BAB II

LANDASAN TEORI

2.1 Pengertian Mobile

Mobile adalah kata sifat yang berarti dapat bergerak atau dapat digerakkan dengan bebas dan mudah. Namun mobile dapat pula diartikan sebuah benda yang berteknologi tinggi dan dapat bergerak tanpa menggunakan kabel. Contohnya seperti smartphone, PDA, dan tablet. Mobile juga bisa diartikan kendaraan bermotor yang dapat bergerak. Mobile bersifat bebas seperti air dan dapat mengalir kemanapun. Mobile dapat berubah dan diubah dengan mudah.Jadi ada gambaran awal tentang mobile. Pengertian yang dibahas disini memang umum tidak sebatas smartphone atau handphone. Mobile bisa saja sebuah program yang dapat menjangkau seluruh tempat bisa diakses dengan mudah dan dapat diganti kapan saja tanpa kesulitan. (Sumber : http://id.wikipedia.org/wiki/Mobile / 8 Oktober 2012).

2.2 Pengertian Technologi

Penggunaan teknologi oleh manusia diawali dengan pengubahan sumber daya alam menjadi alat-alat sederhana. Penemuan prasejarah tentang kemampuan mengendalikan api telah menaikkan ketersediaan sumber-sumber pangan, sedangkan penciptaan roda telah membantu manusia dalam beperjalanan dan mengendalikan lingkungan mereka. Perkembangan teknologi terbaru, termasuk di antaranya mesin cetak, telepon, dan Internet, telah memperkecil hambatan fisik terhadap komunikasi

dan memungkinkan manusia untuk berinteraksi secara bebas dalam skala global. Tetapi, tidak semua teknologi digunakanuntuk tujuan damai pengembangan senjata penghancur yang semakin hebat.

Teknologi telah mempengaruhi masyarakat dan sekelilingnya dalam banyak cara. Di banyak kelompok masyarakat, teknologi telah membantu memperbaiki ekonomi (termasuk ekonomi global masa kini) dan telah memungkinkan bertambahnya kaum urban. Banyak proses teknologi menghasilkan produk sampingan yang tidak dikehendaki, yang disebut pencemar, dan menguras sumber daya alam, merugikan dan merusak Bumi dan lingkungannya. Berbagai macam penerapan teknologi telah memengaruhi nilai suatu masyarakat dan teknologi baru seringkali mencuatkan pertanyaan-pertanyaan etika baru. Sebagai contoh, meluasnya gagasan tentang efisiensi dalam konteks produktivitas manusia, suatu istilah yang pada mulanya hanya menyangku permesinan, contoh lainnya adalah tantangan norma-norma tradisional, bahwa keadaan ini membahayakan lingkungan dan mengucilkan manusia; penyokong paham-paham seperti transhumanisme dan tekno-progresivisme memandang proses teknologi yang berkelanjutan sebagai hal yang menguntungkan bagi masyarakat dan kondisi manusia. Tentu saja, paling sedikit hingga saat ini, diyakini bahwa pengembangan teknologi hanya terbatas bagi umat manusia, tetapi kajian-kajian ilmiah terbaru mengisyaratkan bahwa primata lainnya dan komunitas lumba-lumba tertentu telah mengembangkan alat-alat sederhana dan belajar untuk mewariskan pengetahuan mereka kepada keturunan mereka.(Sumber: http://id.wikipedia.org/wiki/Teknologi/4 Desember 2012).

2.3 Pengertian Peta/Map

Peta/Map adalah gambaran permukaan bumi pada bidang datar dengan skala tertentu melalui suatu sistem proyeksi. Peta bisa disajikan dalam berbagai cara yang berbeda, mulai dari peta konvensional yang tercetak hingga peta digital yang tampil di layar komputer. Istilah peta berasal dari bahasa Yunani *mappa* yang berarti taplak atau kain penutup meja.Namun secara umum pengertian peta adalah lembaran seluruh atau sebagian permukaan bumi pada bidang datar yang diperkecil dengan menggunakan skala tertentu.Sebuah peta adalah representasi dua dimensi dari suatu ruang tiga dimensi.Ilmu yang mempelajari pembuatan peta disebut kartografi. Banyak peta mempunyai skala, yang menentukan seberapa besar objek pada peta dalam keadaan yang sebenarnya.Kumpulan dari beberapa peta disebut atlas. (Sumber : http://id.wikipedia.org/wiki/Peta / 3 Oktober 2012).

2.4 Pengertian Informasi

Menurut Fatta (2007:9), informasi adalah data yang telah diolah menjadi sebuah bentuk yang berarti bagi penerimanya dan bermanfaat dalam pengambilan keputusan saat ini atau mendatang.

Menurut Rochim (2002:3), informasi menyandang arti manfaat, bila kita bisa memanfaatkanya, informasi mengandung makna usaha, untuk mendapatkannya, memahaminya, menggunakannya, menyebarkannya, menyimpannya dan memadukannya dengan informasi lain menjadi suatu bentuk informasi baru.

Menurut Kusrini dan Koniyo (2007:7), informasi adalah data yang sudah diolah menjadi sebuah bentuk yang berarti bagi pengguna, yang bermanfaat dalam pengembalian keputusan saat ini atau mendukung sumber informasi.

Menurut Wahyono (2004:3), informasi adalah data yang telah diolah menjadi bentuk yang berguna bagi penerimanya dan nyata, berupa nilai yang dapat dipahami di dalam keputusan sekarang maupun masa depan.

