BAB 2

TINJAUAN PUSTAKA

2.1. Profil Keuskupan Bandung

Keuskupan dalam Gereja Katolik adalah sebuah wilayah administratif umat Katolik, dipimpin oleh seorang Uskup yang adalah pemimpin umat Katolik di wilayah Keuskupan, Keuskupan merupakan himpunan umat Katolik yang tinggal dalam suatu daerah dengan batas-batas tertentu. Keuskupan umumnya dibagi-bagi menjadi bagian yang lebih kecil, yang disebut Paroki. Paroki adalah himpunan umat Katolik yang dibentuk secara tetap dengan batas-batas kewilayahan tertentu dalam Keuskupan. Keuskupan Bandung adalah wilayah gerejani Gereja Katolik Roma yang terletak di bagian timur dan tengah tanah Parahyangan Jawa Barat dengan luas wilayah 23.315,31 Km2. Selain kota dan kabupaten Bandung, wilayah Keuskupan Bandung meliputi Kabupaten Purwakarta dan Karawang di sebelah barat; Subang, Pamanukan, Indramayu di sebelah utara; Cirebon, Kuningan, Tasikmalaya-Ciamis di sebelah timur; dan Garut di sebelah selatan.

Pada tanggal 12 November 2010, Pk. 18.00 WIB diumumkan secara resmi oleh Tahta Suci bahwa Mgr. Johannes Pujasumarta, Uskup Diosesan Keuskupan Bandung, diangkat menjadi Uskup Diosesan Keuskupan Agung Semarang dan Mgr. Ignatius Suharyo, Uskup Agung, Keuskupan Agung Jakarta diangkat menjadi Administrator Apostolik Keuskupan Bandung hingga terpilih Uskup yang baru. Dengan penuh syukur, pada tanggal 3 Juni 2014, tepat pukul 12.00 waktu Roma (17.00 WIB), Bapa Paus Fransiskus mengumumkan pengangkatan Mgr. Antonius Subianto Bunjamin, OSC, dimana sebelumnya menjabat Provinsial Ordo Salib Suci, kini sebagai Uskup Keuskupan Bandung. Di bawah kepemimpinan beliau, wilayah gereja ini dibagi menjadi beberapa wilayah teritorial yang disebut paroki. Keuskupan Bandung meliputi 23 paroki dan 1 Kuasi; 13 Paroki dan 1 Kuasi Paroki di antaranya terletak di kota Bandung dan sekitarnya. 23 Paroki dan 1 Kuasi, kini menjadi tempat bernaung Umat Katolik Keuskupan Bandung yang saat ini berjumlah kurang lebih 98.069 orang.

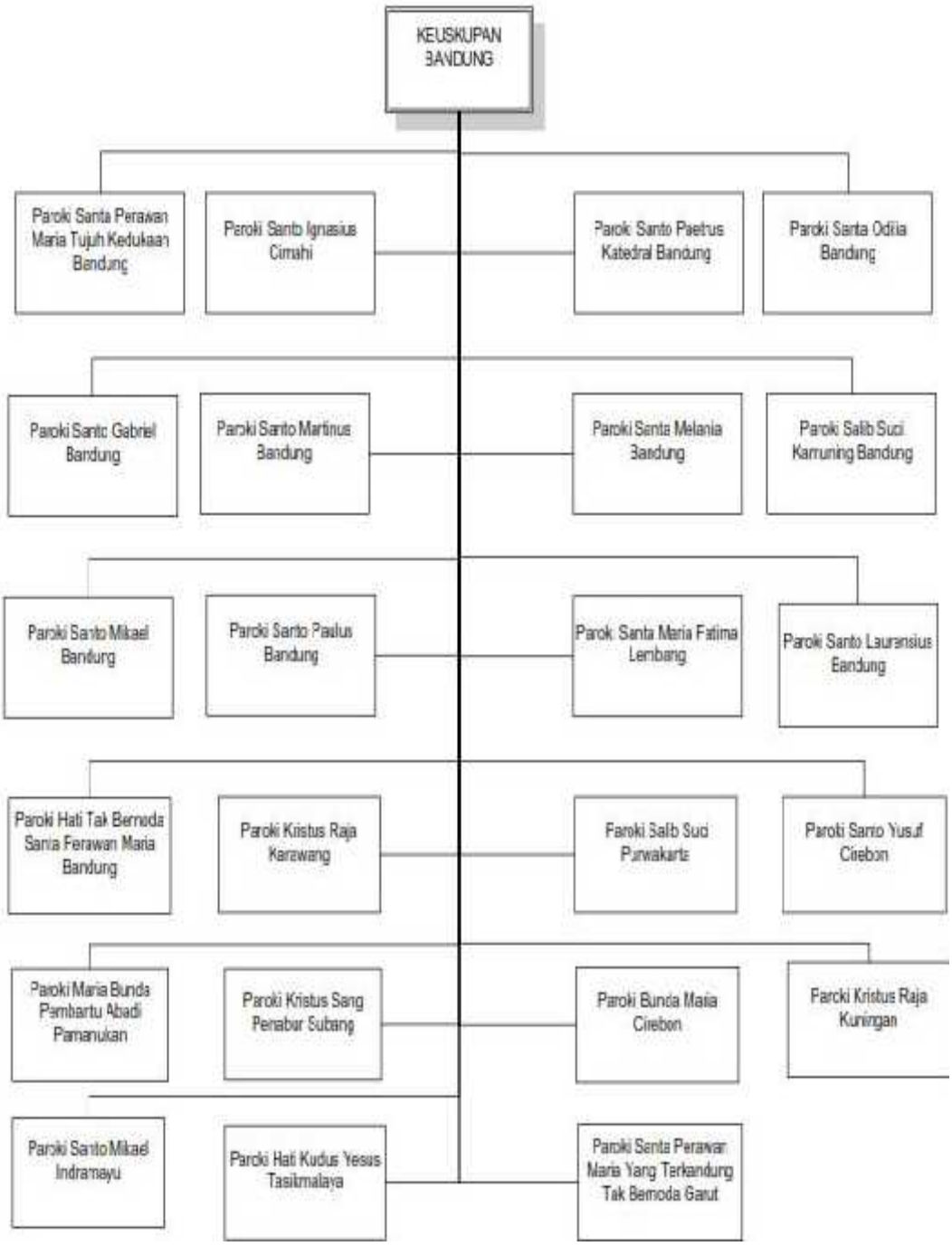
Pelayanan dan penggembalaan di Paroki dipercayakan oleh Uskup kepada para imam (pastor), baik imam diosesan maupun tarekat. Imam diosesan adalah pria yang ditahbiskan menjadi imam katolik dan dengan ketaatannya kepada uskup terikat seumur hidup untuk pelayanan di Keuskupan, dengan tetap terbuka untuk ditugaskan oleh Uskup di wilayah gerejani lain. Imam Tarekat adalah imam yang menjadi anggota suatu tarekat/ordo/lembaga religius Gereja Katolik. Atas kesepakatan dan mandat dari Uskup, Imam Tarekat tersebut ditugaskan oleh pimpinannya untuk tugas pelayanan di Keuskupan. Tarekat imam yang bekerja di Keuskupan Bandung saat ini ialah Ordo Salib Suci (OSC) dan Kongregasi Hati Kudus Yesus dan Maria (SS.CC).

