BAB II

TEORI PENUNJANG

Pada bab ini akan dibahas mengenai teori dan komponen penunjang yang akan digunakan dalam perancangan "Smart Tracking Motorcycle Berbasis GPS".

2.1 Sepeda Motor dan Keamanan

Sepeda motor adalah kendaraan beroda dua yang digerakkan oleh sebuah mesin. Kecepatan tinggi sepeda motor tetap stabil disebabkan oleh gaya giroskopik, sedangkan pada kecepatan rendah, kestabilan atau keseimbangan sepeda motor bergantung kepada pengaturan stang oleh pengendara, karena penggunaan sepeda motor yang mudah dipelajari dan harganya relatif terjangkau, tidak heran jika penggunaan sepeda motor sangat populer di Indonesia. Semakin tinggi tikat penggunaan sepeda motor berbanding lurus dengan kasus pencurian sepeda motor.

Keamanaan kendaraan sepeda motor sangat penting terutama di kota-kota besar khususnya di kota Bandung, dengan beragam kasus pencurian sepeda motor mulai dari pembobolan kunci kontak, adanya cairan yang dapat merapuhkan gembok dan kasus baru-baru ini khususnya di kota besar yaitu pencurian dengan modus pemindahan sepeda motor ke dalam mobil bak terbuka (mobil *pick up*).

2.2 Tracking

Tracking adalah suatu kegiatan untuk mengikuti jejak suatu objek. Pengertian *tracking* atau pemantauan dalam hal ini adalah kegiatan untuk memantau keberadaan sepeda motor berdasarkan posisi yang didapatkan dari peralatan *tracking*. Banyak cara yang digunakan untuk melakukan pemantauan, salah satunya adalah dengan menggunakan modul GPS. Dengan menggunakan modul GPS dapat diketahui keberadaan sepeda motor berdasarkan posisi *latitude*

dan *longitude*, sehingga dari posisi *latitude* dan *longitude* tersebut dapat dibuat menjadi pesan singkat sebagai informasi keberadaan kendaraan. Ada 2 tipe alat *tracking*, yaitu:

1. *Passive Tracking*

   Alat yang digunakan untuk menyimpan data-data seperti lokasi, kecepatan, arah, dan lainya. Ketika sepeda motor kembali maka data yang terdapat pada alat yang dipasang dapat di *download* ke komputer lalu dilakukan evaluasi terhadap data tersebut.
2. *Active Tracking*

   Alat yang digunakan untuk memperoleh data-data seperti lokasi, kecepatan, arah, dan lainnya. Lalu data-data tersebut langsung dikirim secara *real-time* melalui jaringan seluler atau satelit.

2.3 Kapolres Bandung Kota

Kepolisian Resort (Polres) adalah struktur komando Kepolisian Republik Indonesia di daerah kabupaten atau kota. Kepolisian Resort di wilayah perkotaan biasa disebut Kepolisian Resort Kota (Polresta). Kepolisian Resort Kota Besar (Polrestabes) biasanya digunakan untuk ibukota provinsi. Kepolisian Resort dikepalai oleh seorang Kepala Kepolisian Resort (Kapolres), Kepolisian Resort Kota dikepalai oleh seorang Kepala Kepolisian Resort Kota (Kapolresta) dan Kepolisian Resort Kota Besar dikepalai oleh seorang Kepala Kepolisian Resort Kota Besar (Kapolrestabes).

Polres memiliki satuan tugas kepolisian yang lengkap, layaknya Polda, dan dipimpin oleh seorang Komisaris Besar Polisi (Kombes) atau Ajun Komisaris Besar Polisi (AKBP).

Kepolisian Sektor (Polsek) adalah struktur komando Kepolisian Republik Indonesia di tingkat kecamatan. Kepolisian sektor di perkotaan biasanya disebut sebagai Kepolisian Sektor Kota (Polsekta). Kepolisian Sektor dikepalai oleh seorang Kepala Kepolisian Sektor (Kapolsek) dan Kepolisian Sektor Kota dikepalai oleh seorang Kepala Kepolisian Sektor Kota (Kapolsekta).

Polsek maupun Polsekta dipimpin oleh seorang Ajun Komisaris Besar Polisi (AKBP) atau Komisaris Polisi (Kompol), sedangkan di Polda lainnya, Polsek atau Polsekta dipimpin oleh perwira berpangkat Ajun Komisaris Polisi (AKP) [9].

Polres Kota Bandung dibagi menjadi tiga bagian, yaitu Polres Bandung Barat terdiri dari sepuluh Kapolsek, Polres Bandung Timur terdiri dari tiga Kapolsekta, dan Polres Bandung Tengah terdiri dari Sembilan Kapolsekta, daftar Kapolsekta disetiap bagian Polres Bandung;

Tabel 2.1 Polres Bandung Barat

Polsekta	Nama Jalan	Nomor Telepon
Polsekta Sukasari	Jln. Gegerkalong Hilir No. 155 Bandung	022-2013179
Polsekta Cicendo	Jln. Pasirkaliki No. 117 Bandung	022-6011503
Polsekta Andir	Jln. Saritem No. 4 Bandung	022-6012062
Polsekta Astanaanyar	Jln. Astana Anyar No. 298 Bandung	022-52047692
Polsekta Bojongloa Kidul	Jln. Peta No. 104 Bandung	022-5211082
Polsekta Bojongloa Kaler	Jln. Numi Kopo Kencana Bandung	022-6037706
Polsekta Babakan Ciparay	Jln. Soekarno-Hatta Bandung	022-6015345
Polsekta Bandung Kulon	Jln. Holis No. 7 Bandung	022-6011554
Polsekta Cidadap	Jln. Setiabudhi No. 186 Bandung	022-2013521
Polsekta Coblong	Jln. Sangkuriang No. 10 Bandung	022-2502532

