

BAB 2

LANDASAN TEORI

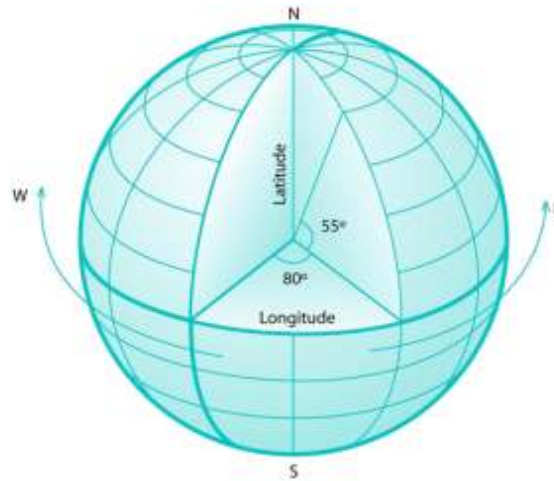
2.1 Global Positioning System(GPS)

Global Positioning System (GPS) adalah sistem yang berfungsi sebagai sistem navigasi global yang dapat menerima informasi dari satelit. Informasi yang diterima lebih dari tiga satelit GPS mengorbit di atas permukaan bumi. Satelit GPS memancarkan sinyal GPS yang memungkinkan penerima sinyal GPS untuk mendapatkan informasi berupa lokasi penerima, arah, dan kecepatan.

Satelit GPS diuji coba dan pertama kali diluncurkan pada tahun 1978. Pada saat itu GPS sudah menjadi alat bantu navigasi diseluruh dunia dan menjadi alat yang penting untuk melakukan pembuatan peta dan survey wilayah. GPS juga menyediakan acuan waktu yang tepat yang digunakan di banyak aplikasi termasuk studi ilmu gempa bumi dan sinkronisasi jaringan telekomunikasi.

Sebuah penerima sinyal GPS mengkalkulasi posisinya dengan mengukur jarak dirinya dengan tiga atau lebih satelit GPS. Dengan mengukur waktu tunda antara pengiriman dan penerimaan sinyal radio dari masing-masing GPS dan mengetahui kecepatan sinyal maka didapat jarak ke masing-masing satelit tersebut. Sinyal tersebut juga berisi informasi mengenai posisi satelit. Dengan menentukan posisi dan jarak berdasarkan paling tidak tiga satelit, penerima GPS dapat menghitung posisinya menggunakan mekanisme *trilateration*. Untuk mendapatkan waktu yang tepat,

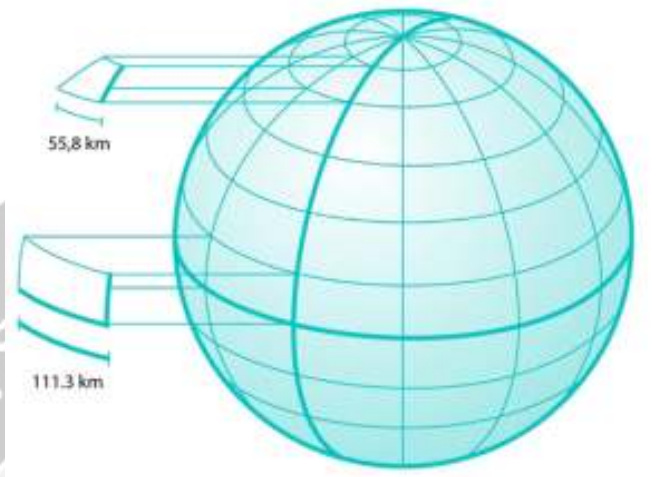
penerima GPS cukup membandingkan dan mengamati satu atau lebih satelit tambahan untuk memperbaiki kesalahan waktu yang diterima. (Wikipedia, en.wikipedia.org/wiki/Global_Positioning_System)



Gambar 2.1 Bumi Dengan Garis Lintang dan Garis Bujur.

