BAB II

LANDASAN TEORI

2.1 Perancangan Perangkat Lunak

2.1.1 Pengertian Perangkat Lunak

Perangkat lunak terdiri dari program, dokumen dan data yang berisi perintah atau program komputer yang bila dieksekusi akan memberikan fungsi dan unjuk kerja seperti yang diinginkan.

Perancangan perangkat lunak adalah disiplin manajerial dan teknis yang berkaitan dengan pembuatan dan pemeliharaan produk perangkat lunak secara sistematis, termasuk pengembangan dan modifikasinya, yang dilakukan pada waktu yang tepat dan dengan mempertimbangkan faktor biaya.

Software engineer bertugas melakukan analisa, rancangan, uji dan verifikasi, dokumentasi, pemeliharaan perangkat lunak, serta pengelolaan proyek. *Software engineer* harus mempunyai keterampilan dan pengalaman seorang *programmer.*

Programmer adalah individu yang bertugas dalam hal rincian implementasi, pengemasan, dan modifikasi algoritma serta struktur data, dituliskan dalam sebuah bahasa pemrograman tertentu.

2.1.2 *Life Cycle* Perangkat Lunak

Siklus hidup (*Life-cycle)* sebuah perangkat lunak mencakup semua kegiatan yang perlu dilakukan untuk mendefinisikan, mengembangkan, menguji, mengantarkan, mengoperasikan, dan memelihara produk perangkat lunak. Beberapa model tersebut yaitu :

1. *Phased Model* (Model Fase)

Model ini membagi *life cycle* ke dalam sederetan kegiatan (fase). Setiap fase membutuhkan informasi masukan, proses, dan produk yang terdefinisi dengan baik. Deretan fase tersebut adalah :

a. Analisa, terdiri dari fase pertama yaitu perencanaan yang menghasilkan dua produk yaitu Pendifinisian Sistem (*System Definition)* dan Perencanaan Proyek (*Poject Plan)* dan fase kedua yaitu penetapan persyaratan yang menghasilkan sebuah produk spesifikasi kebutuhan perangkat lunak (*Software Requirements Specifications).*
b. Perancangan, melakukan identifikasi terhadap komponen perangkat lunak (fungsi, arus data, penyimpanan data), hubungan antar komponen, struktur perangkat lunak (dekomposisi menjadi modul-modul dan antar muka Perangkat Lunak). Fase ini menghasilkan arsitektur rinci, terutama dalam bentuk algoritma-algoritma.
c. Implementasi, adalah terjemahan langsung arsitektur rinci ke dalam bahasa pemrograman tertentu.
d. Pengujian, terdiri dari fase pertama yaitu uji integrasi dengan melakukan pengujian terhadap semua modul dan pengantarmukaan sehingga pada level sistem dapat beroperasi dengan benar, dan fase kedua yaitu uji penerimaan dengan melakukan berbagi pengujian mengacu kepada berbagai persyaratan yang telah ditentukan.
e. Pemeliharaan, terdiri dari fase peningkatan kemampuan, adaptasi terhadap lingkungan pemrosesan, dan melakukan berbagai koreksi atas kesalahan yang terjadi.

2. *Cost Model* (Model Biaya)

Model biaya adalah cara pandang lain model fase sebuah perangkat lunak dengan cara memperhatikan biaya berbagai kegiatan dalam proyek perangkat lunak. Biaya proyek adalah jumlah biaya dari setiap fase proyek. Biaya setiap fase mencakup biaya kegiatan dan penyiapan produk pada fase tersebut ditambah dengan biaya verifikasi konsistensi produk suatu fase terhadap semua fase sebelumnya.

3. *Prototype Model* (Model Prototipe)

Sebuah prototipe adalah model dari sebuah produk perangkat lunak tetapi dengan beberapa keterbatasan, misalnya : keterbatasan kemampuan, keandalan yang rendah, dan kinerja yang tidak efisien.

4. *Succesive Model* (Model Berurutan)

Sebuah model berurutan adalah model perancangan perangkat lunak yang tediri dari fase perencanaan dan analisis, perancangan, pembuatan, penilaian atau pengujian dan pemeliharaan.

2.1.3 Kualitas Perangkat Lunak.

Sebuah perangkat lunak yang berkualitas diukur dari beberapa atribut yaitu :

a. Kegunaan, yaitu pemenuhan terhadap kebutuhan pengguna;
b. Keandalan, yaitu kemampuan melaksanakan fungsi yang diinginkan;
c. Kejelasan, yaitu penulisan program dilakukan secara jelas dan mudah dimengerti;
d. Efisiensi, terutama dalam waktu eksekusi dan penggunaan *memory.*

Menurut Robert Glass suatu *software* memenuhi syarat berkualitas jika *software* yang dihasilkan :

1. Sesuai dengan kebutuhan
2. *User Friendly*
3. *Up to Date*
4. Manajemen file dan database terorganisir dengan baik
5. Tepat waktu dan sesuai anggaran
6. Adanya kepuasan dari pelanggan

2.2 Basis Data

2.2.1 Pengertian Basis Data

Definisi awal dari basis data adalah : kumpulan data yang saling berhubungan dan diorganisasikan sedemikian rupa sehingga dapat dimanfaatkan kembali dengan mudah, basis data memiliki prinsip utama melakukan pengaturan data dengan tujuan *fleksibilitas* dan kecepatan dalam pengambilan data.

Hubungan dalam sebuah basis data biasanya ditunjukkan dengan kunci dari tiap file yang ada. Dalam suatu file terdapat *record-record* yang sejenis, sama besar, sama bentuk, yang merupakan sutau kumpulan entitas yang seragam, sebuah konsep basis data memiliki beberapa hal sebagai berikut:

a. Entitas : merupakan tempat informasi direkam.
b. Atribut : disebut juga data elemen, data *field*, atau data *item* yang digunakan untuk menerangkan suatu entitas dan mempunyai harga tertentu.
c. Data *Value* : informasi atau data aktual yang disimpan pada setiap data, elemen, atau atribut.

d. *File* : kumpulan *record* sejenis yang mempunyai panjang elemen yang sama, atribut yang sama, namun berbeda nilai datanya.
e. *Record* : kumpulan elemen-elemen yang saling berkaitan menginformasikan tentang suatu entitas secara lengkap.

2.2.2 Tujuan dan Kriteria Basis Data.

Penggunaan basis data memiliki tujuan diantaranya adalah sebagai berikut:

a. Kemampuan dalam menangani data dalam jumlah besar.
b. Efisiensi pengelolaan data yang meliputi *speed, space* dan *accuracy.*
c. Kemampuan pemakaian *resource* basis data secara bersama-sama (*sharebility).*
d. Meniadakan duplikasi data yang tidak perlu dan menghindari inkonsistensi data.