Tabel 2.2 Polres Bandung Timur

Polsekta	Nama Jalan	Nomor Telepon
Polsekta Bandung Wetan	Jln. Cihapit No.2 Bandung	022-4205162
Polsekta Sumur Bandung	Jln. Kebun Sirih No. 38 Bandung	022-4235675
Polsekta Cibeunying Kidul	Jln. A. Yani No. 879 Bandung	022-7271128

Tabel 2.3 Polres Bandung Tengah

Polsekta	Nama Jalan	Nomor Telepon
Polsekta Regol	Jln. Moh. Toha No. 81 Bandung	022-5202169
Polsekta Lengkong	Jln. Buahbatu No. 193 Bandung	022-7300958
Polsekta Kiaracondong	Jln. Kiaracondong No. 167 Bandung	022-7312219
Polsekta Cicadas	Jln. Jend.A.Yani No. 10 Bandung	022-7208467
Polsekta Cibiru	Jln. A. Yani No. 10 Bandung	022-7810348
Polsekta Margacinta	Jln. Margacinta Buah Batu	022-7565727
Polsekta Arcamanik	Jln. Batununggal No. 6 Bandung	022-7805022
Polsekta Bandung Kidul	Jln. Bumi Asih No. 2A Cipamokolan	022-7508122
Polsekta Rancasari	Jln. Ciwastra No. 289 Bandung	022-7511153

2.4 Perangkat Keras

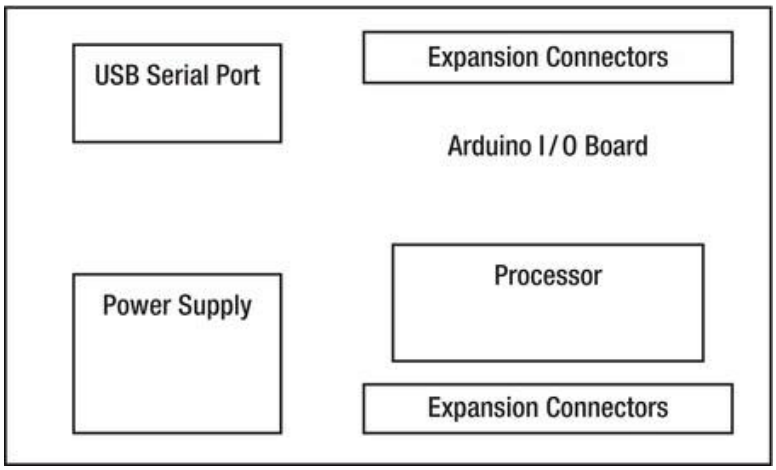
Perangkat keras adalah suatu komponen penunjang dalam pembuatan alat pada penelitin tugas akhir ini, diantaranya mikrokontroler, sensor *accelerometer*, modul GPS dan GSM.

2.4.1 Arduino

Arduino adalah *platform* pembuatan prototipe elektronik yang bersifat *open-source hardware* yang berdasarkan pada perangkat keras dan perangkat lunak yang *fleksibel* dan mudah digunakan. Arduino ditujukan bagi para seniman, desainer, dan siapapun yang tertarik dalam menciptakan objek atau lingkungan yang interaktif. Arduino pada awalnya dikembangkan di Ivrea, Italia.

Platform arduino terdiri dari:

- Arduino *board* biasanya memiliki sebuah *chip* dasar mikrokontroler Atmel AVR ATmega8 berikut turunannya.
- Blok diagram arduino *board*.

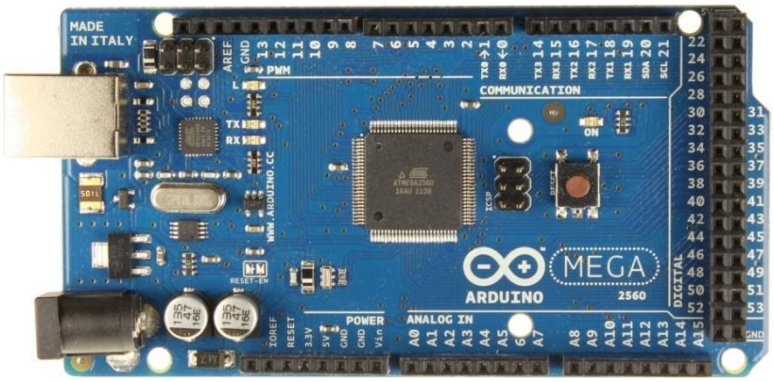


Gambar 2.1 Blok Diagram Arduino *Board*

- *Shield* adalah sebuah papan yang dapat dipasang diatas arduino board untuk menambah kemampuan dari arduino *board*.
- Bahasa pemrograman arduino adalah bahasa pemrograman yang umum digunakan untuk membuat perangkat lunak yang ditanamkan pada arduino *board*. Bahasa pemrograman arduino mirip dengan bahasa pemrograman C++.
- *Arduino Development Environment* adalah perangkat lunak yang digunakan untuk menulis dan meng-*compile* program untuk arduino. *Arduino Development Environment* juga digunakan untuk meng-*upload* program yang sudah di-*compile* ke memori program arduino *board*.

2.4.1.1 Mikrokontroler ATmega2560

ATmega2560 merupakan mikrokontroler 8 bit berbasis arsitektur RISC *(Reduced Instruction Set Computing)* dimana set instruksinya dikurangi baik dari segi ukurannya maupun kompleksitas mode pengalamatannya. Satu instruksi biasnya berukuran 16 bit dan sebagian besar dieksekusi dalam 1 siklus *clock*.



Gambar 2.2 Arduino Mega *Board*

2.4.1.2 Pin Input Output

ATmega2560 memiliki 86 buah I/O yang masing-masing memiliki fungsi khusus yang bisa di program selain sebagai pin I/O biasa. Fungsi khusus tersebut diantaranya adalah sebagai berikut:

1. ADC (*Analog Digital Converter*)

   ATmega2560 memiliki 16 pin ADC yang dapat digunakan untuk mengkonversi nilai tegangan yang berupa analog menjadi data berupa bit-bit biner atau data digital.