(http://sw.nokia.com/id/175bf8e6-alf5-4d3d-6fc936506a6b/MIDP_Location_API_Developers_Guide_v2_0_en.pdf)

Seperti yang terlihat pada gambar 2.1, garis Lintang (*Latitude*) menjadi semakin kecil ketika mendekati kutub. Garis Katulistiwa (0° *Latitude*) memiliki panjang sekitar 111.3 km, sedangkan pada 60° *Latitude* hanya memiliki panjang sekitar 55.8 km. Hal ini mempersulit perhitungan panjang dari garis *Latitude* (http://sw.nokia.com/id/175bf8e6-alf5-4d3d-6fc936506a6b/MIDP_Location_API_Developers_Guide_v2_0_en.pdf)

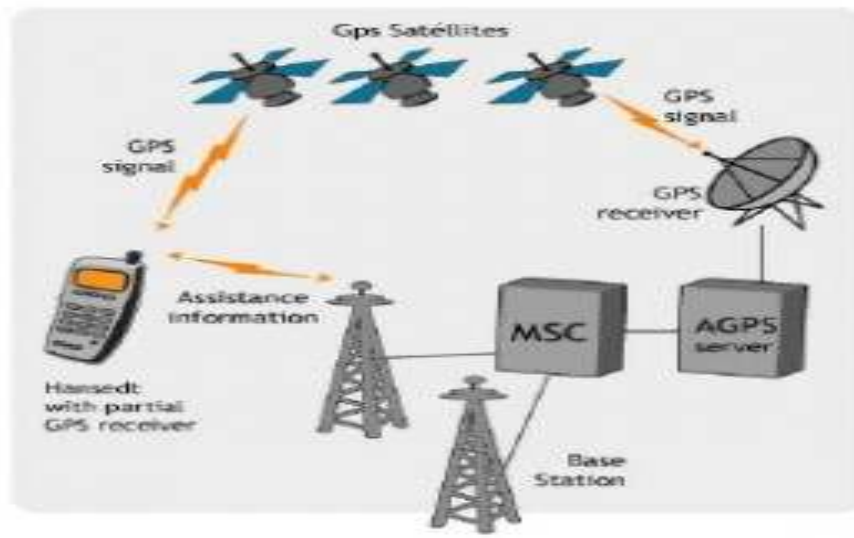


Gambar 2.2 Perbedaan Panjang garis Katulistiwa dengan garis Latitude pada 60° Lintang Utara (http://sw.nokia.com/id/175bf8e6-alf5-4d3d-6fc936506a6b/MIDP_Location_API_Developers_Guide-v2_0_en.pdf)

2.2 Assisted Global Positioning System (A-GPS)

Assisted-Global Positioning System (A-GPS) merupakan penyempurnaan dari GPS sebagai satelit penentu posisi di belahan bumi. Satelit GPS yang dimiliki bumi mempunyai konstelasi 24 satelit dalam enam orbit yang mendekati lingkaran, setiap orbit ditempati oleh 4 buah satelit dengan interval yang tidak sama. Orbit satelit GPS beriklinasi 55° terhadap equator dengan ketinggian rata-rata dari permukaan bumi sekitar 20.000 km.

Metode Advanced Positioning yang terdapat pada A-GPS merupakan metode penentuan posisi yang paling tinggi akurasinya dibandingkan metode penentuan posisi lainnya, sehingga A-GPS jauh lebih efisien dan efektif dalam mengakses informasi dari satelit karena tidak perlu mencari data satu persatu dari ke 24 satelit yang ada, namun A-GPS telah mengetahui sasaran (satelit) mana yang dibutuhkan atau dituju.



Gambar 2.2 A-GPS

A-GPS pertama kali dikeluarkan oleh US FCC (United States Federal Communications Commission), yakni suatu badan komunikasi Amerika. Pada masa itu digunakan untuk mengakomodir panggilan darurat (911) agar penelpon dapat terlacak dengan lebih akurat, baik ketika di dalam maupun di luar gedung.