Selain pelayanan parokial, keuskupan juga menjalankan pelayanan kategorial yang dilaksanakan baik oleh komisi ataupun yayasan keuskupan. Komisi adalah unit kerja keuskupan di bawah koordinasi Dewan Karya Pastoral (DKP) untuk menangani bidang pelayanan tertentu baik yang menyangkut kehidupan internal gereja maupun sosial kemasyarakatan, seperti Komisi Liturgi, Kitab Suci, Pengembangan Sosial Ekonomi Masyarakat, Pendidikan, Perburuhan, Hubungan antar Agama dan Kepercayaan. Yayasan keuskupan adalah yayasan independen yang dibentuk oleh keuskupan untuk melaksanakan tugas pelayanan sosial kemasyarakatan, seperti yayasan pendidikan dan kesehatan. Melalui yayasan-yayasan pendidikan yang telah didirikan, saat ini Keuskupan menjalankan 89 sekolah tersebar di seluruh wilayah keuskupan dari tingkat taman kanak-kanak sampai menengah atas, 2 perguruan tinggi, dan 1 Balai Latihan Kerja. Bidang pelayanan kesehatan terdiri dari 4 rumah sakit dan 6 Balai Pengobatan Paroki. Selain yayasan keuskupan, hadir pula atas izin Uskup Bandung, yayasan-yayasan yang didirikan baik oleh tarekat suster (biarawati) maupun kelompok umat katolik untuk membantu pelayanan sosial dan pendidikan di wilayah Keuskupan Bandung. Mereka menjalankan 26 sekolah dari taman kanak-kanak sampai menengah atas, 3 panti wredha, 1 pusat rehabilitasi narkoba. Saat ini terdapat 5 tarekat suster yang bekerja di Keuskupan Bandung: CB, PI, OP, OSU, dan Ursulin Somasca. Selain itu, terdapat pula pertapaan suster Karmel

(OCD) di Lembang yang tentunya memberikan warna tersendiri bagi Keuskupan Bandung [4].

2.1.1. Struktur Wilayah

Berikut ini adalah struktur wilayah Keuskupan Bandung :



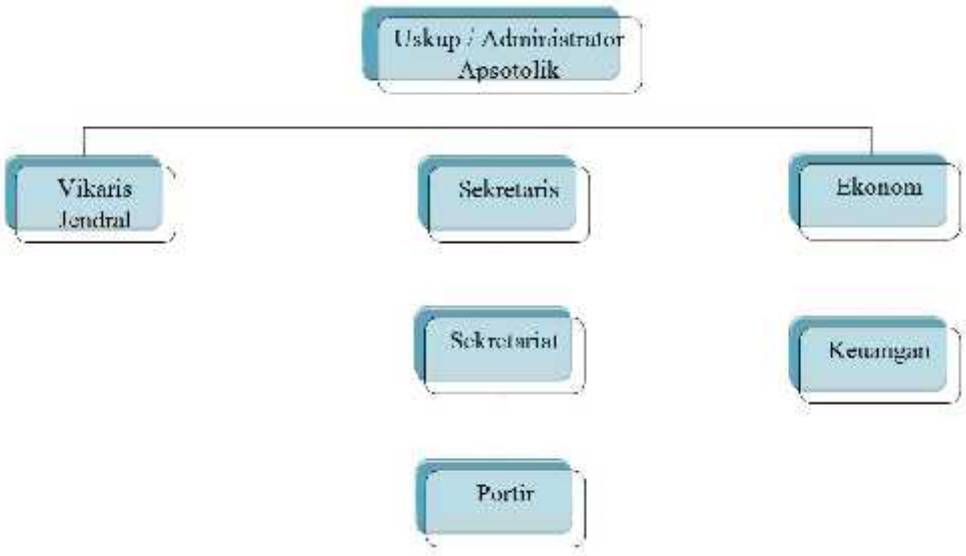
Gambar 2.1 Struktur Wilayah Keuskupan Bandung

Dari Gambar 2.1 terlihat bahwa ada 23 paroki yang berada di wilayah Keuskupan Bandung yakni 13 Paroki yang berada di Kota Bandung dan 10 Paroki yang berada di daerah lainnya.

2.1.2. Struktur Kuria Keuskupan (Struktur Organisasi)

Kuria Keuskupan adalah tim kerja yang membantu Uskup dalam memimpin keuskupan, terutama dalam mengarahkan karya pelayanan pastoral, melaksanakan administrasi Keuskupan dan dalam menjalankan kuasa yudisial.