2. *Timer*

   ATmega2560 didukung oleh 6 *timer* dengan detail 2 *timer* 8 bit dan 4 timer 16 bit.Setiap *timer* tersebut memiliki pewaktuan masing-masing yang merupkan hasil *prescaler* dari *clock* utama mikrokontroler. *Timer* ini bisa diset menjadi mode *Compare* yaitu mode dimana *timer* bisa menghasilkan sinyal PWM dengan perioda dan lebar positif pulsa tertentu.

3. USART

   USART (*Universal Serial Asynchronous Receiver Transmitter*) merupakan fungsi khusus yang tidak kalah penting yang dimiliki ATmega2560. Dengan fungsi ini mikrokontroler dapat berkomunikasi dengan perangkat lain seperti komputer, ponsel, bahkan dengan mikrokontroler lain melalu jalur serial.

4. *Interrupt*

   Digunakan jika terdapat masukan yang perubahan bitnya perlu di respon sesaat itu juga. Dengan memilih salah satu mode misalnya *falling edge* pada pin khusus ini, maka *interrupt* akan terjadi ketika pin tersebut mengalami perubahan nilai bit dari 1 ke 0. Ada mode lain yang dapat membuat *interrupt* terjadi, yaitu mode *rising edge* yang merupakan kebalikan dari mode *falling edge* dan mode *toggle* yang merupakan gabungan antara *falling edge* dan *rising edge.* Sesaat setelah *interrupt* terjadi maka *interrupt service routine* yang merupakan respon pun akan segera dieksekusi.

2.4.2 Modul GPS

Global Positioning Syste (GPS) adalah sistem satelit navigasi dan penetuan posisi yang dimiliki dan dikelola oleh Amerika Serikat. Sistem ini didesain untuk memberikan posisi dan kecepatan 3 dimensi serta informasi mengenai waktu, secara kontinyu di seluruh dunia tanpa bergantung waktu dan cuaca.

Pada saat ini konstelasi satelit yang membawa gelombang-gelombang di atas terdiri dari 24 satelit yang terbagi dalam 6 bidang orbit sedemikian rupa sehingga untuk setiap posisi dimuka bumi dapat menerima minimal 4 buah satelit pada saat yang bersamaan sehingga memungkinkan untuk menentukan posisi setiap saat. GPS terdiri dari segi segmen utama, yaitu:

1. Segmen Angkasa (*Space Segment*)

Stasiun radio angkasa yang dilengkapi dengan antena-antena untuk mengirim dan menerima sinyal-sinyal gelombang. Sinyal-sinyal itu selanjutnya diterima oleh *receiver* GPS di permukaan bumi dan digunakan untuk penentuan posisi, kecepatan maupun waktu. Selain itu satelit juga dilengkapi dengan peralatan untuk mengontrol tingkah laku satelit, serta sensor-sensor untuk mendeteksi peledakan nuklir dan lokasinya.

2. Segmen Sistem Kontrol (*Control System Segment*)

Berfungsi untuk mengontrol dan memantau operasional satelit dan memastikan bahwa satelit berfungsi sebagai mestinya.

Fungsi ini mencakup beberapa tugas, yaitu:

a. Secara kontinyu memantau dan mengontrol sistem satelit.
b. Menemukan dan menjaga waktu sistem GPS.
c. Memprediksi *ephemeris* satelit serta karakter jam satelit.
d. Secara periodik meremajakan (*update*) *navigation message* dari setiap satelit.
e. Melakukan manuver satelit agar tetap berada dalam orbitnya, atau melakukan relokasi untuk menggantikan satelit yang tidak sehat, seandainya diperlakukan.

3. Segmen Pemakai (*User Segment*)

Segmen pemakai terdiri dari pengguna satelit GPS, baik di darat, laut maupun di angkasa. Alat penerima sinyal (*receiver*) diperlukan untuk menerima dan memproses sinyal-sinyal dari satelit GPS untuk digunakan dalam penentuan posisi, kecepatan, maupun waktu. Komponen utama dari *receiver* GPS adalah:

a. Antena dengan *pre-amplifier*.
b. Bagian *Radio Frequency* (RF) dengan pengidentifikasi sinyal dan pemroses sinyal.
c. Pemroses mikro untuk pengontrol *receiver*, data sampling, dan pemroses data.
d. Osilator presesi.
e. Catu daya.
f. Unit perintah dan tampilan.
g. Memori serta perekaman data.

2.4.2.1 Format dan Keluaran GPS

Receiver GPS memiliki format keluaran sebanyak lima jenis yaitu NMEA 0180, NMEA 0182, NMEA 0183, AVIATION, dan PLOTTING. Format data tersebut ditetapkan oleh NMEA (*National Maritime Electronic Association*) dan dapat dikoneksikan ke komputer melalui port komunikasi serial dengan menggunakan kabel RS-232. Data keluaran dalam format NMEA 0183 berbentuk kalimat (s*tring*) yang merupakan rangkaian karakter ASCII 8 bit. Setiap kalimat diawali dengan satu karakter '$', dua karakter *Talker ID*, tiga karakter *Sentence ID*, dan diikuti oleh data *fileds* yang masing-masing dipisahkan oleh koma serta diakhiri oleh *optional cheksum* dan karakter *cariage return/line feed* (CR/LF). Jumlah maksimum karakter dihitung dari awal kalimat ($) sampai dengan akhir kalimat (CR/LF) adalah 82 karakter. Format dasar data NMEA 0183:

NMEA *Output*

Output dari pesawat penerima GPS memiliki struktur tertentu yang disebut NMEA *output*. Contoh struktur *output* beserta cara pembacaannya yaitu sebagai berikut:

$GPRMC,180432,A,4027.027912,N,08704.857070,W,000.04,181.9,131000,1.8, W,D*25, esi4001.

