

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **II.1. Mobile GIS**

Mobile GIS merupakan sebuah integrasi cara kerja perangkat lunak/ keras untuk pengaksesan data dan layanan geospasial melalui perangkat bergerak via jaringan kabel atau nirkabel. Secara umum, mobile GIS diimplementasikan pada dua area aplikasi utama yaitu Layanan Berbasis Lokasi (*Location Based Service*) dan GIS untuk kegiatan lapangan (*Field Based GIS*). Berikut ini hal-hal yang berkenaan dengan aplikasi mobile GIS :

1. Diimplementasikan pada perangkat bergerak dengan keterbatasan ruang penyimpanan, memori, dan resolusi.
2. Dapat diimplementasikan secara mandiri (*stand alone*) dengan menyimpan data dalam perangkat bergerak (untuk aplikasi sederhana), atau disesuaikan dengan arsitektur servernya (untuk aplikasi web GIS).
3. Kemampuan aplikasi mobile GIS, seperti:
  - a. Menampilkan atau melakukan navigasi.
  - b. Mengidentifikasi.
  - c. Pencarian atau *query*.
  - d. Memodifikasi nilai atribut.
  - e. Pemberian tanda atau *redline*.
  - f. Memodifikasi geometri.
  - g. Mengintegrasikan dengan data kantor.

4. Terdapat dua jenis data, yaitu koleksi data (*data collection*) dan navigasi (*navigation*).

Adapun kelebihan sistem koleksi data dengan mobile GIS adalah sebagai berikut:

- a. Dapat diintegrasikan dengan perangkat GPS, *rangefinder*, dan kamera digital.
- b. Sistem koleksi data sangat efisien, yaitu hanya dengan point dan click.
- c. Data spasial dikelola dalam dataset *referensi*.

(Gunita Mustika Hati, dkk : 2013).

## **II.2. Sistem Informasi Geografis**

Sistem informasi geografis adalah sistem berbasis komputer yang digunakan untuk menyimpan dan memanipulasi informasi-informasi geografis. Sistem informasi geografi diciptakan untuk mengumpulkan, menyimpan dan menganalisis obyek atau fenomena dimana lokasi geografis menjadi karakteristik atau kritik penting untuk analisi. Sistem informasi geografi adalah sistem berbasis komputer yang memiliki kemampuan dalam menangani data bereferensi dalam:

1. Masukkan data
2. Manajemen data
3. Manipulasi
4. Analisis

Pada awalnya data geografi hanya disajikan di atas peta dengan menggunakan simbol, garis, dan warna. Elemen-elemen geometri ini dideskripsikan di dalam legenda-nya misalnya, garis hitam tebal untuk jalan

utama, garis hitam tipis untuk jalan sekunder dan jalan-jalan yang berikutnya. Selain itu, berbagai data juga di dapat di-overlay-kan berdasarkan sistem koordinat yang sama. Akibatnya, sebuah peta menjadi media yang efektif baik sebagai alat presentasi maupun sebagai bank tempat penyimpanan data geografis. Tetapi, media peta masih mengandung kelemahan atau keterbatasan. Informasi-informasi yang tersimpan, diproses dan dipresentasikan dengan suatu cara tertentu, dan biasanya untuk tujuan tertentu pula. Tidak mudah untuk mengubah bentuk presentasi ini, sebuah peta selalu menyediakan gambar atau symbol unsure geografi dengan bentuk yang tetap atau statis meskipun diperlukan untuk kebutuhan yang berbeda. (S. Nofan Maulana Rachman : 2012)

### **II.3. Android**

*Android* adalah sistem operasi untuk *mobile device* yang awalnya dikembangkan oleh *Android Inc.* *Google Inc* membeli *Android Inc.* yang merupakan pendatang baru yang membuat peranti lunak untuk ponsel. Pada saat perilis perdana *Android*, 5 November 2007, *Android* bersama *open handset alliance* menyatakan mendukung pengembangan *open source* pada perangkat *mobile*. Di lain pihak, google merilis kode-kode *android* di bawah lisensi *apache*. *Android* dibuat berdasarkan kernel *Linux* yang dimodifikasi. Aplikasi *Android* ditulis dengan bahasa *Java*, menggunakan *Java Core Libraries*. Aplikasi *Android* dijalankan di atas VM bernama *Dalvik Virtual Machine*.

