diinginkan sesuai harapan. Mengenai fitur geofence ini akan dibahas di point selanjutnya.

2.3.4 *Geofence*

Geofence merupakan salah satu fitur yang ada pada GPS *Tracking*. Geofence adalah suatu pembatas atau pagar virtual yang sudah ditentukan dan dapat diterapkan untuk memantau pergerakan kendaraan pada wilayah geografis tertentu.

Ada dua tipe geofence, yang pertama sistem akan memberi laporan atau notifikasi apabila kendaraan memasuki wilayah yang sudah dipagar. Sedangkan yang kedua adalah kebalikannya yaitu, sistem akan memberi laporan atau notifikasi apabila kendaraan keluar dari wilayah yang sudah dipagar. Teknik geofence ini sangat membantu dalam pelacakan kendaraan yang akses

penggunaan kendaraannya terbatas. Berdasarkan *website https://www.google.com/patents/US6064970,* berikut gambaran dari fungsi *geofence* sehingga menghasilkan keluaran yang diinginkan.[14]



Gambar II.4 Flowchart Penerapan Geofence Pada Kendaraan (https://www.google.com/patents/US6064970)

Implementasi *geofence* pada aplikasi ini berguna untuk memberikan data jumlah rit yang sudah dilalui oleh pengemudi mobil angkutan kota tersebut. *Geofence* akan di atur di lokasi terminal-terminal yang terdapat di setiap trayek yang digunakan.

2.4 Pengukuran Tingkat Akurasi GPS

2.4.1 *DRMS (Distance Root Mean Square)*

Menurut referensi dari tesis Nazzla, R. (2016) Akurasi adalah hasil estimasi tingkat kedekatan suatu nilai dari hasil pengukuran terhadap nilai yang sebenarnya. Dalam hal ini akurasi dua dimensi (2D) diekpresikan dalam jarak (m) atau disebut *Distance Root Mean Square* (DRMS). Axelrad (1998) menyatakan bahwa pengukuran yang akurat dari suatu GPS misalnya 25 meter (kepercayaan 95%), ini berarti bahwa dapat dipercaya bahwa 95% pengukuran berada di kisaran kesalahan 25 meter dan kemungkinan 5% terjadi kesalahan yang lebih besar dari 25 meter.[8]

DRMS merupakan selisih jarak antara titik koordinat referensi dengan hasil perolehan data, dimana titik koordinat tersebut merupakan hasil perpotongan sumbu x (*latitude*) dan sumbu y (*longitude*). DRMS merupakan *Radial Error, Means Square Position Error,* dengan probabilitas $\sigma_x = \sigma_y$ sebesar 63%. Dalam rangka menghitung DRMS 2D, perlu mengetahui beberapa persamaan berikut ini;

Misalnya x_i dan y_i, maka rataan nilai pada pengukuran ke n adalah

$$\overline{x} = \frac{1}{n}\sum_{i=1}^{n} x_i \text{ dan } y = \frac{1}{n}\sum_{i=1}^{n} y_i \quad \text{.....[1]}$$

Deviasi pada pengukuran dari rataan adalah

$$\sigma x_i = x_i - \overline{x} \text{ dan } \sigma y_i = y_i - \overline{y} \quad \text{.....[2]}$$

DRMS adalah akar kuadrat dari rata-rata kuadrat *error* yang didefinisikan sebagai berikut:

$$DRMS = \sqrt{(\sigma_x)^2 + (\sigma_y)^2} \text{ atau } \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \text{ ”.....[3]}$$

