

**PENGEMBANGAN APLIKASI SISTEM INFORMASI  
PENJADWALAN PERKULIAHAN ELEKTRONIK  
BERBASIS WEB DENGAN SMS GATEWAY**

**SKRIPSI**

**Diajukan sebagai salah satu syarat untuk memperoleh gelar Sarjana Teknik**

**PAMELA ALFA ADELIA DARMADJI**  
**06 06 04 2834**



**UNIVERSITAS INDONESIA  
FAKULTAS TEKNIK  
PROGRAM STUDI TEKNIK ELEKTRO  
DEPARTEMEN TEKNIK ELEKTRO  
DEPOK  
GENAP 2007/2008**

## HALAMAN PERNYATAAN ORISINALITAS

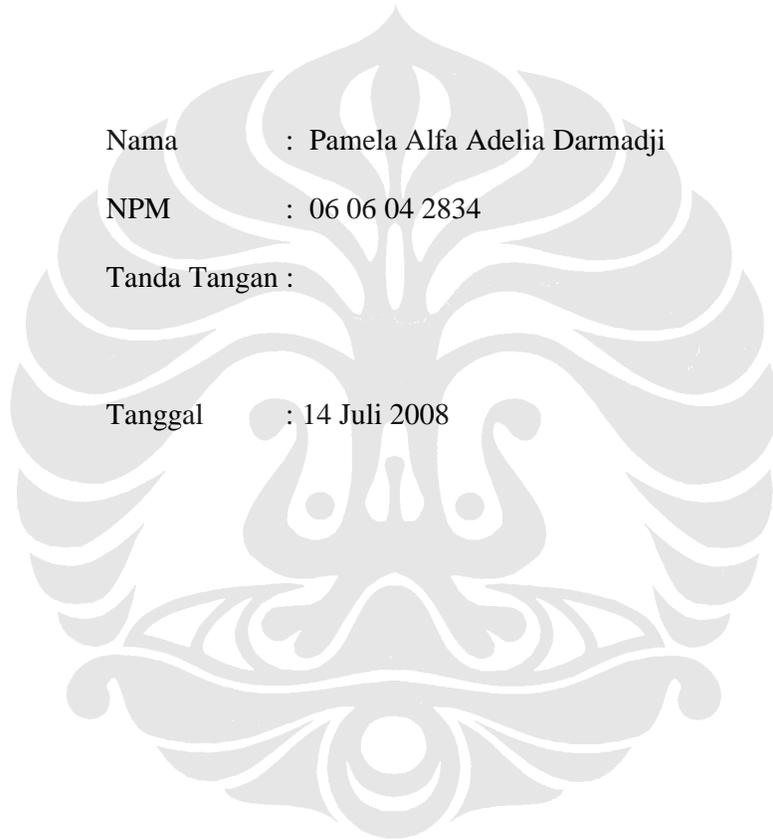
Skripsi ini adalah hasil karya saya sendiri,  
dan semua sumber baik yang dikutip maupun dirujuk  
telah saya nyatakan dengan benar.

Nama : Pamela Alfa Adelia Darmadji

NPM : 06 06 04 2834

Tanda Tangan :

Tanggal : 14 Juli 2008



## HALAMAN PENGESAHAN

Skripsi ini diajukan oleh :  
Nama : Pamela Alfa Adelia Darmadji  
NPM : 06 06 04 2834  
Program Studi : Teknik Elektro  
Judul Skripsi : Pengembangan Aplikasi Sistem Informasi  
Penjadwalan Perkuliahan Elektronik Berbasis Web  
dengan SMS Gateway

Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana Teknik pada Program Studi Teknik Elektro, Fakultas Teknik, Universitas Indonesia.

### DEWAN PENGUJI

Pembimbing : Ir. Purnomo Sidi Priambodo, MSc, Ph.D. ( )

Penguji : Arief Udhiarto ST,MT ( )

Penguji : F. Astha Ekadiyanto ST, M.Sc. ( )

Ditetapkan di : Depok

Tanggal : 14 Juli 2008

## UCAPAN TERIMAKASIH

Puji syukur saya panjatkan kepada Tuhan Yang Maha Esa, karena atas berkat dan rahmat-Nya, saya dapat menyelesaikan skripsi ini. Penulisan skripsi ini dilakukan dalam rangka memenuhi salah satu syarat untuk mencapai gelar Sarjana Teknik Jurusan Teknik Elektro pada Fakultas Teknik Universitas Indonesia. Saya menyadari bahwa, tanpa bantuan dan bimbingan dari berbagai pihak, dari masa perkuliahan sampai pada penyusunan skripsi ini, sangatlah sulit bagi saya untuk menyelesaikan skripsi ini. Oleh karena itu, saya mengucapkan terima kasih kepada:

- (1) Ir. Purnomo Sidi Priambodo, MSc, Ph.D., selaku dosen pembimbing yang telah menyediakan waktu, tenaga, dan pikiran untuk mengarahkan saya dalam penyusunan skripsi ini;
- (2) Orang tua dan keluarga saya yang telah memberikan bantuan dukungan material dan moral; dan
- (3) Sahabat-sahabat yang telah banyak membantu saya dalam menyelesaikan skripsi ini.

Akhir kata, saya berharap Tuhan Yang Maha Esa berkenan membalas segala kebaikan semua pihak yang telah membantu. Semoga skripsi ini membawa manfaat bagi pengembangan ilmu.

Depok, 14 Juli 2008

Penulis

## HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS

---

---

Sebagai sivitas akademik Universitas Indonesia, saya yang bertanda tangan di bawah ini:

Nama : Pamela Alfa Adelia Darmadji  
NPM : 06 06 04 2834  
Program Studi : Teknik Elektro  
Departemen : Teknik Elektro  
Fakultas : Teknik  
Jenis Karya : Skripsi

demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Indonesia **Hak Bebas Royalti Noneksklusif (*Non-exclusive Royalty-Free Right*)** atas karya ilmiah saya yang berjudul :

***Pengembangan Aplikasi Sistem Informasi Penjadwalan Perkuliahan Elektronik Berbasis Web dengan SMS Gateway***

beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini Universitas Indonesia berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan memublikasikan tugas akhir saya tanpa meminta izin dari saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat Di : Depok

Pada Tanggal : 14 Juli 2008

Yang menyatakan,

( Pamela Alfa Adelia Darmadji )

## ABSTRAK

Nama : Pamela Alfa Adelia Darmadji  
Program Studi : Teknik Elektro  
Judul : Pengembangan Aplikasi Sistem Informasi Penjadwalan Perkuliahan Elektronik Berbasis Web dengan SMS Gateway

Dengan berkembang pesatnya teknologi komputer dan informasi, maka peranan komputer dan web dalam menyediakan layanan informasi menjadi faktor penting untuk menunjang kerja berbagai institusi dan perorangan. Selain itu teknologi telekomunikasi – khususnya jaringan seluler – telah menjadikan kontak antar perorangan dapat dilakukan dengan mudah dari mana pun dan kapan pun. Aktivitas keseharian terasa lebih mudah dengan adanya perangkat telekomunikasi yang disebut dengan telepon seluler. Maraknya penggunaan ponsel untuk berkirim SMS, memunculkan gagasan untuk membuat layanan informasi berbasis SMS. Penelitian ini difokuskan pada perancangan aplikasi sistem informasi penjadwalan perkuliahan elektronik berbasis web dan dengan memanfaatkan teknologi SMS Gateway.

Analisis dilakukan dengan cara mempelajari bagaimana sistem informasi penjadwalan pada umumnya bekerja, komunikasi sistem informasi tersebut dengan SMS Gateway, dan program antarmuka *user* untuk dapat mengakses aplikasi. Pengembangan aplikasi ini diharapkan dapat meningkatkan efektifitas penyampaian informasi jadwal perkuliahan yang ditujukan kepada dosen dan mahasiswa.

Kata kunci :  
SMS Gateway, SMPP, Aplikasi, Kalender Elektronik, PDU, Java

## **ABSTRACT**

Nama : Pamela Alfa Adelia Darmadji  
Study Program: Electrical Engineering  
Title : Application Development of Study Electronic Scheduler Web Based Using SMS Gateway

The rapid growth of computer and information technology makes the role of computer and web in providing information service to be an important factor to support either institution or individual activity. Communication technology – especially cellular network – also has made communicating become easier between people anywhere and anytime. Daily activity becomes easier by having telecommunication device which called cell phone. The great amount usage of cell phone in sending short message has developed an idea in creating information system based on SMS. This paper is focused on designing web based electronic study scheduler information system application using SMS gateway technology and web based.

The analysis done by examining how the scheduler information system works, how the system communicate with SMS gateway, and user interface to access the application. The development expected to improve the effectiveness in providing schedule information for lecturers and students.

Key words :

SMS Gateway, SMPP, Application, Electronic Scheduler, PDU, Java

# DAFTAR ISI

	Halaman
HALAMAN JUDUL	i
HALAMAN PERNYATAAN ORISINALITAS	ii
LEMBAR PENGESAHAN	iii
KATA PENGANTAR	iv
LEMBAR PERSETUJUAN PUBLIKASI KARYA ILMIAH	v
ABSTRAK	vi
ABSTRACT	vii
DAFTAR ISI	viii
DAFTAR GAMBAR	x
DAFTAR TABEL	xi
DAFTAR SINGKATAN	xii
<b>BAB I PENDAHULUAN</b>	1
1.1. LATAR BELAKANG	1
1.2. TUJUAN	2
1.3. BATASAN MASALAH	2
1.4. METODOLOGI PENELITIAN	2
1.5. SISTEMATIKA PENULISAN	3
<b>BAB II LANDASAN TEORI</b>	4
2.1. SHORT MESSAGE PEER-TO-PEER PROTOCOL (SMPP)	4
2.1.1. Format PDU	6
2.1.2. SMPP Session	12
2.2. PEMROGRAMAN JAVA	17
2.3. KONSEP APLIKASI MULTI TIER	19
2.3.1. Perbedaan Web Based Programming dengan System Programming	20
2.3.2. Teknologi Alternatif	20
2.4. SQL QUERY	21
2.4.1. Query Pengelolaan Database	22
2.4.2. Query Satu Tabel	23

<b>BAB III PERANCANGAN SISTEM</b>	25
3.1. SISTEM APLIKASI KALENDER ELEKTRONIK PADA SERVER	25
3.1.1. Server Aplikasi (dan Modem SMS Gateway)	25
3.1.2. Sistem Database	27
3.2. TERMINAL USER/PENGGUNA	28
3.3. PEMODELAN SISTEM	32
3.4. DIAGRAM ALIR SISTEM	33
3.5. PERANCANGAN PEMROGRAMAN JAVA	35
3.5.1. Web Pages	36
3.5.2. Web.xml	36
3.5.3. Source Packages	37
<b>BAB 4. PENGUJIAN SISTEM</b>	39
4.1. PERSIAPAN PENGUJIAN SISTEM	39
4.2. PENGUJIAN SISTEM	41
4.2.1 Pengujian SMS Gateway	41
4.2.2 Pengujian Desktop User	42
4.3. ANALISA PENGUJIAN SISTEM	43
<b>BAB 5. KESIMPULAN</b>	51
<b>DAFTAR REFERENSI</b>	52

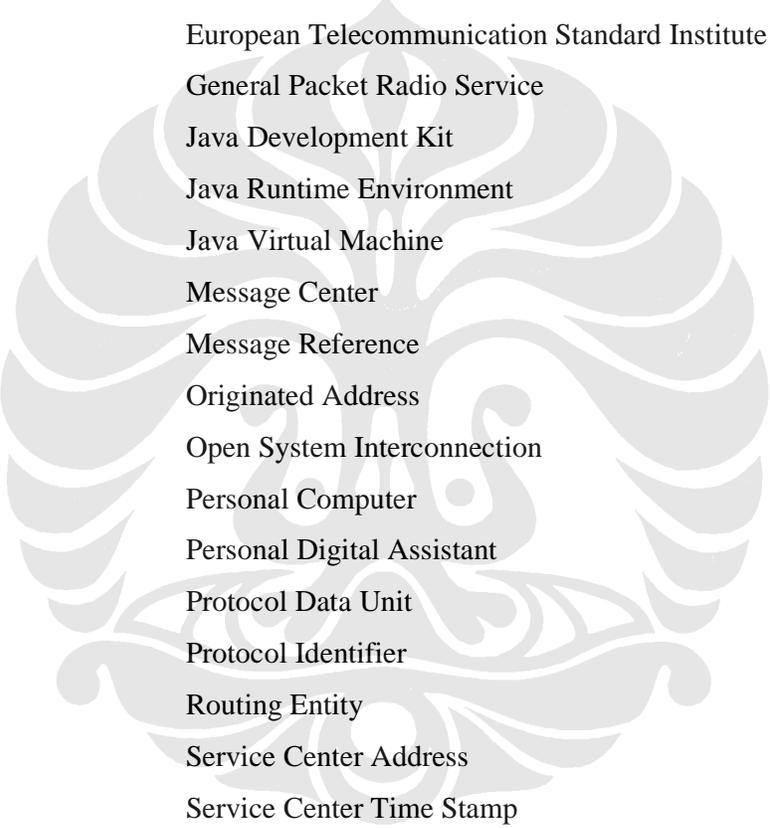
## DAFTAR GAMBAR

	Halaman
<b>Gambar 2.1</b>	Skema Format SMS PDU Pengirim 7
<b>Gambar 2.2</b>	PDU Type Pengirim 8
<b>Gambar 2.3</b>	Skema Format SMS PDU Penerima 10
<b>Gambar 2.4</b>	PDU Type Penerima 11
<b>Gambar 2.5</b>	Contoh operasi protokol SMPP 15
<b>Gambar 2.6</b>	Proses pengekseskuan program Java 18
<b>Gambar 3.1</b>	Skema sistem keseluruhan pada aplikasi sistem informasi penjadwalan perkuliahan elektronik 26
<b>Gambar 3.2</b>	Hubungan database pada server database sistem aplikasi informasi penjadwalan perkuliahan elektronik. 28
<b>Gambar 3.3</b>	Duplikasi Tabel Matkul dan Tabel Matkul harian 29
<b>Gambar 3.4</b>	Tampilan Awal dan Menu Login ke Sistem 30
<b>Gambar 3.5</b>	Tampilan Login User 30
<b>Gambar 3.6</b>	Contoh Tampilan Tabel Daftar Mahasiswa 31
<b>Gambar 3.7</b>	Contoh Tampilan Tabel Daftar Matakuliah 31
<b>Gambar 3.8</b>	Tampilan Menu Dosen 31
<b>Gambar 3.9</b>	Tampilan Menu Daftar Mata kuliah 32
<b>Gambar 3.10</b>	Tampilan Menu Peserta Mata Kuliah 32
<b>Gambar 3.11</b>	Tampilan Menu Peserta Mata Kuliah 33
<b>Gambar 3.12(a)</b>	Diagram Alir Web Server 34
<b>Gambar 3.12(b)</b>	Diagram Alir SMS Server 35
<b>Gambar 3.13</b>	Contoh Alur Pengekseskuan Program Java menu login 35
<b>Gambar 4.1</b>	Skema Pengujian Sistem Aplikasi Informasi Penjadwalan Perkuliahan Elektronik 40
<b>Gambar 4.2</b>	Halaman Login 43
<b>Gambar 4.3</b>	Tampilan Menu Edit Mata Kuliah 48
<b>Gambar 4.4</b>	Tampilan Menu Edit pada User 'Dosen' 49
<b>Gambar 4.5</b>	Tampilan Menu Penambahan Dosen 49
<b>Gambar 4.6</b>	Hasil Penambahan Dosen 49