Tabel 2.4 NMEA *Output* GPRS MC

Bidang Data	Nilai	Deskripsi
1	180432	UTC dari penentuan posisi dalam format hhmmss.ss (18 jam, 4 mnt dan 32 detik)
2	A	Status (A-data *valid*, V-peringatan)
3	4027.027912	*Latitude* geografis dalam format ddmm.mmmmm.m (40 derajat dan 27.027912 menit)
4	N	Arah dari *latitude* (N–North, S–South)
5	08704.857070	*Longitude* geografis dalam format dddmm.mmmmm.m (87 derajat dan 4.85707 menit)

6	W	Arah dari *longitude* (*E–East, W–West*)
7	000.04	Kecepatan diatas tanah (0.04 knot)
8	181.9	*Track made good* (*heading*) (181.9º)
9	131000	Tanggal dalam format ddmmyy (Oktober 13, 2000)
10	1.8	Variasi kompas (1.8º)
11	W	Arah dari variasi kompas (E-*East*, W-*West*)
12	D	Mode indikasi (A-*autonomous*, D-*differensial*, N-data tidak *valid*)
13	esi4001	Id perangkat yang disertakan pada saat penyetingan oleh *installer*

2.4.3 Modul GSM

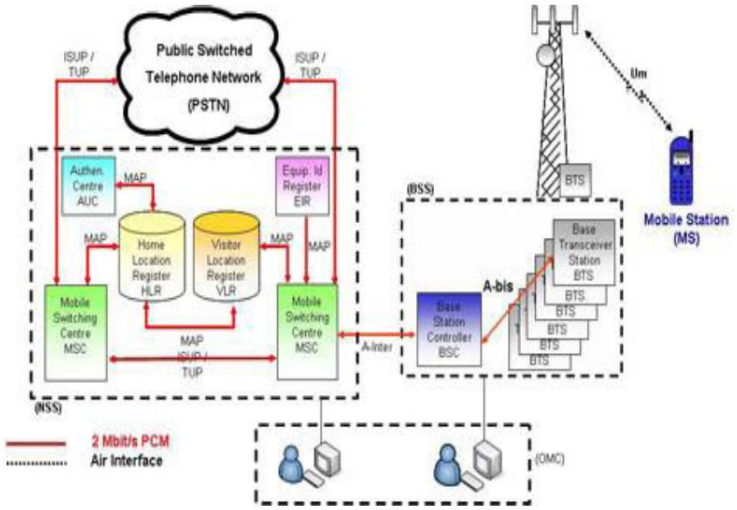
Global System for Mobbile (GSM) adalah sistem standar selular pertama di dunia yang menggunakan digital modulation dan *network level architectures* and *service*. Pada awalnya sistem GSM ini dikembangkan untuk melayani sistem seluler pan-Eropa dan menjanjikan jangkauan *network* yang lebih. Pada perkembangaannya sistem GSM ini mengalami kemajuan pesat dan menjadi standar yang paling populer di seluruh dunia untuk sistem seluler.

Penggunaan alokasi frekuensi 900 MHz oleh GSM ini diambil berdasarkan rekomendasi GSM (*Gropue Special Mobile*) *mitteci* yang merupakan salah satu grup kerja pada *Conference Europe'ene Postes des Telecommunication* (CEPT). Namun pada akhirnya untuk alasan *marketing* GSM berubah namanya menjadi *The Global System for Mobile Communication*, sedangkan standar teknisnya diambil dari *European Technical Standards Institute* (ETSI)

GSM pertama kali diperkenalakan di Eropa pada tahun 1991 kemudian pada akhir 1993, beberapa negara non Amerika seperti Amerika Selatan, Asia dan Australia mulai mengadopsi GSM yang akhirnya menghasilkan standar baru yang mirip yaitu DCS1800, yang mendukung *Personal Communiction Service* (PCS) pada frekuensi 1,8 Ghz sampai 2 Ghz.

2.4.3.1 Arsitektur Jaringan GSM

Secara garis besar terdiri dari 4 subsistem yang terkoneksi dan berinteraksi antar sistem dan dengan *user* melalui *network interface*, subsistem tersebut adalah:



Gambar 2.3 Arsitektur Jaringan GSM

Arsitektur jaringan GSM terdiri atas:

1. *Mobile System*

Merupakan perangkat yang digunakan oleh pelanggan untuk melakukan pembicaraan. Terdiri atas *Mobile Equipment* (ME) dan *Subscriber Identity Module* (SIM). ME merupakan terminal transmisi radio yang dilengkapi dengan *International Mobile Equipment Identity* (IMEI), sedangkan SIM berisi nomor identitas pelanggan untuk masuk ke jaringan operator GSM.

2. *Base Station*

Terdiri atas *Base Station Controller, Base Transceiver Station* dan *Transcoder* (XCDR) Dimana fungsi dari BSS adalah mengontrol tiap–tiap BTS yang terhubung kepada BS. Fungsi BTS adalah untuk berhubungan langsung dengan MSC dan juga berfungsi sebagai pengirim dan penerima sinyal. XCDR berfungsi untuk mengkompres data atau suara keluaran dari MSC (64 Kbps) menjadi 16 Kbps ke arah BSC dan sebaliknya untuk efisiensi kanal transmisi.

3. *Network Subsystem*

NSS berfungsi sebagai *switching* pada jaringan GSM, memanajemen jaringan, sebagai *interface* antara jaringan GSM dengan jaringan lainnya. Komponen NSS pada jaringan GSM terdiri dari:

a. *Mobile Switching Center* (MSC) bertugas mengatur komunikasi antar pelanggan dan *user* jaringan telekomunikasi lainnya.

b. *Home Location Register* (HLR) merupakan *database* yang berisi data pelanggan yang tetap suatu wilayah cakupan. Data-data tersebut antara lain, layanan pelanggan, *service* tambahan dan informasi mengenai lokasi pelanggan yang paling akhir.

c. *Visitor Location Register* (VLR) merupakan *database* yang berisi informasi sementara mengenai pelanggan yang melakukan *mobile* (*roaming*) dari area cakupan lain.

d. *Authentication Center* (AuC) berisi *database* yang bersifat rahasia yang disimpan dalam bentuk format kode untuk pengamanan dan pengontrolan penggunaan sistem seluler yang sah dan mencegah pelanggan yang melakukan kecurangan.

e. *Equipment Identity Register* (EIR) merupakan *database* terpusat yang berfungsi untuk validasi *Internasional Mobile Equipment Identity* (IMEI).