Pada masa sekarang ini, deteksi posisi bukan lagi hanya untuk kebutuhan militer ataupun eksplorasi, namun sudah menjadi kebutuhan masyarakat sipil untuk menentukan lokasi dan penggunaan deteksi telah menjadi kebutuhan di dunia telekomunikasi.

Adapun kegunaan A-GPS antara lain

1. A-GPS menawarkan solusi terakurat dari metode-metode yang telah ada sebelumnya. A-GPS merupakan layanan yang menggabungkan sistem GPS dengan layanan GSM.
2. Layanan ini berguna untuk menjembatani kekurangan dan kelebihan GPS dan LBS. LBS adalah Location

Based Service yang sebenarnya adalah salah satu layanan tambahan dari selular GSM.

3. A-GPS menjadikan proses akses informasi menggunakan satelit menjadi lebih mudah dan cepat.

2.2 Sistem Koordinat Geografi

Sistem koordinat geografi digunakan untuk menunjukkan suatu titik di Bumi berdasarkan *Latitude* (garis lintang) dan *Longitude* (garis bujur). Garis Lintang adalah garis vertical yang mengukur sudut antara suatu titik dengan garis katulistiwa. Pada gambar 2.1, garis horizontal menunjukkan garis lintang yang berada diantara dua kutub bumi. Titik di utara garis katulistiwa dinamakan garis Lintang Utara sedangkan titik di selatan katulistiwa dinamakan Lintang Selatan. Kutub utara bumi berada pada 90° Lintang Utara ($+90^\circ$), kutub selatan bumi berada pada 90° Lintang Selatan (-90°), sedangkan garis katulistiwa berada pada 0° . Lokasi yang berada di bawah katulistiwa memiliki *Latitude* positif (0° sampai dengan 90°) dan lokasi yang berada di bawah katulistiwa memiliki *Latitude* negatif (0° sampai dengan -90°).

Garis Bujur yaitu horizontal yang mengukur sudut antara suatu titik dengan titik nol di bumi yaitu Greenwich di London Britania Raya yang merupakan titik bujur 0° atau 360° yang diterima secara internasional. Titik di barat bujur 0° dinamakan bujur barat dan memiliki *Longitude* negatif (0° sampai dengan -180°), sedangkan titik di timur 0° dinamakan bujur timur dan memiliki *Longitude* Positif (0° sampai dengan $+180^\circ$).

Suatu titik di bumi dapat di deskripsikan dengan menggabungkan kedua pengukuran tersebut. (Wikipedia Indonesia)

2.3 Aplikasi-aplikasi Berbasis GPS

Seiring perkembangan teknologi, semakin banyak aplikasi berbasis GPS yang dikembangkan. Aplikasi ini dikembangkan di berbagai bidang, yaitu : Militer, Navigasi, Survey dan Pemetaan, dan bidang lain.

2.3.1 Militer

Di bidang militer, aplikasi yang menggunakan sistem GPS mampu memberikan target yang akurat yang mencakup penentuan target misil, kendali jarak jauh atau perlengkapan perang sejenis. Maka dari itu untuk mencegah penggunaan GPS oleh musuh, pemerintah AS membatasi ekspor perangkat GPS. Biasanya manufaktur GPS Amerika hanya dapat mengekspor satelit GPS dengan batasan fungsi yaitu: ketinggian diatas 18kilometer dan hanya bisa berkeliling dengan kecepatan maksimal 514 m/s.

2.3.2 Navigasi

Banyak aplikasi GPS yang sudah dipasarkan yang sebagian besar memiliki fungsi sebagai navigasi atau petunjuk arah, diantaranya:

1. Aplikasi yang dapat menampilkan peta bergerak dan informasi tentang lokasi, kecepatan, arah, tempat tujuan dan jalan terdekat ke suatu tempat.
2. Aplikasi yang digunakan pada penerbangan, yang membantu autopilot dalam melakukan navigasi. Pilot menggunakan data GPS untuk mendapatkan informasi posisi pesawat pada saat di udara. GPS juga

digunakan untuk menghitung kecepatan angin, ketinggian awan dan informasi tempat-tempat tertentu seperti bandara alternatif atau puncak gunung, sehingga membantu pilot dalam mengambil keputusan.