## DAFTAR TABEL

	Halaman
<b>Tabel 2.1</b> Service Center Address (SCA)	8
<b>Tabel 2.2</b> Destination Address (DA)	9
<b>Tabel 2.3</b> Validity Period (VP)	10
<b>Tabel 2.4</b> Service Center Time Stamp (SCTS)	12
<b>Tabel 2.5</b> Hal yang dapat dilakukan pada <i>Query</i> satu tabel	23
<b>Tabel 4.1</b> Kebutuhan Pengujian Sistem	39
<b>Tabel 4.2</b> Pengaturan Port Siemens C55	40
<b>Tabel 4.3</b> Format SMS PDU Pengirim Sesuai dengan Pengujian	44
<b>Tabel 4.4</b> SCA PDU Pengirim oleh Sistem	44
<b>Tabel 4.5</b> DA PDU Pengirim oleh Sistem	44
<b>Tabel 4.6</b> User Data PDU Pengirim oleh Sistem	46
<b>Tabel 4.7</b> Hasil Pengujian dengan Menambah Jumlah User	47
<b>Tabel 4.8</b> Hasil Pengujian Mengubah bit per second modem SMS Gateway	47

## DAFTAR SINGKATAN



CBD	Cell Broadcast Center
DA	Destination Address
DCS	Data Coding Scheme
DDL	Data Definition Language
DML	Data Manipulation Language
ESME	External Short Messaging Entity
ETSI	European Telecommunication Standard Institute
GPRS	General Packet Radio Service
JDK	Java Development Kit
JRE	Java Runtime Environment
JVM	Java Virtual Machine
MC	Message Center
MR	Message Reference
OA	Originated Address
OSI	Open System Interconnection
PC	Personal Computer
PDA	Personal Digital Assistant
PDU	Protocol Data Unit
PID	Protocol Identifier
RE	Routing Entity
SCA	Service Center Address
SCTS	Service Center Time Stamp
SMPP	Short Message Peer-to-Peer Protocol
SMS	Short Message Service
SMSC	Short Message Service Center
SQL	Data Communication Equipment
UD	User Data
UDL	User Data Length
USSD	Unstructured Supplementary Services Server
VP	Validity Period

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Berkembang pesatnya teknologi komputer dan web yang ada sekarang ini memaksa kita untuk mengikuti arus perkembangan teknologi tersebut. Tidak dapat dipungkiri lagi bahwa peranan komputer dan web dalam menyediakan layanan informasi merupakan faktor penting untuk berbagai institusi maupun perorangan. Dengan sebuah layanan yang *simple* dan tidak terbatas oleh jarak, berbagai macam lembaga maupun perorangan menyediakan dan mengambil data dan informasi melalui komputer dan web.

Disamping teknologi informasi tersebut di atas, terdapat teknologi lain yang mendukung teknologi tersebut, yaitu teknologi telekomunikasi. Kontak dengan orang lain dapat dilakukan dengan mudah dari mana pun dan kapan pun. Teknologi ini terdiri dari teknologi suara (telepon) dan data (mencakup GPRS dan SMS). Aktivitas keseharian terasa lebih mudah dengan adanya perangkat telekomunikasi yang disebut dengan telepon seluler. Selain digunakan untuk berkomunikasi suara dengan lawan bicara, ponsel juga memiliki kemampuan lain, yaitu dapat digunakan untuk saling berkiriman pesan melalui teks yang lebih populer dengan nama SMS. Maraknya penggunaan ponsel untuk berkiriman SMS maka muncul gagasan untuk membuat layanan informasi berbasis SMS. Dalam waktu singkat beberapa jenis layanan informasi yang berbasis SMS bermunculan.

Salah satu aplikasi yang akan penulis kembangkan dalam penelitian ini adalah sistem aplikasi informasi penjadwalan perkuliahan elektronik. Dengan aplikasi informasi penjadwalan perkuliahan elektronik ini, pengumuman pembatalan kelas atau penggantian jadwal kelas yang masih manual (misalnya pengumuman yang hanya di tempel di dinding pengumuman, dan berlaku jika mahasiswa/mahasiswi tersebut datang ke kampus) dapat dibuat menjadi penjadwalan elektronik, sehingga mis-informasi dapat dihindari. Aplikasi informasi penjadwalan perkuliahan elektronik ini dirancang untuk dapat diakses via intranet, dan sistem dapat mengirimkan SMS *reminder* sebagai pengingat bahwa hari tertentu ada jadwal perkuliahan. Dengan adanya aplikasi ini,

diharapkan dapat meningkatkan efektifitas dan efisiensi dari penyampaian informasi pengumuman tersebut.

## 1.2 Tujuan

Tujuan utama dari tugas akhir ini adalah pembuatan aplikasi informasi penjadwalan perkuliahan elektronik yang memiliki fitur sebagai berikut :

- Aplikasi dapat meningkatkan efektifitas penyampaian informasi jadwal perkuliahan yang diutamakan ditujukan kepada dosen pengajar dan mahasiswa.
- Aplikasi informasi penjadwalan perkuliahan elektronik ini dapat mengirimkan *reminder* berupa sms ke mahasiswa-mahasiswa dan dosen dan sms update jika terdapat perubahan jadwal perkuliahan.
- Dosen juga dapat mengakses aplikasi melalui intranet, untuk memberi tahu jika dosen tersebut tidak dapat hadir, dan aplikasi akan mengirimkan pesan ke mahasiswa berupa pesan sms.

## 1.3 Batasan Masalah

Penelitian ini dibatasi pada pengembangan aplikasi informasi penjadwalan perkuliahan elektronik, termasuk di dalamnya fitur-fitur yang dimiliki oleh informasi penjadwalan perkuliahan tersebut.

## 1.4 Metodologi Penelitian

1. Studi literatur untuk mendukung penelitian ini, meliputi pencarian referensi serta pendalaman materi.
2. Analisa dan studi kebutuhan sistem aplikasi informasi penjadwalan perkuliahan elektronik secara menyeluruh.
3. Perancangan perangkat lunak aplikasi informasi penjadwalan perkuliahan elektronik.
4. Pengujian aplikasi.

## 1.5 Sistematika Penulisan

Sistematika penulisan penelitian ini disusun atas lima bab, yaitu :

**BAB I : PENDAHULUAN**

Bab ini membahas mengenai latar belakang, tujuan, batasan masalah, metodologi penelitian, dan sistematika penulisan laporan penelitian.

**BAB II : LANDASAN TEORI**

Bab ini mengemukakan dasar-dasar teori yang digunakan untuk melandasi penelitian mengenai pengembangan aplikasi informasi penjadwalan perkuliahan elektronik, serta mendukung pembuatan tugas akhir ini.

**BAB III : PERANCANGAN SISTEM**

Bab ini membahas mengenai perancangan dan pembuatan sistem, yaitu alur kerja sistem, detil setiap bagian sistem, dan pembangunan program, baik dari sisi server maupun klien.

**BAB IV : PENGUJIAN SISTEM**

Bab ini berisi mengenai pengujian kerja sistem secara keseluruhan.

**BAB V : PENUTUP**

Bab ini berisi kesimpulan dari analisis program dan kegiatan dalam penyusunan tugas akhir.

## BAB II

### LANDASAN TEORI

Bab ini menguraikan dasar-dasar teori yang digunakan untuk mendukung penelitian mengenai aplikasi informasi penjadwalan perkuliahan elektronik, termasuk perancangannya. Ada pun pokok-pokok yang dibahas adalah SMPP (*Short Message Peer-To-Peer Protocol*), dasar pemrograman Java, dan dasar SQL (*Structure Query Language*).

#### 2.1 Short Message Peer-To-Peer Protocol (SMPP)

SMPP merupakan sebuah protokol standar industri yang digunakan dalam pertukaran pesan pendek antara *External Short Messaging Entity* (ESME), *Routing Entity* (RE), dan *Message Center* (MC). *Message center* merupakan terminology generik untuk menyebutkan beberapa entitas seperti *Short Message Service Center* (SMSC), *GSM Unstructured Supplementary Services Server* (USSD), atau *Cell Broadcast Center* (CBD). ESME merupakan entitas yang berada di luar jaringan komunikasi wireless, berfungsi sebagai terminal penyedia layanan berbasis SMS seperti *WAP Proxy Server*, *Email Gateway*, atau *Voice Mail Server*. *Routing Entity* merupakan nama generik yang diberikan untuk menyebut beberapa entitas dalam system yang berfungsi melakukan *routing* SMS [2].

Protokol SMPP merupakan sebuah protokol yang berjalan pada lapisan aplikasi, seperti halnya protokol-protokol lain dalam konteks komunikasi data dalam jaringan komputer seperti HTTP, FTP, Rlogin, WAP, dan lain-lain. Satuan paket data yang dipertukarkan pada lapisan aplikasi dalam protokol SMPP disebut dengan PDU (*Protocol Data Unit*). Dalam protokol SMPP terdapat beberapa macam format PDU dimana penggunaan masing-masing PDU tersebut harus sesuai dengan fungsinya. Sebagai contoh, untuk mengirim pesan pendek, harus digunakan PDU dengan format *submit\_sm*, *deliver\_sm*, atau *data\_sm*.

Secara umum, cara pengiriman data dari satu titik ke titik lain yang dilakukan oleh protokol SMPP, menganut prinsip-prinsip sebagai berikut :

- Dalam komunikasi SMPP antar dua titik, salah satu titik harus bertindak sebagai server dan titik lainnya sebagai klien.
- Inisiatif koneksi dan pembentukan sebuah *session* dilakukan oleh klien. Jenis *session* yang dipilih sepenuhnya diserahkan kepada klien (otorisasi diterima atau tidak tetap dipegang oleh server). Terdapat tiga buah sesi, yaitu penerima (*Receiver*) – bila klien ingin dapat menerima data, Pengirim (*Transmitter*) – bila klien ingin dapat mengirimkan paket data, atau *Tranceiver* – bila klien ingin dapat mengirim dan menerima paket data.
- Bila server ingin mengirimkan paket data kepada klien harus menggunakan format PDU *deliver\_sm*, sedangkan bila klien ingin mengirimkan paket data kepada server maka harus menggunakan format PDU *submit\_sm*. Format PDU yang dapat digunakan bersama untuk saling bertukar data yang berisi pesan pendek adalah *data\_sm*.
- Sebuah proses transaksi pengiriman paket data (apa pun jenis format PDU-nya) terdiri dari dua tahap, yaitu pengiriman paket data utama dan respons pada arah sebaliknya. Sebagai contoh, bila server mengirimkan data menggunakan format PDU *deliver\_sm* maka klien harus membalasnya dengan menggunakan format PDU *deliver\_sm\_resp*. Sebaliknya, jika klien mengeluarkan *submit\_sm* maka server harus membalasnya dengan *submit\_sm\_resp*.

Pada umumnya sebuah *Message Center* akan bertindak sebagai SMPP server, sedangkan ESME akan bertindak sebagai SMPP klien. *Message Center* merupakan sebuah entitas yang bersifat tetap, baik secara fungsi maupun secara fisik, sehingga lebih cocok untuk menjadi server SMPP. Server cenderung bersifat pasif dan menunggu klien untuk melakukan koneksi. ESME merupakan sebuah entitas yang berfungsi pada level aplikasi dan tidak berkontribusi langsung pada sebuah system layanan SMS. ESME dapat dianggap sebagai *end user* dalam konfigurasi layanan SMS, sehingga keberadaannya bersifat tidak tetap. Jika ESME ada, maka layanan konten SMS dapat diadakan. Akan tetapi jika ESME tidak ada, layanan SMS tetap dapat berfungsi sekalipun tidak ada entitas penyedia konten SMS.

Operasi protokol SMPP, berdasarkan karakteristik format PDU-nya, dikategorikan dalam beberapa kelompok, yaitu :

- *Session Management*  
Meliputi operasi-operasi dalam protokol SMPP yang berfungsi dalam pembentukan session antara ESME dan *Message Center*, sekaligus menangani bentuk-bentuk error yang mungkin terjadi pada proses pembentukan session tersebut.
- *Message Submission*  
Meliputi operasi-operasi dalam protokol SMPP yang khusus ditujukan bagi pengiriman pesan pendek dari ESME ke *Message Center* (dalam hal ini *Message Center* berlaku sebagai SMPP Server).
- *Message Delivery*  
Meliputi operasi-operasi dalam protokol SMPP yang khusus diperuntukkan bagi pengiriman pesan pendek dari *Message Center* ke ESME (dalam hal ini ESME berlaku sebagai SMPP Klien).
- *Message Broadcast*  
Meliputi operasi-operasi dalam protokol SMPP yang dikhususkan bagi keperluan pengiriman pesan pendek secara broadcast dalam suatu cakupan *Message Center*.
- *Ancillary Operation*  
Meliputi operasi-operasi dalam protokol SMPP yang berfungsi menyediakan fungsi-fungsi tambahan seperti pembatalan pengiriman pesan pendek, pemeriksaan kiriman, dan penggantian pesan yang telah dikirimkan.

### **2.1.1 Format PDU**

*Protocol Data Unit* digunakan dalam pengiriman dan penerimaan SMS. Dalam pengiriman dan penerimaan pesan SMS terdapat dua mode, yaitu mode teks dan mode PDU. Mode teks adalah format pesan dalam bentuk teks asli yang dituliskan pada saat akan mengirim pesan. Sesungguhnya mode teks ini adalah hasil pengkodean dari mode PDU. Sedangkan mode PDU adalah format pesan dalam bentuk oktet heksadesimal dan oktet semidesimal dengan panjang mencapai 160 (7 bit) atau 140 (8 bit)

karakter. Di Indonesia tidak semua operator GSM maupun terminal mendukung mode teks, sehingga mode yang digunakan adalah mode PDU [6]. Pada pengiriman pesan terdapat dua jenis *mobile*, yaitu *mobile terminated* (telepon seluler penerima) dan *mobile originated* (telepon seluler pengirim).