4. *Network Management System*

Operation and Maintenance Center (OMC) sebagai pusat pengontrolan operasi dan pemeliharaan jaringan. Fungsi utamanya mengawasi alarm perangkat dan perbaikan terhadap kesalahan operasi.

Network Management Centre (NMC) berfungsi untuk pengontrolan operasi dan pemeliharaan jaringan yang lebih besar dari OMC.

2.4.4 Sensor Accelerometer

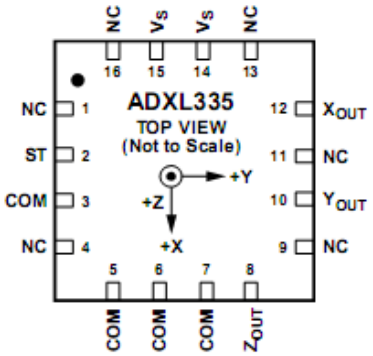
Sensor *accelerometer* adalah alat yang digunakan untuk mengukur percepatan, mendeteksi dan mengukur getaran dan mengukur percepatan akibat gravitasi (*inklinasi*). Sensor *accelerometer* mengukur percepatan akibat gerakan benda yang melekat padanya. *Accelerometer* dapat digunakan untuk mengukur getaran pada kendaraan bermotor, mesin, bangunan, dan instalasi pengamanan. Sensor *accelerometer* juga dapat diaplikasikan pada pengukuran aktivitas gempa bumi dan peralatan-peralatan elektronik, seperti permainan 3 dimensi, *mouse* komputer, dan telepon. Untuk aplikasi yang lebih lanjut, sensor ini banyak digunakan untuk keperluan navigasi. Percepatan merupakan suatu keadaan berubahnya kecepatan terhadap waktu. Bertambahnya suatu kecepatan dalam suatu rentang waktu disebut juga percepatan (*acceleration*). Jika kecepatan semakin berkurang dari pada kecepatan sebelumnya, disebut *deceleration*. Percepatan juga bergantung pada arah/orientasi karena merupakan penurunan kecepatan yang merupakan besaran vektor. Berubahnya arah pergerakan suatu benda akan menimbulkan percepatan pula.

2.4.4.1 Accelerometer ADXL335

Accelerometer ADXL335 merupakan *accelerometer* 3-axis (sumbu x, y, dan z). Sumbu arah gerak yang dideteksi untuk diukur adalah sumbu arah gerak x, y, dan z. Gambar 2.4 adalah gambar modul *accelerometer* ADXL335. Sementara gambar 2.5 adalah gambar deskripsi pin-nya. *Accelerometer* ADXL335 memiliki 16 buah kaki yang masing-masing fungsinya dijelaskan pada tabel 2.5.



Gambar 2.4 Modul *Accelerometer* ADXL335



Gambar 2.5 Konfigurasi Pin *Accelerometer* ADXL335

Tabel 2.5 Deskripsi Pin *Accelerometer* ADXL335

Pin	Nama	Keterangan
1	NC	*No Connect*
2	ST	*Self Test*
3	COM	*Common*
4	NC	*No Connect*
5	COM	*Common*
6	COM	*Common*
7	COM	*Common*
8	Zout	Z *Channel Output*
9	NC	*Self Test*
10	Yout	Y *Channel Output*
11	NC	*No Connect*
12	Xout	X *Channel Output*
13	NC	*No Connect*
14	Vs	*1.8v-3.6v*
15	Vs	*1.8v-3.6v*
16	NC	*No Connect*
EP	*Expose Pad*	*for Mechanical Integrity*

Fitur yang dimiliki oleh *accelerometer* ADXL335 adalah sebagai berikut: *Accelerometer* 3-axis dalam 1 chip IC. Dimensi, 4mm×4mm×1.45mm. Operasi dengan *power supply* tunggal 1.8 volt–3.6 volt.

Bandwidth kerja *accelerometer* dapat diatur dengan menggunakan kapasitor Cx, Cy, dan Cz yang dipasang secara seri dengan pin X*out*, Y*out*, dan Z*out*. Rentang *bandwidth* dari 0.5 Hz hingga 1600 Hz untuk sumbu X dan Y, sedangkan 0.5 Hz hingga 550 Hz untuk sumbu Z. Rentang *bandwith* tersebut dapat dipilih sesuai dengan aplikasi sistem yang dibangun.

2.5 Perangkat Lunak

Perangkat lunak adalah aplikasi penunjang dalam pembuatan program yang akan digunakan dalam memprogam sebuah alat dan membuat aplikasi *interface*, di bawah ini adalah perangkat lunak penunjang dalam penelitian tugas akhir ini.