Pada saat ini, sudah banyak vendor *smartphone* yang memproduksi berbasis *android*, hal ini terjadi karena android adalah sistem operasi yang *open*

*source* sehingga bebas di distribusikan dan dipakai oleh *vendor* manapun. *Android* itu sendiri sangat lengkap baik dari segi sistem operasi, aplikasi dan tool pengembangan, *market* aplikasi serta dukungan yang sangat tinggi dari komunitas *open source* didunia. (Eko Subiyantoro : 2013)

#### **II.4. Google Map API**

Maps adalah aplikasi dan teknologi layanan pemetaan berbasis web yang menampilkan citra satelit beresolusi tinggi untuk peta jalan, rute sepeda, pejalan kaki, mobil serta penentuan lokasi bisnis di kota-kota berbagai negara seluruh dunia yang di sediakan oleh *Digital Globe* dengan satelitnya *QuickBird*, serta data dari *Geographic Information System (GIS)* buatan *Tele Atlas*, *NAVTEQ*, dan *MapABC*. Kemudian Google membuat *Google Maps API* untuk para pengembang web yang ingin menyertakan *Google Maps* ke situs mereka secara gratis. (Eko Tresno Prabowo, dkk : 2012).

#### **II.5. MySQL**

MySQL (bisa dibaca dengan mai-es-ki-el atau bisa juga mai-se-kuel) adalah suatu perangkat lunak database relasi (*Relational Database Management System* atau DBMS), seperti halnya *ORACLE*, *POSTGRESQL*, *MSSQL*, dan sebagainya. *SQL* merupakan singkatan dari *Structure Query Language*, didefinisikan sebagai suatu sintaks perintah-perintah tertentu atau bahasa program yang digunakan untuk mengelola suatu database. Jadi MySQL adalah softwarenya dan SQL adalah bahasa perintahnya. (Anisya ; 2013).

## II.6. Haversine

Posisi di bumi dapat direpresentasikan dengan posisi garis lintang (*latitude*) dan bujur (*longitude*). Untuk menentukan jarak antara dua titik di bumi berdasarkan letak garis lintang dan bujur, ada beberapa rumusan yang digunakan. Semua rumusan yang digunakan berdasarkan bentuk bumi yang bulat (*spherical earth*) dengan menghilangkan faktor bahwa bumi itu sedikit elips (*elipsodial factor*). (Wahyuni Eka Sari. dkk : 2013)

$$\begin{aligned}\Delta lat &= lat2 - lat1 \\ \Delta long &= long2 - long1 \\ a &= \sin^2(\Delta lat/2) + \cos(lat1) \cdot \cos(lat2) \cdot \sin^2(\Delta long/2) \\ c &= 2 \cdot \text{atan2}(\sqrt{a}, \sqrt{1-a}) \\ d &= R \cdot c\end{aligned}$$

**Gambar II.1. Rumus Haversine**

Sumber (Wahyuni Eka Sari. dkk : 2013)

Keterangan :

R = jari-jari bumi sebesar 6371(km)

$\Delta lat$  = besaran perubahan latitude

$\Delta long$  = besaran perubahan longitude

c = kalkulasi perpotongan sumbu

d = jarak (km)

## II.7. Eclipse

*Eclipse* adalah sebuah IDE (*Integrated Development Environment*) untuk pengembangan java/android yang *free*. Versi *eclipse* yang ada sekarang sudah banyak seperti *Eclipse Helios* (*eclipse* versi 3.6), *Eclipse Galileo* (*eclipse* versi 3.5), dan *Eclipse Ganymede* (*eclipse* versi 3.4), kita dapat melakukan

instalasi salah satu versi, versi 3.4 sudah support dengan *Android Development Tools* (ADT) untuk membuat *eclipse* dapat digunakan untuk *coding project Android*. . (Nazruddin Saffat : 2011 : 16)