### SMS PDU Pengirim (*Mobile Originated*)

SMS PDU Pengirim adalah pesan yang dikirimkan dari telepon seluler ke terminal yang kemudian dikirimkan ke SMSC. Pada prinsipnya, apabila kita mengirimkan pesan ke nomor tujuan, pesan itu akan melalui SMSC. Pesan yang dikirimkan oleh terminal masih dalam bentuk teks, sedangkan dalam pengiriman ke SMSC harus dalam bentuk PDU. Untuk itu, sebelum dikirim, SMS Server akan melakukan perubahan dari format teks menjadi format PDU. Gambar 2.1 merupakan skema dari format PDU pengirim yang telah diatur dan ditetapkan oleh ETSI (*European Telecommunication Standard Institute*).

SCA	PDU Type	MR	DA	PID	DCS	VP	UDL	UD
-----	----------	----	----	-----	-----	----	-----	----

Gambar 2.1 Skema Format SMS PDU Pengirim

Berikut adalah penjelasan dari skema format SMS PDU Pengirim :

- **Service Center Address (SCA)**

SCA adalah informasi dari alamat / nomor SMSC. SCA memiliki tiga komponen utama, yaitu *len*, tipe nomor, dan nomor pusat layanan (*service center*). Dalam pengiriman pesan SMS, nomor SMSC tidak dicantumkan.

Tabel 2.1 *Service Center Address (SCA)*

Oktet	Keterangan
Len	Panjang Informasi SMSC dalam oktet
Tipe Nomor	Format Nomor SMSC 81 hexa = format lokal 91 hexa = format internasional
Nomor Pusat Layanan	Nomor SMSC dari operator pengirim

- **PDU Type**

Nilai *default* dari PDU Type untuk SMS pengirim adalah 11 hexa, yang memiliki arti bahwa 11 hexa = 00010001. Lihat Gambar 2.2.

Bit No.	7	6	5	4	3	2	1	0
Nama	RP	UDHI	SRR	VPF	VPF	RD	MTI	MTI
Nilai	0	0	0	1	0	0	0	1

Gambar 2.2 PDU Type Pengirim

Keterangan :

RP : *Reply Path*. Parameter yang menunjukkan bahwa alur jawaban ada.

UDHI : *User Data Header Indicator*. Bit ini bernilai 1 jika data pengirim dimulai dengan suatu judul/tema.

SRR : *Status Report Request*. Bit ini bernilai 1 jika laporan status pengiriman diminta.

VPF : *Validity Period Format*. Format dari batas waktu pengiriman jika pesan gagal diterima.

00 → Jika pesan tidak disimpan di SMSC

10 → Format relatif (satu oktet)

01 → Format enhanced (tujuh oktet)

11 → Format absolut (tujuh oktet)

RD : *Reject Duplicates*. Parameter yang menandakan ya atau tidaknya pusat layanan akan menerima suatu pengiriman pesan SMS untuk suatu pesan yang masih disimpan dalam pusat layanan tersebut. Ia memiliki MR dan DA yang sama sebagai pesan dikirimkan dari OA yang sama.

MTI : *Message Type Indicator*. Bit bernilai 0 untuk menunjukkan bahwa PDU ini adalah suatu SMS-Deliver.

- **Message Reference (MR)**

*Message Reference* adalah acuan dari pengaturan pesan SMS. Untuk membiarkan pengaturan pesan SMS dilakukan sendiri oleh telepon seluler tujuan, maka nilai yang diberikan adalah "00".

- **Destination Address (DA)**

DA adalah alamat/nomor tujuan, yang terdiri atas panjangnya nomor tujuan (*len*), format dari nomor tujuan (tipe nomor), dan nomor tujuan, hal ini dapat dilihat pada Tabel 2.2.

Tabel 2.2 *Destination Address*

Oktet	Keterangan	Contoh
Len	Panjang Informasi SMSC dalam oktet	0C
Tipe Nomor	Format Nomor SMSC 81 hexa = format lokal 91 hexa = format internasional	91
Nomor Pusat Layanan	Nomor SMSC dari operator pengirim	628168919080 → 261868919080

- **Protocol Identifier (PID)**

PID adalah tipe atau format dari cara pengiriman pesan, yang biasanya diatur dari telepon seluler pengirim. Nilai *default* PID adalah 00 = "Standard Text".

- **Data Coding Scheme (DCS)**

DCS adalah rencana dari pengkodean data untuk menentukan kelas dari pesan tersebut apakah berupa SMS teks standar (00), *Flash SMS* (01), atau *blinking SMS* (10).

- **Validity Period (VP)**

VP adalah lama waktu pesan SMS disimpan di SMSC apabila pesan tersebut gagal diterima oleh telepon seluler penerima. Lihat Tabel 2.3 mengenai perhitungan *Validity Period*.

Tabel 2.3 *Validity Period*

Waktu VP	Nilai VP
5 menit - 720 menit (12 jam)	$(\text{Waktu VP} / 5) - 1$
12,5 jam - 24 jam	$143 + ((\text{Waktu VP} - 12) * 2)$
2 - 30 hari	$166 + \text{Waktu VP}$
Lebih dari 4 minggu	$192 + \text{Waktu VP}$

- **User Data Length (UDL)**  
UDL adalah panjangnya pesan SMS yang akan dikirimkan dalam bentuk teks standar.
- **User Data (UD)**  
UD adalah isi pesan yang akan dikirimkan dalam format heksadesimal. Pengkodean dari nilai teks standar menjadi heksadesimal dilakukan dengan bantuan *Default Alphabet* yang telah dibakukan oleh ETSI GSM.

### SMS PDU Penerima (*Mobile Terminated*)

SMS PDU Penerima adalah terminal menerima pesan yang datang atau masuk dari SMSC ke telepon seluler dalam format PDU. Pada prinsipnya pesan yang kita terima dari SMSC masih dalam format PDU setelah itu SMS Server yang menerima pesan akan melakukan pengkodean menjadi teks. Cara pengkodean format PDU sudah diatur dan distandarkan oleh ETSI. Gambar 2.3 merupakan skema format SMS PDU penerima.

SCA	PDU Type	OA	PID	DCS	SCTS	UDL	UD
-----	----------	----	-----	-----	------	-----	----

Gambar 2.3 Skema Format SMS PDU Penerima

Berikut adalah penjelasan dari skema format SMS PDU Pengirim :

- **Service Center Address (SCA)**  
Sama seperti pada PDU pengirim, SCA adalah alamat/nomor dari SMSC.
- **PDU Type**  
Nilai *default* dari PDU Type untuk SMS-Deliver adalah 04 Hexa, yang memiliki arti 04 hexa = 00000100 seperti yang terlihat pada Gambar 2.4.

Bit No.	7	6	5	4	3	2	1	0
Nama	RP	UDHI	SRI	<nn>	<nn>	MMS	MTI	MTI
Nilai	0	0	0	0	0	1	0	0

Gambar 2.4 PDU Type Penerima

Keterangan :

RP : *Reply Path*. Parameter yang menunjukkan bahwa alur jawaban ada.

UDHI : *User Data Header Indicator*. Bit ini bernilai 1 jika data pengirim dimulai dengan suatu judul/tema.

SRI : *Status Report Indicator*. Bit ini bernilai 1 jika suatu laporan akan dikembalikan ke SME.

MMS : *More Message to Send*. Bit ini bernilai 0 jika ada pesan lebih yang akan dikirim.

MTI : *Message Type Indicator*. Bit bernilai 0 untuk menunjukkan bahwa PDU ini adalah suatu SMS-Deliver.

▪ **Originator Address (OA)**

OA adalah alamat/nomor dari pengirim, yang terdiri atas panjangnya nomor pengirim (*len*), format dari nomor pengirim, dan nomor pengirim.

▪ **Protocol Identifier (PID)**

PID adalah tipe atau format dari cara pengiriman pesan, yang biasanya diatur oleh telepon seluler pengirim.

▪ **Data Coding Scheme (DCS)**

DCS adalah rencana dari pengkodean data untuk menentukan kelas dari pesan tersebut apakah berupa SMS teks standar (00), *Flash SMS* (01), atau *blinking SMS* (10).

▪ **Service Center Time Stamp (SCTS)**

SCTS adalah waktu dari penerimaan pesan oleh SMSC penerima. SCTS terdiri atas tahun, bulan, tanggal, jam, menit, detik, dan zona waktu (lihat Tabel 2.4).

Tabel 2.4 *Service Center Time Stamp*

Nama	Nilai	Hasil
Tahun	80	08 (2008)
Bulan	10	01 (Januari)
Tanggal	60	06
Jam	61	16
Menit	22	22
Detik	02	20
Zona Waktu	82	28, dimana 1 unit = 15 menit, jadi $(15 \times 28) / 60 = 7$ jam. Sehingga menjadi GMT + 07.00 = WIB

Dari kolom 'Hasil' pada Tabel 2.4, maka dapat terlihat bahwa pesan diterima oleh SMSC pada tanggal 16 Januari 2008 pukul 16:22':20" WIB.

- ***User Data Length (UDL)***  
UDL adalah panjang dari pesan yang diterima dalam bentuk teks standar.
- ***User Data (UD)***  
UD adalah pesan yang diterima dalam format heksadesimal. Pengkodean dari nilai heksa decimal menjadi teks standar dilakukan dengan bantuan tabel kode ASCII.

### 2.1.2 SMPP Session

Dalam teknologi komunikasi data, SMPP merupakan protokol yang berjalan dalam lapisan aplikasi pada model interkoneksi OSI (*Open System Interconnection*). Pada implementasinya, *user data* yang dipertukarkan biasanya menggunakan buffer sebagai antarmukanya. Melalui buffer inilah aplikasi melakukan pengiriman dan penerimaan data. Pada aplikasi berbasis protokol SMPP, masing-masing titik yang saling berkomunikasi harus menganggap bahwa data-data yang dipertukarkan adalah sebuah PDU dengan format yang saling disepakati. Masing-masing titik akan dapat saling berkirim PDU bila sebuah session yang disepakati telah terbentuk [2].

Sebelum sebuah session dapat terbentuk, pertama kali yang harus dilakukan adalah membentuk koneksi antara titik yang ingin berkomunikasi, misalnya antara *Message Center* dan ESME. Pembentukan koneksi dapat dilakukan melalui koneksi protokol komunikasi data seperti TCP/IP atau X.25. Untuk koneksi menggunakan protokol TCP/IP, port yang digunakan untuk aplikasi SMPP adalah port 2775.

Komponen operasi protokol SMPP adalah sebagai berikut :

- ***Status Session***  
Seperti telah disebutkan sebelumnya, session hanya mungkin terbentuk setelah terbentuk koneksi.
- ***Open***  
Status ini terjadi apabila telah terjadi koneksi antar ESME dan MC, namun *bind request* belum dikirimkan oleh ESME. Pada kondisi ini belum terjadi pertukaran informasi.
- ***Bound\_TX***

Status ini terjadi apabila antara ESME dan MC telah terbentuk koneksi dan PDU *bind\_transmitter* telah dikirimkan oleh ESME yang kemudian dibalas oleh MC dengan mengirimkan *bind\_transmitter\_resp* dengan status OK yang menyatakan session telah disetujui untuk dibuka. Status ini memungkinkan ESME untuk mengirimkan pesan pendek pada MC atau ESME lain. Operasi lain yang dapat dilakukan oleh ESME adalah mengirimkan PDU *replace\_sm*, *query\_sm*, atau *cancel\_sm* terhadap pesan pendek yang telah dikirimkan sebelumnya.

- *Bound\_RX*

Status ini terjadi apabila antara ESME dan MC telah terbentuk koneksi dan PDU *bind\_receiver* telah dikirimkan oleh ESME yang kemudian dibalas oleh MC dengan mengirimkan *bind\_receiver\_resp* dengan status OK yang menyatakan bahwa session telah disetujui untuk dibuka. Status ini memungkinkan ESME untuk menerima pesan pendek dari MC atau ESME lain.

- *Bound\_TRX*

Status ini terjadi apabila antara ESME dan MC telah terbentuk koneksi dan PDU *bind\_transceiver* telah dikirimkan oleh ESME yang kemudian dibalas oleh MC dengan mengirimkan *bind\_transceiver\_resp* dengan status OK yang menyatakan session telah disetujui untuk dibuka. Status ini memungkinkan ESME untuk mengirimkan/menerima pesan pendek ke/dari MC atau ESME lain. Session ini merupakan gabungan atau kombinasi antara *bound\_TX* dan *bound\_RX*.

- *Unbound*

Status ini terjadi apabila pada saat satu atau lebih sesi *bind* sedang terbentuk, ESME atau MC kemudian mengirimkan PDU *unbind* yang dibalas dengan *unbind\_resp* dengan status OK. PDU *unbind* merupakan sebuah PDU berisi request untuk menutup session SMPP.