3. Aplikasi yang digunakan di bidang kelautan, yang memberikan informasi mengenai kondisi danau, laut dan samudra di seluruh dunia. Selain itu terdapat aplikasi yang digunakan untuk pencarian lokasi kapal yang tenggelam.
4. Aplikasi yang digunakan di bidang pertambangan dan pertanian. Di bidang pertambangan pengeboran dapat dilakukan secara otomatis melalui kendali jarak jauh dengan bantuan GPS. Di bidang pertanian, juga menggunakan kendali jarak jauh oleh petani untuk alat pembajak dimana petani cukup melihat secara visual tampilan lahan sawah yang akan dibajak.
5. Aplikasi yang digunakan pada bidang olahraga, khususnya rally dan touring. Aplikasi ini memungkinkan pengendara untuk menentukan rute perjalanan dengan mengikuti petunjuk yang diberikan. Aplikasi ini biasanya menampilkan gambaran nyata mengenai medan yang akan dilewati, seperti gurun, rawa-rawa, pegunungan dan sebagainya.
6. Aplikasi yang digunakan oleh pejalan kaki atau seorang untu menentukan posisinya di lingkungan perkotaan maupun di pedesaan. Di area yang terisolasi, kemampuan aplikasi ini sangat

dibutuhkan ketika seseorang tersesat atau kehilangan komunikasi.

2.3.3 Survey dan Pemetaan

Untuk pemetaan dan survey aplikasi yang menggunakan sistem berbasis GPS sangat dibutuhkan, yaitu:

1. Aplikasi survey lokasi. Aplikasi ini digunakan untuk memberikan informasi wilayah sebagai referensi seseorang yang akan melakukan survey, membuat bangunan, jalan raya dan sebagainya.
2. Aplikasi pemetaan dan Sistem Informasi Geografi (GIS). Aplikasi ini memungkinkan untuk menampilkan gambaran permukaan bumi secara nyata untuk keperluan pembuatan peta, pendidikan, prakiraan cuaca dan sejenisnya. Aplikasi ini memiliki kemampuan mengurangi perbedaan posisi dengan asli sampai kurang dari 1 meter.
3. Aplikasi yang digunakan pada ilmu geologi dan geofisika. Aplikasi ini digunakan untuk menganalisis bentuk permukaan bumi, seperti gunung berapi yang aktif atau zona yang tidak bisa dijangkau, aplikasi ini mampu memberi gambaran perbedaan permukaan bumi dari waktu ke waktu sebagai interpretasi penyebab perubahan bentuk suatu tempat.
4. Aplikasi yang digunakan pada ilmu arkeologi. Aplikasi ini ahli arkeologi dalam menggali suatu lokasi. Aplikasi ini menampilkan peta tiga dimensi yang menunjukkan detail dimana masing-masing *artifact* telah ditemukan.

2.3.4 Penggunaan Di Bidang Lain

Adapun penggunaan lain dari aplikasi berbasis GPS, adalah:

1. Sebagai acuan penunjuk waktu. Banyak sistem di dunia menggunakan GPS sebagai acuan waktu yang akurat. Sistem ini biasanya dilengkapi sensor yang memberikan informasi tentang waktu peristiwa-peristiwa di bumi.
2. Aplikasi game berbasis GPS. Ketersediaan perangkat GPS yang portable telah mendorong dunia game untuk menciptakan game berbasis GPS, seperti Geocaching. Game ini menggunakan GPS untuk bepergian sepanjang garis lintang dan bujur yang spesifik untuk mencari objek-objek yang tersembunyi.
3. Aplikasi GPS Kompas. Sistem GPS ini digunakan untuk mendapatkan informasi arah mata angin. Sistem ini menggunakan tiga pasang antenna yang membentuk setiga untuk mendapatkan tiga pembacaan yang terpisah dari masing-masing satelit. Dengan posisi antenna dan beda fase dari sinyal yang diterima maka orientasi dari dua antenna dapat dihitung.
4. Aplikasi pemantau distribusi (GPS Trakcing). Aplikasi ini mampu menampilkan pergerakan seseorang, posisi barang dalam waktu tertentu dengan tujuan untuk menganalisis distribusi barang yang dilakukan distributor.
5. Aplikasi pembantu pemasaran. Banyak perusahaan pemasaran yang sudah mengubah mengombinasikan

GIS (*Geographical Information System*) dengan sistem pendukung keputusan yang membantu perusahaan untuk menentukan dimana tempat yang potensial untuk membuka cabang baru dan pemasangan iklan berdasarkan pola pemakaian jalan konsumen dan faktor lainnya.

2.4 J2ME

Java 2 Micro Edition (J2ME) merupakan sebuah kombinasi yang terbentuk antara sekumpulan interface Java yang sering disebut dengan JAVA API (*Application Programming Interface*) dengan JVM (*Java Virtual Machine*) yang didesain khusus untuk alat, yaitu JVM dengan ruang yang terbatas. Kombinasi tersebut kemudian digunakan untuk membangun aplikasi-aplikasi yang dapat berjalan di atas alat (dalam hal ini *mobile device*) (Raharjo, Hermantyo, Haryono, 2007).

Komponen-komponen J2ME terdiri dari *Java Virtual Machine* (JVM) yang digunakan untuk menjalankan aplikasi Java pada emulator atau *handheld device*, Java API (*Application Programming Interface*) dan tools lain untuk pengembangan aplikasi Java semacam emulator *Java Phone*, emulator Motorola dari *J2ME Wireless Toolkit*. Dalam pengembangan aplikasi *wireless* dengan Java, J2ME dibagi menjadi dua buah bagian, yaitu:

1. Lapisan Konfigurasi (*Configuration Layer*)

Virtual machine yang berupa konfigurasi yang menyediakan fungsi dasar dengan karakteristik yang sama. Contohnya: fungsi koneksi jaringan dan manajemen memori. J2ME mempunyai dua konfigurasi utama yaitu:

a. **Connected Limited Device Configuration (CLDC)**

CLDC diperlukan untuk pengembangan aplikasi *wireless* dengan MIDP. Implementasinya CLDC digunakan untuk program *Java* pada perangkat keras dengan ukuran memori yang terbatas, pada 160 sampai dengan 512 *Kilobyte*.

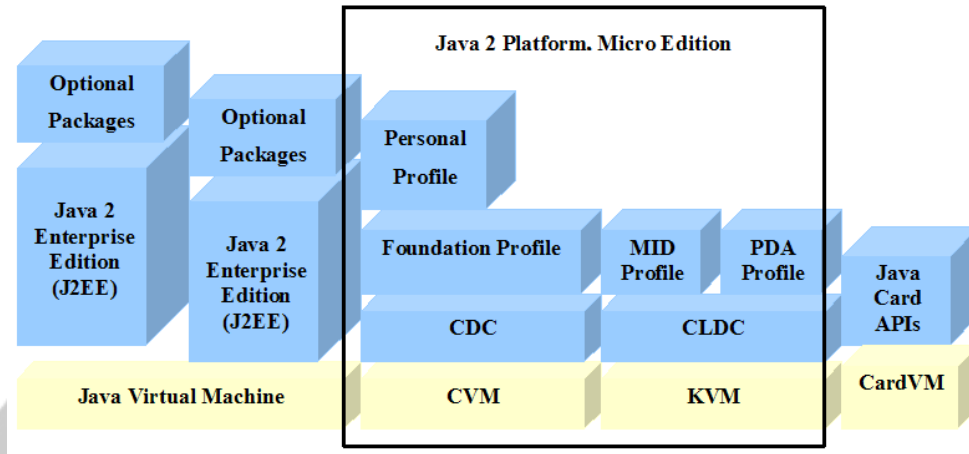
b. **Connected Device Configuration (CDC)**

CDC mampu menggunakan seluruh fitur *Java 2 Virtual Machine*. CDC juga mampu menggunakan hampir seluruh fitur *J2SE*. Besarnya permintaan CDC akan sumber daya, membuat CDC tidak bisa digunakan di banyak piranti *mobile*.