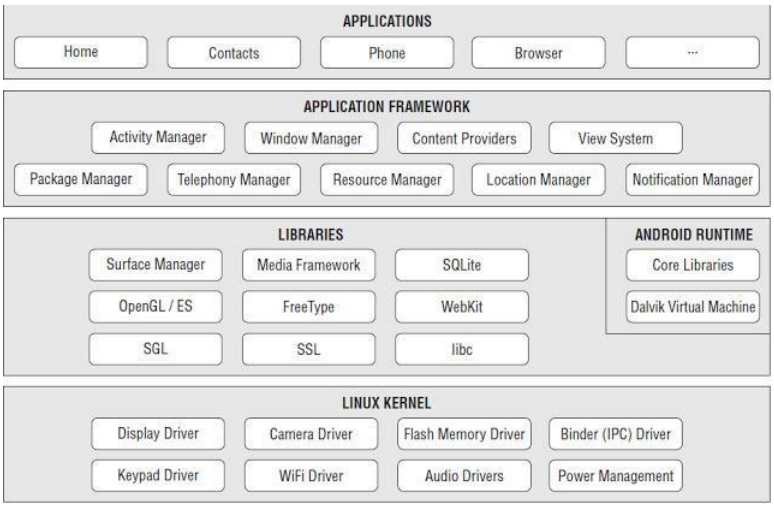
2.5.1 Android

Android adalah sistem operasi yang digunakan di *smartphone* dan juga tablet PC. Fungsinya sama seperti sistem operasi Symbian di Nokia, iOS di Apple dan BlackBerry OS. Android tidak terikat ke satu merek *handphone* saja, beberapa vendor terkenal yang sudah memakai Android antara lain Samsung, Sony Ericsson, HTC, Nexus, dan Motorolla.

Android pertama kali dikembangkan oleh perusahaan bernama Android Inc., dan pada tahun 2005 di akuisisi oleh raksasa Internet Google. Android dibuat dengan basis kernel Linux yang telah dimodifikasi, dan untuk setiap *release*-nya diberi kode nama berdasarkan nama hidangan makanan. Keunggulan utama Android adalah gratis dan *open source*, yang membuat *smartphone* Android dijual lebih murah dibandingkan dengan Blackberry atau iPhone meski fitur (*hardware*) yang ditawarkan Android lebih baik. Beberapa fitur utama dari Android antara lain *WiFi hotspot*, *Multi-touch*, *Multitasking*, GPS, accelerometers, *support java*, mendukung banyak jaringan (GSM/EDGE, IDEN, CDMA, EV-DO, UMTS,Bluetooth, Wi-Fi, LTE & WiMAX) serta juga kemampuan dasar**.**

Arsitektur Android

Arsitektur android menunjukkan komponen-komponen utama yang terdapat pada sistem operasi android. Berikut ini arsitektur android yang dilihat pada Gambar 2.6.



Gambar 2.6 Arsitektur Android

1. *Applications* dan *Widgets*

Applications dan *Widgets* ini adalah layer dimana pengguna berhubungan dengan aplikasi saja, dimana biasanya pengguna men-*download* aplikasi dan pengguna melakukan instalasi dan menjalankan aplikasi tersebut. Di *layer* terdapat aplikasi inti termasuk klien *email*, program SMS, kalender, peta, *browser*, kontak dan lain-lain. Semua aplikasi dikembangkan menggunakan bahasa pemrograman java.

2. *Applications Framework*

Applications frameworks ini adalah *layer* dimana para pengembang aplikasi melakukan pengembangan atau 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.

3. *Libraries*

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

- *Libraries* media untuk pemutaran media *audio* dan *video*.
- *Libraries* untuk manajemen tampilan.
- *Libraries graphics* mencakup SGL dan OpenGL untuk grafis 2D dan 3D.
- *Libraries* SQLite untuk dukungan *database*.
- *Libraries* SSL dan WebKit terintegrasi dengan web *browser* dan *security*.
- *Libraries* LiveWebcore mencakup modern web *browser* dengan *engine embedded wev view*.
- *Libraries* 3D yang mencakup implementasi OpenGL ES 1.0 API's.

4. *Android Run Time*

Layer yang membuat aplikasi android dapat dijalankan dimana dalam prosesnya menggunakan implementasi linux. *Dalvik Virtual Machine* (DVM) merupakan mesin yang membuat dasar kerangka aplikasi android. Didalam android *run time* dibagi kedalam dua bagian yaitu:

- *Core Libraries*: Aplikasi android dibangun dalam bahasa java, sementara dalvik sebagai virtual mesinnya bukan *virtual machine java*, sehingga diperlukan sebuah *libraries* yang berfungsi untuk menterjemahkan bahasa java atau c yang ditangani oleh *core libraries*.

- *Dalvik Virtual Machine*: Virtual mesin berbasis register yang dioptimalkan untuk menjalankan fungsi-fungsi secara efiesien, dimana merupakan pengembangan yang mampu membuat linux kernel untuk menjalankan *threading* dan manajemen tingkat rendah.

5. *Linux Kernel*

Linuk kernel adalah *layer* dimana inti dari operating sistem dari android itu berada. Berisi file-file sistem yang mengatur sistem *processing, memory, resource, drivers* dan sistem-sistem android lainnya. Linux kernel yang digunakan android adalah linux kernel *release*. Linux Kernel menyediakan *Driver* Layar, kamera, *keypad*, *WiFi*, *flash memory*, audio dan IPC (*Interprocess Communication*) untuk mengatur aplikasi dan keamanan. Kernel juga bertindak sebagai lapisan abstrak antara *hardware* dan *software stack*-nya.

2.5.2 Java

Java adalah bahasa pemrograman berorientasi objek murni yang dibuat berdasarkan kemampuan-kemampuan terbaik bahasa pemrograman objek sebelumnya (C++, Ada, Simula). Java merupakan bahasa pemrograman yang dapat dijalankan di berbagai komputer termasuk telepon genggam (*smartphone*).