Uksup : Mgr. Antonius Subianto Bunjamin, OSC.
Vikaris Jendral : R.D. Yustinus Hilman Pujiatmoko.
Sekretaris : R.P. Ignatius Eddy Putranto, OSC.
Ekonom : R.D. Antonius Sulastijana.



Gambar 2.2. Struktur Kuria Keuskupan Bandung

Berdasarkan Gambar 2.2 berikut ini jobdeks atau tugas dari masing-masing bagian :

Tabel 2.1 Jobdesk

No	Jabatan	Jobdesk
1	Uskup/Administrator Apostolik	Memimpin penggembalaan umat katolik di Keuskupan Bandung.
2	Vikaris Jendral	Mewakili sebagian tugas atau wewenang Uskup di Keuskupan Bandung.
3	Sekretaris	Mengatur atau mengelola arsip-arsip atau keadministrasian Uskup

		dan Keuskupan Bandung.
4	Ekonom	mengelola harta-benda keuskupan dibawah otoritas Uskup dan dari pendapatan keuskupan yang telah ditetapkan melakukan pengeluaran-pengeluaran yang diperintahkan Uskup atau orang-orang lain yang ditugaskan dengan legitim olehnya.
5	Sekretariat	Di bagian sekretarian ini terdiri dari 3 orang dimana masing-masing meliliki tugas seperti satu orang bertugas untuk membuat Surat Keputusan (SK) tugas penempatan para pastor sesuai keputusan uskup, mengelola data Gereja, mengelola data baptis, komuni, dan krisma umat; satu orang mengelola data umat paroki, data sekolah, data yayasan, data lembaga hidup bakti, dan lain-lain; satu orang mengelola data majalah komuni, mengelola data kendaraan se-Keuskupan dan pengurusannya.
6	Keuangan	membuat laporan keuangan Keuskupan dan Paroki-paroki, juga komisi-komisi yang ada di Keuskupan Bandung.
7	Portir	Menerima tamu, *telphone* dan surat-surat untuk Kuria, juga mengirim surat dari Kuria.

2.2. Landasan Teori

Landasan teori merupakan penjelasan berbagai konsep dasar dan teori-teori yang berkaitan dalam pembangunan aplikasi *mobile* Keuskupan berbasis Android. Dimulai dengan teori – teori informatika umum sampai dengan *tools* pembangun aplikasi. Hal ini digunakan untuk mengetahui gambaran tempat penelitian Tugas Akhir serta menjesm'bv laskan teori-teori yang berhubungan dengan masalah yang dibahas.

2.2.1. Informasi

Informasi adalah data yang diolah menjadi bentuk yang berguna dan menjadi berarti bagi penerimanya. Kegunaan informasi adalah untuk mengurangi ketidakpastian di dalam proses pengambilan keputusan tentang suatu keadaan. Suatu informasi dikatakan bernilai bila manfaatnya lebih efektif dibandingkan dengan biaya untuk mendapatkan informasi tersebut. Informasi merupakan sesuatu yang harus dimiliki oleh suatu perusahaan atau organisasi untuk bisa tetap eksis dan bertahan, karena kurangnya informasi yang dimiliki akan

mengakibatkan perusahaan atau organisasi akan hancur dan berakhir. Kualitas informasi sangat dipengaruhi atau ditentukan oleh beberapa hal yaitu :

a. Relevan *(Relevancy)*
b. Akurat *(Accurancy)*
c. Tepat waktu *(Time liness)*
d. Ekonomis *(Economy)*
e. Efisien *(Efficiency)*
f. Ketersediaan *(Availability)*
g. Dapat dipercaya *(Reliability)*
h. Konsisten

2.2.2. Android

Android adalah sistem operasi untuk telepon selular (*mobile)* seperti pada *smartphone* dan komputer tablet. Sistem operasi Android berbasiskan Linux yang bersifat terbuka (*opensource)* bagi pengembang untuk menciptakan aplikasi maupun mengembangkan Android dengan *tools* yang disediakan yaitu Android SDK (*Software Development Kit)* dan API (*Application Programming Interface)* dengan menggunakan bahasa pemrograman Java. Dikembangkan Oleh Google Inc, HTC, Intel, Motorola, Qualcom, T-Mobile, dan Nvidia yang tergabung dalam *Open Handset Alliance* (OHA). Dengan tujuan mendukung sebuah standar terbuka pada perangkat seluler. Kode-kode yang dirilis oleh Google untuk Android berada dibawah *license Apache*, sebuah *license* perangkat lunak dan standar terbuka bagi perangkat seluler [5]. Android dipuji sebagai “platform mobile pertama yang lengkap, terbuka dan bebas:

1. Lengkap (*Complate Platform*) : para desainer dapat melakukan pendekatan yang komprehensif ketika mereka sedang mengembangkan *platform* Android. Android merupakan sistem operasi yang aman dan banyak menyediakan *tools* dalam membangun *software* dan memungkinkan untuk peluang pengembangan aplikasi.

2. Terbuka (*Open source*) : *platform* Android disediakan melaui lisensi *open source*. Pengembangan dapat dengan bebas untuk mengembangkan aplikasi. Android sendiri menggunakan Linux karnel 2.6.
3. Bebas (*free platform*) : Android adalah *platform* atau aplikasi yang bebas untuk develop. Tidak ada lisensi atau royalti untuk dikembangkan pada *platform* Android. Tidak ada biaya keanggotaan yang diperlukan. Tidak diperlukan biaya pengujian. Tidak ada kontrak yang diperlukan. Aplikasi Android dapat di distribusikan dan diperdagangkan dalam bentuk apa pun.