## **II.8. Konsep UML (*Unified Modelling Language*)**

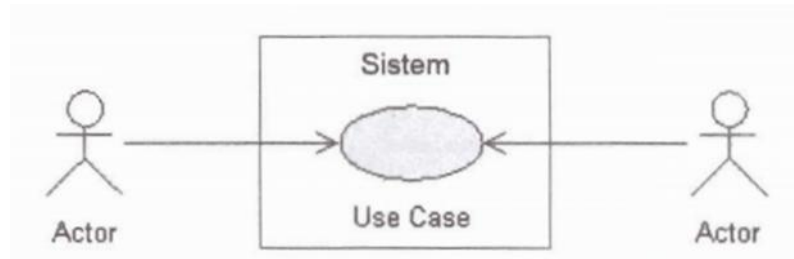
UML (*Unified Modelling Language*) adalah salah satu alat bantu yang sangat handal di dunia pengembangan sistem yang berorientasi obyek. Hal ini disebabkan karena UML menyediakan bahasa pemodelan *visual* yang memungkinkan bagi pengembang sistem untuk membuat cetak biru (*blueprint*) atas visi mereka dalam bentuk yang baku, mudah dimengerti, serta dilengkapi dengan mekanisme yang efektif untuk berbagi (*sharing*) dan mengkomunikasikan rancangan mereka dengan yang lain. (Radiant Victor Imbar dan Yuliusman Kurniawan : 2012).

### **II.8.1 Diagram – diagram UML**

#### *1. Use Case Diagram*

*Use-case* adalah konstruksi untuk mendeskripsikan bagaimana sistem akan terlihat di mata pengguna potensial. *Use-case* terdiri dari sekumpulan skenario yang dilakukan oleh seorang aktor (orang, perangkat keras, urutan waktu atau sistem yang lain). Sedangkan *use-case* diagram memfasilitasi komunikasi di antara analis dan pengguna serta diantara analis dan klien. Diagram *use case* menunjukkan 3 aspek dari sistem yaitu : *actor*, *use-case*, dan *system boundary*. *Actor* adalah pengguna sistem, biasanya mewakili peran orang, sistem yang lain atau alat yang berkomunikasi dengan *use-case*. *Use Case* adalah tugas yg

dilakukan oleh *actor*. Sekumpulan *use-case* biasanya dikelompokkan dalam suatu group yang disebut *System Boundary*. (Radiant Victor Imbar dan Yuliusman Kurniawan : 2012 )



**Gambar II.2. Usecase Model**

Sumber : (Radiant Victor Imbar dan Yuliusman Kurniawan : 2012).

## 2. Activity Diagram

*Activity diagrams* menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity diagram* juga dapat menggambarkan proses *paralel* yang mungkin terjadi pada beberapa eksekusi (Radiant Victor Imbar dan Yuliusman Kurniawan : 2012).

Simbol	Keterangan
●	Titik Awal
⦿	Titik Akhir
▭	Activity
◇	Pilihan untuk pengambilan keputusan

**Gambar II.3. Simbol-simbol pada Activity Diagram**

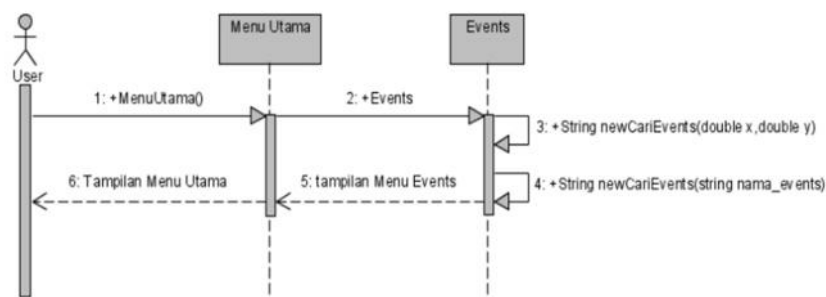
Sumber : (Radiant Victor Imbar dan Yuliusman Kurniawan : 2012).

### 3. *Class Diagram*

*Class diagram* menggambarkan struktur statis dari kelas dalam sistem anda dan menggambarkan atribut, operasi dan hubungan antara kelas. *Class diagram* membantu dalam memvisualisasikan struktur kelas-kelas dari suatu sistem dan merupakan tipe diagram yang paling banyak dipakai. Selama tahap desain, *class diagram* berperan dalam menangkap struktur dari semua kelas yang membentuk arsitektur sistem yang dibuat. (Haviluddin : 2011).

### 4. *Sequence diagram*

*Sequence diagram* menjelaskan interaksi objek yang disusun berdasarkan urutan waktu. Secara mudahnya *sequence diagram* adalah gambaran tahap demi tahap, termasuk kronologi (urutan) perubahan secara logis yang seharusnya dilakukan untuk menghasilkan sesuatu sesuai dengan *use case diagram*. (Haviluddin : 2011)



**Gambar II.4. Contoh *Sequence Diagram***

Sumber : (S. Nofan Maulana Rachman : 2012 : 9)