2. Lapisan Profile (Profile Layer)

Menyediakan lingkungan pustaka-pustaka API untuk membangun aplikasi *mobile* (Mardiono, 2006). *J2ME* mempunyai beberapa profil (Agung, 2005) antara lain:

- a. *Mobile Information Device Profile (MIDP)*
- b. *Foundation Profile (FP)*
- c. *Personal Profile*
- d. *Personal Digital Assistance (PDA) Profile*



Gambar 2.3 Java 2 Micro Edition (Sun Microsystem)

Dilihat dari Gambar 2.3 di atas, maka J2ME (Java 2 *Micro Edition*) mempunyai lapisan konfigurasi dan *profile* yang didukung oleh *Java Virtual Machine* (*CVirtual Machine* dan *K-Virtual Machine*).

2.5 Java API Berbasis Lokasi (Location API)

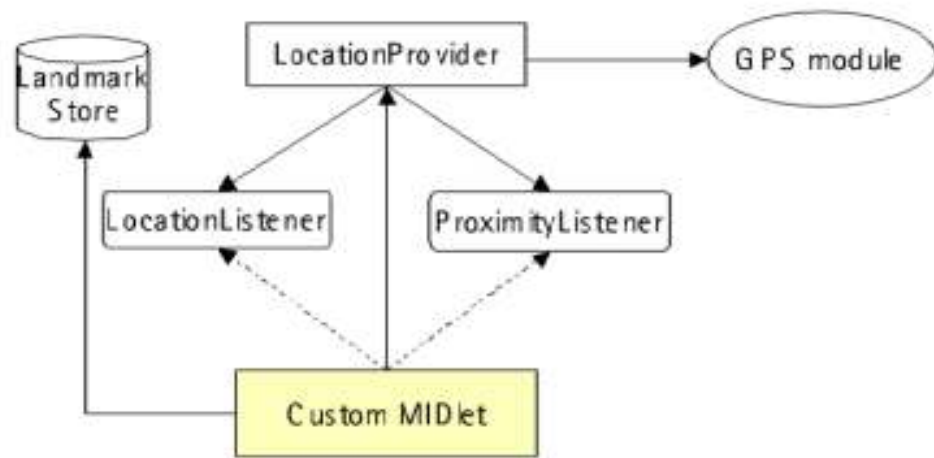
Location API dikhususkan untuk penggunaan pada platform Java 2 *Micro Edition*. API (*Application Programming Interface*) merupakan paket referensi atau library tambahan yang bisa digunakan dan dikembangkan di dalam pembuatan aplikasi berbasis lokasi (*Location Based Service*) pada perangkat mobile, seperti handphone, smartphone, PDA dan sebagainya.

Dengan pertimbangan alami dan keterbatasan yang dimiliki perangkat mobile, Location API menyediakan cara yang alami dan memungkinkan informasi lokasi bisa diterima secara mudah dan cepat. Lebih dari itu, Location API merupakan paket yang lengkap sebagai antarmuka dengan perangkat lain untuk kemudahan

penggunaan. Ada tiga layanan utama yang dimiliki oleh Location API, yaitu:

1. Mendapatkan informasi tentang lokasi suatu device.
2. Memungkinkan untuk membuat, mengedit, menyimpan dan menerima tempat-tempat tertentu yang pernah dikunjungi.
3. Memungkinkan untuk menerima informasi tentang orientasi suatu device.