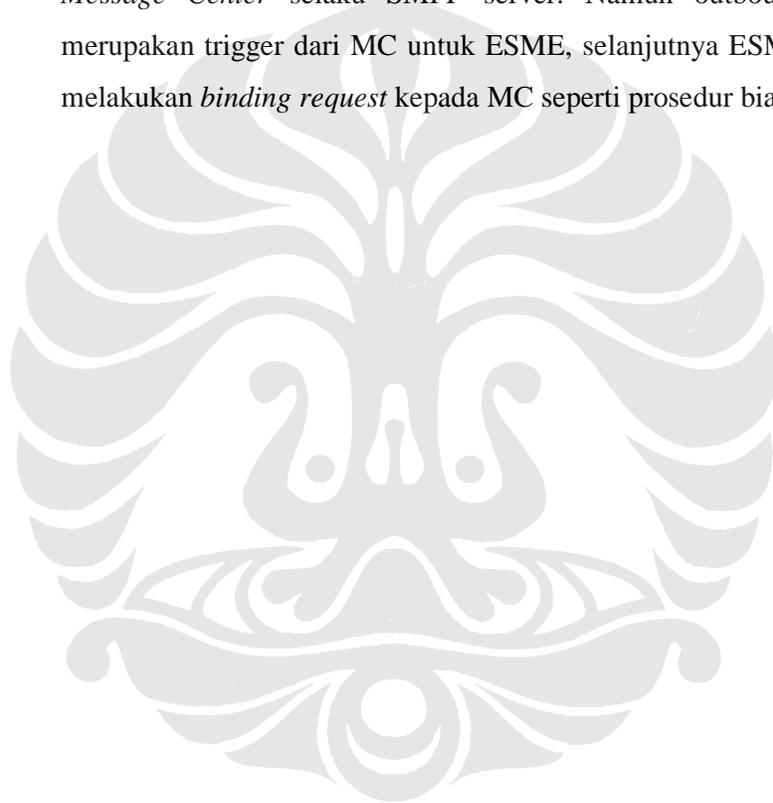
- *Closed*

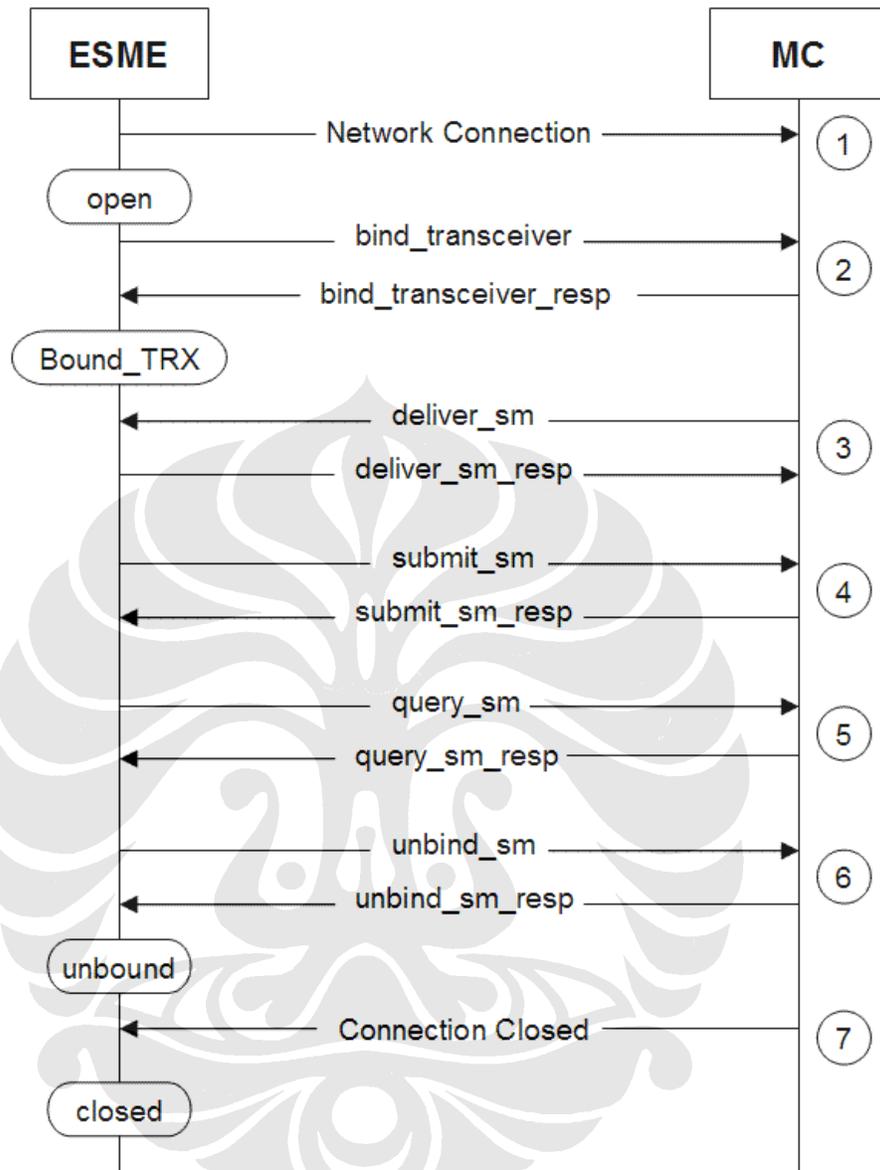
Status ini terjadi apabila koneksi antara ESME dan MC telah diputuskan. Biasanya terjadi secara berurutan setelah status *unbound* tercapai akibat salah satu titik mengirim request untuk menutup session yang ada. Status

*closed* juga dapat terjadi apabila dideteksi terjadi error pada jaringan, baik yang diketahui maupun yang tidak yang menyebabkan terputusnya proses komunikasi.

- *Outbound*

Pada operasi normal, inisiatif pembentukan koneksi maupun session dilakukan oleh titik yang bertindak sebagai SMPP cliend (dalam hal ini adalah ESME). Status outbound merupakan kasus khusus dimana inisiatif pembentukan koneksi dan pembangunan sebuah session dilakukan oleh *Message Center* selaku SMPP server. Namun outbound ini hanya merupakan trigger dari MC untuk ESME, selanjutnya ESME tetap harus melakukan *binding request* kepada MC seperti prosedur biasa.





Gambar 2.5. Contoh operasi protokol SMPP

Gambar 2.5 merupakan salah satu contoh operasi protokol SMPP. Operasi pada Gambar 2.5 tersebut merupakan operasi standar yang biasa dilakukan untuk pengiriman SMS biasa, dan dapat berubah sesuai dengan kondisi dan proses yang terjadi pada suatu sistem.

Berikut adalah penjelasan untuk setiap operasi yang dilakukan :

**Langkah 1 : Pembentukan koneksi**

MC (dalam hal ini PC) selaku SMPP server memiliki sebuah port TCP yang berstatus *listening* pada nomor port tertentu. Dengan posisi ini, PC siap untuk menerima permintaan koneksi dari *host* lain. Permintaan koneksi dilakukan oleh ESME dengan prosedur *3-way handshaking* yang ada dalam TCP. Dalam TCP/IP, proses-proses ini sudah diakomodasi oleh sistem secara otomatis. Setelah tahap pembentukan koneksi berhasil, emaka titik yang berhubungan sudah dapat untuk saling berkirim data aplikasi yang berbentuk PDU.

### **Langkah 2 : Pembentukan sesi**

Pada langkah ini, ESME mengirimkan sebuah PDU ke MC dengan jenis *bind\_transceiver*. Kemudian PDU yang diterima oleh MC akan di ekstrak. Sebagai respon, MC akan mengirimkan PDU respons yaitu *bind\_transceiver\_resp*. Respon ini harus dikirimkan agar status dapat diketahui, apakah OK atau tidak OK. Setelah proses ini selesai, maka ESME dan MC akan membentuk sebuah sesi dengan status *Bound\_TR*. Selanjutnya masing-masing dapat saling berkirim pesan pendek.

### **Langkah 3 : Pengiriman pesan pendek dengan *deliver\_sm***

PDU *deliver\_sm* merupakan PDU yang dikhususkan untuk pengiriman pesan pendek dari SMPP server ke SMPP klien, dalam hal ini dari MC ke ESME. Respon yang dikirimkan kembali adalah *deliver\_sm\_resp*, yang menyatakan satu proses pengiriman pesan pendek dari MC ke ESME selesai.

### **Langkah 4 : Pengiriman pesan pendek dengan *submit\_sm***

PDU *submit\_sm* merupakan PDU yang dikhususkan untuk pengiriman pesan pendek dari SMPP klien ke SMPP server, dalam hal ini dari ESME ke MC. Respon yang dikirimkan kembali adalah *submit\_sm\_resp*, yang menyatakan satu proses pengiriman pesan pendek dari ESME ke MC selesai.

### **Langkah 5 : *Query* pesan pendek**

*Query* pesan pendek merupakan fasilitas untuk mengecek status pesan pendek sebelumnya apakah proses permintaan *query* sudah dapat dinyatakan selesai atau belum.

### **Langkah 6 : Penutupan sesi**

Tahap ini merupakan penutupan sesi yang terbentuk sebelumnya. ESME akan mengirimkan PDU *unbind* dan direspon dengan PDU *unbind\_resp* oleh MC. Maka baik ESME maupun MC akan menutup sesi.

### **Langkah 7 : Pembubaran koneksi**

Pembubaran koneksi merupakan sebuah proses yang menyatu dengan pembubaran sesi. Pada saat sesi di tutup, maka koneksi juga akan dibubarkan. Pada status ini tidak ada lagi aktivitas, dan seluruh proses dinyatakan selesai. Hingga terdapat SMS permintaan lain dari pengguna.

## **2.2 Pemrograman Java**

Java merupakan bahasa pemrograman berorientasi objek yang dikembangkan oleh Sun Microsystem 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. Bahasa ini juga dirancang untuk pemrograman di internet sehingga dirancang agar aman dan portabel [1].

Java juga merupakan salah satu bahasa pemrograman baru yang menjanjikan banyak kemudahan bagi *programmer* awam maupun yang sudah senior.

Pemrograman Java memiliki *platform independent*, yang berarti program yang ditulis dalam bahasa Java dapat dengan mudah dipindahkan antar berbagai jenis sistem operasi dan berbagai jenis arsitektur komputer. Aspek ini sangat penting untuk dapat mencapai tujuan Java sebagai bahasa pemrograman internet dimana sebuah program ini akan dijalankan oleh berbagai jenis komputer dengan berbagai jenis sistem operasi. Berbeda dengan bahasa C dan C++, semua tipe data dalam bahasa Java mempunyai ukuran yang konsisten di semua jenis platform [5].

Dalam Java ditemukan adanya istilah JVM (*Java Virtual Machine*), yang merupakan sebuah aplikasi yang berjalan di atas sebuah sistem operasi dan menerjemahkan *bytecode* program Java dan mengeksekusinya, sehingga secara

konsep bisa dianggap sebagai sebuah interpreter. Proses pengekseskuan program java dapat dilukiskan seperti pada Gambar 2.6. Dengan cara ini, sebuah program Java yang telah dikompilasi akan dapat berjalan di platform mana saja, asalkan komputer tersebut telah terinstal JVM.



Gambar 2.6. Proses Pengekseskuan Program Java

Dua macam pemrograman Java yang dikenal adalah *Java Runtime Environment* (JRE), yaitu interpreter Java tanpa kompilernya, dan *Java Development Kit* (JDK), yang digunakan untuk mengembangkan program Java. Jika pengguna hanya ingin menjalankan *bytecode* Java, cukup dengan JRE saja.

Selain compiler dan interpreter, bahasa Java sendiri memiliki *library* yang cukup besar yang dapat mempermudah pengguna dalam membuat sebuah aplikasi dengan cepat. *Library* ini sudah mencakup grafik, desain *user interface*, kriptografi, jaringan, suara, *database*, dan lain-lain.

Java merupakan bahasa pemrograman berorientasi objek, yang berarti bahasa ini memiliki teknik untuk mengorganisir program dan dapat dilakukan dengan hampir semua bahasa pemrograman. Namun Java sendiri telah mengoptimalkan teknik pemrograman berorientasi objek.

Sedikit perbandingan lain dengan bahasa C dan C++, adalah Java banyak mewarisi konsep orientasi objek dari C++ namun dengan menghilangkan aspek-aspek kerumitan dalam bahasa C++ tanpa mengurangi kekuatannya. Hal ini mempermudah programmer pemula untuk mempelajari Java. Namun di balik kemudahan yang ditawarkan Java, luasnya fasilitas *library* Java itu sendiri membuat seorang programmer membutuhkan waktu yang tidak singkat untuk dapat menguasai penggunaan *library-library* tersebut.

## 2.3 Konsep Aplikasi Multi Tier

Aplikasi internet berbasis web merupakan salah satu penerapan aplikasi multi tier. Aplikasi multi tier adalah aplikasi yang dibagi menjadi beberapa bagian yang menjalankan fungsi masing-masing [1]. Secara umum, ada tiga bagian utama dari aplikasi multi tier, yaitu :

- *Client side presentation*

*Client side presentation* mengatur bagaimana aplikasi berinteraksi dengan *user*. Yang dimaksud dengan interaksi antara lain adalah bagaimana data ditampilkan, bagaimana fungsi dan fitur aplikasi ditampilkan.

Dalam aplikasi berbasis web, *client side presentation* dibuat dengan bahasa HTML, CSS, dan JavaScript. Beberapa tool yang digunakan untuk membuat klien side presentation diantaranya Microsoft Frontpage, Macromedia Dreamweaver, dan sebagainya.

Contoh dari *client side presentation* berbasis web adalah tampilan aplikasi email yang kita buka dengan browser.

- *Server side business logic*

Server side business logic, sering disebut juga *middle tier*, adalah bagian yang bertanggung jawab atas cara kerja aplikasi. Di dalamnya *user* mengatur bagaimana fungsi dan fitur aplikasi dapat bekerja dengan baik. Dalam aplikasi berbasis web, ada beberapa alternatif yang dapat digunakan, tergantung dari jenis *platform* yang digunakan.

- *Backend storage*

Bagian ini mengatur cara penyimpanan data. Penyimpanan data merupakan materi yang cukup kompleks dalam pembangunan aplikasi, karena kecepatan, keutuhan, dan keamanan data merupakan faktor kritis dalam aplikasi.

Ada banyak solusi database yang tersedia di pasaran. Pada umumnya database yang digunakan bertipe relasional (*Relational Database Management System – RDBMS*). Manajemen data dilakukan dengan bahasa SQL (*Structured Query Language*).

### 2.3.1 Perbedaan Web Based Programming dengan System Programming

Pembuatan aplikasi berbasis web berbeda dengan pembuatan aplikasi berbasis windows (visual programming), misalnya Visual Basic, Delphi, atau KDevelop. Dalam visual programming, kecepatan dan kinerja aplikasi ditingkatkan dengan mengoptimasi penggunaan memori, manajemen proses, dan pengaturan *Input-Output*. Pada pemrograman berbasis web, faktor yang menentukan kinerja aplikasi adalah kecepatan akses database dan kecepatan akses jaringan dan internet.

Perbedaan kedua, adalah cara aplikasi berjalan. Pada aplikasi visual, aplikasi dibangun dengan menggunakan tool tertentu, kemudian dikompilasi. Hasilnya dapat langsung digunakan dalam komputer. Aplikasi berbasis web tidak dapat dijalankan langsung di komputer. Untuk menjalankannya, dibutuhkan engine tertentu, dalam hal ini web server.

### 2.3.2 Teknologi Alternatif

Teknologi server side yang digunakan dalam simulasi ini adalah PHP. Walaupun demikian, terdapat beberapa teknologi alternatif yang dapat menjadi pertimbangan, yaitu :

- **CGI Script**

CGI *Script* dapat dibuat dengan berbagai bahasa pemrograman, misalnya *Perl* atau *Phyton*. Teknologi ini pernah sangat populer di masa awal berkembangnya aplikasi berbasis web. Tetapi saat ini banyak ditinggalkan orang karena tidak efisien, tidak fleksibel, dan keamanannya rendah.

- **Proprietary API**

Teknologi ini adalah bahasa pemrograman yang disediakan masing-masing webserver, misalnya ISAPI atau NSAPI. Teknologi ini memiliki kelemahan, yaitu hanya dapat berjalan di webserver tertentu, sehingga mengurangi portabilitas.

- **ASP**

Merupakan solusi server side programming dari Microsoft. Teknologi ini banyak digunakan oleh para programmer yang berlatang belakang Visual Basic. Database pasangannya adalah MS SQL Server. Operating system

pasangannya adalah Window 2000 Server yang menjalankan webserver Microsoft IIS.