2.4.2 *Kesalahan dan Bias pada GPS*

Posisi yang ditunjukkan oleh suatu GPS mempunyai faktor kesalahan atau juga disebut tingkat akurasi. Distorsi atau ingkat akurasi pada GPS *smartphone* kurang lebih 5 sampai 15 meter. Tingkat akurasi ini sifatnya berubah-ubah karena disebabkan banyak faktor. Seperti dikemukakan oleh Wishnu (2012), faktor apa saja yang dapat mempengaruhi tingkat akurasi pada GPS, yaitu :

a. Kesalahan *Ephemeris*. Terjadi jika satelit tidak dapat mentransmisikan posisinya di orbit dengan tepat.
b. Keadaan *Ionosphere*. *Ionosphere* berada pada jarak sekitar 43-50 mil di atas permukaan bumi. Satelit yang melewati ionosphere akan menjadi lambat dikarenakan adanya plasma (gas dengan tingkat kepadatan rendah). Walaupun GPS *receiver* berusaha untuk mengkoreksi/memperbaiki faktor keterlambatan yang terjadi tetap saja aktivitas tertentu dari plasma bisa menyebabkan kesalahan perhitungan.
c. Keadaan *Troposphere*. *Troposphere* adalah bagian terendah dari atmosfer sampai dengan ketinggian sekitar 11 mil dari permukaan tanah. Variasi pada temperatur, tekanan, dan kelembaban bisa menyebabkan perbedaan kecepatan penerimaan gelombang radio.
d. Kesalahan Waktu. Karena penempatan jam atom pada setiap GPS *receiver* tidak berjalan sebagaimana mestinya. Kesalahan waktu dari GPS *receiver* yang tidak presisi dapat menimbulkan ketidak akurasian.
e. Kesalahan *Multipath*. Terjadi karena sinyal satelit membentur permukaan keras (seperti bangunan atau tebing) sebelum mencapai GPS receiver. Hal tersebut bisa menyebabkan terjadinya delay sehingga perhitungan jarak menjadi tidak akurat.
f. Buruknya Sinyal Satelit. Keadaan langit yang terhalang akan menyebabkan GPS sulit menerima data satelit. Sebuah sinyal satelit yang pada hari tertentu diterima dengan sangat bagus belum tentu pada hari lain bisa diterima dengan kualitas yang sama walaupun user berdiri pada tempat yang sama. Hal tersebut dikarenakan posisi dari satelit yang terus bergerak atau bisa juga disebabkan faktor penghalang lain seperti pohon, gedung bertingkat, dan sebagainya. [7]

2.5 Google *Maps Application Programming Interface* (API)

Peta digital yang digunakan pada aplikasi ini adalah Google *Maps* yang akan diintegrasikan pada android. Maka dibutuhkan *Application Programming Interface* (API) dari Google *Maps* tersebut.

Google Maps Application Programming Interface (API) merupakan suatu fitur aplikasi yang dikeluarkan oleh google untuk memfasilitasi pengguna yang ingin mengintegrasikan *Google Maps* ke dalam *website* masing-masing dengan menampilkan data *point* milik sendiri. Agar aplikasi *Google Maps* dapat muncul di *website* tertentu, diperlukan adanya API *key*. API *key* merupakan kode unik yang digenerasikan oleh google untuk suatu *website* tertentu, agar *server Google Maps* dapat mengenali. *Google Maps* API telah menyediakan *template* dasar yang dapat digunakan oleh pengguna untuk dikembangkan lebih lanjut.[9] Pada Google *Maps* API terdapat 4 jenis pilihan model peta yang disediakan oleh Google, diantaranya adalah:

1. *Roadmap,* untuk menampilkan peta biasa 2 dimensi.
2. *Satellite*, untuk menampilkan foto satelit.
3. *Terrain*, untuk menunjukkan *relief* fisik permukaan bumi dan menunjukkan seberapa tingginya suatu lokasi, contohnya akan menunjukkan gunung dan sungai.
4. *Hybrid*, akan menunjukkan foto satelit yang diatasnya tergambar pula apa yang tampil pada *roadmap* (jalan dan nama kota).

2.6 Android

Aplikasi ini berbasis android, dimana dibutuhkan minimal dua *smartphone* dengan sistem operasi android. Satu *smartphone* digunakan oleh *user*, dan *smartphone* yang lain berfungsi sebagai *background*. Karena begitu pentingnya peranan android pada aplikasi ini, maka akan lebih baik jika kita membahas mengenai konsep android terebih dahulu. Beberapa penjelasan mengenai android akan dibahas di bawah ini.