Beberapa kriteria penting dalam membangun suatu basis data, yaitu:

a. Basis data bersifat *data oriented* dan bukan *program oriented*.
b. Basis data memiliki sifat dapat dikembangkan dengan mudah baik *volume*, ukuran maupun struktur basis datanya.
c. Basis data yang baik dapat memenuhi kebutuhan sistem-sistem baru secara mudah.
d. Dalam perkembangannya ada semacam standardisasi tipe file *database* sehingga basis data dapat digunakan oleh beberapa program aplikasi tanpa perlu mengubah basis data itu sendiri.

2.2.3 Persyaratan Basis Data

Suatu basis data yang baik memiliki beberapa ketentuan yang harus diperhatikan pada pembuatan file *database*, diantaranya :

a. Redudansi dan inkonsistensi data.

   Redudansi berarti melakukan penyimpanan data yang sama di beberapa tempat. Hal ini menyebabkan pemborosan atau *in-efensiensi* dan menimbulkan *inkonsistensi* data karena bila terjadi perubahan terhadap data maka data harus dirubah dibeberapa tempat.

b. Keamanan data.

   Penerapan aturan yang berhubungan dengan keamanan sistem menyebabkan tidak setiap pemakai sistem basis data diperbolehkan untuk mengakses semua data. Keamanan dapat diatur dan disesuaikan baik di tingkat basis data atau aplikasi.

c. Data integrity

   Integritas data menyangkut bagaimana mengatur kaitan antara file database agar dapat melakukan transaksi-transaksi ecara efisien.

d. Data Access

   Manajemen pengolahan data atau *Database Management System* (DBMS) dilakukan supaya data dalam basis data harus siap diakses oleh siapa saja yang membutuhkan dan mempunyai hak untuk mengaksesnya.

e. Data independence

   Sebuah program dalam sistem basis data, harus dipisah dengan *database* yang ada. Ini artinya perintah DBMS bebas terhadap *database* karena apapun perubahan terhadap *database*, semua perintah akan stabil tanpa ada yang perlu diubah, *data independence* dapat dibagi menjadi dua bagian yakni *Phisycal Data Independence* untuk mengubah pola fisik database tanpa mengakibatkan suatu

aplikasi program ditulis kembali dan *Logical Data Independence* untuk mengubah pola konseptual tanpa mengakibatkan suatu aplikasi ditulis kembali.

f. Isolasi Data

Isolasi data dilakukan dengan membuat satu format data yang sama dalam satu database.

g. Multi-User Support

Suatu sistem basis data harus mampu memberi dukungan kepada pemakaian program untuk banyak penguna (*multi-user*).

2.2.4 Abstraksi Data

Merupakan tingkatan dalam memandang sebuah database, dibagi kedalam tiga kelompok, yakni :

a. Level Fisik, merupakan level abstraksi paling rendah karena menggambarkan bagaimana data disimpan dalam kondisi sebenarnya.
b. Level Konseptual, menggambarkan data apa yang disimpan dalam *database* dan hubungan relasi yang terjadi antara data dari keseluruhan *database*.
c. Level Pandangan Pemakai (*View Level*), menggambarkan hanya sebagian saja yang dilihat dan dipakai dari keseluruhan *database*.

2.3 Interaksi Manusia Dan Komputer

2.3.1 Pengertian Interaksi Manusia dan Komputer

Interaksi Manusia dan Komputer (IMK) adalah suatu bidang ilmu yang terbuka untuk dipengaruhi dan mempengaruhi berbagai disiplin ilmu lainnya yang

cakupannya meliputi teknik dan ilmu komputer sampai dengan ilmu pengetahuan tentang manusia seperti psikologi, linguistik dan ergonomis.

Sistem komputer terdiri dari 3 aspek yaitu perangkat keras *(Hardware)*, perangkat lunak (*Software)*, dan teknisi (*Brainware)* ketiga aspek tersebut harus saling bekerjasama agar sebuah program komputer dapat bekerja dengan sempurna, untuk merancang suatu sistem IMK yang sempurna perancang harus mengetahui aspek teknik dari sistem komputer tersebut, tetapi juga harus mengerti bagaimana manusia mengolah informasi.

Panca indera manusia terdiri dari persepsi, kognitif dan pengendalian motorik, ketiga aspek tersebut memegang peranan penting dalam sembarang sistem manusia dan komputer, ketiga aspek ini memberikan kontribusi nyata untuk mendapatkan sistem yang efisien dan efektif secara keseluruhan, baik ditinjau dari sisi manusia ataupun komputer.

Suatu perangkat lunak yang baik dituntut untuk memberi kemudahan bagi pengguna atau *user friendly.* Istilah *user friendly* ini digunakan menunjuk kepada kemampuan yang dimiliki oleh suatu perangkat lunak atau program aplikasi yang mudah dioperasikan dan mempunyai sejumlah kemampuan lain sehingga penguna merasa betah dalam mengoperasikan program tersebut.

2.3.2 Jenis Program Aplikasi

Program aplikasi atau perangkat lunak dikelompokkan menjadi 2 jenis yaitu:

a. *Special Purpose Software*

   Yaitu program aplikasi yang digunakan untuk keperluan khusus dengan pengguna yang khusus pula.

b. *Pubblic Software*

Yaitu program aplikasi yang digunakan oleh banyak pengguna, biasanya model antar muka untuk jenis aplikasi ini menggunakan *customization interface* sehingga pengguna dapat menggunakan program aplikasi dengan antar muka sesuai dengan selera pribadi.

Untuk merancang *interface* (antar muka) suatu program biasanya dilakukan beberapa pendekatan dengan pengguna diantaranya yaitu:

a. *User Centered Design Approach*, yaitu perancangan antar muka program yang melibatkan pengguna dimana perancang dan pengguna duduk bersama untuk merancang antar muka program yang diinginkan pengguna.
b. *User Design Approach*, yaitu jenis perancangan antar muka program dimana pengguna sendiri yang merancangan antar muka program yang diinginkan.

2.3.3 Strategi Pengembangan Antar Muka

Antar muka pengguna secara ilmiah terbagi menjadi 4 komponen yaitu :

a. Model Pengguna

Yaitu model konseptual yang diinginkan oleh pengguna dalam memanipulasi informasi dan proses yang di aplikasikan pada informasi tersebut, model ini memungkinkan seorang pengguna untuk mengembangkan pemahaman yang mendasar tentang apa yang dikerjakan oleh program.

b. *Command Language* (Bahasa Perintah)

Yaitu peranti untuk memanipulasi model pengguna sehingga dengan cepat dapat dioperasionalkan.

c. Umpan Balik

   Yaitu kemampuan sebuah program yang membantu pengguna untuk mengoperasikan program sendiri, biasanya berbentuk pesan penjelasan atau indikasi adanya objek terpilih.

d. Tampilan Informasi

   Tampilan infromasi menunjukkan status informasi atau program ketika pengguna melakukan suatu tindakan.