- **JavaServlet / JSP**

Teknologi server side Java, seperti telah dibahas sebelumnya, memiliki banyak keunggulan dan kemudahan pemrograman. Tetapi karena murni berorientasi objek, banyak programmer pemula kesulitan menggunakannya. Selain itu, java juga adalah bahasa pemrograman yang relatif rumit bagi pemula. Keunggulannya adalah sangat portabel. Dapat dipindahkan dengan mudah dari Windows ke UNIX, dan sebaliknya.

- **Coldfusion**

Bahasa pemrograman ini mirip dengan HTML, menggunakan tag untuk membentuk blok-blok programnya.

- **PHP**

Banyak digunakan oleh programmer berlatar belakang C/C++ karena kemiripan syntaxnya. Open Source, karenanya gratis dan bebas. Database pasangannya biasanya adalah MySQL, dijalankan bersama webserver Apache di atas operating system Linux. Semuanya gratis dan bebas.

## 2.4 SQL Query

SQL (*Structured Query Language*) adalah bahasa yang khusus digunakan untuk mengoperasikan database [1]. Pada dasarnya perintah-perintah SQL terbagi menjadi dua kelompok, yaitu :

1. DDL (*Data Definition Language*)

Bahasa yang digunakan untuk mendefinisikan data. Pernyataan-pernyataan di sini berkaitan dengan pembuatan tabel, penghapusan tabel, dan lain-lain. Misalnya *create table*, *alter table*, *drop table*, dan sebagainya.

2. DML (*Data Manipulation Language*)

Bahasa yang digunakan untuk memanipulasi/memodifikasi data. Pernyataan-pernyataan di sini berkaitan dengan penambahan data, penghapusan data, menampilkan data, dan lain-lain. Misalnya *select*, *insert*, *update*, dan sebagainya.

Untuk memudahkan, SQL *query* akan dikelompokkan menjadi tiga:

- Query untuk mengelola database
- Query untuk mengakses data dalam satu tabel
- Query yang melibatkan lebih dari satu tabel

#### 2.4.1 Query Pengelolaan Database

Yang termasuk ke dalam kelompok ini adalah query yang bertujuan untuk :

- Membuat database
- Menghapus database
- Membuat tabel
- Memodifikasi tabel
- Menghapus tabel
- Menambah *user*
- Mengatur permission
- Menghapus *user*

Membuat database dilakukan dengan perintah sebagai berikut :

```
CREATE DATABASE <nama database>
```

Contohnya : Untuk aplikasi *Content Management*, kita akan membuat database *ContentManager* dengan query sebagai berikut :

```
CREATE DATABASE ContactManager
```

Menghapus database dilakukan dengan perintah :

```
DROP DATABASE <nama database>
```

Contoh :

```
DROP DATABASE ContactManager
```

Membuat tabel dilakukan dengan memberikan perintah sebagai berikut:

```
CREATE TABLE <nama tabel> (
<nama kolom> <tipe data>,
<nama kolom> <tipe data>,
... ,
<nama kolom> <tipe data>
);
```

contohnya:

```
CREATE TABLE userTable(
  UserId INT (3),
  UserName VARCHAR (50),
  Password VARCHAR (50),
  NamaLengkap VARCHAR (50)
);
```

Menghapus tabel dilakukan dengan menggunakan query

```
DROP TABLE <nama tabel>;
```

Contoh :

```
DROP TABLE userTable;
```

## 2.4.2 Query Satu Tabel

Query satu tabel digunakan untuk mengelola data dalam satu tabel. Beberapa hal yang dapat dilakukan pada satu tabel dapat dilihat pada Tabel 2.5 berikut :

Tabel 2.5. Hal yang dapat dilakukan pada *query* satu tabel

Tujuan	Query
Memasukkan data	INSERT
Memodifikasi data	UPDATE
Mengambil data	SELECT
Menghapus data	DELETE
Menghitung banyaknya data	COUNT
Menghitung penjumlahan data	SUM
Menghitung nilai minimal	MIN
Menghitung nilai maksimal	MAX
Menghitung nilai rata-rata	AVG

Untuk memasukkan data ke dalam database, kita menggunakan perintah INSERT. Aturan penulisannya adalah :

```
INSERT INTO <nama tabel> (<nama kolom 1>, < nama
kolom
1>, ..)
VALUES (<isi kolom 1>, <isi kolom 2>, ..);
```

```
INSERT INTO userTable VALUES (
1,
'lala',
'inipassword',
'Lala Darmadji'
);
```

Untuk mengubah data dalam database, digunakan perintah UPDATE, dengan aturan penulisan sebagai berikut :

```
UPDATE <nama tabel>
SET <nama kolom>=<isi kolom>
WHERE <kriteria>
```

Contoh :

```
UPDATE userTable
SET password='pas5word'
WHERE UserName='lala'
```

Untuk menampilkan data dalam tabel, gunakan perintah SELECT. Aturan penulisannya adalah sebagai berikut:

```
SELECT <nama kolom>
FROM <nama tabel>
WHERE <kriteria>
```

Contoh :

```
SELECT *
FROM UserTable
WHERE UserId=1
```

Untuk menghapus data dari dalam tabel, gunakan perintah DELETE. Aturan penulisannya adalah :

```
DELETE FROM <nama tabel> WHERE <kriteria>
```

Contoh :

```
DELETE FROM UserTable
WHERE NamaLengkap='Lala Darmadji'
```

## **BAB III**

### **PERANCANGAN SISTEM**

Dalam bab ini akan dijelaskan mengenai rancangan sistem aplikasi yang dikembangkan dalam penelitian ini. Sistem aplikasi informasi penjadwalan perkuliahan elektronik ini dipasang pada server, dan informasi dapat diakses melalui terminal *users* atau pengguna berupa PDA, browser intranet di PC, dan telepon seluler yang dapat menerima sms reminder oleh sistem. Aplikasi yang dipasang di server terdiri atas komponen sebagai berikut :

1. Server aplikasi dan modem sebagai sms gateway
2. Database server

Sedangkan yang dimaksud terminal user/pengguna adalah sebagai berikut :

1. PDA (*Personal Digital Assitant*), digunakan untuk mengakses database server, terhubung melalui intranet dengan menggunakan fasilitas WiFi.
2. Aplikasi User (desktop), digunakan untuk mengakses database server, terhubung melalui intranet, dan dibuka di desktop dengan menggunakan browser internet.
3. Telepon seluler (*handphone*), digunakan untuk menerima sms reminder/update jadwal yang dikirim oleh server aplikasi.

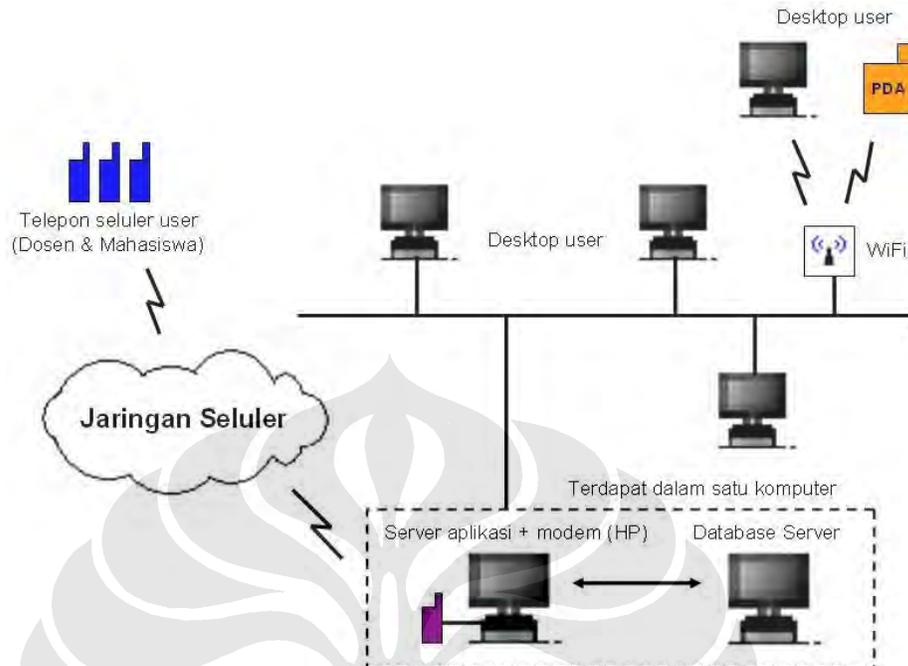
Skema hubungan kerja dalam sistem aplikasi secara keseluruhan ditunjukkan pada Gambar 3.1. Skema Sistem Keseluruhan dari Aplikasi Sistem Informasi Penjadwalan Perkuliahan Elektronik.

#### **3.1 Sistem Aplikasi Informasi Penjadwalan Elektronik Pada Server**

Seperti telah dijelaskan sebelumnya, sistem aplikasi pada server ini terdiri dari dua bagian, yaitu server aplikasi dan database server. Masing-masing fungsi bagian sistem tersebut akan dibahas dalam sub-sub bab berikut.

##### **3.1.1 Server Aplikasi (dan Modem SMS Gateway)**

Server aplikasi merupakan komponen utama dalam sistem informasi penjadwalan perkuliahan elektronik ini. Program utama terdapat pada server ini dan terhubung dengan sistem database.



Gambar 3.1. Skema sistem keseluruhan dari aplikasi sistem penjadwalan perkuliahan elektronik

Server aplikasi adalah bagian utama dari sistem informasi penjadwalan perkuliahan elektronik. Server tersebut mengambil data dari database server, seperti terlihat pada Gambar 3.1. Dalam penelitian ini, kedua fungsi server (aplikasi dan database) terletak dalam satu komputer. Sinkronisasi data akan diatur oleh database server, dan data yang valid adalah data yang terakhir kali dimasukkan.

Dalam menjalankan tugasnya, server aplikasi terhubung dengan modem telepon seluler yang berfungsi sebagai SMS gateway. Aplikasi ini dapat mengirimkan sms *reminder* kepada pengguna sesuai dengan keinginan melalui SMS Gateway tersebut. Misalnya *admin* ingin agar sms reminder dikirimkan 30 menit sebelum perkuliahan dimulai, maka sms reminder akan dikirim pada mahasiswa atau dosen 30 menit sebelum perkuliahan dimulai.

Selain mengirim sms reminder, server aplikasi juga berfungsi untuk mengirimkan sms update jika terdapat update jadwal pada databasenya,

misalnya perubahan jadwal perkuliahan yang telah diupdate oleh *admin* atau dosen perkuliahan tersebut sendiri via intranet.

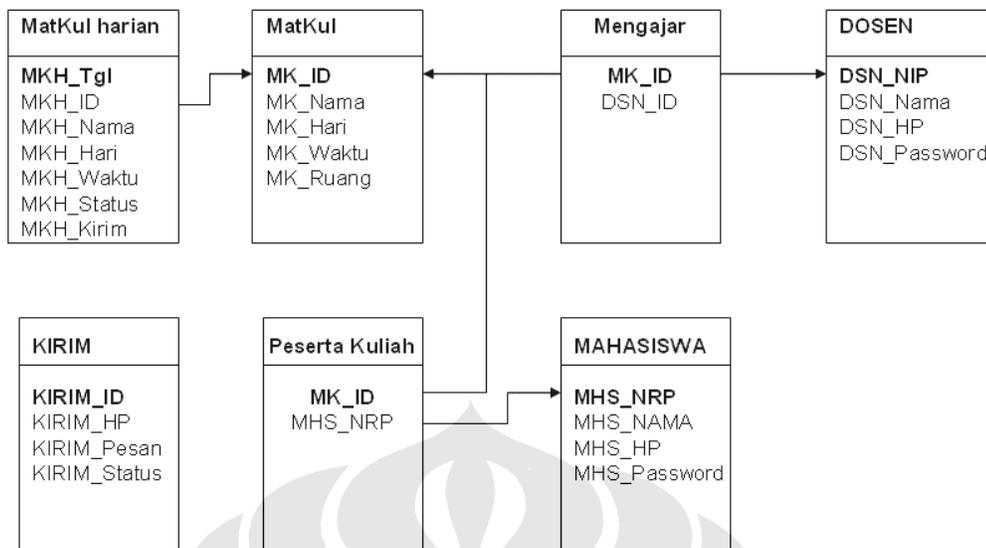
Modem yang digunakan sebagai SMS Gateway adalah sebuah telepon seluler. Pada penelitian ini, penulis menggunakan telepon seluler merk *Siemens* tipe C55 sebagai modem SMS gateway.

### 3.1.2 Sistem Database

Sistem database merupakan pusat berkumpulnya seluruh data yang dimasukkan ke dalam sistem yang merupakan informasi bagi pengguna. Update oleh pengguna (dalam hal ini adalah dosen) dapat dilakukan melalui desktop aplikasi user atau PDA, kemudian update data tersebut akan dicatat dalam database. Informasi ini akan segera diproses oleh server aplikasi, yang kemudian akan mengirimkan sms update kepada pengguna.

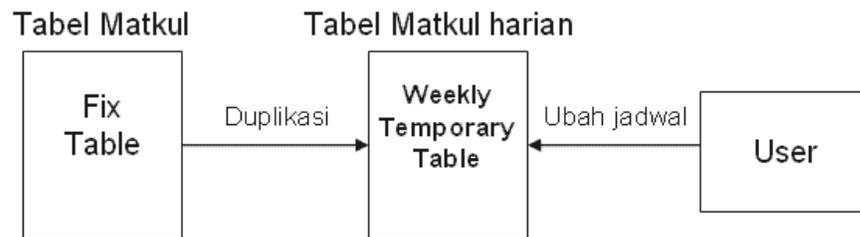
Pada sistem database tersebut terdapat informasi berupa berbagai *field* seperti mata kuliah, dosen pengajar, mahasiswa yang terdaftar, nomor telepon seluler pengguna, dan jadwal perkuliahan yang data awalnya diinputkan oleh administrator. Sistem database ini kemudian dapat diakses oleh server aplikasi untuk manajemen jadwal. Pada dasarnya kedua jenis server tersebut (database dan aplikasi) terletak pada satu komputer yang sama. Selanjutnya *user* dapat mengupdate jadwal perkuliahan dengan menggunakan *browser* internet yang mengakses server database melalui server aplikasi.

Hubungan database yang ada pada server ini ditunjukkan pada Gambar 3.2. Sistem database ini menggunakan database MySQL. Setiap user (dalam hal ini dosen dan mahasiswa) dapat mengakses aplikasi, namun sebelumnya harus melakukan login terlebih dahulu. Baik dosen maupun mahasiswa masing-masing akan mendapatkan *username* dan *password* agar dapat mengakses aplikasi. Lihat Gambar 3.4 Tampilan Login User.



Gambar 3.2. Hubungan database pada server database sistem aplikasi informasi penjadwalan perkuliahan elektronik.