Di bawah ini adalah beberapa keunggulan Android dibandingkan dengan OS Mobile lain:

1. Multitasking, HP Android bisa menjalankan berbagai aplikasi secara bersamaan. Kemudahan dalam Notifikasi – Setiap ada SMS, Email, atau bahkan artikel terbaru dari RSS Reader, akan selalu ada notifikasi di Home Screen Ponsel Android, tak ketinggalan Lampu LED Indikator yang berkedipkedip.
2. Arsitektur komponen dasar android terinspirasi dari teknologi internet Mashup. Bagian dalam sebuah aplikasi dapat digunakan oleh aplikasi lainnya, bahkan dapat diganti dengan komponen lain yang sesuai dengan aplikasi yang dikembangkan.
3. Banyak dukungan service, kemudahan dalam menggunakan berbagai macam layanan pada aplikasi seperti penggunaan layanan pencarian lokasi, database SQL, browser dan penggunaan peta. Semua itu sudah tertanam pada android sehingga memudahkan dalam pengembangan aplikasi.
4. Siklus hidup aplikasi diatur secara otomatis, setiap program terjaga antara satu sama lain oleh berbagai lapisan keamanan, sehingga kerja sistem menjadi lebih stabil. Pengguna tak perlu kawatir dalam menggunakan aplikasi pada perangkat yang memorinya terbatas.

Sebuah studi yang dilakukan oleh Bank of America mengungkapkan, hingga september 2014, market share OS dunia masih dikuasai Android (47,06%), dan IOS (43,86%). Selebihnya dibagi merata oleh Java ME (2,78%), Symbian

(2,56%), Windows Phone (2,38%), Blackberry (1,22%), Kindle (0,08%), Samsung (0,03%), Windows Mobile (0,01%), dan LG (0,01%) [6].

2.2.3. Global Positioning System (GPS)

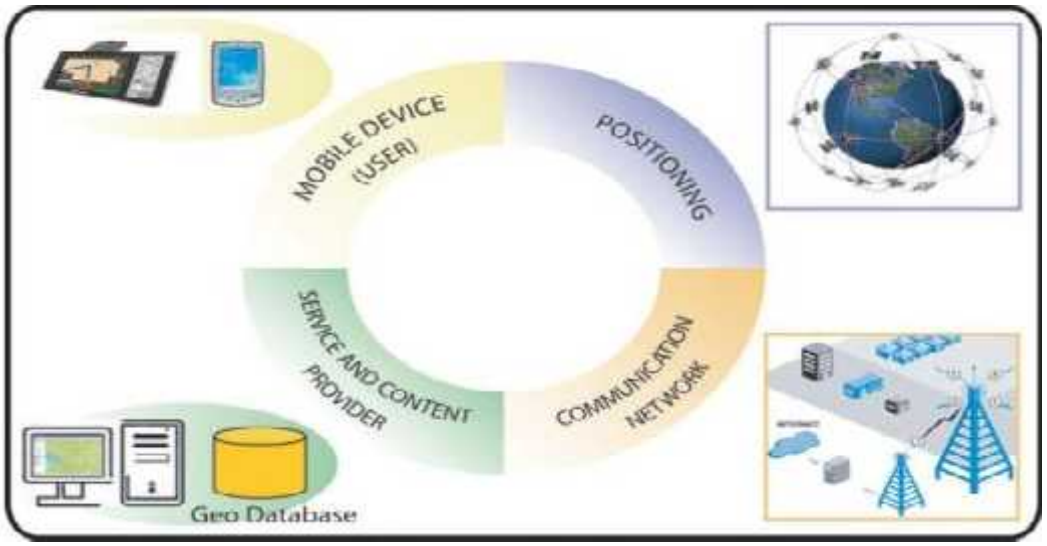
GPS (Global Positioning System) adalah sistem satelit navigasi dan penentuan posisi yang dimiliki dan dikelola oleh Amerika Serikat. Sistem ini didesain untuk memberikan posisi dan kecepatan tiga-dimensi serta informasi mengenai waktu, secara kontinyu di seluruh dunia tanpa bergantung waktu dan cuaca, bagi banyak orang secara simultan. Saat ini GPS sudah banyak digunakan orang di seluruh dunia dalam berbagai bidang aplikasi yang menuntut informasi tentang posisi, kecepatan, percepatan ataupun waktu yang teliti. GPS dapat memberikan informasi posisi dengan ketelitian bervariasi dari beberapa millimeter (orde nol) sampai dengan puluhan meter.

Beberapa kemampuan GPS antara lain dapat memberikan informasi tentang posisi, kecepatan, dan waktu secara cepat, akurat, murah, dimana saja di bumi ini tanpa tergantung cuaca. Hal yang perlu dicatat bahwa GPS adalah satu-satunya sistem navigasi ataupun sistem penentuan posisi dalam beberapa abad ini yang memiliki kemampuan handal seperti itu. Ketelitian dari GPS dapat mencapai beberapa mm untuk ketelitian posisinya, beberapa cm/s untuk ketelitian kecepatannya dan beberapa nanodetik untuk ketelitian waktunya. Ketelitian posisi yang diperoleh akan tergantung pada beberapa faktor yaitu metode penentuan posisi, geometri satelit, tingkat ketelitian data, dan metode pengolahan datanya.

2.2.4. Location Based Service (LBS)

Location Based Service atau disingkat LBS adalah kemampuan untuk mencari lokasi geografis dari *mobile device* dan menyediakan layanan berdasarkan lokasi yang diperolehnya.Konsep LBS ini menghasilkan layanan informasi yang dapat dicocokkan dengan lokasi keberadaan calon penerima informasi. Hal ini menyebabkan peningkatan nilai informasi dikarenakan penerima dapat mengasosiasikan pengetahuan atau informasi yang didapat keberadaannya [7].

2.2.4.1. Komponen LBS



Gambar 2.3. Komponen LBS

Berikut ini komponen pendukung utama dalam teknologi Layanan Berbasis Lokasi seperti yang ditunjukkan pada Gambar 2.3, yaitu antara lain:

1. Piranti *Mobile*.

Piranti *Mobile* adalah salah satu komponen penting dalam LBS. Piranti ini berfungsi sebagai alat bantu (*tool*) bagi pengguna untuk meminta informasi. Hasil dari informasi yang diminta dapat berupa teks, suara, gambar dan lain sebagainya.Piranti *mobile* yang dapat digunakan bisa berupa PDA, *smartphone*, *laptop*.Selain itu, piranti *mobile* dapat juga berfungsi sebagai alat navigasi di kendaraan seperti halnya alat navigasi berbasis GPS.