2.3.4 Urutan Perancangan Perangkat Lunak

Langkah-langkah urutan perancangan suatu perangkat lunak dikerjakan dengan melakukan tahapan sebagai berikut :

1. Pemilihan Ragam Dialog

   Pemilihan ragam dialog dipengaruhi oleh karakteristik populasi pengguna (pemula, menenggah, dan mahir) tipe dialog yang diperlukan, kendala teknologi yang ada untuk mengimplementasikan ragam dialog interaktif dimana ragam dialog tersebut satu sama lain harus saling mendukung.

2. Perancangan Struktur Dialog

   Melakukan analisis dan menentukan model pengguna untuk membentuk struktur dialog yang sesuai.

3. Perancangan format pesan

   Meliputi perancangan tata letak tampilan dan keterangan tekstual.

4. Perancangan penanganan kesalahan

   Dilakukan untuk menghindari adanya *abnormal termination* (eksekusi program berhenti karena terjadi kesalahan), bentuk penanganan kesalahan tersebut yaitu :

a. Validasi pemasukan data : mekanisme untuk pemasukan ulang data yang salah.
b. Proteksi pengguna : program memberi peringatan ketika pengguna melakukan tindakan tak disengaja.
c. Pemulihan dari kesalahan : membatalkan tindakan yang baru saja dilakukan.
d. Penampilan pesan kesalahan yang tepat dan sesuai dengan kesalahan yang terjadi pada waktu itu.

5. Perancangan Struktur Data

   Menentukan struktur data yang dapat digunakan untuk menyajikan dan mendukung fungsionalitas komponen dan antar muka yang diperlukan.

2.4 *Global System For Mobile Communications* (GSM)

2.4.1 Sejarah GSM

Komunikasi bergerak (*mobile communication*) mulai dirasakan perlu sejak orang semakin sibuk pergi ke sana kemari dan memerlukan alat telekomunikasi yang siap dipakai sewaktu-waktu di mana saja ia berada. Kebutuhan ini ternyata tidak dibiarkan begitu saja oleh para *engineer* telekomunikasi. Mereka telah memikirkan standardisasi untuk komunikasi bergerak ini, salah satunya adalah GSM (*Global System for Mobile communications*)

Alokasi spektrum frekuensi untuk GSM awalnya dilakukan pada tahun 1979. Spektrum ini terdiri atas dua buah *sub-band* masing-masing sebesar 25MHz, antara 890MHz - 915MHz dan 935MHz - 960MHz. Sebuah *sub-band* dialokasikan untuk frekuensi *uplink* dan *sub-band* yang lain sebagai frekuensi *downlink.*

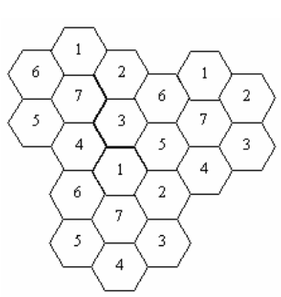
Kedua *sub-band* tersebut dibagi lagi menjadi kanal-kanal, sebuah kanal pada satu *sub-band* memiliki pasangan dengan sebuah kanal pada *sub-band* yang lain. Tiap *sub-band* dibagi menjadi 124 kanal, yang kemudian masing-masing diberi nomor yang dikenal sebagai *Absolute Radio Frequency Channel Number* (ARFCN). Jadi sebuah *handphone* yang dialokasikan pada sebuah ARFCN akan beroperasi pada satu frekuensi untuk mengirim dan satu frekuensi untuk menerima sinyal.

2.4.2 Teknik Modulasi dan *Bandwidth* GSM

Teknik modulasi yang digunakan pada GSM adalah *Gaussian Minimum Shift Keying* (GMSK). Teknik ini bekerja dengan melewatkan data yang akan dimodulasikan melalui Filter Gaussian. Filter ini menghilangkan sinyal-sinyal harmonik dari gelombang pulsa data dan menghasilkan bentuk yang lebih bulat pada ujung-ujungnya. *Bandwidth* yang dialokasikan untuk tiap frekuensi pembawa pada GSM adalah sebesar 200 kHz.

2.4.3 Struktur Sistem Seluler

Pembagian area dalam kumpulan sel-sel merupakan prinsip penting GSM sebagai sistem telekomunikasi selular. Sel-sel tersebut dimodelkan sebagai bentuk heksagonal seperti pada gambar berikut.

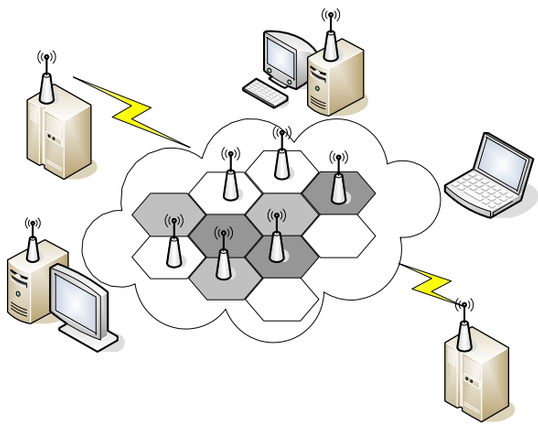


Gambar 2.1 Pembagian area sel frekuensi GSM

Tiap sel mengacu pada satu frekuensi pembawa atau kanal tertentu. Pada kenyataannya jumlah kanal yang dialokasikan terbatas, sementara jumlah sel bisa saja berjumlah sangat banyak. Untuk memenuhi hal ini, dilakukan teknik pengulangan frekuensi (*frequency re-use*).

Jelas bahwa semakin besar jumlah himpunan kanal, semakin sedikit jumlah kanal tersedia per-sel dan oleh karenanya kapasitas sistem menurun. Namun, peningkatan jumlah himpunan kanal menyebabkan jarak antara sel yang berdekatan kanal semakin jauh, dan ini mengurangi resiko terjadi interferensi sehingga desain sistem GSM memerlukan kompromi antara kualitas dan kapasitas.

Bagian paling rendah dari sistem GSM adalah *Mobile Station* (MS). Bagian ini berada pada tingkat pelanggan dan *portable*. Pada tiap sel terdapat *Base Transceiver Station* (BTS). BTS ini berfungsi sebagai *station* penghubung dengan MS dan merupakan sistem yang langsung berhubungan dengan *handphone*. BTS pada dasarnya hanya merupakan "pesuruh" saja. Otak yang mengatur lalu-lintas trafik di BTS adalah *Base Station Controller* (BSC). *Location Updating*, penentuan BTS dan proses *handover* pada percakapan ditentukan oleh BSC ini. Beberapa BTS pada satu region diatur oleh sebuah BSC. Seluruh BSC ini dihubungkan dengan *Mobile Switching Center* (MSC). MSC merupakan pusat penyambungan yang mengatur jalur hubungan antar BSC maupun antara BSC dan jenis layanan telekomunikasi lain (PSTN, operator GSM lain, AMPS, dll). Saat ini teknik *switching* terus berkembang, dan begitu pula pada layanan GSM. Beberapa operator GSM di Indonesia telah menerapkan *Intelegent Network* lanjutan dalam teknik *switching*nya.