Sistem database ini dirancang berdasarkan pada jadwal rutin mingguan. Namun permasalahan dengan sistem database tersebut adalah jika pada saat terjadi perubahan pada salah satu jadwal pada database (sebagai contoh, jadwal kuliah yang setiap minggu diadakan pada pukul 16:00:00, pada minggu kedua diubah menjadi pukul 15:00:00), maka perubahan tersebut akan secara permanen diaplikasikan pada minggu-minggu berikutnya. Padahal sebenarnya perubahan tersebut hanya berlaku pada minggu saat perubahan tersebut dilakukan. Untuk menghindari hal tersebut, maka penulis merancang satu tabel khusus lain yang disebut dengan tabel MatKul harian (jadwal matakuliah mingguan). Tabel ini duplikat harian dari tabel MatKul (tabel jadwal matakuliah yang berlaku secara permanen dalam satu semester), sehingga pada saat pengguna merubah data pada database tersebut, data yang berubah adalah data yang terdapat pada tabel MatKul harian, sedangkan data pada tabel MatKul tidak akan berubah dan tetap terulang pada minggu-minggu seterusnya. Gambar duplikasi kedua tabel database tersebut dapat dilihat pada Gambar 3.3.



Gambar 3.3 Duplikasi Tabel Matkul dan Tabel Matkul harian

### 3.2 Terminal User (Pengguna)

Terminal user atau pengguna pada sistem ini dapat berupa PDA, desktop, dan telepon seluler. Dengan terminal tersebut, user dapat terhubung dengan sistem, bahkan melakukan perubahan jadwal (dilakukan oleh dosen). Berikut adalah uraian dari berbagai terminal user yang dapat digunakan untuk mengakses server database.

- **PDA**

PDA digunakan untuk mengakses aplikasi sistem informasi penjadwalan perkuliahan elektronik. User terhubung dengan database server menggunakan *web browser* melalui intranet yang dapat diakses dengan menggunakan WiFi. Dengan PDA, pengguna dapat melihat jadwal perkuliahan dan dapat melakukan perubahan (update) terhadap jadwal tersebut. Hal ini tergantung pada otoritas yang diberikan pada pengguna tersebut.

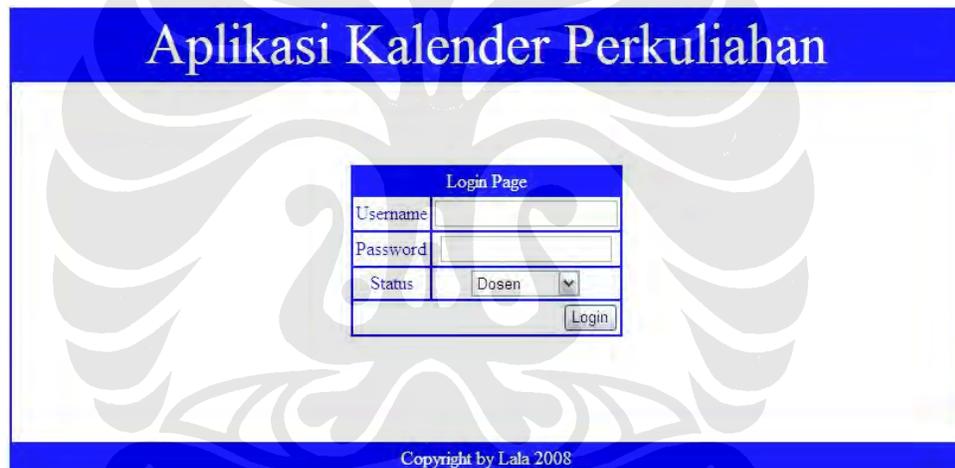
- **Dekstop User**

Aplikasi user merupakan antarmuka sistem yang dapat diakses oleh pengguna dari PC, asalkan PC tersebut dapat terhubung dengan intranet. Sama seperti PDA, aplikasi user memungkinkan pengguna untuk mengakses database dan melakukan perubahan terhadap jadwal yang ada dalam database tersebut.

- **Telepon Seluler**

Telepon seluler digunakan untuk menerima sms reminder dari server aplikasi, juga sms mengenai adanya perubahan jadwal (jika terdapat update pada database server, misalnya mengenai perubahan waktu perkuliahan, perubahan ruangan, atau pembatalan perkuliahan).

Untuk dapat mengakses sistem, setiap user akan diberikan *login profile* dan password. Gambar 3.4 dan Gambar 3.5 merupakan halaman muka dan halaman login untuk user yang hendak mengakses aplikasi. Gambar 3.6 dan Gambar 3.7 merupakan contoh tabel mahasiswa dan tabel mata kuliah pada aplikasi sistem informasi penjadwalan perkuliahan elektronik. Sedangkan Gambar 3.8 merupakan tampilan menu dosen pada saat user telah login. terlihat terdapat tulisan 'edit' dan 'delete' yang dapat di klik jika user tersebut ingin mengubah isi database atau menghapus database. Gambar 3.9 adalah tampilan menu mata kuliah pada saat user telah login. Berbeda dengan tampilan menu dosen, pada menu mata kuliah ini terdapat menu 'peserta' yang berisi mengenai detail dari matakuliah tersebut seperti terlihat pada Gambar 3.10.



Gambar 3.4 Tampilan Awal dan Menu Login ke Sistem



Gambar 3.5 Tampilan Login User

Daftar Mahasiswa			
No	NIP	Nama	HP
1	5102100005	Aris Kumara Prabhawa	08170022022
2	5102100006	Radityo Anggoro	08563472644
3	0606042834	Pamela	081808261746

Gambar 3.6 Contoh Tampilan Tabel Daftar Mahasiswa

Daftar Mata Kuliah				
ID	Nama	Hari	Waktu	Ruang
FIS01	Fisika Dasar	Senin	07:00:00	K101
MAT01	Matematika Dasar	Selasa	10:00:00	K102
PRO01	Pemrograman Dasar	Rabu	12:00:00	IK208

Gambar 3.7 Contoh Tampilan Tabel Daftar Matakuliah

Aplikasi Kalender Perkuliahan

Welcome

Nama : Aryotomo Markam  
ID : AM01  
HP : 08170022022  
Status : Dosen

[Logout](#)

---

Menu

[Dosen](#)

[Mahasiswa](#)

[Mata Kuliah](#)

Daftar Dosen					
No	NIP	Nama	HP	Edit	Delete
1	AM01	Aryotomo Markam	08170022022	<a href="#">Edit</a>	<a href="#">Delete</a>
2	WH01	Wahyu Haryadi	08170022021	<a href="#">Edit</a>	<a href="#">Delete</a>
3	BA	Betty Alisjhabana	08170022021	<a href="#">Edit</a>	<a href="#">Delete</a>
4	PD	Pamela Darmaji	08170022021	<a href="#">Edit</a>	<a href="#">Delete</a>

Tambah Dosen

NIP

Nama

HP

Password

Gambar 3.8 Tampilan Menu Dosen



Gambar 3.9 Tampilan Menu Daftar Mata kuliah



Gambar 3.10 Tampilan Menu Peserta Mata Kuliah

### 3.3 Pemodelan Sistem

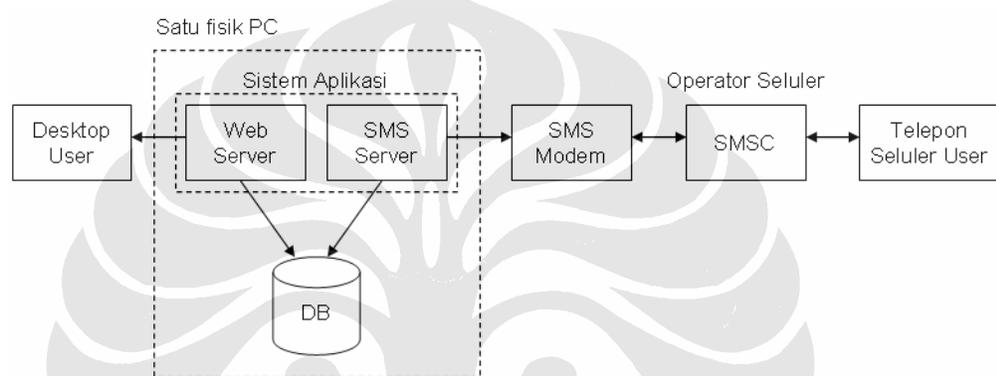
Sebagai bagian dari persyaratan sistem dan kegiatan perancangan, sistem aplikasi informasi penjadwalan perkuliahan elektronik terdiri dari beberapa komponen yang berkaitan. Hubungan masing-masing komponen ditunjukkan dengan diagram blok pada Gambar 3.11.

Komponen fungsional pada sistem dijelaskan sebagai berikut:

- o Komponen Web Server yaitu komponen sistem yang berfungsi menghubungkan user dengan database sistem. pada web server, terdapat

program desktop user yang merupakan antarmuka user untuk dapat mengakses database. Selain perancangan antarmuka user, pada komponen ini juga dirancang program yang dapat melakukan perubahan isi database.

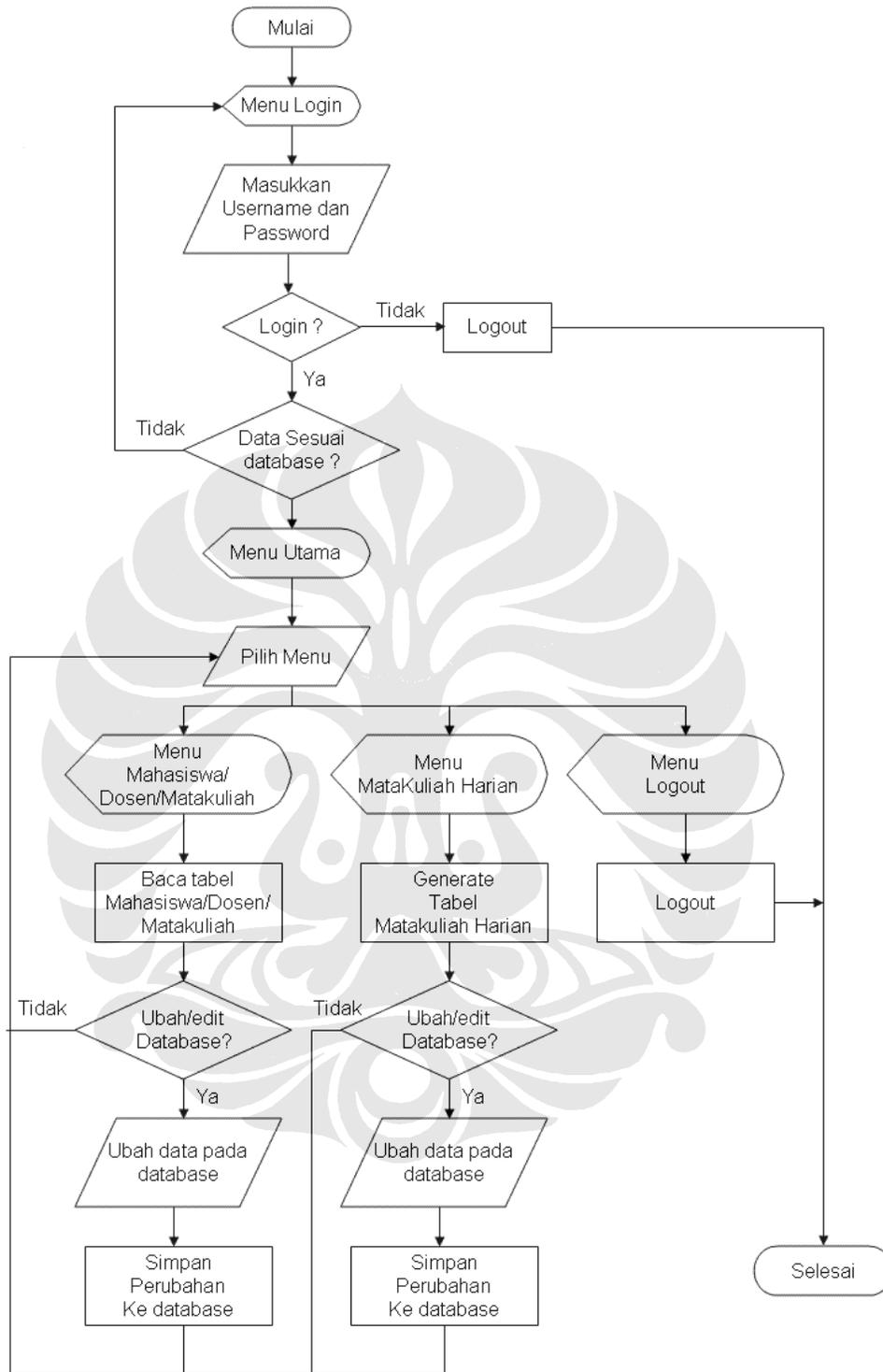
- o Komponen SMS Server yaitu komponen yang memungkinkan sistem untuk dapat berkomunikasi dengan user melalui teknologi SMS.
- o Database merupakan komponen sistem yang menyimpan seluruh data dan penghubung antara kedua komponen sistem lainnya, yaitu web server dan SMS server.



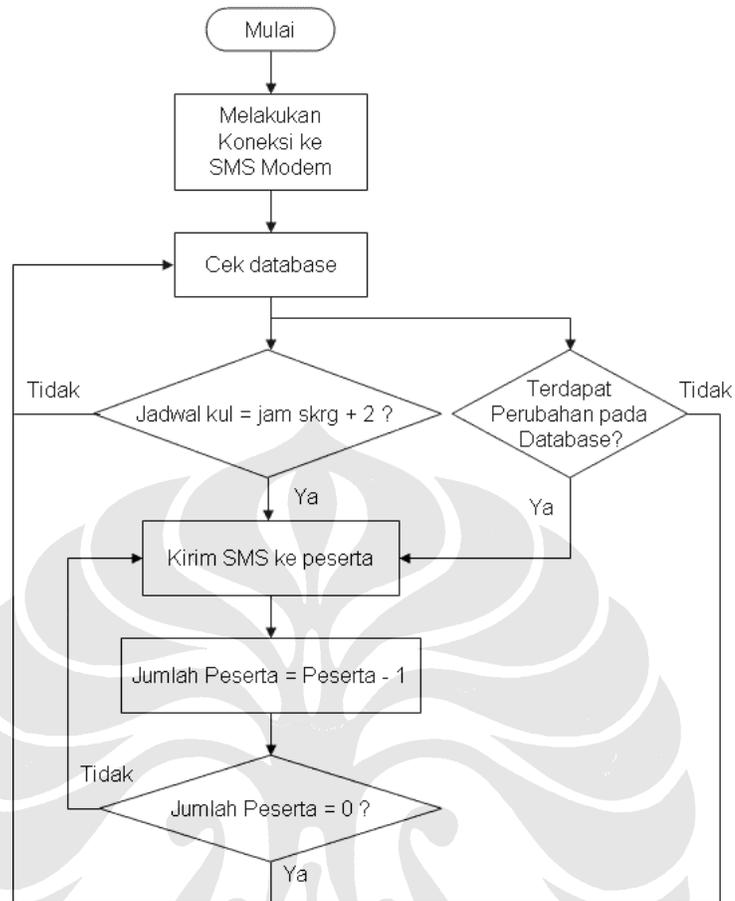
Gambar 3.11 Diagram Blok Sistem

### 3.4 Diagram Alir Sistem

Diagram alir sistem menunjukkan bagaimana sistem melakukan proses eksekusi dari pengguna memberikan masukan dan sistem merespon masukan tersebut. Secara lengkap diagram alir sistem direpresentasikan pada Gambar 3.12 (a) dan Gambar 3.12 (b). Gambar 3.12 (a) merupakan diagram alir dari web server, sedangkan Gambar 3.12 (b) merupakan diagram alir SMS server.