2. Jaringan Komunikasi.

Komponen kedua adalah jaringan komunikasi.Komponen ini berfungsi sebagai jalur penghubung yang dapat mengirimkan data-data yang dikirim oleh pengguna dari piranti *mobile*-nya untuk kemudian dikirimkan ke penyedia layanan dan kemudian hasil permintaan tersebut dikirimkan kembali oleh penyedia layanan kepada pengguna.

3. Komponen *Positioning* (Penunjuk Posisi/Lokasi).

Setiap layanan yang diberikan oleh penyedia layanan biasanya akan berdasarkan pada posisi pengguna yang meminta layanan tersebut. Oleh karena itu diperlukan komponen yang berfungsi sebagai pengolah/pemroses yang akan

menentukan posisi pengguna layanan saat itu. Posisi pengguna tersebut bisa didapatkan melalui jaringan komunikasi *mobile* atau juga menggunakan *Global Positioning System* (GPS).

4. Penyedia layanan dan aplikasi.

Penyedia layanan merupakan komponen LBS yang memberikan berbagai macam layanan yang bisa digunakan oleh pengguna. Sebagai contoh ketika pengguna meminta layanan agar bisa tahu posisinya saat itu, maka aplikasi dan penyedia layanan langsung memproses permintaan tersebut, mulai dari menghitung dan menentukan posisi pengguna, menemukan rute jalan, mencari data di Yellow Pages sesuai dengan permintaan, dan masih banyak lagi yang lainnya.

5. Penyedia data dan konten.

Penyedia layanan tidak selalu menyimpan seluruh data dan informasi yang diolahnya.Karena bisa jadi berbagai macam data dan informasi yang diolah tersebut berasal dari pengembang/pihak ketiga yang memang memiliki otoritas untuk menyimpannya.Sebagai contoh basis data geografis dan lokasi bisa saja berasal dari badan-badan milik pemerintah atau juga data-data perusahaan/bisnis/industri bisa saja berasal dari Yellow Pages, maupun perusahaan penyedia data lainnya.

2.2.4.2. Unsur Utama LBS

Dua unsur utama dari *Location Based Service* adalah:

1. *Location Manager* (*API Maps*) : Menyediakan perangkat bagi sumber atau source untuk LBS, *Application Programing Interface* (API) *maps* menyediakan fasilitas untuk menampilkan atau memanipulasi peta. Paket ini berada pada “com.google.android.maps”.
2. *Location Providers* (*API Location*): Menyediakan teknologi pencarian lokasi yang digunakan oleh perangkat. API Location berhubungan dengan data GPS (*Global Positioning System*) dan data lokasi *real-time*. *API Location* berada pada paket Android yaitu dalam paket “android.location”. Lokasi, perpindahan,

serta kedekatan dengan lokasi tertentu dapat ditentukan melalui *Location Manager*.

Dalam menentukan posisi dari sebuah handphone yang sedang aktif, secara umum terdapat tiga tingkat metode yang digunakan saat ini, yaitu :

A. **Metode *Basic Positioning yang Berbasis* Pada *Cell Identification* (Cell ID)**

Penentuan posisi didasarkan pada daerah geografis yang tercakup oleh sebuah *cell* berhubungan dengan daerah cakupan dari sinyal radio. Ketika sebuah *handphone* terhubung secara aktif dengan sebuah base station, berarti handphone tersebut diasumsikan berada dalam *cell* dari *base station* tersebut. Untuk mengukur jarak dan arah *handset* dari *base station* tidak dapat diketahui dengan pasti. Oleh karena itu, untuk lebih meningkatkan lagi akurasi hasil pencarian, metode *Cell ID* ini seringkali dikombinasikan dengan metode lain misalnya :

a. *Timing Advanced* (TA), dengan menggunakan TA ini, metode *Cell ID* akan ditambahkan sebuah fungsionalitas untuk menghitung *Round Trip Time* (RTT), yaitu waktu transmisi sebuah frame (dari *base station* ke *handphone*) dan waktu penerimaan sebuah *frame* (dari handphone ke base station). Dengan tambahan metode ini, jarak antara *handphone* dan *base station* dapat ditentukan dengan keakuratan 50 m.

b. *Network Measurement Report* (NMR), dengan berdasar pada besar kecilnya sinyal (*Received Signal Strength*) yang diterima *handphone* yang ada di suatu "*sector cell*", maka posisi itu dapat ditentukan lebih akurat.

B. **Metode *Enhanced Positioning***

Pada umumnya metode ini menggunakan pendekatan *Observe Time Difference* (OTD). Dalam jaringan GSM yang sering digunakan adalah *Enhanced-OTD* (E-OTD). E-OTD adalah metode pencarian posisi yang berdasarkan pada waktu. Untuk menentukan posisi relatif, sebuah *handphone* harus aktif terhadap *tiga base station* dan perlu ditentukan terlebih dahulu jarak *handphone* terhadap masing-masing *base station* berdasarkan waktu yang ditempuh oleh sebuah sinyal dari *handphone* ke masing-masing base station. Selanjutnya, dengan menggunakan rumus matematika untuk triangulasi, maka dapat ditentukan posisi

dari *handphone* yang sedang aktif tersebut. Dengan menggunakan metode ini akurasinya akan meningkat hingga memiliki ketelitian sampai kurang dari 50m.