Gambar 2.2 Jaringan sistem selular

2.5 *Short Message Service* (SMS)

2.5.1 Perkembangan Teknologi Ponsel

Sistem komunikasi telepon seluler terdiri dari dua bagian yaitu *base station* dan telepon. Hubungan antara *base station* dan telepon menggunakan sistem rasio berdaya rendah, area komunikasi sistem *wireless* dibagi-bagi dalam bentuk sel-sel yang memiliki diameter sekitar 8 mil. Setiap sel dilayani oleh sebuah *tower* pengirim dan penerima sinyal. Sistem ini kemudian dikenal dengan nama sistem seluler.

Tipe komunikasi *wireless* jarak jauh ini telah berkembang mulai dari komuniksi satu arah *Global Positioning System* (GPS), kemudian disusul dengan sistem *Pager* (Radio panggil) hingga lahirlah komunikasi dua arah generasi pertama atau 1G berupa layanan seluler analog yaitu *Advanced Mobile Phone System* (AMPS). Setelah itu muncul layanan komunikasi telepon seluler digital generasi kedua (2G) seperti layanan *Global System for Mobile Communication* (GSM) hingga saat ini perkembangan layanan komunikasi generasi ketiga (3G)

berupa *wireless digital broadband* yang mampu mentransmisikan file audio-video mulai mengisi pasar komunikasi.

2.5.2 Sejarah SMS

Short Message Service (SMS) merupakan sebuah layanan yang banyak diaplikasikan pada sistem komunikasi tanpa kabel, dikembangkan dan distandarisasi oleh suatu badan yang bernama *European Telecomunication Standards Institute* (ETSI) sebagai bagian dari pengembangan GSM fase 2, yang terdapat pada dokumentasi GSM 03.40 dan GSM 03.38. Fitur SMS ini memungkinkan perangkat Stasiun Seluler Digital (*Digital Cellular Terminal*, seperti ponsel) untuk dapat mengirim pesan dalam bentuk *alphanumeric* dan menerima pesan-pesan teks dengan panjang sampai dengan 160 karakter melalui jaringan GSM. (ETSI, 1996)

Isu SMS pertama kali muncul dibelahan Eropa pada sekitar tahun 1991 bersama sebuah teknologi komunikasi *wireless* yang saat ini cukup banyak pengunanya, yaitu *Global System for Mobile comunication* (GSM). Dipercaya bahwa pesan pertama yang dikirimkan menggunakan SMS dilakukan pada bulan Desember 1992, dikirimkan dari sebuah *Personal Computer* (PC) ke telepon *mobile* (bergerak) dalam jaringan GSM milik Vodafone Inggris. Perkembangannya kemudian merambah ke benua Amerika, dipelopori oleh beberapa operator komunikasi bergerak berbasis digital seperti BellSouth Mobility, PrimeCo, Nextel, dan beberapa operator lain.

Layanan SMS merupakan layanan yang bersifat *nonreal time* dimana sebuah *short message* dapat di-*submit* ke suatu tujuan, tidak peduli apakah tujuan tersebut aktif atau tidak. Karakteristik utama SMS adalah SMS merupakan sebuah sistem pengiriman data dalam paket yang bersifat *out-of-band* dengan *bandwidth* kecil,

dengan karakteristik ini pengiriman suatu *burst data* yang pendek dapat dilakukan dengan efisiensi yang sangat tinggi.

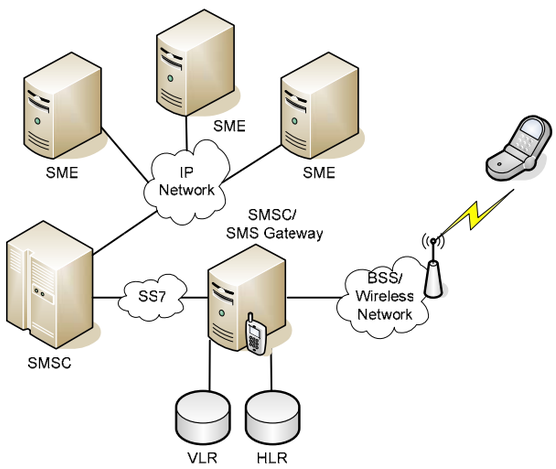
2.5.3 Mekanisme Distribusi Pesan SMS

Terdapat empat macam mekanisme distribusi pesan SMS oleh aplikasi SMS, yaitu:

a. *Pull*, yaitu pesan yang dikirimkan ke pengguna berdasarkan permintaan pengguna.
b. *Push – Event based*, yaitu pesan yang diaktivasi oleh aplikasi berdasarkan kejadian yang berlangsung
c. *Push - Schedule*, yaitu pesan yang diaktivasi oleh aplikasi berdasarkan waktu yang telah terjadwal.
d. *Push – Personal Profile*, yaitu pesan yang diaktivasi oleh aplikasi berdasarkan *profile* dan *preference* dari pengguna.

2.5.4 Arsitektur dan Elemen Layanan Jaringan SMS

Untuk implementasi layanan SMS, operator menyediakan apa yang disebut SMS Center (SMSC). Secara fisik SMSC dapat berwujud sebuah PC biasa yang mempunyai interkonektivitas dengan jaringan GMS.



Gambar 2.3 Arsitektur dasar jaringan SMS

2.5.4.1 Elemen dasar jaringan SMS

1. *Short Messaging Entities* (SME), suatu piranti yang dapat menerima dan mengirim pesan pendek.
2. *Short Message Service Center* (SMSC), kombinasi perangkat lunak dan perangkat keras yang brtanggung jawab memperkuat, menyimpan dan meneruskan pesan pendek antar SME dan piranti bergerak (*mobile phone).*
3. *Signaling System 7* (SS7), protokol *signalling* yang umum di gunakan dalam jeringan telepon seluler.
4. *Base Station System* (BSS), berfungsi mengendalikan satu atau lebih BTS dan bertanggung jawab dalam pemberian sumber data dan transmisi sinyal radio elektromagnetis antara MSC dan *mobile phone*.
5. *Home Location Register* (HLR), basis data yang digunakan untuk penyimpanan permanen, pengelolaan dan profil layanan.

6. *Visitor Location Register* (VLR), basis data yang berisi informasi temporal mengenai pelanggan yang berasal dari suatu HLR yang roaming ke HLR lainnya.