Berdasarkan definisi diatas dapat disimpulkan bahwa informasi adalah data yang telah diolah menjadi sesuatu yang bermanfaat bagi penggunanya itu sendiri.

2.4.1 Ciri-ciri Informasi

Ciri-ciri informasi menurut Azhar Susanto (2008 : 38) menyatakan bahwa:

1. Akurat

   Informasi harus mencerminkan keadaan yang sebenarnya.

2. Tepat waktu

   Informasi harus tersedia atau ada pada saat informasi tersebut diperlukan.

3. Relevan

   Informasi yang diberikan harus sesuai dengan yang dibutuhkan oleh suatu individu.

4. Lengkap

   Informasi harus diberikan secara lengkap.

2.4.2 Fungsi Informasi

Selain itu informasi mempunyai berbagai fungsi, menurut Azhar Susanto (2008 : 38) antara lain sebagai berikut:

1. Menambah pengetahuan.

   Adanya informasi yang akan menambah pengetahuan bagi penerimanya yang dapat digunakan sebagai baha pertimbangan yang mendukung proses pengambilan keputusan.

2. Mengurangi ketidakpastian.

   Adanya informasi yang akan mengurangi ketidakpastian karena apa yang akan terjadi dapat diketahui sebelumnya, sehingga menghindari keraguan pada saat pengambilan keputusan

3. Mengurangi resiko kegagalan.

   Adanya informasi akan mengurangi resiko kegagalan karena apa yang akan terjadi dapat diantisipasi dengan baik, sehingga kemungkinan terjadinya kegagalan akan dapat dikurangi dengan pengambilan keputusan yang tepat.

4. Mengurangi keanekaragaman/variasi yang tidak diperlukan.

   Adanya informasi akan mengurangi keanekaragaman yang tidak diperlukan, karena keputusan yang di ambil lebih terarah.

5. Memberi standar, aturan-aturan, ukuran dan keputusan keputusan yang menentukan pencapaian sasaran dan tujuan.

   Adanya informasi yang akan memberikan standar, aturan, ukuran, dan keputusan yang lebih terarah untuk mencapai sasara dan tujuan yang telah ditetapkan secara baik berdasarkan informasi yang diperoleh.

2.5 Geographic Information System

Sistem Informasi Geografi (SIG) atau Geographic Information System (GIS) adalah suatu sistem informasi yang dirancang untuk bekerja dengan data yang bereferensi spasial atau berkoordinat geografi atau dengan kata lain suatu SIG adalah suatu sistem basis data dengan kemampuan khusus untuk menangani data yang bereferensi keruangan (spasial) bersamaan dengan seperangkat operasi kerja (Barus dan Wiradisastra, 2000). Sedangkan menurut Anon (2001) Sistem Informasi geografi adalah suatu sistem Informasi yang dapat memadukan antara data grafis (spasial) dengan data teks (atribut) objek yang dihubungkan secara geogrfis di bumi (*georeference*). Disamping itu, SIG juga dapat menggabungkan data, mengatur data dan melakukan analisis data yang akhirnya akan menghasilkan keluaran yang dapat dijadikan acuan dalam pengambilan keputusan pada masalah yang berhubungan dengan geografi.

Sistem Informasi Geografis dibagi menjadi dua kelompok yaitu sistem manual (analog), dan sistem otomatis (yang berbasis digital komputer). Perbedaan yang paling mendasar terletak pada cara pengelolaannya. Sistem Informasi

manual biasanya menggabungkan beberapa data seperti peta, lembar transparansi untuk tumpang susun *(overlay)*, foto udara, laporan statistik dan laporan survey lapangan.Kesemua data tersebut dikompilasi dan dianalisis secara manual dengan alat tanpa komputer. Sedangkan Sistem Informasi Geografis otomatis telah menggunakan komputer sebagai sistem pengolah data melalui proses digitasi. Sumber data digital dapat berupa citra satelit atau foto udara digital serta foto udara yang terdigitasi. Data lain dapat berupa peta dasar terdigitasi (Nurshanti, 1995).

Pengertian GIS/SIG saat ini lebih sering diterapkan bagi teknologi informasi spasial atau geografi yang berorientasi pada penggunaan teknologi komputer. Dalam hubungannya dengan teknologi komputer, Arronoff (1989) dalam Anon (2003) mendifinisikan SIG sebagai sistem berbasis komputer yang memiliki kemampuan dalam menangani data bereferensi geografi yaitu pemasukan data, manajemen data (penyimpanan dan pemanggilan kembali), memanipulasi dan analisis data, serta keluaran sebagai hasil akhir (*output*). Sedangkan Burrough, 1986 mendefinisikan Sistem Informasi Geografis (SIG) sebagai sistem berbasis komputer yang digunakan untuk memasukkan, menyimpan, mengelola, menganalisis dan mengaktifkan kembali data yang mempunyai referensi keruangan untuk berbagai tujuan yang berkaitan dengan pemetaan dan perencanaan. Komponen utama Sistem Informasi Geografis dapat dibagi kedalam 4 komponen utama yaitu: perangkat keras (*digitizer*, *scanner*, *Central Procesing Unit* (*CPU*), *hard-disk*, dan lain-lain), perangkat lunak (ArcView, Idrisi, ARC/INFO, ILWIS, MapInfo, dan lain-lain), organisasi

(manajemen) dan pemakai (*user*). Kombinasi yang benar antara keempatkomponen utama ini akan menentukan kesuksesan suatu proyek pengembangan Sistem Informasi Geografis.