C. **Metode *Advanced Positioning***

Pada umumnya menggunakan teknologi *Assisted-Global Positioning* System (A-GPS). A-GPS juga merupakan metode yang berbasis pada waktu. Pada metode ini, akan dilakukan pengukuran waktu tiba dari sebuah sinyal yang dikirim dari tiga buah satelit GPS. Hal ini berarti handset harus memiliki fasilitas untuk mengakses GPS. A-GPS juga menghasilkan akurasi secara vertikal dan estimasi jarak yang baik. Akurasinya pun sampai kurang dari 10m

2.2.5. Rumus Haversine

Rumus Haversine adalah persamaan pada navigasi, yang memberikan jarak lingkaran besar antara dua titik bumi berdasarkan garis bujur (longitude) dan garis lintang (latitude). Rumus Haversine diperkenalkan oleh matematikawan asal Inggris yang bernama Prof. James Inman pada tahun 1835. Perhitungan dari rumus ini juga cukup akurat dimana rumus ini mengabaikan ketinggian bukit dan kedalaman lembah di permukaan bumi. Secara matematis dapat dituliskan didalam persamaan yang ditunjukkan pada gambar 2.4 sebagai berikut :

$$Jarak = 2r.\arcsin\left\{\sqrt{\sin^2\left(\frac{Lat_1 - Lat_2}{2}\right) + \cos(Lat_1).\cos(Lat_2).\sin^2\left(\frac{Long_1 - Long_2}{2}\right)}\right\}$$

Gambar 2.4 Rumus Haversine

Dimana : *r* adalah konstanta radius bumi yaitu 6.371 Km.

Lat1 = Nilai *Latitude* dari lokasi awal (satuan radian).

Long1 = Nilai *longitude* dari lokasi awal (satuan radian).

Lat2 = Nilai *Latitude* dari lokasi tujuan (satuan radian).

Long2 = Nilai *longitude* dari lokasi tujuan (satuan radian).

2.2.6. Database

Database dapat diartikan sebagai kumpulan data yang terdiri atas satu atau lebih tabel yang terintegrasi satu sama lain, di mana setiap pemakai (user) diberi

wewenang (otorisasi) untuk dapat mengakses (mengubah, menghapus, menganalisis, menambah, memperbaiki) data dalam tabel-tabel tersebut [8]. Adapun fungsi database diantaranya adalah sebagai berikut :

a. Salah satu komponen penting dalam sistem informasi, karena merupakan dasar dalam menentukan informasi
b. Menentukan kualitas informasi. Informasi dapat dikatakan bernilai apabila manfaatnya lebih efektif dibandingkan dengan biaya mendapatkannya.
c. Mengurangi duplikasi data (*data redundancy*).
d. Hubungan data dapat ditingkatkan.
e. Mengurangi pemborosan tempat simpanan luar.

2.2.7. MySQL

MySQL adalah sebuah perangkat lunak sistem manajemen basis data *SQL* (bahasa Inggris : *Database Management System*) atau *DBMS* yang *multithread*, *multi-user*, dengan sekitar 6 juta instalasi di seluruh dunia. *MySQL AB* membuat *MySQL* tersedia sebagai perangkat lunak gratis di bawah lisensi *GNU General Public License (GPL)*, tetapi mereka juga menjual dibawah lisensi komersial untuk kasus-kasus dimana penggunaannya tidak cocok dengan penggunaan *GPL*[8].

Pada MySQL sebuah database mengandung satu atau sejumlah tabel. Tabel terdiri dari sejumlah baris dan setiap baris mengandung satu atau beberapa kolom. Untuk mengelolah data MySQL ada beberapa cara yaitu melalui *prompt* DOS *(tool command line)* dan dapat juga menggunakan program *utility* seperti:

1. PHP MyAdmin
2. MySQLGUI
3. MySQL Manager Java Based
4. MySQL Administrator for windows

2.2.8. WAMP

WAMP adalah perangkat lunak gratis, yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program. Fungsinya adalah sebagai server yang berdiri sendiri (localhost), yang terdiri atas program Apache HTTP

Server, MySQL database, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl. Program ini tersedia dalam GNU General Public License dan gratis, merupakan web server yang mudah digunakan yang dapat melayani tampilan halaman web yang dinamis.

2.2.9. UML (Unified Modeling Language)

Unified Modeling Language (UML) adalah bahasa spesifikasi standar untuk mendokumentasikan, menspesifikasikan, dan membangun sistem perangkat lunak. UML adalah himpunan struktur dan teknik untuk pemodelan desain program berorientasi objek (OOP) serta aplikasinya. UML adalah metodologi untuk mengembangkan sistem OOP dan sekelompok perangkat tool untuk mendukung pengembangan sistem tersebut. UML mulai diperkenalkan oleh *Object Management Group*, sebuah organisasi yang telah mengembangkan model, teknologi, dan standar OOP sejak tahun 1980-an. Sekarang UML sudah mulai banyak digunakan oleh para praktisi OOP. UML merupakan dasar bagi perangkat (tool) desain berorientasi objek dari IBM. UML adalah suatu bahasa yang digunakan untuk menentukan, memvisualisasikan, membangun, dan mendokumentasikan suatu sistem informasi. UML dikembangkan sebagai suatu alat untuk analisis dan desain berorientasi objek oleh Grady Booch, Jim Rumbaugh, dan Ivar Jacobson. Namun demikian UML dapat digunakan untuk memahami dan mendokumentasikan setiap sistem informasi. Penggunaan UML dalam industri terus meningkat. Ini merupakan standar terbuka yang menjadikannya sebagai bahasa pemodelan yang umum dalam industri peranti lunak dan pengembangan sistem.

UML terdiri dari 13 jenis diagram yaitu Activity, Class, Communication, Component, Composite Structure, Deployment, Interaction Overview, Object, Package, Sequence, State Machine (Statechart), Timing dan Use Case [9]. Berikut ini adalah beberapa diagram yang sering digunakan dalam pembangunan aplikasi berorientasi objek :

1. *Use Case* Diagram

Use case diagram digunakan untuk memodelkan bisnis proses berdasarkan perspektif pengguna sistem. *Use case* diagram terdiri atas diagram untuk *use case* dan *actor*. *Actor* merepresentasikan orang yang akan mengoperasikan atau orang yang berinteraksi dengan sistem aplikasi. *Use case* merepresentasikan operasi-operasi yang dilakukan oleh aktor. *Use case* digambarkan berbentuk elips dengan nama operasi dituliskan di dalamnnya. *Actor* yang melakukan operasi dihubungkan dengan garis lurus ke *use case*.