Gambar dibawah menjelaskan struktur umum dari suatu aplikasi midlet yang menggunakan Modul GPS sebagai penyedia informasi lokasi. Setelah aplikasi midlet dibuat dan mampu berjalan dengan baik di lingkungan komputer (yang dilengkapi JVM), maka aplikasi tersebut bisa di uji dan dipasang di perangkat mobile yang mendukung Location api.



Gambar 2.4 contoh model aplikasi yang menggunakan Location API

Dalam pemrograman berbasis Java, Location API merupakan salah satu *runtime* spesifikasi Java (JSR-

179) dan didefinisikan di dalam package `javax.microedition.location`. Semua kelas pada Location API sudah dijadikan dalam satu paket yang sama. Tabel 2.1. sampai 2.5 memberikan penjelasan masing-masing kelas yang dimiliki Location API yang digolongkan berdasarkan fungsinya.

Tabel 2.1 kelas yang dibutuhkan dalam memilih penyedia informasi lokasi

Kelas	Definisi
Criteria	Digunakan untuk memilih penyedia informasi lokasi (Location Provider) yang terdeteksi
LocationProvider	Merepresentasikan sumber dari informasi lokasi (misal: Modul GPS)

Tabel 2.2 Kelas yang berhubungan dengan penyedia informasi lokasi

Kelas	Definisi
LocationListener	Kelas yang menerima perubahan informasi yang diberikan <i>Location Provider</i>
ProximityListener	Kelas yang digunakan untuk mendeteksi kedekatan posisi dengan suatu tempat.

Tabel 2.3 Kelas yang dibutuhkan sebagai pengukuran informasi lokasi

Kelas	Definisi
AddressInfo	Memberikan informasi alamat di suatu lokasi
Coordinate	Merepresentasikan koordinat sebagai nilai <i>latitude</i> , <i>longitude</i> , <i>altitude</i> suatu lokasi
Location	Merepresentasikan paket informasi suatu lokasi (<i>koordinat</i> , <i>kecepatan gerak device</i> , <i>alamat</i>)
LocationException	Melemparkan eksepsi (kendali kesalahan) ketika <i>Location API</i> mengalami <i>error</i> atau kesalahan
QualifiedCoordinates	Merepresentasikan koordinat sebagai nilai <i>latitude</i> , <i>longitude</i> , <i>altitude</i> suatu lokasi yang didapat dari <i>Location Provider</i>

Tabel 2.4 Kelas yang berhubungan dengan pengelolaan *Landmark*

Kelas	Definisi
Landmark	Merepresentasikan lokasi yang sudah diketahui datanya (<i>nama</i> , <i>alamat</i> , <i>koordinat</i>)

LandmarkException	Melempar eksepsi pada operasi yang berhubungan dengan pengelolaan <i>landmark</i>
LandmarkStore	Sepagai penyimpan data <i>landmark</i> dan pengelolaan <i>landmark</i> (simpan, edit, hapus dan <i>load</i>)

Tabel 2.5 Kelas yang berhubungan dengan informasi arah

Kelas	Definisi
Orientation	Merepresentasikan arah fisik dari suatu device

2.6 Google Maps API

Google menciptakan *Google Maps API* untuk memfasilitasi pembangunan suatu aplikasi peta berbasis web (*Google Maps*) yang memiliki data sesuai dengan kebutuhan pengguna.

Dengan menggunakan *Google Maps API* dimungkinkan untuk menyisipkan seluruh fasilitas dan teknologi yang ada pada *Google Maps* ke dalam suatu web site. Hal utama yang dibutuhkan dalam pembangunan aplikasi menggunakan *API* ini adalah *API key*. *API key* adalah nilai kunci atau penanda yang harus ada bila kita menggunakan fasilitas *Google Maps API*. Setiap web site yang menggunakan fasilitas *Google Maps API* memiliki nilai *API key* yang berbeda-beda.