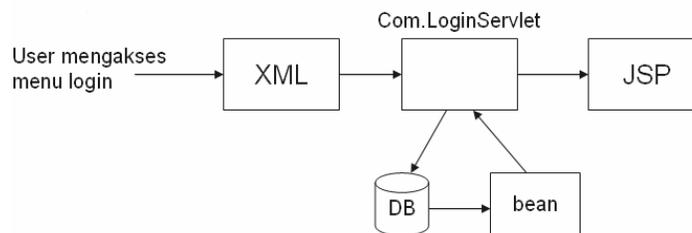
Gambar 3.12 (a) Diagram Alir Web Server



Gambar 3.12 (b) Diagram Alir SMS Server

### 3.5 Perancangan Perangkat Lunak

Sistem aplikasi informasi penjadwalan perkuliahan penulis menggunakan bahasa pemrograman Java. Dalam pemrograman, terdapat beberapa bagian, dimana setiap bagian memiliki fungsi tersendiri. Gambar 3.13 merupakan contoh dari proses pengekseskuan program Java yang diimplementasikan pada sistem, yaitu menu login pada halaman *index*.



Gambar 3.13 Contoh Alur Pengekseskuan Program Java pada menu login

Dari Gambar 3.13 terlihat bahwa pada saat user mengakses menu login, maka program akan langsung merujuk ke `web.xml`, yang kemudian akan melakukan mapping terhadap halaman login tersebut, pada *source packages* com, ditemukan *link* dari halaman login, yaitu `com.LoginServlet`. Maka `login.servlet` akan mengecek ke database apakah username dan password yang digunakan telah sama (*match*). Jika ya, maka informasi tersebut akan ditampung pada *bean* yang kemudian diteruskan lagi ke com. selanjutnya com akan mengantarkan user ke *JavaServer Pages* (JSP) yang dituju, dalam hal ini adalah halaman menu utama yang menandakan bahwa user tersebut telah berhasil melakukan login.

### 3.5.1 Web Pages

Web pages berisi mengenai program untuk membuat tampilan web. file yang terdapat dalam bagian ini misalnya adalah `home.jsp`. Disini terdapat pengaturan warna latar belakang, desain tampilan web, dan lain-lain.

### 3.5.2 Web.xml

`Web.xml` berfungsi sebagai penghubung antar halaman web, yang melakukan *mapping*, kemana halaman web tersebut terhubung dengan halaman web yang lain. berikut adalah contoh `web.xml`.

```
<servlet>
  <servlet-name>LoginServlet</servlet-name>
  <servlet-class>com.LoginServlet</servlet-class>
</servlet>
<servlet>
  <servlet-name>MatkulListServlet</servlet-name>
  <servlet-class>com.MatkulListServlet</servlet-class>
</servlet>
```

Dari *listing* program di atas, dapat diketahui bahwa `LoginServlet` terhubung dengan `com.LoginServlet`, artinya, `LoginServlet` tersebut terhubung dengan `LoginServlet` yang ada pada bagian com. Begitu juga dengan `MatkulListServlet` terhubung dengan `MatkulListServlet` yang ada pada bagian com.

### 3.5.3 Source Packages

*Source Packages* berisi bagian-bagian utama dari program. Pada *source packages* terdapat tiga bagian, yaitu *Bean*, *Com*, dan *Connection*.

- **Bean**

Bean merupakan bagian yang berfungsi untuk menampung setiap informasi, dimana isinya sama dengan field database yang telah dibuat. *Listing* program dibawah, merupakan *listing* program pada Dosen.java.

```
public class Dosen {
    private String DSN_NIP;
    private String DSN_NAMA;
    private String DSN_HP;
    private String DSN_PASSWORD;

    public String getDSN_NIP() {
        return DSN_NIP;
    }
    public void setDSN_NIP(String DSN_NIP) {
        this.DSN_NIP = DSN_NIP;
    }
    public String getDSN_NAMA() {
        return DSN_NAMA;
    }
    public void setDSN_NAMA(String DSN_NAMA) {
        this.DSN_NAMA = DSN_NAMA;
    }
    public String getDSN_HP() {
        return DSN_HP;
    }
    public void setDSN_HP(String DSN_HP) {
        this.DSN_HP = DSN_HP;
    }
    public String getDSN_PASSWORD() {
        return DSN_PASSWORD;
    }
    public void setDSN_PASSWORD(String DSN_PASSWORD) {
        this.DSN_PASSWORD = DSN_PASSWORD;
    }
}
```

- **Com**

Com berisi program-program untuk pengekseskuan setiap halaman web. Misalnya pada *listing* program DosenDelete.java berikut. halaman ini

mengambil data dari database MySQL dan terkoneksi dengan database MySQL tersebut.

```
import connection.ConnectDB
import java.sql.Connection;
import java.sql.PreparedStatement;
import javax.servlet.*;
import javax.servlet.http.*;
```

- **Connection**

Pada bagian ini, terdapat *listing* program ConnectDB.java, dimana *listing* program ini berisi hubungan antara program Java dengan database MySQL. berikut adalah *listing* programnya.

```
package connection;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;

public class ConnectDB {
    private static Connection conn=null;

    public ConnectDB(){
        try{
            Class.forName("com.mysql.jdbc.Driver");
            String url="jdbc:mysql://localhost:3306/TA";
            conn = DriverManager.getConnection(url, "root", "");
        }catch(Exception ex){
            System.out.println("Ex : " + ex);
        }
    }

    public Connection getConnection(){
        return conn;
    }
}
```

## BAB IV

### PENGUJIAN SISTEM

Pengujian sistem yang dilakukan adalah cara kerja aplikasi secara keseluruhan, yaitu dimana sistem dapat mengirimkan sms pengingat sesuai dengan jadwal yang terdapat dalam database, adanya masukan *update* dari seorang *user* (dalam hal ini user dosen) melalui aplikasi *user*, dan sms update yang kemudian akan dikirimkan oleh sistem kepada user lain (dalam hal ini user mahasiswa). Modem yang digunakan sebagai modem SMS *Gateway* adalah telepon seluler merk Siemens tipe C55, dan program dibuat dengan menggunakan bahasa pemrograman Java.

#### 4.1 Persiapan Pengujian Sistem

Sub bab ini akan menguraikan mengenai persiapan pengujian sistem informasi penjadwalan perkuliahan elektronik. Hal-hal yang perlu dipersiapkan dapat dilihat pada Tabel 4.1 berikut

Tabel 4.1 Kebutuhan Pengujian Sistem

No	Peralatan	Jumlah
1	Telepon seluler sebagai modem SMS <i>gateway</i>	1
2	PC/Laptop : - Server (server aplikasi, server database) - Aplikasi <i>user</i>	1 1
3	Telepon seluler / PDA ( <i>Personal Digital Assitant</i> ) sebagai terminal <i>user</i>	1

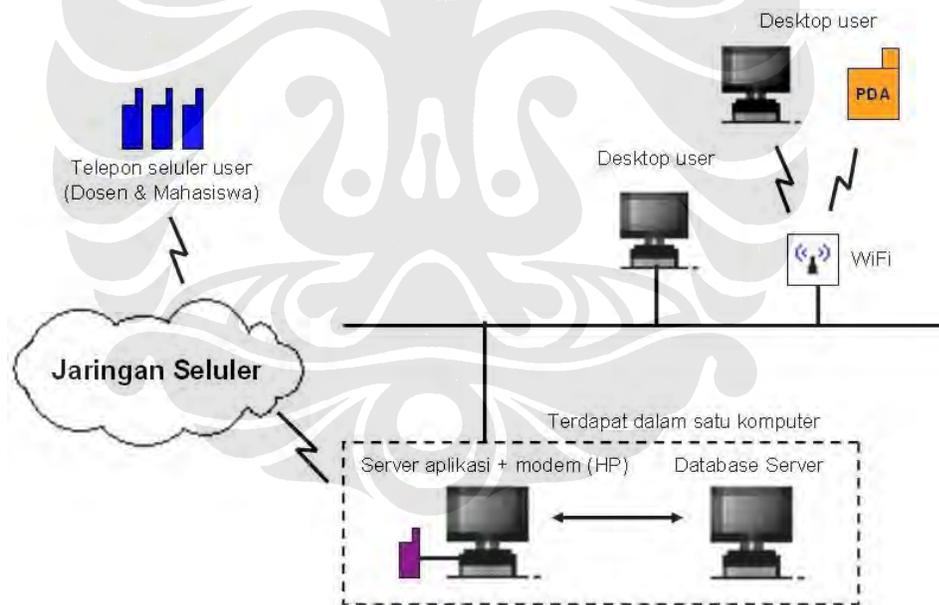
Seperti yang telah dijelaskan sebelumnya, bahwa telepon seluler yang digunakan sebagai modem SMS *gateway* adalah telepon seluler merk Siemens dengan tipe C55.

Satu PC atau Laptop dibutuhkan sebagai server yang didalamnya terdapat server aplikasi dan server database, dan terdapat satu PC atau Laptop yang berperan sebagai desktop user. Untuk dapat mengakses aplikasi, asalkan PC atau

Laptop user terhubung dengan intranet, sistem aplikasi dapat diakses melalui web browser (biasanya PC atau Laptop telah terinstal *browser* seperti Internet Explorer, Mozilla Firefox, atau Oprah). Modem SMS *gateway* dihubungkan ke PC atau Laptop server, diikuti dengan mengatur pengaturan *port* pada *Hyper Terminal*. Pengaturan *port* yang dilakukan dapat dilihat pada Tabel 4.2 Pengaturan port Siemens C55.

Tabel 4.2 Pengaturan Port Siemens C55

Parameter Port	Nilai
Bits per Second	19200
Data Bits	8
Parity	None
Stop Bits	1
Flow Control	None



Gambar 4.1 Skema Pengujian Sistem Aplikasi Informasi Penjadwalan Perkuliahan Elektronik

Telepon seluler yang digunakan sebagai terminal user adalah telepon seluler yang biasa digunakan oleh *user*. Sistem akan mengirimkan SMS

pengingat jadwal perkuliahan atau *update* jadwal (jika terdapat *update*) ke terminal *user*.

PDA juga merupakan salah satu perangkat pendukung aplikasi – sebagai terminal *user* – karena selain dapat berfungsi sebagai telepon seluler, PDA juga dapat membuka beberapa aplikasi komputer dasar, salah satunya adalah web browser. Untuk dapat terkoneksi dengan sistem, maka PDA harus dihubungkan dengan intranet, dan dapat langsung mengakses web aplikasi.

Skema dari pengujian sistem informasi penjadwalan perkuliahan elektronik yang dilakukan oleh penulis dapat dilihat pada Gambar 4.1.

## 4.2 Pengujian Sistem

Pengujian sistem aplikasi informasi penjadwalan perkuliahan elektronik dilakukan dengan prosedur awal sebagai berikut :

1. Diawali dengan melakukan pengaturan koneksi server dengan intranet Fakultas Teknik Universitas Indonesia, yaitu dengan mengeset *IP Address* dengan *IP Address* yang digunakan dalam intranet FT UI.
2. Mempersiapkan seluruh aplikasi yang diperlukan untuk melakukan pengujian, meliputi perangkat lunak sistem aplikasi, database server, dan server *SMS Gateway*.
3. Memastikan Laptop *user* atau PDA *user* telah terkoneksi dengan server sistem aplikasi informasi penjadwalan perkuliahan elektronik, yaitu dengan mengeset *IP Address* agar dapat terkoneksi dengan intranet FT UI.

### 4.2.1 Pengujian SMS Gateway

Pada pengujian *SMS Gateway*, yang diuji adalah pengiriman SMS pengingat oleh system aplikasi informasi penjadwalan perkuliahan elektronik kepada *user* yang bersangkutan, dan pengiriman *SMS update* jika terdapat *update* pada database sistem (misalnya, seorang dosen merubah jadwal kuliah).

Prosedur-prosedur yang dilakukan dalam pengujian *SMS Gateway* ini adalah sebagai berikut :

1. Mengisi database sesuai dengan jadwal yang diinginkan.

2. Selanjutnya sistem akan mengirimkan SMS pengingat secara otomatis kepada grup dosen dan mahasiswa yang terkait. Misalnya, SMS tersebut mengingatkan bahwa sore hari ini pukul 16:00:00 terdapat kuliah Matematika Dasar 1.
3. Salah satu *user* Dosen melakukan perubahan (*update*) terhadap database, misalnya memajukan jadwal kuliah, yang sebelumnya pukul 17:00:00 menjadi pukul 15:00. Perubahan dilakukan oleh *user* pada tampilan aplikasi *user* yang terhubung dengan intranet dan sistem aplikasi.
4. Kemudian sistem akan mengirimkan SMS *update* tersebut kepada *user* terkait, sehingga *user-user* tersebut mengetahui bahwa telah terjadi perubahan jadwal.
5. Pengujian juga dilakukan dengan percobaan untuk mengirimkan SMS dengan jumlah *user* yang berbeda.

#### 4.2.2 Pengujian Desktop User

Aplikasi *user* merupakan tampilan berbasis web yang dapat digunakan oleh *user* untuk dapat terhubung dengan sistem aplikasi. Antarmuka yang dapat digunakan oleh *user* adalah PC, Laptop, maupun PDA. Prosedur pengujian dilakukan dengan langkah-langkah sebagai berikut :

1. Memastikan bahwa PC/Laptop/PDA *user* telah terhubung dengan server aplikasi. Hal ini dapat diketahui dengan cara melakukan *pinging* yang merujuk ke *IP Address* yang digunakan oleh server.
2. User mengakses sistem dengan cara mengetikkan IP server aplikasi pada *web browser*-nya.
3. Pada tampilan awal, *user* akan diminta untuk melakukan login ke sistem aplikasi dan memilih status, apakah *user* tersebut login sebagai dosen atau mahasiswa. Lihat Gambar 4.1.