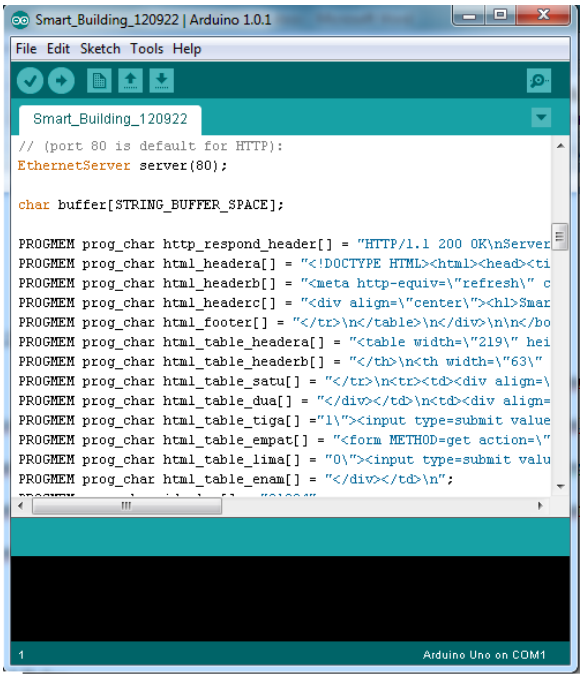
Selain itu, banyak juga program-program lain yang dapat digunakan untuk membuat program java, seperti Eclipse, NetBeans, JBuilder, JCreator, J++, dan sebagainya. Saat ini distribusi Java dan kelas pendukungnya dibagi dalam tiga bagian yang masing-masing memiliki konsentrasi tersendiri yaitu:

1. *Java 2 Standart Edition* (J2SE), untuk aplikasi desktop.
2. *Java 2 Enterprise Edition* (J2EE), untuk aplikasi server.
3. *Java 2 Micro Edition* (J2ME), untuk piranti dengan kemampuan terbatas.

2.5.3 Android Development Enviroment

Arduino Development Environment terdiri dari *editor teks* untuk menulis kode, sebuah area pesan, sebuah konsol, sebuah *toolbar* dengan tomol-tombol untuk fungsi yang umum dan beberapa menu. *Arduino Development Environment* terhubung ke arduino *board* untuk meng-*upload* program dan juga untuk berkomunikasi dengan arduino *board*.

Perangkat lunak yang ditulis menggunakan *Arduino Development Environment* disebut *sketch. Sketch* ditulis pada editor teks. *Sketch* disimpan dengan file berekstensi ino. Area pesan memberikan memberikan informasi dan pesan *error* ketika kita menyimpan atau membuka *sketch.* Konsol menampilkan *outputteks* dari *Arduino Development Environment* dan juga menampilkan pesan *error* ketika kita mengkompile *sketch.* Pada sudut kanan bawah dari jendela *Arduino Development Environment* menunjukkan jenis *board* dan *port* serial yang sedang digunakan. Tombol *toolbar* digunakan utuk mengecek dan mengupload *sketch,* membuat, membuka atau menyimpan *sketch*, dan menampilkan serial monitor.



Gambar 2.7 *Arduino Development Environment*

Berikut ini adalah tombol-tombol *toolbar* serta fungsinya:

- *Verify* mengecek *error* pada kode program.
- *Upload* meng-*compile* dan meng-*upload* program ke Arduino *board*.
- *New* membuat *sketch* baru.
- *Open* merupakan sebuah menu dari seluruh *skatch* yang berada di dalam *sketchbook.*

- *Save* menyimpan *sketch.*
- Serial monitor membuka serial monitor.

Dalam lingkungan arduino digunakan sebuah konsep yang disebut *sketchbook,* yaitu tempat standar untuk menyimpan program. *Sketch* yang ada pada *sketchbook* dapat dibuka dari menu *File >Sketchboo*k atau dari tombol *open* pada *toolbar*. Ketika pertama kali menjalankan *arduino development environment*, sebuah direktori akan dibuat secara otomatis untuk tempat penyimpana *sketchbook*. Kita dapat melihat atau mengganti lokasi dari direktori tersebut dari menu *File>Preferences*.

Serial monitor menampilkan data serial yang sedang dikirim dari arduino *board.* Untuk mengirim data ke *board,* masukkan teks dan klik tombol *send* atau tekan *enter* pada *keyboard*.

Sebelum meng-*upload* program, kita perlu mengatur jenis *board* dan *port* serial yang sedang kita gunakan melalui menu *Tools>Board* dan *Tools>Serial Port*. Pemilihan *board* berguna untuk mengeset parameter (contohnya: kecepatan mikrokontroler dan *baud rate*) yang digunakan ketika meng-*compile* dan meng-*upload sketch*.

Setelah memilih *board* dan *port* serial yang tepat, tekan tombol *upload* pada *toolbar* atau pilih menu *File>Upload*. Arduino *board* akan me-*reset* secara otomatis dan proses *upload* akan dimulai. Pada kebanyakan *board,* LED RX dan TX akan berkedip ketika program sedang di-*upload*. *Arduino development environment* akan menampilkan pesan ketika proses *upload* telah selesai, atau menampilkan pesan *error*.

Ketika sedang meng-*upload* program, arduino *bootloader* sedang digunakan, Arduino *bootloader* adalah sebuah program kecil yang telah ditanamkan pada mikrokontroler yang berada pada arduino *board*. *Bootloader* ini mengijinkan kita meng-*upload* program tanpa menggunakan perangkat keras tambahan.

Bahasa Pemrograman

Arduino *board* merupakan perangkat yang berbasiskan mikrokontroler. Perangkat lunak merupakan komponen yang membuat sebuah mikrokontroler dapat bekerja. Arduino *board* akan bekerja sesuai dengan perintah yang ada dalam perangkat lunak yang ditanamkan.

Bahasa pemrograman arduino adalah bahasa pemrograman utama yang digunakan untuk membuat program untuk arduino *board*. Bahasa pemrograman arduino menggunakan bahasa pemrograman C sebagai dasarnya.