2. *Sequence* Diagram

   Sequence diagram menjelaskan secara detil urutan proses yang dilakukan dalam sistem untuk mencapai tujuan dari use case: interaksi yang terjadi antar class, operasi apa saja yang terlibat, urutan antar operasi, dan informasi yang diperlukan oleh masing-masing operasi[10].

3. *Class* Diagram

   Class diagram merupakan diagram yang selalu ada di permodelan sistem berorientasi objek. Class diagram menunjukkan hubungan antar class dalam sistem yang sedang dibangun dan bagaimana mereka saling berkolaborasi untuk mencapai suatu tujuan.

4. Aktivity Diagram

   Activity Diagram adalah teknik menggambarkan logika prosedural, proses bisnis dan jalur kerja. Dalam beberapa hal, diagram ini memainkan peran mirip sebuah diagram alir, tetapi perbedaan prinsip antara diagram ini dan notasi diagram alir adalah diagram ini mendukung *behavior* paralel.

   Activity Diagram lebih memfokuskan diri pada eksekusi dan alur sistem dari pada bagaimana sistem itu dirakit. Diagram ini menunjukkan aktifitas sistem dalam bentuk kumpulan aksi-aksi.

5. *Collaboration* Diagram

   Collaboration diagram dipakai untuk memodelkan interaksi antar objek di dalam sistem. Berbeda dengan *sequence* diagram yang lebih menonjolkan kronologis dari operasi-operasi yang dilakukan, collaboration diagram

lebih fokus pada pemahaman atas keseluruhan operasi yang dilakukan oleh objek.

6. State Diagram

   Menggambarkan semua *state* (kondisi) yang dimiliki oleh suatu object dari suatu *class* dan keadaan yang menyebabkan *state* berubah. Kejadian dapat berupa *object* lain yang mengirim pesan. *Stateclass* tidak digambarkan untuk semua *class*, hanya yang mempunyai sejumlah *state* yang terdefinisi dengan baik dan kondisi *class* berubah oleh *state* yang berbeda.

2.2.10. Astah* Community 6.1

Astah* dahulu dikenal dengan nama JUDE (*Java and UML Developer's Environment*) adalah alat permodelan UML yang diciptakan oleh perusahaan Jepang *Change Vision*. Astah * bersifat *multi-platform* atau dapat dipakai pada berbagai sistem operasi/*platform*. Astah* dapat menggambarkan permodelan dari bahasa pemrograman Java, C++ , dan C#.

2.2.11. HTML (Hyper Text Markup Language)

HTML (*Hyper Text Markup Language*) dikenal sebagai bahasa *standar* untuk membuat dokumen *web.* Sesungguhnya *HTML* justru tidak dibuat untuk mempublikasikan informasi di *web*, namun oleh karena kesederhanaan serta kemudahan penggunanya, HTML kemudian dipilih orang untuk mendistribusikan informasi di *web.*Perintah-perintah HTML diletakkan dalam *file* berekstensi *.*html* dan ditandai dengan mempergunakan *tag* (tanda) berupa karakter " < " dan " > ". Struktur sebuah dokumen HTML pada dasarnya dibagi menjadi dua bagian besar, yaitu *header* dan *body*. Masing-masing ditandai oleh pasangan *container tag* <*head*> dan <*body*>. Bagian *head* berisikan judul dokumen dan informasi-informasi dasar lainnya sedangkan bagian *body* adalah data dokumennya. Pengaturan format teks dan pembentukan link dilakukan terhadap objeknya langsung dengan ditandai oleh *tag-tag* HTML [10].

2.2.12. PHP

PHP (*Hypertext Preprocessor*), merupakan bahasa pemograman *web* yang bersifat *serverside*. Hal tersebut mengandung arti bahwa PHP merupakan bahasa berbentuk *script* yang disimpan dan dijalankan di komputer *server* (*web server*) sedang hasilnya dikirimkan ke komputer *client* (*web browser*) dalam bentuk *script* HTML. Keuntungan penggunaan PHP, kode yang menyusun program tidak perlu dibagikan ke pemakai, yang berarti kerahasiaan kode dapat dilindungi. Penggunaan PHP memungkinkan *web* dapat dibuat dinamis sehingga perawatan situs *web* tersebut menjadi lebih mudah dan efisien. Hal menarik yang didukung PHP adalah kenyataan bahwa PHP bisa digunakan untuk mengakses berbagai *database* seperti Access, Oracle, MY\ySQL, dan lain-lain. PHP merupakan perangkat lunak *Open Source* yang disebarkan dan dilisensikan secara gratis serta dapat diunduh secara bebas dari situs resminya[10].

2.2.13. Adobe Dreamwaver

Software Adobe Dreamwaver adalah tool yang ideal untuk web designer, coders, dan application developers untuk semua tingkatan. Fungsi pekodean membuatnya mudah untuk menavigasi melalui halaman situs yang kompleks pada saat waktu mendesain. Fitur Adobe Dreamwaver terbaru mengembangkan tool layout yang membawa Expedited workflows, dan konsepsi komputer ke persetujuan klien. Selanjutnya, inovasi di seluruh Dreamwaver juga dapat membantu tim dan individu developer sama mencapai tingkat berikutnya dalam fungsionalitas dan kinerja [11].

2.2.14. Java

Java adalah bahasa pemrograman berorientasi objek yang dikembangkan olehSun Microsystems sejak tahun 1991.Bahasa ini dikembangkan dengan model yang mirip dengan bahasa C++ dan Smalltalk, namun dirancang agar lebih mudah dipakai dan *platform independent*, yaitu dapat dijalankan di berbagai jenis sistem operasi dan arsitektur komputer [12].