Aplikasi SIG dapat digunakan untuk berbagai kepentingan selama data yang diolah memiliki refrensi geografi, maksudnya data tersebut terdiri dari fenomena atau objek yang dapat disajikan dalam bentuk fisik serta memiliki lokasi keruangan (Indrawati, 2002).

Tujuan pokok dari pemanfaatan Sistem Informasi Geografis adalah untuk mempermudah mendapatkan informasi yang telah diolah dan tersimpan sebagai atribut suatu lokasi atau obyek.Ciri utama data yang bisa dimanfaatkan dalam Sistem Informasi Geografis adalah data yang telah terikat dengan lokasi dan merupakan data dasar yang belum dispesifikasi (Dulbahri, 1993).

Data-data yang diolah dalam SIG pada dasarnya terdiri dari data spasial dan data atribut dalam bentuk digital, dengan demikian analisis yang dapat digunakan adalah analisis spasial dan analisis atribut. Data spasial merupakan data yang berkaitan dengan lokasi keruangan yang umumnya berbentuk peta. Sedangkan data atribut merupakan data tabel yang berfungsi menjelaskan keberadaan berbagai objek sebagai data spasial.

2.6 Pengertian clothing

Clothing Company adalah istilah yang digunakan untuk perusahaan yang memproduksi pakaian jadi dibawah brand mereka sendiri. Awalnya barang yg diproduksi berupa kaos, tapi seiring perkembangan kebutuhan dan permintaan customer, produknya pun meluas hingga ke celana, dompet, tas dan aksesoris lainnya. Istilah clothing lokal sendiri dimulai dengan berdirinya 347 boardrider.co pada tahun 1996 (sekarang 347/eat). Clothing ini mengambil nama dari lokasi pertamanya yang terletak di Jalan Dago no.347. Disusul kemudian oleh Ouval Research pada tahun 1997 (meskipun pada awalnya telah dirintis oleh salah satu foundernya, Arif Maskom pada 1993 dengan merilis M Clothing). Serta ada pula Airplane, Harder, No Labels (NL's), Monik, dan Two Clothes yang berdiri 1998. Serta masih banyak lagi clothing serta distro lainnya yang jika disebutkan di sini akan sangat banyak memakan ruang dan waktu.(Sumber: http://www.bisnis.pusatkaospolosmurah.com/pengertian-distro-dan-clothing-company/10 Oktober 2012).

2.7 Pengertian Distro

Distro berasal dari kata Distribution Store yang bisa diartikan sebagai tempat/outlet/toko yang secara khusus mendistribusikan produk dari suatu komunitas. Biasanya berasal dari komunitas music band-band independent atai intilahnya band indie dan komunitas skateboard. Produk produknya biasanya terdiri

dari album-album band indie sampai ke pernak perniknya spt kaos dan aksesoris dan produk apparel untuk skateboard. Di Bandung pun distro pertama kali dibuka untuk menjual produk dari band-band luar khususnya band underground serta perlengkapan dan apparel untuk skateboard. Dimulai dengan adanya Reverse di Jalan Sukasenang, yang menjual berbagai kaos musik luar dan apparel skateboard. Reverse juga dikenal sebagai markas musisi pelopor indie label waktu itu. Kemudian ada juga Hobbies yang mengkhususkan diri pada produk-produk skateboard, serta Mossy yang khusus hanya menjual kaos band-band luar. Sedangkan distro yang pertama menjual produk dari clothing lokal sendiri adalah Anonim. Kemudian Flashy serta Cynical md di Jakarta. Saat ini istilah distro kemudian dikenal sebagai toko/retail yang khusus hanya menjual produk dari berbagai clothing lokal serta merchandise band indie lokal.(Sumber:http://www.bisnis.pusatkaospolosmurah.com/pengertian-distro-dan-clothing-company/10 Oktober 2012).

2.8 Metode Pendekatan Sistem

Metode pendekatan sistem yang digunakan oleh penulis adalah pendekatan dengan *Object Oriented* yang menggunakan OOAD (*Object Oriented Analisis and Design*).

2.8.1. Pengertian OOAD

OOAD adalah metode analisis yang memeriksa requirements dari sudut pandang kelas kelas dan objek yang ditemui dalam ruang lingkup permasalahan yang mengarahkan arsitektur software yang didasarkan pada manipulasi objek-objek system atau subsistem. OOAD merupakan cara baru dalam memikirkan suatu masalah dengan menggunakan model yang dibuat menurut konsep sekitar dunia nyata. Dasar pembuatan adalah objek yang merupakan kombinasi antara struktur data dan perilaku dalam satu entitas.

2.8.2. Konsep Dasar OOAD

OOAD mencakup analisis dan desain sebuah sistem dengan pendekatan objek, yaitu analisis berorientasi objek (OOA) dan desain berorientasi objek (OOD).OOA adalah metode analisis yang memerika requirement (syarat/keperluan) yang harus dipenuhi sebuah sistem dari sudut pandang kelas-kelas dan objek-objek yang ditemui dalam ruang lingkup tertentu.Sedangkan OOD adalah metode untuk mengarahkan arsitektur software yang didasarkan pada manipulasi objek-objek sistem atau subsistem.