Menurut Safaat H (2011), android adalah sebuah sistem operasi untuk perangkat mobile berbasis Linux yang mencakup sistem operasi, middleware dan aplikasi.[2] Android merupakan sistem operasi dengan *open sources*. Dengan sistem distribusi *open sources* yang digunakan memungkinkan para pengembang untuk menciptakan beragam aplikasi menarik yang dapat dinikmati oleh para penggunanya, seperti *game, chatting* dan lain-lain.

Pengembang aplikasi pada umumnya menggunakan bahasa pemrograman Java dengan menggunakan SDK (*Software Development Kit*). SDK ini terdiri atas seperangkat *tools* pengembangan, termasuk *debugger*, *library software*, emulator *handset* yang berbasis QEMU, dokumentasi, kode sampel, dan tutorial. Didukung secara resmi oleh Google menjadi IDE (*Integrated Development Environment*) yaitu Android *Studio*, yang menggunakan *plug-in* ADT (*Android Development Tools*).

Dari penjelasan diatas, dapat disimpulkan bahwa android bersifat *open sources.* Sehingga penulis memilih android sebagai basis aplikasi yang akan dibuat, karena dalam pembuatan dan pengembangan aplikasinya menjadi lebih mudah. Selain itu, smartphone yang berbasis android juga relatif terjangkau oleh semua kalangan. Hal ini memungkinkan penggunaan aplikasi ini akan bermanfaat dan dapat digunakan oleh pemilik armada angkutan kota.

2.6.1 *Java*

Menurut Avestro (2007) "Sebagai sebuah bahasa pemrograman, Java dapat membuat seluruh bentuk aplikasi, *desktop*, *web* dan lainnya, sebagaimana dibuat dengan menggunakan bahasa pemrograman konvensional yang lain. Java adalah bahasa pemrograman yang berorientasi objek (OOP) dan dapat dijalankan pada berbagai *platform* sistem operasi. Perkembangan Java tidak hanya terfokus pada satu sistem operasi, tetapi dikembangkan untuk berbagai sistem operasi dan bersifat *open source*. Sebagai sebuah peralatan pembangun, teknologi Java menyediakan banyak *tools: compiler, interpreter*, penyusun dokumentasi, paket kelas dan sebagainya.

Aplikasi dengan teknologi Java secara umum adalah aplikasi serba guna yang dapat dijalankan pada seluruh mesin yang memiliki Java *Runtime Environment* (JRE). Terdapat dua komponen utama dari *Deployment Environment*. Yang pertama adalah JRE, yang terdapat pada paket J2SDK, mengandung kelas–kelas untuk semua paket teknologi Java yang meliputi kelas dasar dari Java, komponen GUI dan sebagainya. Komponen yang lain terdapat pada *Web Browser*. Hampir seluruh *Web Browser* komersial menyediakan *interpreter* dan *runtime environment* dari teknologi Java. Berdasarkan white paper resmi dari SUN, Java memiliki karakteristik berikut:

1. Sederhana
   Bahasa pemrograman Java menggunakan sintaks mirip dengan C++ namun sintaks pada Java telah banyak diperbaiki terutama menghilangkan penggunaan pointer yang rumit dan multiple inheritance. Java juga menggunakan *automatic memory allocation* dan *memory garbage collection.*