Pembangunan aplikasi peta yang menggunakan *Google Maps API* ini menggunakan bahasa *Javascript*. Baik itu

bahasa Javascript umum maupun bahasa Javascript khusus yang dikembangkan oleh Google untuk pembangunan Google Maps (Google Javascript). Dengan menggunakan Javascript ini dapat dibangun suatu aplikasi peta berbasis web yang interaktif.

Google terus-menerus melakukan pengembangan dan penyempurnaan API ini, agar dapat dengan mudah digunakan dan memiliki layanan yang baik.

Google Maps API menyediakan kelas-kelas, tipe dan fungsi-fungsi yang dapat digunakan untuk membangun suatu aplikasi peta Google Maps.

Tabel 2.6 memberikan penjelasan masing-masing kelas dasar yang dimiliki Google Maps API.

Kelas	Definisi
GMap2	Merupakan kelas paling utama dari Google Maps API. Digunakan untuk menciptakan peta.
GMapOptions	Merepresentasikan argumen-argumen untuk kelas GMap2.
GInfoWindow	Kelas yang digunakan untuk menciptakan jendela informasi.
GInfoWindowOptions	Merepresentasikan argumen-argumen untuk kelas GInfoWindow.
GMarker	Kelas yang digunakan untuk menciptakan tanda (<i>marks</i>) pada peta.
GMarkerOptions	Merepresentasikan argumen-argumen untuk kelas GMarker.
GPolyline	Kelas yang digunakan untuk menciptakan garis atau <i>polyline</i> pada peta.

GPolylineOptions	Merepresentasikan argumen-argumen untuk kelas GPolyline.
GPolygon	Kelas yang digunakan untuk menciptakan bidang luasan pada peta.
GPolygonOptions	Merepresentasikan argumen-argumen untuk kelas GPolygon.
GIcon	Kelas yang digunakan untuk menciptakan <i>Icon</i> pada suatu <i>Marks</i> .
GLatLng	Merepresentasikan koordinat langitude, longitude.
GMapType	Merepresentasikan tipe peta yang akan ditampilkan.
GMapTypeOptions	Merepresentasikan argumen-argumen untuk kelas GMapType.
GEventListener	Kelas yang digunakan untuk mendengar dan menangani event-event.

Tabel 2.1 Kelas Dasar Google Maps API.

2.7 Framework CodeIgniter

CodeIgniter merupakan salah satu *open source framework* yang digunakan oleh *script* pemrograman web PHP (*PHP Hypertext Preprocessor*) dalam mengembangkan aplikasi web dinamis. *CodeIgniter* dapat mempercepat developer untuk membuat sebuah aplikasi web. Ada banyak library dan helper yang berguna didalamnya dan tentunya mempermudah proses development. *Codeigniter* pada awalnya ditulis oleh mas Rick Ellis, pendiri dan CEO EllisLab.com, perusahaan yang mengembangkan *codeigniter* (Upton, 2007). Saat ini, *codeigniter* dikembangkan oleh

komunitas dan disebarakan ke seluruh dunia dengan lisensi bebas. Metode yang digunakan oleh *framework CodeIgniter* disebut *Model - View - Controller* atau yang disingkat dengan sebutan *MVC*. *MVC* memisahkan antara logika pemrograman dengan presentasi. Hal ini dapat terlihat dari adanya minimalisir *script* presentasi (*HTML, CSS, JavaScript, dsb*) yang dipisahkan dari *PHP script*. Didalam folder *CodeIgniter*, *MVC* dapat kita temukan dalam folder *application*. Selain *framework CodeIgniter*, masih banyak *framework* lain yang juga menggunakan *PHP*, misalnya *CakePHP, Symphoni Project, Zend, Seagull, Prado*, dan lain sebagainya. *CodeIgniter* dirilis pertama kali pada 28 Februari 2006. Versi yang sudah mulai stabil dirilis pada 26 Juni 2008, yaitu versi 1.6.3. Sekarang versi terakhir yang telah dikeluarkan adalah 1.7.1.