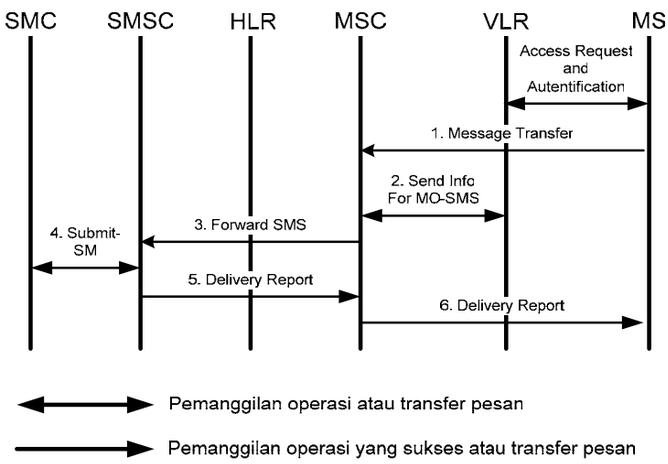
2.5.4.2 Elemen Layanan

SMS terdiri dari beberapa elemen layanan yang relevan terhadap penerimaan dan pengiriman pesan pendek yaitu:

1. *Message Expiration*, SMSC akan menyimpan dan mencoba mengirimkan kembali pesan yang mengalami kegagalan sampai pengiriman tersebut berhasil.
2. *Priority*, Untuk memberi tanda pesan-pesan yang penting dan membedakannya dari pesan biasa.

Sistem SMS memiliki dua layanan dasar *point-to-point* bagi pelanggan yaitu:

a. *Mobile-Oriented* (MO) *Short Message*, dikirimkan dari *mobile phone* yang MO-*Capable* ke SMSC dan dapat ditujukan ke *mobile phone* lainnya. Pada layanan ini selalu ada laporan yang dikirimkan ke *mobile phone*, baik yang mengkonfirmasikan pengiriman oesan pendek ke SMSC ataupun mengkonfirmasikan kegagalan pengiriman dan mengidentifikasikan penyebabnya.



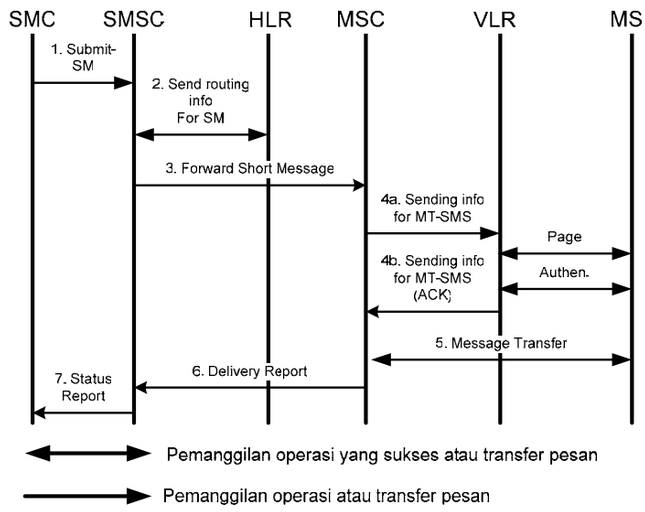
Gambar 2.4 Skenario pengiriman MO-SM

Keterangan :

1. MS mengirimkan SM ke MSC
2. MSC menginterogasi VLR utuk membuktikan bahwa pengiriman pesan tersebut tidak melanggar permintaan layanan pembatasan yang telah ditetapkan.
3. MSC mengirimkan pesan pendek ke SMSC dengan menggunakan operasi *Forward Short Message.*
4. SMSC mengirimkan pesan pendek ke SME
5. SMSC memberitahu MSC mengenai keberhasilan operasi *Forward Short Message.*
6. MSC mengembalikan hasil dari operasi MO-SM ke MS.

b. *Mobile Terminated* (MT) *Short Message*, dikirimkan dari SMSC ke *mobile phone* dan dapat sampai ke SMSC dari *mobile phone* lainnnya melalui MO-SM, pada layanan ini juga terdapat laporan yang diberikan kepada SMSC yang isinya

bisa berupa konfirmasi pengiriman pesn pendek ke *mobile phone* maupun informasi kegagalan pengiriman pesan.



Gambar 2.5 Skenario pengiriman MT-SM

Keterangan :

a. Pesan pendek dikirimkan dari SME ke SMSC.

b. Setelah menyelesaikan pengolahan internalnya, SMSC menginterogasi HLR dan menerima informasi *routing* untuk pelanggan *mobile*.

c. SMSC mengirimkan pesan pendek ke MSC dengan menggunakan operasi *Forward Short Message.*.

d. MSC mengambil informasi pelanggan dari VLR. Operasi ini dapat melibatkan prosedur autentifikasi.

e. MSC mengirimkan pesan pendek ke MS.

f. MSC mengembalikan hasil dari operasi *Forward Short Message* ke SMSC.

g. Jika diminta oleh SME, SMSC akan mengembalikan laporan status yang mengindikaikan pengiriman pesan pendek.

2.5.5 Jenis Aplikasi SMS

1. Aplikasi SMS untuk Konsumen Personal

Aplikasi untuk konsumen personal hingga saat ini masih mendominasi hingga 90% pengguna yang memakai aplikasi ini, aplikasi ini antara lain yaitu:

a. Pesan SMS antar pengguna (*person to person messaging*)
b. Notifikasi *Voice* dan *Fax-Mail*
c. Notifikasi *e-mail* internet
d. *Unified Messaging,* merupakan penggabungkan tiga macam *messaging* yaitu *voicemail, faxmail*, dan *e-mail*, dalam satu *accoun*t *unified messaging box*.

Pusat informasi merupakan layanan yang menyediakan informasi umum dan penting bagi pengguna. Informasi disajikan melalui dua mekanisme, yaitu *push-based* dimana infomasi diaktivasi oleh aplikasi dan *pull-based* yaitu informasi diminta (di-*request*) oleh pengguna.