2.8.2.1 Objek (*Object)*

1. Objek adalah benda secara fisik dan konseptual yang ada di sekitar kita. Sebuah objek memiliki keadaan sesaat yang disebut state.

2. State dari sebuah objek adalah kondisi dari objek atau himpunan keadaan yang menggambarkan objek tersebut. State dinyatakan dengan nilai dari atribut objeknya.
3. Atribut adalah nilai internal suatu objek yang mencerminkan karakteristik objek, kondisi sesaat, koneksi dengan objek lain dan identitas.
4. Behaviour (perilaku objek) mendefinisikan bagaimana sebuah objek bertindak dan memberi reaksi. Behaviour ditentukan oleh himpunan semua atau beberapa operasi yang dapat dilakukan oleh objek tersebut, yang dicerminkan oleh interface, service, dan method dari objek tersebut.
5. Interface adalah pintu untuk mengakses service dari objek
6. Service adalah fungsi yang dapat dikerjakan oleh sebuah objek
7. Method adalah mekanisme internal objek yang mencerminkan perilaku objek tersebut

2.8.2.2 Kelas (*Class)*

Class adalah himpunan objek yang sejenis yaitu mempunyai sifat (atribut), perilaku umum (operasi), relasi umum dengan objek lain dan semantik umum. Class adalah abstraksi dari objek dalam dunia nyata.Class menetapkan spesifikasi perilaku dan atribut dari objek tersebut.

2.8.2.3 Kotak Hitam (*Black Box)*

Sebuah objek adalah kotak hitam. Konsep ini menjadi dasar implementasi objek. Dalam operasi OO hanya developer yang dapat memahami detail proses yang ada didalam kotak tersebut, sedangkan user tidak perlu mengetahui apa yang dilakukan yang penting mereka dapat menggunakan objek untuk memproses kebutuhan mereka. Kotak hitam berisi kode dan data.

1. Encapsulation, yaitu proses menyembunyikan detail implementasi sebuah objek. Untuk mengakses data objek tersebut adalah melalui interface. Untuk berkomunikasi dengan objek digunakan message.
2. Message adalah permintaan agar objek menerima untuk membawa metode yang ditunjukkan oleh perilaku dan mengembalikan result dari aksi tersebut kepada objek pengirim (sender)

2.8.2.4 Asosiasi dan Agregasi

1. Asosiasi adalah hubungan yang mempunyai makna antara sejumlah objek. Asosiasi digambarkan dengan sebuah garis penghubung diantara objeknya. Contohnya : Asosiasi karyawan dengan unit kerja. Setiap karyawan bekerja di satu unit kerja, sedangkan unit kerja dapat memiliki beberapa karyawan.
2. Agregasi adalah bentuk khusus sebuah asosiasi yang menggambarkan seluruh bagian pada suatu objek merupakan bagian dari objek yang lain. Contohnya : Kopling dan piston adalah bagian dari mesin, sedangkan mesin, roda, body merupakan bagian dari sebuah mobil.

2.8.3 Pengertian Object

Object adalah sesuatu yang dapat dilihat, disentuh atau dirasakan dan digunakan pengguna serta akan disimpan data dan perilakunya. Bisa berupa:

1. Orang, tempat, benda atau kejadian
2. Pegawai, pelanggan, guru, dosen, mahasiswa, murid.
3. Gudang, kantor, bangunan, ruangan.

2.8.4 Karakteristik Object

1. Identitas berarti bahwa data diukur mempunyai nilai tertentu yang membedakan entitas disebut Objek.
2. Objek dapat kongkrit, seperti halnya arsip dalam sistem, atau konseptual seperti kebijakan penjadualan dalam multiprocessing pada sistem operasi.
3. Setiap objek mempunyai sifat yang melekat pada identitasnya.
4. Dua objek dapat berbeda walaupun bila semua nilai atributnya identik.

2.8.4.1 Kelas Object

1. Kelas merupakan gambaran sekumpulan Objek yang terbagi dalam atribut, operasi, metode, hubungan, dan makna yang sama.
2. Suatu kegiatan mengumpulkan data (atribut) dan perilaku (operasi) yang mempunyai struktur data sama ke dalam satu grup.
3. Kelas Objek merupakan wadah bagi Objek. Dapat digunakan untuk menciptakan Objek.

2.8.5 Metodologi Berorientasi Objek

Metodologi adalah cara systematis untuk mengerjakan analisys and design. Dengan metodologi, pihak yang membangun system software dapat merencanakan dan mengulangi pekerjaan dilain waktu. Metodologi juga menghilangkan perbedaan notasi untuk suatu hal yang sama karena setiap oarng akan berbicara dalam bahasa yang sama.

2.8.5.1 Karakteristik Metodologi Berorientasi Objek

1. Encapsulation (pengkapsulan) merupakan dasar untuk pembatasan ruang lingkup program terhadap data yang diproses, data dan prosedur atau fungsi dikemas bersama-sama dalam suatu objek, sehingga prosedur atau fungsi lain dari luar tidak dapat mengaksesnya, data terlindung dari prosedur atau objek lain, kecuali prosedur yang berada dalam objek itu sendiri
2. Inheritance (pewarisan) adalah teknik yang menyatakan bahwa anak dari objek akan mewarisi data/atribut dan metode dari induknya langsung. Atribut dan metode dari objek induk diturunkan kepada anak objek, demikian seterusnya, Inheritance mempunyai arti bahwa atribut dan operasi akan dimiliki bersama di antara kelas yang mempunyai hubungan secara hirarki, suatu kelas dapat ditentukan secara umum, kemudian ditentukan spesifik menjadi subkelas. Setiap subkelas mempunyai hubungan atau mewarisi semua sifat yang dimiliki oleh kelas induknya, dan ditambah dengan sifat unik yang dimilikinya.