2. Berorientasi objek (*Object Oriented*)
   Java mengunakan pemrograman berorientasi objek yang membuat program dapat dibuat secara modular dan dapat dipergunakan kembali. Pemrograman berorientasi objek memodelkan dunia nyata kedalam objek dan melakukan interaksi antar objek-objek tersebut.
3. Dapat didistribusi dengan mudah
   Java dibuat untuk membuat aplikasi terdistribusi secara mudah dengan adanya *libraries networking* yang terintegrasi pada Java.
4. Interpreter
   Program Java dijalankan menggunakan *interpreter* yaitu *Java Virtual Machine* (JVM). Hal ini menyebabkan *source code* Java yang telah dikompilasi menjadi Java *bytecodes* dapat dijalankan pada *platform* yang berbeda-beda.
5. Robust
   Java mempuyai reliabilitas yang tinggi. *Compiler* pada Java mempunyai kemampuan mendeteksi eror secara lebih teliti dibandingkan bahasa pemrograman lain. Java mempunyai *runtime-Exception handling* untuk membantu mengatasi eror pada pemrograman.
6. Aman
   Sebagai bahasa pemrograman untuk aplikasi internet dan terdistribusi, Java memiliki beberapa mekanisme keamanan untuk menjaga aplikasi tidak digunakan untuk merusak sistem komputer yang menjalankan aplikasi tersebut.
7. *Architecture Neutral*
   Program Java merupakan *platform independent*. Program cukup mempunyai satu buah versi yang dapat dijalankan pada platform yang berbeda dengan *Java Virtual Machine*.
8. Portabel
   Source code maupun program Java dapat dengan mudah dibawa ke platform yang berbeda-beda tanpa harus dikompilasi ulang.
9. *Performance*
   Performance pada Java sering dikatakan kurang tinggi. Namun *performance* Java dapat ditingkatkan menggunakan kompilasi Java lain seperti buatan Inprise, Microsoft ataupun Symantec yang menggunakan *Just In Time Compilers* (JIT).
10. *Multithreaded*
    Java mempunyai kemampuan untuk membuat suatu program yang dapat melakukan beberapa pekerjaan secara sekaligus dan simultan.
11. Dinamis
    Java didesain untuk dapat dijalankan pada lingkungan yang dinamis. Perubahan pada suatu *class* dengan menambahkan *properties* ataupun *method* dapat dilakukan tanpa menggangu program yang menggunakan *class* tersebut."[1]

Pada umumnya banyak pengembang android yang menggunakan bahasa pemrograman Java untuk membuat berbagai aplikasi android, serta jika melihat kelebihan-kelebihan dari bahasa pemrograman Java yang telah dipaparkan diatas,

maka penulis menggunakan Java sebagai bahasa pemrograman untuk membuat aplikasi ini.

Dalam pembuatan aplikasi ini banyak *software* yang dapat digunakan, tetapi penulis menggunakan *software* Android Studio untuk pembuatan aplikasi ini karena selain dapat menggunakan bahasa pemrograman Java, Android Studio juga lebih mudah dan memiliki banyak fitur yang dapat digunakan.

2.6.2 *Android Studio*

Android Studio adalah *Integrated Development Environment* (IDE) untuk pengembangan aplikasi Android, berdasarkan IntelliJ IDEA . Android Studio merupakan IDE yang dibuat oleh Google untuk menggantikan Eclipse dalam pembuatan program aplikasi android. Android Studio memiliki fitur lebih banyak untuk meningkatkan produktivitas saat membuat aplikasi Android, misalnya:

1. Sistem pembuatan berbasis Gradle yang fleksibel.
2. Emulator yang cepat dan kaya fitur.
3. Lingkungan yang menyatu untuk pengembangan bagi semua perangkat Android.
4. *Instant Run* untuk mendorong perubahan ke aplikasi yang berjalan tanpa membuat APK baru.
5. Template kode dan integrasi GitHub untuk membuat fitur aplikasi yang sama dan mengimpor kode contoh.
6. Alat penguji dan kerangka kerja yang ekstensif.
7. Alat Lint untuk meningkatkan kinerja, kegunaan, kompatibilitas versi, dan masalah-masalah lain.
8. Dukungan C++ dan NDK.
9. Dukungan bawaan untuk Google Cloud Platform, mempermudah pengintegrasian Google Cloud Messaging dan App Engine.