2. Aplikasi SMS untuk Korporat.

Aplikasi SMS untuk korporat adalah aplikasi SMS yang dimodifikasi berdasarkan kebutuhan terhadap pelayanan informasi dan komunikasi dalam sebuah perusahaan atau kelompok orang. Aplikasi yang dapat dikembangkan antara lain:

a. *E-mail* Korporat adalah layanan bagi karyawan perusahaan yang memiliki *account* di *e-mail server* perusahaan untuk dapat memantau datangnya *e-mail* melalui pesan SMS.
b. Program *Affinity,* merupakan jenis layanan yang pada awalnya merupakan kerja sama dari *operator network* dengan perusahaan-perusahaan yang memiliki

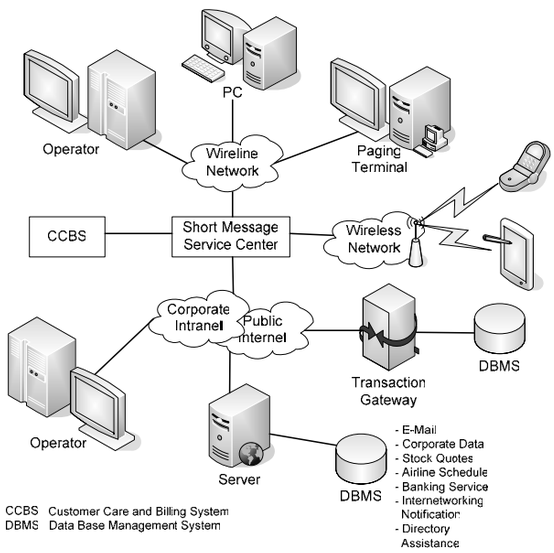
pelanggan cukup besar, layanan ini pada dasarnya merupakan bagian dari pelayanan terhadap konsumen atau *customer service*.

c. *Mobile Banking*, merupakan salah satu proggram *affinity* untuk layanan perusahaan perbankan. Layanan ini memberikan kemudahan kepada pelanggan nasabah sebuah bank dalam bertransaksi melalui rekeningnya

d. *E-Commerce*, merupakan perdagangan elektronik dimana transaksi perdagangannya dapat dilakukan melalui pesan SMS.

e. Penentuan lokasi (*Positioning*), aplikasi ini mengintegrasikan aplikasi G*lobal Positioning System* (GPS) yang berbasis satelit dengan pesan melalui SMS yang dapat memberitahukan kepada orang lain mengenai keberadaan seseorang.

f. *Remote Monitoring*, layanan dibangun dengan aplikasi yang dapat secara otomatis melaporkan kejadian dimana kinerja suatu alat mencapai titik kritis tertentu, maka pada saat itu juga aplikasi akan secara elektronis segera mengirim pesan pemberitahuan melalui SMS ke *handset administrator* atau penanggung jawab alat tersebut.



Gambar 2.6 Infrastruktur Jaringan Inovatif (NewNet, 1999)

2.6 Pengiriman Paket Data SMS

2.6.1 Arsitektur Jaringan SMS

Untuk implementasi layanan SMS, operator menyediakan apa yang disebut sebagai *SMS Center* (SMSC). Secara fisik SMSC dapat berwujud PC biasa yang mempunyai interkonektivitas dengan jaringan GSM, salah satu implementasi SMSC *Open Source* adalah Kannel, yang digunakan untuk membangun WAP dan SMS *Gateway.*

2.6.2 *Protocol Data Unit* (PDU)

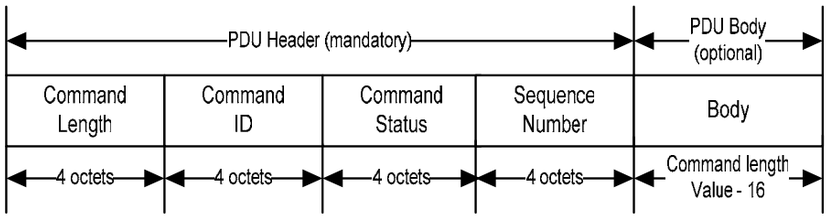
Satuan paket data yang dipertukarkan pada lapisan aplikasi dalam protokol SMS Manager disebut *Protocol Data Unit* (PDU). Dalam protokol tersebut terdapat beberapa macam format PDU dimana penggunaan masing-masing PDU terebut harus sesuai dengan fungsinya, sebagai contoh untuk mengirim sebuah pesan, harus digunakan PDU dengan format ***submit_sm, deliver_sm,*** atau ***data_sm.***

Umumnya cara pengiriman paket data dari satu titik ke titik lain, slah satu titik harus bertindak sebagai server dan titik lainnya sebagai *client*, Inisiatif dan pembentukan sebuah *session* (jenis permintaan atau perintah) dilakukan oleh client. Jenis session yang dipilih sepenuhnya diserahkan kepada *client* (otorisasi diterima atau tidak tetap dipegang oleh server), terdapat tiga buah *session* yang dapat dipilih yaitu:

1. *Receiver* (RX), bila *client* ingin dapat menerima paket data.
2. *Transmitter* (TX), bila *client* ingin dapat mengirimkan paket data.
3. *Transceiver* (TRX), bila *client* ingin dapat mengirim dan menerima paket data.

2.6.3 Format Umum PDU

Secara umum format PDU dalam protokol SMS Manager adalah sebagai berikut:



Gambar 2.7 Format umum PDU

Tabel 2.1. Format PDU

	SMPP PDU Field	Size (Octets)	Type	Description
PDU HEADER	*Command_length*	4	Integer	Overall size of PDU Including header and body
	Command_id	4	Integer	Identifies the PDU
	Command_status	4	Integer	Use to carry an error code
	Sequence_number	4	Integer	Use to uniquely identified PDU session
BODY	Standard Parameters	Var.	Mixed	The body part of PDU
	TLV Parameter	Var.	mixed	

Enambelas oktet pertama disebut PDU *header*, merupakan bagian yang bersifat mandatori atau harus selalu ada dalam setiap PDU. Sisanya disebut PDU *Body,* merupakan bagian yang bersifat opsional (tergantung jenis PDU). Body PDU terdiri dari parameter standar dan parameter tambahan yang disebut parameter *Tag-Length_Value* (TLV), berikut penjelasan masing-masing bagian tersebut :

1. ***command_Length***

   Command_length menyatakan panjang yang sesungguhnya dari PDU meliputi PDU Header dan PDU Body, termasuk didalamnya oktet dari *command_length* sendiri. Posisinya terletak pada awal atau field pertama dari setiap PDU.

2. *command_id*

command_id merupakan parameter yang mendgindikasikan jenis operasi dari protokol SMS Manager yang sedang dilakukan. Parameter ini dikodekan dalam 4 oktet dengan tipe data integer.Nilai parameter ini berkisar antara 0x00000000 hingga 0x00000001FF untuk parameter yang termasuk kategori request, kategori respons nilainya berkisar antara 0x80000000 hingga 0x800001FF, berikut contoh beberapa command_id :

Tabel 2.2 Contoh *command_id*

Command_id	Nilai
Bind_receiver	0x00000001
Bind_transmitter	0x00000002
Query_sm	0x00000003
Submit_sm	0x00000004
Deliver_sm	0x00000005
Unbind	0x00000006
Replace_sm	0x00000007
Cancel_sm	0x00000008
Bind_transceiver	0x00000009
Outbind	0x0000000B
Data_sm	0x00000103
Broadcast_sm	0x00000111
Bind_receiver_resp	0x80000001
Query_sm_resp	0x80000003
Submit_sm_resp	0x80000004
Deliver_sm_resp	0x80000005
Unbind_resp	0x80000006
Data_sm_resp	0x80000103
Broadcast_sm_resp	0x80000111
Query_broadcast_sm_resp	0x80000112

3. ***command_status***

   command_satus merupakan paramater yang menunjukkan status operasi dalam protokol dan hanya relevan untuk PDU-PDU kategori *respons*. Untuk PDU-PDU kategori *request*, parameter ini diabaikan.