3. Polymorphism (polimorfisme) yaitu konsep yang menyatakan bahwa sesuatu yang sama dapat mempunyai bentuk dan perilaku berbeda, Polimorfisme mempunyai arti bahwa operasi yang sama mungkin mempunyai perbedaan dalam kelas yang berbeda, kemampuan objek-objek yang berbeda untuk melakukan metode yang sesuai dalam merespon pesan yang sama, seleksi dari metode yang sesuai, bergantung pada kelas yang seharusnya menciptakan objek.

2.9 Teknik Pemodelan OOAD

Model Objek :

1. Model objek Menggambarkan struktur statis dari suatu objek dalam sistem dan relasinya
2. Model objek berisi diagram objek. Diagram objek adalah graph dimana nodenya adalah kelas yang mempunyai relasi antar kelas.

Model Dinamik

1. Model dinamik menggambarkan aspek dari sistem yang berubah setiap saat.
2. Model dinamik dipergunakan untuk menyatakan aspek kontrol dari sistem.
3. Model dinamik berisi state diagram. State diagram adalah graph dimana nodenya adalah state dan arc adalah tarnsisi antara state yang disebabkan oleh event.

Model Fungsional

1. Model fungsional menggambrakan transformasi nilai data di dalam sistem.
2. Model fungsional berisi data flow diagram. DFD adalah suatu *graph* dimana *node* nya menyatakan proses dan *arc* nya adalah aliran data.

2.10.Alat Bantu Dalam Analisis dan Perancangan

2.10.1 UML (*Unified Modeling Languages*)

UML adalah bahasa untuk mengspesifikasi, memvisualisasi,membangun dan mendokumentasikan artefacts (bagian dari informasiyang digunakan atau dihasilkan oleh proses pembuatan perangkatlunak, seperti pada pemodelan bisnis dan sistem non perangkat lunak lainnya. (Modul Unaki,2003, hal: 2)

2.10.2 Notasi-notasi dalam Unified Modelling Languaage (UML)

Notasi Dasar Merupakan bentuk pemodelan yang digunakan untuk mempresentasikan elemen-elemen dalam UML, terdiri dari :

1. Notasi Struktual, Terdiri dari :

   a. Notasi Aktor

   Merupakan segala sesuatu yang berinteraksi dengansystem aplikasi komputer. Aktor dapat berupa orang, perangkat, atau objek lain dalam sistem yang sama.

b. Notasi Kelas

Merupakan pembentuk utama dari sistem berorientasiobjek karena kelas menunjukan kumpulan objek yangmempunyai attribute dari operasi yang sama. Kelas digunakan untuk mengabstraksikan elemen-elemendari sistem yang sedang dibuat.

c. Notasi Antar Muka (interface)

Merupakan kumpulan operasi tanpa implementasi darisuatu kelas. Implementasi operasi dalam antar muka dijabarkan oleh operasi dalam kelas, sehingga keberadaan anatar muka selalu disertai oleh kelas yang mengimplementasikan operasinya.

d. Notasi Kolaborasi

Mempresentasikan sekelompok peranan dan elemenlainya yang bekerja bersama untuk menyediakan perilaku bersama dari semua elemen yang ada.

e. Notasi Use Case

Gambaran fungsionalitas dari suatu sistem, sehingga pengguna sistem paham dan mengerti mengenai kegunaan sistem yang akan dibangun

f. Notasi kelas aktif

Mirip dengan sebuah kelas, hanya saja objek -objeknya merepresentasikan elemen-elemen yangmemiliki konkuren dengan elemen-elemen yang lain.

g. Notasi Komponen

Merupakan bentuk fisik yang dapat dipindahkandari sebuah sistem yang membentuk danmenyediakan realisasi dari sekumpulan antar muka.

h. Notasi Node

Merupakan elemen fisik yang harus ada pada saataplikasi dijalankan dan mempresentasikan sebuahsumber daya komputasi, yang memiliki memori,dan kemampuan pengolahan data.

2.10.3 Objek Desain

Sruktur sebuah sistem berbasis objek dapat di identifikasikan dan di klasifikasikan dalam tiga tipe objek yaitu :

a. Objek entity Merupakan sebuah objek yang berisi informasiyang berhubungan dengan kehidupan nyatayang bersifat menetap dan disimpan dalam suatudatabase yang memungkinkan pemanggilan danmanipulasi di wktu yang akan datang.
b. Objek Antar MukaMerupakan sebuah objek yang menyediakan peralatan dimana pengguna dapat berkomunikasi dengan system

c. Objek Control Merupakan sebuah objek yang berisi logikaaplikasi yang bukan merupakan tanggung jawabobjek entity. Contoh logika tersebut adalahaturan dan perhitungan bisnis yang melibatkan banyak objek.