Struktur setiap program dalam arduino *board* terdiri dari dua fungsi utama yaitu *setup*() dan *loop*(). Instruksi yang berada dalam fungsi *setup*() dieksekusi hanya sekali, yaitu ketika arduino *board* pertama kali dihidupkan. Biasanya instuksi yang berada pada fungsi *setup*() merupakan konfigurasi dan inisialisasi dari arduino *board*. Instruksi yang berada pada fungsi *loop*() dieksekusi berulang-ulang hingga arduino *board* dimatikan (catu daya diputus). Fungsi *loop*() merupakan tugas utama dari arduino *board*. Jadi setiap program yang menggunakan bahasa pemrograman arduino memilliki struktur sebagai berikut:

```
void setup()
{
  // perintah-perintah untuk konfigurasi dan inisialisasi arduino board
}

void loop()
{
  //perintah-perintah utama arduino board
}
```

Program diatas dapat dianalogikan dalam bahasa C sebagai berikut:

```
void setup(void); // prototipe fungsi setup
void loop(void); // prototipe fungsi loop
int main(void) {
setup(); //
while(1) {
loop(); // ulangi terus menerus
}
return 0; //bagian ini tidak akan pernah dieksekusi
}
```

\

2.5.4 Eclipse IDK

Eclipse merupakan sebuah IDE (*Integrated Development Environment*) yang difokuskan pada pengembangan perangkat lunak dan dapat dijalankan disemua *platform.*

Berikut adalah sifat dari eclipse:

1. *Multi-platform*: Dapat berjalan pada berbagai *platform* seperti, Microsoft Windows, Linux, Solaris, AIX, HP-UX, dan Mac OS X.
2. *Multi-language*: Eclipse dikembangkan dengan bahasa pemrograman java, akan tetapi eclipse mendukung pengembangan aplikasi berbasis bahasa pemrograman lainnya, seperti C/C++, Cobol, Phyton, Perl, PHP, dan lain sebagainya.
3. *Multi-role*: Selain sebagai IDE untuk pengembangan aplikasi, eclipse pun bisa digunakan untuk aktivitas dalam siklus pengembangan perangkat lunak, seperti dokumentasi, tes perangkat lunak, pengembangan web, dan lain sebagainya

Eclipse pada saat ini merupakan salah satu IDE yang difavoritkan karena gratis dan *open source,* yang berarti setiap orang boleh melihat kode program perangkat lunak. Selain itu, kelebihan dari eclipse yang membuatnya popular adalah kemampuannya untuk dapat dikembangkan oleh pengguna dengan komponan yang dinamakan *plug-in*.

2.5.5 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* android menggunakan bahasa pemrograman java. Sebagai

platform aplikasi-netral, android memberikan kesempatan unutk membuat aplikasi yang kita butuhkan yang bukan merupakan aplikasi bawaan *hadphone* atau *smartphone*. Beberapa fitur-fitur android yang paling penting adalah:

1. *Framework*: aplikasi yang mendukung pengganti komponen dan *reusable*.
2. *Dalvik Virtual Machine* dioptimalkan untuk perangkat *mobile*.
3. *Integrated browser* berdasarkan *engine open source* WebKit.
4. Grafis yang dioptimalkan dan didukung oleh libraries grafis 2D dan grafis 3D.
5. SQLite untuk penyimpanan data.
6. Media *Support* yang mendukung audio, video, dan gambar (MPEG4, H.264, MP3, AAC, AMR, JPG, PING, GIF), GSM *Telephony* (tergantung *hardware*).
7. Bluetooth, EDGE, 3G, dan WiFi.
8. Kamera, GPS, Kompas, dan *Accelerometer.*
9. Lingkungan *Development* yang lengkap dan termasuk pernagkat emulator, *tools* untuk *debugging*, profil dan kinerja memori, dan *plugin* untuk IDE Eclipse.

Untuk *source* SDK Android ini dapat dilihat dan didownload langsung di situs resmi pengembang SDK Android di http://www.developer.android.com.

2.5.6 Android Development Tool

Android Development Tools (ADT) adalah *plugin* untuk Eclipse IDE yang dirancang khusus untuk memberikan *integrated environment* yang kuat untuk membuat aplikasi android. ADT memberikan kemampuan kepada eclipse untuk membuat projek baru android secara cepat, membuat aplikasi *user interface*, menambahkan komponen berdasarkan android *framework* API, melakukan *debugging* aplikasi yang di buat dengan menggunakan Android SDK *tools* dan bahkan melakukan distribusi aplikasi yang dibuat.

Pembuatan aplikasi android dengan eclipse beserta ADT sangat dianjurkan karena merupakan cara tercepat untuk memulai membuat projek android. Dengan

disediakannya *project setup,* serta *tools* yang sudah terintegrasi di dalamnya, *custom* XML editor, dan *debugging* beserta *output* dalam emulator yang sudah disediakan android SDK, mempermudah para pengembang aplikasi android dalam pembuatan aplikasinya.

2. 6 Perhitungan Presentase Akurasi

Setelah alat berhasil di selesaikan, maka tahap selanjutnya adalah proses pengukuran pada alat, setelah diperoleh data pengukuran kemudian dilakukan perhitungan guna melakukan performa alat, terdapat 4 rumusan untuk mengetahui perhitungan eror [14]:

1. *Error Absolute*

   Eror atau kesalahan adalah penyimpanan nilai dari suatu pengukuran terhadap harga sebenarnya, dapat dinyatakan dalam:

   e=|Yn-Xn|…….(1)

   Dimana, e = *error absolute*, Yn = nilai sebenarnya, dan Xn = nilai hasil pengukuran.

2. Presentase Eror

   Data-data hasil pengukuran kemudian dihitung untuk mengetahui nilai eror dengan menggunakan persamaan (2).

   Presentase Eror %= $\frac{Yn-\mathrm{Xn}}{Yn}\ x\ 100\%$ ……. (2)

3. Perhitungan Akurasi

   Akurasi didefinisikan sebagai keterdekatan hasil pengukuran suatu alat ukur terhadap suatu nilai yang benar. Untuk mendapatkan nilai akurasi relatif digunakan persamaan sebagai berikut:

   A=1- $\frac{Yn-Xn}{Yn}$ ……. (3)

   Dimana, Xn = nilai hasil pengukuran, Yn = nilai sebenarnya, dan A = akurasi.

4. Presentase Akurasi

   Akurasi dapat pula dinyatakan dalam prosen akurasi sebagai berikut:

   Presentase Akurasi = 100% - presentase eror…….(4)

   Nilai akurasi pada pengujian alat ini dinyatakan dalam persen dengan menggunakan persamaan (4).