4. ***sequence_number***

   sequence_number merupakan parameter yang digunakan untuk menandai sebuah PDU secara unik agar dapat dibedakan dengan PDU lain.

5. ***Parameter Standar***

   Parameter standar merupakan parameter yang terdapat pada PDU *Body* pada PDU-PDU tertentu (PDU yang non-header only, yaitu PDU *submission* dan *delivery*). Formatnya sendiri bervariasi tergantung PDU-nya dan merupakan kombinasi data dengan tipe integer.

6. **Parameter TLV**

   Tag-Length-Value, merupakan parameter dalam protokol yang dapat dibongkar-pasang sesuai dengan kebutuhan. Satu-satunya syarat hanyalah parameter ini, bila disertakan, harus diletakkan setelah paramater standar.

2.6.4 Contoh Penggunaan PDU

Langkah-langkah untuk menentukan kode PDU yang dibutuhkan untuk mengirimkan *(send)* SMS ke SMS *Center* :

a. Nomor SMS Center

   Terdiri dari tiga *subheader*, yaitu:

1. Jumlah pasangan heksa desimal SMS *Center* dalam bilangan heksa.
2. Kode nasional dan kode internasional, nomor kode nasional yang digunakan adalah 81 sedangkan nomor kode internasional adalah 91.
3. Nomor SMS *Center*, dalam pasangan bilangan heksa yang saling dipertukarkan dengan aturan tertentu, jika tersisa satu angka heksa yang tak memiliki pasangan maka angka tersebut dipasangkan dengan huruf F didepannya.

Contoh konversi nomor ponsel kedalam kode PDU dengan menggunakan :

1. Kode Nasional

   SMS Center menggunakan nomor 0856000000 maka :

   1.a. Terdapat 6 pasang angka heksa desimal.

   1.b. Kode nasional 81, ada 1 pasang

   1.c. 80-56-00-00-00

   Maka kode PDU yang didapat : 06818056000000

2. Kode Internasional

   SMS Center menggunakan nomor 62811000000, maka:

   2.a. Terdapat 7 pasang angka heksa desimal

   2.b. Kode internasional 91, ada 1 pasang

   2.c. 26-18-01-00-00-F0

   Maka kode PDU yang didapat : 07912618010000F0

b. Tipe SMS, untuk pengiriman atau *send* tipe SMS adalah 1 dalam heksa 01
c. Nomor referensi SMS, nomor ini dibiarkan dulu 0 atau dalam heksa 00, ponsel secara otomatis akan memberi nilai nomor referensinya.
d. Nomor ponsel penerima, terdiri atas tiga bagian *subheader* yaitu:

1. Jumlah bilangan desimal ponsel yang dituju dalam bilangan heksa
2. Kode nasional atau kode internasional, kode *subheader* nasional adalah 81 dan internasional adalah 91.
3. Nomor ponsel yang dituju dalam pasangan heksa yang saling dipertukarkan menurut aturan tertentu.

Contoh untuk nomor ponsel yang dituju adalah : 081321272326, maka konversi kedalam heksa desimal yaitu dengan cara:

1. Kode Nasional:

   1.a. terdapat 12 angka, berarti dalam heksa adalah 0C

   1.b. kode nasional adalah 81

   1.c 80-31-12-72-32-62

   Maka kode *subheader* PDU yang didapat adalah : 0C81803112723262

2. Kode Internasional

   2.a 62 81 32 12 17 32 6

   2.b. Terdapat 13 angka, berarti dalam heksa adalah 0D

   2.c. Kode internasional adalah 91

   2.d. 261823212732F6

   Maka kode *subheader* PDU yang didapat adalah : 0D91261823212732F6

e. Bentuk SMS

   00 : dikirim sebagai SMS

   01 : dikirim sebagai Telekomunikasi

f. Skema encoding data Input dan Output (I/O)

   Terdapat dua skema yaitu ;

   1. Skema 7 bit, ditandai dengan angka 00

2. Skema 8 bit, ditandai lebih besar dari nol (dalam heksa)

g. Jangka Waktu atau validasi sebelum SMS *expired* atau kadaluarsa, jika bagian ini diloncat itu berarti berlakunya SMS tidak dibatasi, sedangkan jika diisi dengan suatu bilangan integer yang kemudian diubah ke dalam bilangan heksa, maka bilangan tersebut mewakili jumlah waktu validasi SMS tersebut.

Tabel 2.3 Tabel validitas waktu SMS

Integer (INT)	**Jangka Waktu Validasi**
0-143	(INT+1)x 5 menit (berarti 5 sampai dengan 12 jam)
144-167	12 jam + (INT-143) x 30 menit
168-196	(INT-166) x 1 hari
197-255	(INT-192) x 1 minggu

h. Isi SMS

Terdiri dari dua subheader yaitu:

1. Panjang Isi (Jumlah huruf dalam SMS), misal untuk kata “hello” terdiri dari 5 huruf.
2. Isi berupa pasangan bilangan heksa desimal, skema 7 bit dapat dilihat pada tabel dibawah ini.

Tabel 2.4 Tabel konversi heksa desimal

				B7	0	0	0	0	1	1	1	1
				B6	0	0	1	1	0	0	1	1
				B5	0	1	0	1	0	1	0	1
B4	**B3**	**B2**	**B1**		**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
0	0	0	0	**0**	@	Δ	SP	0	-	P		p
0	0	0	1	**1**			!	1	A	Q	a	q
0	0	1	0	**2**	$	Φ	“	2	B	R	b	r
0	0	1	1	**3**		Γ	#	3	C	S	c	s
0	1	0	0	**4**				4	D	T	d	t
0	1	0	1	**5**			%	5	E	U	e	u
0	1	1	0	**6**			&	6	F	V	f	v
0	1	1	1	**7**			.	7	G	W	g	w

1	0	0	0	**8**			(	8	H	X	h	x
1	0	0	1	**9**			)	9	I	Y	i	y
1	0	1	0	**10**	LF		*	:	J	Z	j	z
1	0	1	1	**11**			+	;	K		k	
1	1	0	0	**12**			‘	<	L		l	
1	1	0	1	**13**	CR		_	=	M		m	
1	1	1	0	**14**				>	N		n	
1	1	1	1	**15**			/	?	O		o	

Contoh pengiriman kata “hello”, dengan langkah-langkah sebagai berikut:

1. Mengubah kata “hello” menjadi 7 bit
2. Mengubah ke dalam kode 8 bit.

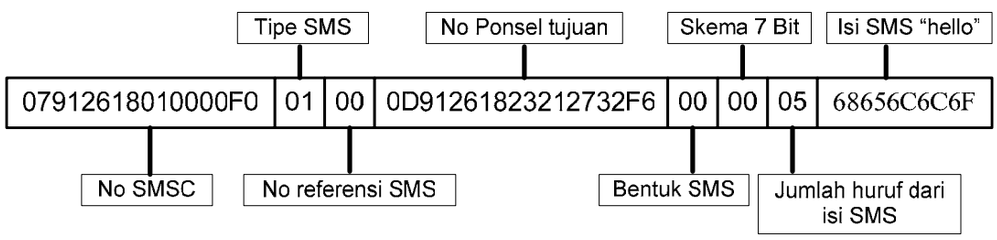
Langkah Pertama		
BIT	**7**	
h	110	1000
e	110	0101
l	110	1100
l	110	1100
o	110	1111

Langkah Kedua		
	6	**8**
h	0110	1000
	6	**5**
e	0110	0101
	6	**C**
l	0110	1100
	6	**C**
l	0110	1100
	6	**F**
o	0110	1111

Maka konversi dari kata “hello”kedalam bentuk heksadesimal adalah: 68656C6C6F

Setelah selesai dari langkah a sampai dengan langkah h, maka tinggal menggabungkan subheader-subheader tersebut, sehingga didapat PDU untuk mengirim SMS :

07912618010000F0 01 00 0D91261823212732F6 00 00 05 68656C6C6F



Gambar 2. 8 Contoh SubHeader kode PDU

Kode PDU ini akan diterima oleh SMS Center dan kemudian mengirimkannya ke nomor tujuan. Langkah-langkah untuk menentukan *subheader* PDU untuk menerima (*receive)* SMS tidak begitu jauh berbeda dari pengiriman SMS yaitu terdiri dari:

a. Nomor SMS Center
b. Tipe SMS, tipe SMS untuk menerima adalah 04
c. Nomor ponsel pengirim
d. Bentuk SMS
e. Skema Encoding
f. Validitas tanggal dan waktu SMS, jika tak dibatasi dilambangkan dengan 00
g. Batas waktu validasi
h. Isi SMS

2.7 AT-*Command*

AT-*Command* atau *Attention Command* yaitu perintah AT (Hayes AT-*Command)* yang digunakan untuk berkomunikasi dengan terminal (modem) melalui gerbang serial pada komputer. Dengan penggunaan perintah AT, dapat diketahui atau dibaca kondisi dari terminal, seperti mengetahui kondisi sinyal, kondisi baterai, mengirim pesan, membaca pesan, menambah item pada daftar telepon, dan sebagainya. Beberapa jenis ponsel memiliki *extended* AT *Command* yang bisa digunakan untuk mengambil informasi jenis, model hp, nomor *Internasional Mobile Station Equipment Identity* (IMEI) , SIM *Subscriber Identification Number* (IMSI), status batere, kekuatan sinyal, nama operator, lokasi dan cell ID. Pada tabel 2.1

diperlihatkan beberapa jenis perintah Hayes yang berhubungan dengan penanganan pesan-pesan AT-Command GSM 07.07.

Tabel 2.5. Contoh Perintah AT-Command GSM 07.07

AT-Command	Singkatan	Fungsi
ATE1	*Activate Command*	Mengaktivasi At-Command 07.07
AT+C.....	*Attention + Command.....*	Awal perintah hayes untuk....
AT+COPS	*Operator Selection*	Memilih operator jaringan
AT+CLCK	*Lock*	Menampilkan penguncian telepon
AT+CBC	*Battery Charge*	Menampilkan level batere
AT+CSQ	*Signal Quality*	Menampilkan kualitas sinyal
AT+CCLK	*Clock Mode*	Pengaturan Jam
AT+CPBR	*Phone Book Read*	Membaca daftar buku telepon
AT+CPBS	*Phone Book Selection*	Menampilkan isi buku telepon
AT+CCSA	*Service Center Address*	Menampilkan Lokasi *Service Center*
AT+CMGS	*Message Get Send*	Mengirimkan SMS
AT+CMGL	*Message Get List*	Membaca isi *inbox* dan *outbox* SMS
AT+CMGD	*Message Get Deleted*	Menghapus *inbox* dan *outbox* SMS
AT+CGMI	*Get Manufactured Identification*	Menampilkan merk *handphone*
AT-CGMM	*Get Mobile Model*	Menampilkan ID model Handphone
AT+CGSN	*Get Serial Number*	Menampilkan serial nomor IMEI
AT+CPWD	*Change Password*	Merubah *password*
AT+CLIP	*List Incoming Phone*	Menampilkan nomor telepon masuk
AT+CPBW	*Phone Book Write*	Memasukan data pada buku telepon
AT+CRSM	*Restricted SIM Access Mode*	Memblokir akses SIM
AT+CAOC	*Advice Of Charge*	Menampilkan biaya panggilan
AT+CALM	*Alert Sound Mode*	Pengaturan jenis nada dering
AT+CRSL	*Ringer Sound Level*	Pengaturan volume nada dering
AT+CMUT	*Mute Control*	Pengaturan tanpa suara
AT+CLVL	*Level Volume*	Pengaturan volume pengeras suara

AT+CVIB	*Vibration Mode*	Pengaturan nada getar
AT+CLCC	*List Curent Call*	Menampilkan penelepon terakhir
AT+CGMR	*GSM Mobile Version*	Menampilkan versi ponsel GSM
AT+CHLD	*Call Hold and Multiparty*	Menahan telepon masuk
AT+CPAS	*Query the telephone status*	Status telepon
AT+CPIN	*Enter PIN and query lock*	Memasukan PIN PUK atau SIM
AT+CHUP	*Terminate Call*	Untuk mengakhiri telepon masuk
AT+CEER	*Query the reason for disconnection of last call*	Menanyakan alasan pemutusan telepon terakhir.
AT+CREG	*Network Registration*	Registrasi Jaringan
AT+CCFC	*Call Forwarding*	Menampilkan telepon masuk
AT+CMEE	*Expanded Error Messages*	Menampilkan Pesan Error pada telepon
AT+CSCS	*Select TE character set*	Memilih set karakter TE mana yang dipakai.
AT+CSSN	*Supplementary service notifications*	Pemberitahuan penambahan service
AT+CRSM	*Restricted SIM Acces*	Membatasi akses SIM
AT+CACM	*Accumulated call meter*	Menghitung banyaknya panggilan
AT+CAMM	*Accumulated call meter maximum*	Menghitung maksimum banyaknya panggilan
AT+COPN	*Read operator names*	Membaca nama operator
AT+CPUC	*Price per unit and currency table*	Harga per unit dan skema mata uang.