2.10.4 Bagian bagian UML

Bagian-bagian utama dari UML adalah view, diagram, model element, dan general mechanism.

a. View

View digunakan untuk melihat sistem yang dimodelkan dari beberapa aspek yang berbeda. View bukan melihat grafik, tapi merupakan suatu abstraksi yang berisi sejumlah diagram.Beberapa jenis view dalam UML antara lain: use case view, logical view, component view, concurrency view,dan deployment view.

b. Use case view

Mendeskripsikan fungsionalitas sistem yang seharusnya dilakukan sesuai yang diinginkan external actors. Actor yang berinteraksi dengan sistem dapat berupa user atau sistem lainnya.View ini digambarkan dalam use case diagramsdan kadang-kadang dengan activity diagrams. View ini digunakan terutama untuk pelanggan, perancang (designer), pengembang (developer), dan penguji sistem (tester).

c. Logical view

Mendeskripsikan bagaimana fungsionalitas dari sistem, struktur statis (class, object,danrelationship) dan kolaborasi dinamis yang terjadi ketika object mengirim pesan ke object lain dalam suatu fungsi tertentu.View ini digambarkan dalam class diagrams untuk struktur statis dan dalam state, sequence, collaboration, dan activity diagram untuk model dinamisnya. View ini digunakan untuk perancang (designer) dan pengembang (developer).

d. Component view

Mendeskripsikan implementasi dan ketergantungan modul. Komponen yang merupakan tipe lainnya dari code module diperlihatkan dengan struktur dan ketergantungannya juga alokasi sumber daya komponen dan informasi administrative lainnya.View ini digambarkan dalam component view dan digunakan untuk pengembang (developer).

e. Concurrency view

Membagi sistem ke dalam proses dan prosesor.View ini digambarkan dalam diagram dinamis (state, sequence, collaboration, dan activity diagrams) dan diagram implementasi (component dan deployment diagrams) serta digunakan untuk pengembang (developer), pengintegrasi (integrator), dan penguji (tester).

f. Deployment view

Mendeskripsikan fisik dari sistem seperti komputer dan perangkat (nodes) dan bagaimana hubungannya dengan lainnya.View ini digambarkan dalam deployment diagramsdan digunakan untuk pengembang (developer), pengintegrasi (integrator), dan penguji (tester).

g. Diagram

Diagram berbentuk grafik yang menunjukkan simbol elemen model yang disusun untuk mengilustrasikan bagian atau aspek tertentu dari sistem. Sebuah diagram merupakan bagian dari suatu view tertentu dan ketika digambarkan biasanya dialokasikan untuk view tertentu. Adapun jenis diagram antara lain :

1. Use Case Diagram

Use case adalah abstraksi dari interaksi antara system dan actor. Use case bekerja dengan cara mendeskripsikan tipe interaksi antara user sebuah system dengan sistemnya sendiri melalui sebuah cerita bagaimana sebuah system dipakai. Use case merupakan konstruksi untuk mendeskripsikan bagaimana system akan terlihat di mata user. Sedangkan use case diagram memfasilitasi komunikasi diantara analis dan pengguna serta antara analis dan client.

2. Class Diagram

Class adalah dekripsi kelompok obyek-obyek dengan property, perilaku (operasi) dan relasi yang sama. Sehingga dengan adanya class diagram dapat memberikan pandangan global atas sebuah system. Hal tersebut tercermin dari class- class yang ada dan relasinya satu dengan yang lainnya. Sebuah sistem biasanya mempunyai beberapa class diagram. Class diagram sangat membantu dalam visualisasi struktur kelas dari suatu system.

3. Component Diagram

Component software merupakan bagian fisik dari sebuah system, karena menetap di komputer tidak berada di benak para analis. Komponent merupakan implementasi software dari sebuah atau lebih class. Komponent dapat berupa source code, komponent biner, atau executable component. Sebuah komponent berisi informasi tentang logic class atau class yang diimplementasikan sehingga membuat pemetaan dari logical view ke component view.Sehingga component diagram merepresentasikan dunia riil yaitu component software yang mengandung component, interface dan relationship.

4. Deployment Diagram

Menggambarkan tata letak sebuah system secara fisik, menampakkan bagian-bagian software yang berjalan pada bagian-bagian hardware, menunjukkan hubungan komputer dengan perangkat (nodes) satu sama lain dan jenis hubungannya. Di dalam nodes,executeable component dan object yang dialokasikan untuk memperlihatkan unit perangkat lunak yang dieksekusi oleh node tertentu dan ketergantungan komponen.

5. 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. State class tidak digambarkan untuk semua class, hanya yang mempunyai sejumlah state yang terdefinisi dengan baik dan kondisi class berubah oleh stateyang berbeda.

6. Sequence Diagram

Sequence Diagram digunakan untuk menggambarkan perilaku pada sebuah scenario. Kegunaannya untuk menunjukkan rangkaian pesan yang dikirim antara object juga interaksi antaraobject, sesuatu yang terjadi pada titik tertentu dalam eksekusi sistem.

7. Collaboration Diagram

Menggambarkan kolaborasi dinamis seperti sequence diagrams. Dalam menunjukkan pertukaran pesan, collaboration diagrams menggambarkan object dan hubungannya (mengacu ke konteks). Jika penekannya pada waktu atau urutan. Gunakan sequence diagrams, tapi jika penekanannya pada konteks gunakan collaboration diagram.