2.2.14.1.Kelebihan

Berikut ini kelebihan dari Java :

1. *Multiplatform*. Kelebihan utama dari Java ialah dapat dijalankan di beberapa *platform* / sistem operasi komputer, sesuai dengan prinsip tulis sekali, jalankan di mana saja. Dengan kelebihan ini pemrogram cukup menulis sebuah program Java dan dikompilasi (diubah, dari bahasa yang dimengerti manusia menjadi bahasa mesin / *bytecode*) sekali lalu hasilnya dapat dijalankan di atas beberapa platform tanpa perubahan. Kelebihan ini memungkinkan sebuah program berbasis java dikerjakan diatas *operatingsystem* Linux tetapi dijalankan dengan baik di atas Microsoft Windows. *Platform* yang didukung sampai saat ini adalah Microsoft Windows, Linux, Mac OS dan Sun Solaris. Penyebanya adalah setiap sistem operasi menggunakan programnya sendiri-sendiri (yang dapat diunduh dari situs Java) untuk meninterpretasikan *bytecode* tersebut.
2. OOP (*Object Oriented Programming* - Pemrogram Berorientasi Objek).
3. Perpustakaan Kelas Yang Lengkap, Java terkenal dengan kelengkapan *library*/perpustakaan (kumpulan program program yang disertakan dalam pemrograman java) yang sangat memudahkan dalam penggunaan oleh para pemrogram untuk membangun aplikasinya. Kelengkapan perpustakaan ini ditambah dengan keberadaan komunitas Java yang besar yang terus menerus membuat perpustakaan-perpustakaan baru untuk melingkupi seluruh kebutuhan pembangunan aplikasi.
4. Bergaya C++, memiliki sintaks seperti bahasa pemrograman C++ sehingga menarik banyak pemrogram C++ untuk pindah ke Java. Saat ini pengguna Java sangat banyak, sebagian besar adalah pemrogram C++ yang pindah ke Java. Universitas-universitas di Amerika Serikat juga mulai berpindah dengan mengajarkan Java kepada murid-murid yang baru karena lebih mudah dipahami oleh murid dan dapat berguna juga bagi mereka yang bukan mengambil jurusan komputer.
5. Pengumpulan sampah otomatis, memiliki fasilitas pengaturan penggunaan memori sehingga para pemrogram tidak perlu melakukan pengaturan

memori secara langsung (seperti halnya dalam bahasa C++ yang dipakai secara luas).

2.2.14.2.Kekurangan

Berikut ini adalah kekurangan dari Java :

1. Tulis sekali, jalankan di mana saja - Masih ada beberapa hal yang tidak kompatibel antara *platform* satu dengan *platform* lain. Untuk J2SE, misalnya *SWT-AWT bridge* yang sampai sekarang tidak berfungsi pada Mac OS X.
2. Mudah didekompilasi. Dekompilasi adalah proses membalikkan dari kode jadi menjadi kode sumber. Ini dimungkinkan karena kode jadi Java merupakan *bytecode* yang menyimpan banyak atribut bahasa tingkat tinggi, seperti nama-nama kelas, metode, dan tipe data. Hal yang sama juga terjadi pada Microsoft .NET *Platform*. Dengan demikian, algoritma yang digunakan program akan lebih sulit disembunyikan dan mudah dibajak/di-*reverse-engineer*.
3. Penggunaan memori yang banyak. Penggunaan memori untuk program berbasis Java jauh lebih besar daripada bahasa tingkat tinggi generasi sebelumnya seperti C/C++ dan Pascal (lebih spesifik lagi, Delphi dan Object Pascal). Biasanya ini bukan merupakan masalah bagi pihak yang menggunakan teknologi terbaru (karena trend memori terpasang makin murah), tetapi menjadi masalah bagi mereka yang masih harus berkutat dengan mesin komputer berumur lebih dari 4 tahun.

2.2.14.3.Tahap Kompilasi Java

Tahapan kompilasi java adalah sebagai berikut :

1. Tulis / Ubah. Pemrogram menulis program dan menyimpannya di media dalam bentuk berkas '.java'.
2. Kompilasi. Pengkompilasi membentuk *bytecodes* dari program menjadi bentuk berkas '.class'.
3. Muat. Pemuat kelas memuat *bytecodes* ke memori.

4. Verifikasi. Peng-verifikasi memastikan *bytecodes* tidak mengganggu sistem keamanan Java.
5. Jalankan. Penerjemah menerjemahkan *bytecodes* ke bahasa mesin.

2.2.15. Ecipse IDE

Eclipse merupakan komunitas *open source* yang bertujuan menghasilkan *platform* pemrograman terbuka. Eclipse terdiri dari *framework* yang dapat dikembangkan lebih lanjut, peralatan bantu untuk membuat dan me*manage software* sejak awal hingga diluncurkan. *Platform* Eclipse didukung oleh ekosistem besar yang terdiri dari vendor teknologi, *start-up* inovatif, universitas, riset institusi serta individu. Banyak orang mengenal Eclipse sebagai IDE (integrated development environment) untuk bahasa Java, tapi Eclipse lebih dari sekedar IDE untuk Java. Secara umum Eclipse digunakan untuk membangun software inovatif berstandar industri, dan alat bantu beserta *framework*-nya membantu pekerjaan menjadi lebih mudah.

2.2.16. Java Development Kit (JDK)

JDK (*Java Development Kit*) adalah sebuah perangkat peralatan yang digunakan untuk membangun perangkat lunak dengan menggunakan bahasa pemrograman Java. JDK berjalan diatas sebuah *virtual machine* yang dinamakan JVM (Java *Virtual Machine*).

Dokumentasi JDK berisi spesifikasi API, deskripsi fitur, panduan pengembang, referensi halaman untuk perkakas JDK dan utilitas, demo, dan *link* ke informasi terkait. Dokumentasi ini juga tersedia dalam bundel yang dapat diunduh dan diinstal pada komputer.