Android Studio memakai Gradle sebagai landasan sistem, dengan kemampuan spesifik Android yang disediakan oleh *Android Plugin for Gradle*. Sistem ini bisa dijalankan sebagai alat terpadu dari menu Android Studio dan

secara independen dari baris perintah. Para pengembang dapat menggunakan fitur-fitur sistem untuk melakukan hal-hal berikut:

1. Menyesuaikan, mengonfigurasi, dan memperluas proses pembuatan.
2. Membuat beberapa APK untuk aplikasi Android dengan aneka fitur menggunakan proyek dan modul yang sama.
3. Menggunakan kembali kode dan sumber daya pada seluruh rangkaian sumber.

Dengan menerapkan fleksibilitas Gradle, semua dapat dicapai tanpa mengubah *file* sumber inti aplikasi. *File* versi Android Studio diberi nama `build.gradle`. *File* ini adalah teks biasa yang menggunakan konfigurasi *Groovy* dengan elemen yang disediakan oleh *plugin Android for Gradle*. Masing-masing proyek memiliki *file* level atas untuk seluruh proyek dan *file* level modul terpisah untuk setiap modul. Saat mengimpor proyek yang sedang dikerjakan Android Studio otomatis menghasilkan file yang diperlukan. [13]

Android Studio dalam pembuatan aplikasi ini berfungsi untuk membuat tampilan serta instruksi-instruksi untuk menghasilkan keluaran diinginkan. Kemudian menghasilkan *services* yang dibutuhkan seperti memunculkan Google *Maps*, dan lain-lain.

2.7 *Database* MySQL

Untuk menyimpan data yaitu berupa koordinat dan informasi-informasi lainnya, maka penulis menggunakan MySQL sebagai *database*. *Database* MySQL ini akan menjadi sebuah *web server* yang menggunakan *hosting* dan *domain* sehingga *database* ini dapat aktif selama 24 jam.

MySQL merupakan salah satu produk RDBMS (*Relational Database Management System*) yang bisa dipakai secara gratis dan *open sources*. Data yang ingin disimpan akan diperlakukan RDBMS sebagai tabel-tabel yang saling berhubungan atau dapat dihubungkan maupun berdiri sendiri dalam *database*. MySQL banyak digunakan programmer *web* sebagai program *database* karena memiliki kekuatan dan stabilitas yang tinggi sebagai media penyimpanan data. [4]

Beberapa kelebihan menggunakan MySQL sebagai *database* diantaranya adalah :

1. MySQL merupakan *database* tercepat saat ini.
2. MySQL mempunyai performa tinggi, namun merupakan *database* yang simpel sehingga mudah di-*set-up* dan di konfigurasi.
3. MySQL merupakan *database* yang bisa digunakan secara gratis.
4. MySQL mengerti bahasa SQL (*Structured Query Language*) yang merupakan pilihan sistem *database* modern.
5. MySQL memiliki kapabilitas yang tinggi karena *server* dapat diakses secara simultan.
6. *Database* MySQL dapat diakses dari semua tempat di internet dengan hak akses tertentu.
7. MySQL merupakan *database open source*.[5]

2.8 Deskripsi Aplikasi Pengawasan Rute Angkutan Kota (APRA)

"Aplikasi Pengawasan Aktifitas Rute Kendaraan Angkutan Kota menggunakan Teknologi GPS Berbasis Android" atau aplikasi ini penulis beri nama APRA, merupakan sistem yang memanfaatkan teknologi GPS yang ada pada android untuk mendapatkan dan menampilkan posisi dimana kendaraan angkutan kota berada. Pada sistem ini terdiri dari aplikasi *user* dan aplikasi *background*. Aplikasi *background* berfungsi sebagai alat perekam jejak atau *tracker* berupa data koordinat lalu menyimpannya ke dalam *database*. *User* adalah pemilik kendaraan angkutan kota yang menggunakan aplikasi ini untuk mengawasi aktifitas yang terjadi di jalur trayek angkutan kota tersebut. *User* akan menerima informasi yang dikirim secara periodik oleh aplikasi *background* untuk mendapatkan lokasi kendaran angkutan kota.