8. Activity Diagram

Menggambarkan rangkaian aliran dari aktivitas, digunakan untuk mendeskripsikan aktifitas yang dibentuk dalam suatu operasi sehingga dapat juga digunakan untuk aktifitas lainnya seperti use case atau interaksi.

2.10.5 Tujuan Penggunaan UML

Memberikan bahasa pemodelan yang bebas dari berbagai bahas pemrograman dan proses rekayasa. Menyatukan praktek-praktek terbaik yang terdapat dalam pemodelan. Memberikan model yang siap pakai, bahasa pemodelan visual yang ekspresif untuk mengembangkan dan saling menukar model dengan mudah dan dimengerti secara umum.UML bisa juga berfungsi sebagai sebuah (blue print) cetak biru karena sangat lengkap dan detail. Dengan cetak biru ini maka akan bias diketahui informasi secara detail tentang coding program atau bahkan membaca program dan menginterpretasikan kembali ke dalam bentuk diagram (reserve enginering).Perangkat lunak yang mendukung pembuatan diagaram UML.

1. StarUML (http://staruml.sourceforge.net/en/)

   StarUML adalah sebuah proyek open source untuk mengembangkan cepat, fleksibel, extensible, featureful, dan bebas-tersedia UML / platform MDA berjalan pada platform Win32.Tujuan dari proyek StarUML adalah untuk membangun sebuah alat pemodelan perangkat lunak dan juga platform yang menarik adalah pengganti alat UML komersial seperti Rational Rose, Bersama dan sebagainya

2. Acceleo (http://www.acceleo.org/pages/home/en)

Acceleo adalah generator kode yang mengubah model menjadi kode. Acceleo mudah digunakan dan menyediakan “dari rak” generator (Jee,. Bersih, Php …) dan templateeditor untuk Eclipse.

3. ArgoUML (http://argouml.tigris.org/)

ArgoUML adalah open source UML modeling tool terkemuka dan termasuk dukungan untuk semua diagram UML standar 1,4. Ini berjalan pada setiap platform Java dan tersedia dalam bahasa sepuluh. ArgoUML ditulis seluruhnya di Jawa dan menggunakan Java Kelas Foundation. Hal ini memungkinkan ArgoUML untuk berjalan di hampir semua platform.

2.10.6 Tools Yang Mendukung UML

a. Rational Rose

Dalam penelitian ini, *Rational Rose* berfungsi sebagai *tool* untuk pemodelansistem yang menggambarkan proses-proses yang ada pada sistem ini. *Rational Rose* adalah kakas (*tools)* pemodelan visual untuk pengembangan system berbasis objek yang sangat handal untuk digunakan sebagai bantuan bagi para pengembang dalam melakukan analisis dan perancangan sistem.

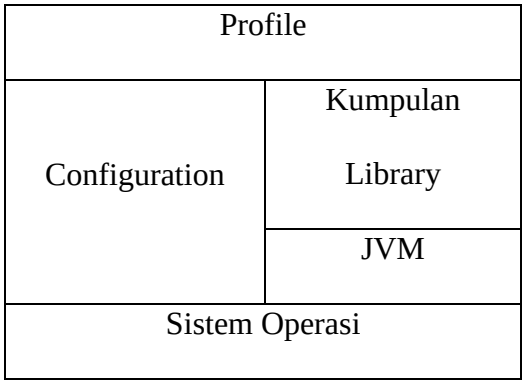
Rational Rose digunakan untuk melakukan pemodelan sistem sebelum pengembang menulis kode-kode dalam bahasa pemrograman

tertentu. Ia juga membantu analisis system dengan cara pengembang membuat diagram *use case* untuk melihat fungsionalitas sistem secara keseluruhan sesuai dengan harapan dan keinginan pengguna. Kemudian, ia juga menuntut pengembang untuk mengembangkan *Interaction* diagram untuk melihat bagaimana objek-objek saling bekerjasama dalam menyediakan fungsionalitas yang diperlukan (Adi Nugroho, 2005:20).

2.11 Pengertian J2ME

Java2 Micro edition atau yang biasa disebut J2ME adalah lingkungan pengembangan yang didesain untuk meletakkan perangkat lunak java pada barang elektronik beserta perangkat pendukungnya. Pada j2me, jika perangkat lunak berfungsi baik pada sebuah perangkat maka belum tentu juga berfungsi baik pada perangkat yang lain. J2ME membawa java ke dunia informasi, komunikasi, dan perangkat komputasi selain perangkat komputer *desktop* yang biasanya lebih kecil dibandingkan perangkat komputer *desktop*.J2ME biasa digunakan pada telepon selular, *pager, personal disital assistans* (PDA's) dan sejenisnya.

J2ME adalah bagian dari j2se, karena itu tidak semua *library* yang ada pada j2se dapat digunakan pada j2me. Tetapi j2me mempunyai beberapa *library* khusus yang tidak dimiliki j2se, arsitektur j2me dapat dilihat pada gambar



Gambar 2.3 Arsitektur J2ME

(Sumber M. Sahalahuddin, Rosa A.S, Pemrograman J2ME, Refisi Kedua, Informatika)

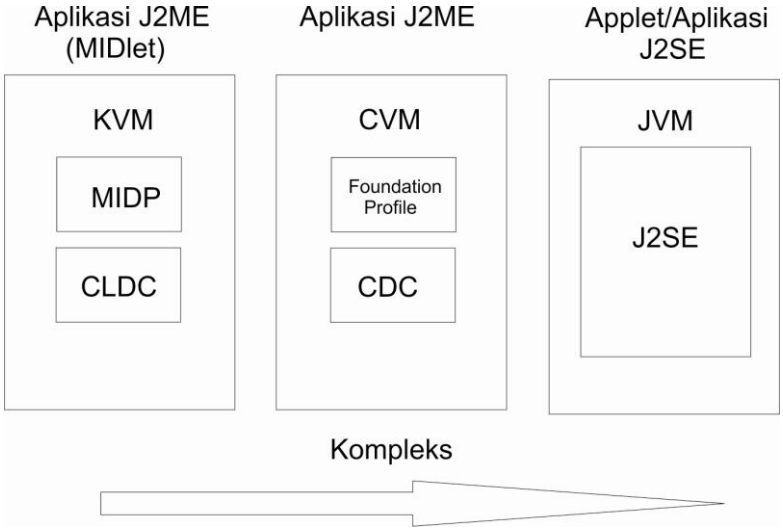
Teknologi j2me juga memiliki beberapa keterbatasan, terutama jika diaplikasikan pada ponsel.J2ME sangat tergantung pada perangkat (device) yang digunakan, bisa dari segi merk ponsel, maupun kemampuan ponsel, dan dukungannya terhadap teknologi j2me pada ponsel tersebut tidak dapat mengakses kamera. Keterbatasan lainnya adalah pada ukuran aplikasi, karena memori pada ponsel sangat terbatas. Sebagian besar ponsel tidak mengijinkan aplikasi j2me menulis pada file karena alasan keamanan.

Confituration merupakan java library minimum dan kapabilitas yang dipunya oleh para pengembang j2me, yang dimaksudnya sebuah *mobile device* dengan kemampuan java akan dioptimalkan untuk menjadi sesuai. *Configuration* hanyalah mengatur hal-hal tentang kesamaan sehingga dapat dijadikan ukuran kesesuaian antar-*device*. Misalnya sebuah lampu sepedah dirancang sedemikian rupa sehingga dapat digunakan oleh berjenis-jenis sepeda.Dalam j2me telah

didefinisikan dua buah konfigurasi yaitu CLDC (*Connected Limited Device Configuration*) untuk perangkat kecil CDC (*Connected Device Configuration*) untuk perangkat yang lebih besar.

Profile berbeda dengan *configuration*, *profile* membahas sesuatu yang spesifik untuk sebuah perangkat. Sebagai contoh misalnya, sebuah sepeda dengan merk tertentu tentu mempunyai cirri spesifik dengan sepeda lain.

Dalam j2me terdapat dua buah *profile* yaitu MIDP dan *foundation profile*. Keterhubungan antara *configuration* dan *profile* yang ada pada j2me beserta jenis mesin virtualnya dapat dilihat pada gambar



Gambar 2.4 Hubungan J2ME dan J2SE

(Sumber M. Sahalahuddin, Rosa A.S, Pemrograman J2ME, Refisi Kedua, Informatika)

2.12 PengertianNetBeans

Pengembangan NetBeans diawali dari Xelfi, sebuah proyek mahasiswa tahun 1997 di bawah bimbingan Fakultas Matematika dan Fisika Universitas Charles, Praha. Sebuah perusahaan kemudian dibentuk untuk proyek tersebut dan menghasilkan versi komersial NetBeans IDE hingga kemudian dibeli oleh Sun Microsystem pada tahun 1999. Sun kemudian menjadikan NetBeans open source pada bulan Juni tahun 2000. Sejak itu komunitas NetBeans terus berkembang.

Platform NetBeans adalah framework yang dapat digunakan kembali (reusable) untuk menyederhanakan pengembangan aplikasi desktop. Ketika aplikasi berbasis platform NetBeans dijalankan, kelas Main dari platform dieksekusi. Modul-modul yang tersedia ditempatkan di sebuah registry di dalam memori, dan tugas startup modul dijalankan. Secara umum, kode modul dimuatkan ke dalam memori hanya ketika ia diperlukan.

NetBeans IDE adalah IDE open source yang ditulis sepenuhnya dengan bahasa pemrograman Java menggunakan platform NetBeans. NetBeans IDE mendukung pengembangan semua tipe aplikasi Java (J2SE, web, EJB, dan aplikasi mobile). Fitur lainnya adalah sistem proyek berbasis Ant, kontrol versi,dan refactoring.

NetBeans Mobility Pack adalah alat untuk mengembangkan aplikasi yang berjalan pada perangkat bergerak (mobile), umumnya telepon seluler, tetapi juga mencakup PDA, dan lain-lain. NetBeans Mobility Pack

digunakan untuk menulis, menguji, dan debugging aplikasi untuk perangkat bergerak yang menggunakan teknologi berplatform Java Micro Edition (platform Java ME). Paket ini mengintegrasikan dukungan terhadap Mobile Information Device Profile (MIDP) 2.0, Connected Limited Device Configuration (CLDC) 1.1, dan Connected Device Configuration (CDC). Emulator dari pihak ketiga dapat diintegrasikan dengan mudah untuk lingkungan pengujian yang lebih kokoh. NetBeans Mobility Pack saat ini tersedia dalam dua klaster yang berbeda, yang satu memuat CDC dan yang lainnya CLDC.