Login Page	
Username	0606042834
Password	••••••••
Status	Dosen
	<input type="button" value="Dosen"/> <input type="button" value="Mahasiswa"/> <input type="button" value="Login"/>

Gambar 4.2. Halaman Login

4. *User* dapat melihat isi database dengan mengklik *link* yang diinginkan yang terdapat pada tampilan web.
5. Login sebagai Dosen, kemudian membuka halaman mata kuliah. Mengubah atau mengupdate jadwal kuliah melalui menu *edit*.
6. Masih dengan status dosen, buka halaman dosen. Mengubah nomor telepon dosen melalui menu edit. Kemudian logout.
7. Login sebagai Mahasiswa, kemudian membuka halaman mahasiswa. Mengubah atau mengupdate nomor telepon mahasiswa. Untuk keluar, tekan logout.

#### 4.3 Analisa Pengujian Sistem

Dalam pengiriman dan penerimaan pesan pendek (SMS) terdapat dua mode, yaitu mode teks dan mode PDU. Mode teks adalah format pesan dalam bentuk teks asli yang dituliskan pada saat akan mengirim pesan, dalam hal ini adalah SMS permintaan pengguna mengenai jadwal ujian kuliah. Mode PDU adalah format pesan dalam bentuk octet heksadesimal dan octet semidesimal dengan panjang mencapai 160 (7 bit) atau 140 (8 bit) karakter.

Pesan yang dikirimkan oleh terminal masih dalam bentuk teks, sedangkan dalam pengiriman ke SMSC harus dalam bentuk PDU. Untuk itu sebelum informasi dikirim, SMS Server akan melakukan perubahan dari format teks menjadi format PDU, proses ini sering disebut *encodec*. Begitu juga pada sistem penerimaan SMS pada sistem. Pesan yang datang dari SMSC (dalam bentuk PDU) akan dikodekan di dalam SMS Server menjadi mode teks. Proses ini sering disebut *decodec*.

Format PDU diset didalam program SMS Server sesuai dengan format PDU yang diinginkan. Format PDU Pengirim yang digunakan pada sistem memiliki format seperti yang ditunjukkan pada Tabel 4.3.

Tabel 4.3 Format SMS PDU Pengirim Sesuai dengan Pengujian

SCA	PDU Type	MR	DA	PID	DCS	VP	UDL	UD
-----	----------	----	----	-----	-----	----	-----	----

Dalam pengujian yang dilakukan, sistem mengirimkan SMS pengingat ke salah satu ponsel *user* dengan nomor 081808261746. Nomor telepon yang digunakan oleh sistem untuk mengirimkan SMS adalah 08158919080. Misalnya SMS pengingat tersebut dikirim pada tanggal 19 Juni 2008 pukul 13:00:00 WIB. Maka format PDU-nya akan menjadi sebagai berikut :

- SCA

Tabel 4.4. SCA PDU Pengirim oleh Sistem

Oktet	Hasil
Len	0
Tipe Nomor	<tidak ada>
Nomor Service Center	<tidak ada>

Dari Tabel 4.4 dapat terlihat bahwa nilai SCA-nya adalah 00.

- PDU *Type*

Nilai *default* dari tipe PDU untuk SMS pengirim adalah 11.

- MR

Pada pengujian ini, penulis membiarkan pengaturan pesan SMS dilakukan sendiri oleh telepon seluler, maka nilai MR-nya adalah 00.

- DA

Tabel 4.5. DA PDU Pengirim oleh Sistem

Oktet	Nilai	Hasil
Len	12	0C
Tipe Nomor	Format Internasional	91
Nomor Service Center	6281808261746	2618808261746

Pada Tabel 4.5 dapat dilihat bahwa hasil pada *Destination Address* adalah 0C91261808261746

- PID  
Nilai *default* dari PID adalah 00 yaitu “*Standard Text*”. Pada sistem ini, pesan SMS yang dikirimkan menggunakan format teks standar, sehingga pada *Protocol Identifier* hasilnya adalah 00.
- DCS  
Pada sistem ini, pesan SMS yang dikirimkan berupa teks standar (bukan berupa *Flash SMS* atau *Blinking SMS*) sehingga nilai pada *Data Coding Scheme* adalah 00.
- VP  
*Validity Period* dalam kebanyakan sistem seluler adalah 5 hari, dimana hal ini berarti lama waktu pesan SMS disimpan di SMSC apabila pesan tersebut gagal diterima oleh telepon seluler penerima adalah 5 hari. Sehingga nilai VP pada sistem ini adalah  $166 + 5 = 171d = AB$  h. Jadi hasil dari *Validity Period* adalah AB.
- UDL  
*User Data Length* merupakan panjangnya pesan SMS yang akan dikirim dalam bentuk teks standar. Pada sistem, panjang pesan SMS yang dikirim mencapai 120 karakter. Sehingga untuk mempermudah analisa, penulis akan mengambil contoh pesan SMS yang lebih pendek, yaitu “Pesan Pendek”. Pesan ini memiliki 12 karakter (0C h), sehingga nilai UDL-nya adalah 0C.
- UD  
Pengkodean dari nilai teks standar menjadi heksadesimal dilakukan dengan bantuan *Default Alphabet* yang dibakukan oleh ETSI GSM. Sehingga pengkodeannya dapat dilihat pada Tabel 4.6. Dari tabel tersebut terlihat bahwa hasil dari pengkodean yang juga merupakan nilai UD adalah D0F23CEC06C1CB6E72790D.

Tabel 4.6. User Data PDU Pengirim oleh Sistem

Nilai	Dec	Septet (7 bit)	Oktet (8 bit)	Hasil
P	80	1010000	1 1010000	D0
E	101	110010 1	11 110010	F2
S	115	11100 11	001 11100	3C
A	97	1100 001	1110 1100	EC
N	110	110 1110	00000 110	06
<spasi>	32	01 00000	110000 01	C1
P	112	1 110000	1100101 1	CB
E	101	1100101		
N	110	1101110	0 1101110	6E
D	100	110010 0	01 110010	72
E	101	11001 01	011 11001	79
K	107	1101 011	0000 1101	0D

Dari pembahasan tersebut, maka diperoleh hasil untuk pengiriman SMS dalam format PDU oleh sistem adalah sebagai berikut :

0011000C912618229888040000AB0CD0F23CEC06C1CB6E72790Dn

Saat melakukan pengujian SMS Gateway, sistem dihadapkan dengan beberapa *user*, dalam percobaan dilakukan dengan jumlah *user* yang beragam, lima, tujuh, dan sepuluh *user*. Untuk mengirimkan SMS ke *user-user* tersebut memerlukan waktu yang berbeda pula. Waktu yang diperlukan untuk mengirimkan SMS ke lima *user* adalah 1 menit. Waktu yang diperlukan untuk mengirimkan SMS ke tujuh *user* adalah 1 menit 20 detik. Dan waktu yang diperlukan sistem untuk dapat mengirimkan SMS ke sepuluh *user* adalah 2 menit 12 detik (lihat Tabel 4.7). Dengan hasil ini maka dapat disimpulkan semakin banyak *user* yang harus dikirimkan SMS, maka semakin lama pula waktu yang diperlukan oleh sistem untuk mengirimkan SMS keseluruhan *user* tersebut. Hal ini dikarenakan sistem mengirimkan SMS tersebut secara berurutan, sehingga semakin banyak *user* yang akan dikirimkan SMS, pesan tersebut akan ter-*query* dan menunggu hingga satu per satu SMS terkirim. Mengingat jumlah mahasiswa yang dapat melebihi limapuluh mahasiswa dalam satu kelas, disarankan agar setiap matakuliah memiliki paling banyak 5

mahasiswa sebagai wakil kelas yang dianggap dapat menyebarkan pesan tersebut ke mahasiswa-mahasiswa lainnya.

Tabel 4.7. Hasil Pengujian dengan Menambah Jumlah User

Jumlah User	Waktu Pengiriman SMS
5	1 menit
7	1 menit 20 detik
10	2 menit 12 detik

Penulis juga melakukan pengujian dengan mengubah *bit per second* dari parameter modem SMS Gateway. Hal ini dilakukan hanya sebagai pengetahuan tambahan dan pembuktian bahwa tidak semua standar PDU string dapat bekerja dengan modem SMS gateway tertentu (dalam hal ini adalah telepon seluler). Sehingga perlu untuk mengetahui parameter *default* dari modem SMS Gateway yang digunakan. Hasil pengujian ini dapat dilihat pada Tabel 4.8 Dengan bit per second sebesar 19200 SMS terkirim dengan baik. Begitu juga pada bit per second diset pada nilai 8600. Namun pada saat nilai bit per second modem di set pada 1200, SMS menjadi error dan tidak terkirim. Hal ini disebabkan karena pengaturan untuk setiap merk dan jenis telepon seluler dapat berbeda-beda tergantung dari kapasitas dan kecepatan dari telepon seluler tersebut. Dan nilai parameter bit per second yang diset pada nilai 1200 tidak cocok dengan tipe telepon seluler yang penulis gunakan sebagai modem.

Tabel 4.8. Hasil Pengujian Mengubah bit per second modem SMS Gateway

Bit per Second	Status SMS
1200	Error
8600	Terkirim
19200	Terkirim

Setelah melakukan pengujian, terlihat bahwa sistem dapat berfungsi sesuai dengan fungsinya. Pada pengujian SMS Gateway, sistem dapat mengirimkan SMS pengingat dengan baik sesuai dengan waktu yang telah diatur, misalnya jadwal kuliah yang dijadwalkan pada pukul 17:00:00 WIB, maka sistem

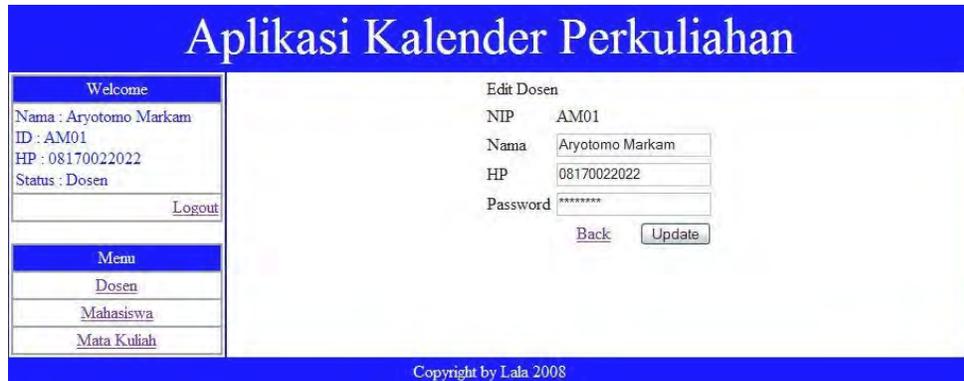
dapat mengirimkan SMS pengingat dua jam sebelumnya, yaitu pukul 15:00:00 WIB. Begitu pula saat terdapat *update* pada database, dimana dosen mengubah jadwal kuliah dari pukul 17:00:00 WIB menjadi pukul 16:00:00 WIB, sistem mengirimkan SMS *update* yang berisi pemberitahuan mengenai perubahan jadwal kuliah kepada mahasiswa yang bersangkutan.

Pada pengujian web, sistem juga dapat bekerja dengan baik, dimana *user* dapat melakukan login, mengubah database sistem, dan melihat isi database sistem. Lihat Gambar 4.3 dan Gambar 4.4 yang merupakan tampilan desktop *user* untuk menu merubah jadwal mata kuliah dan informasi dosen. Gambar 4.5 adalah tampilan untuk menambahkan user 'Dosen'. setelah di *submit*, maka data yang dimasukkan akan direkam di database.

Pengguna yang login dengan status Dosen dapat mengubah jadwal matakuliah dan nomor telepon dosen. Sedangkan pengguna yang login dengan status mahasiswa akan dapat mengubah nomor telepon mahasiswa. Dapat disimpulkan bahwa dalam pengujian ini, sistem telah berfungsi sesuai dengan fungsinya.

The screenshot displays the 'Aplikasi Kalender Perkuliahan' interface. On the left, a 'Welcome' section shows user details: Nama: Aryotomo Markam, ID: AM01, HP: 08170022022, and Status: Dosen, with a 'Logout' button. Below this is a 'Menu' section with links for 'Dosen', 'Mahasiswa', and 'Mata Kuliah'. The main area is titled 'Edit Mata Kuliah' and contains the following form fields: ID (FIS01), Nama (Fisika Dasar), Hari (Senin), Waktu (09:00:00), and Ruang (IK302). At the bottom of the form are 'Back' and 'Update' buttons. The footer indicates 'Copyright by Lala 2008'.

Gambar 4.3. Tampilan Menu Edit Mata Kuliah



Gambar 4.4. Tampilan Menu Edit pada User 'Dosen'



Gambar 4.5. Tampilan Menu Penambahan Dosen



Gambar 4.6. Hasil Penambahan Dosen

Sistem aplikasi informasi penjadwalan perkuliahan elektronik ini dirancang sesuai dengan hasil pencarian informasi mengenai sistem apa yang sebenarnya dibutuhkan oleh mahasiswa dan dosen untuk dapat meningkatkan efisiensi dan efektifitas dari penyampaian informasi jadwal perkuliahan. Namun tetap ada beberapa permasalahan yang masih belum dapat teratasi dengan sistem ini, yaitu :

- Jika sewaktu-waktu provider penyedia layanan SMS *down* dan tidak dapat memberikan pelayanannya dalam jangka waktu tertentu.
- Terjadi *bottleneck* jika user yang akan dikirimkan SMS oleh sistem semakin banyak, seperti yang ditunjukkan pada Tabel 4.7 hasil pengujian penambahan jumlah user.

Ada suatu gagasan agar kecepatan pengiriman pesan ditingkatkan agar dapat menangani jumlah user yang cukup besar. gagasan tersebut adalah dengan menambahkan beberapa SMS *Gateway* dengan beberapa provider penyedia layanan SMS yang berbeda pada sistem, dimana salah satu fungsinya adalah untuk efisiensi biaya (pengiriman pesan pendek ke sesama operator biasanya memiliki tarif yang lebih murah, sehingga user akan dikelompokkan sesuai dengan operatornya). Sementara ini, sistem belum mampu untuk menyediakan fitur ini.

## BAB V

### KESIMPULAN

Dari penelitian yang telah dilakukan, maka dapat diambil beberapa kesimpulan sebagai berikut :

1. Aplikasi ini dapat dikembangkan lebih lanjut tidak hanya sebagai informasi penjadwalan perkuliahan, namun dapat menjadi informasi penjadwalan elektronik untuk berbagai instansi, kantor, organisasi, dan lain-lain.
2. Pengembangan tidak hanya sebatas pada intranet, dapat juga dikembangkan agar aplikasi dapat menggunakan jaringan internet.
3. Koneksi ke modem SMS Gateway pada pemasangan pertama kali harus selalu di cek apakah sudah terhubung dengan baik atau belum.
4. Standar PDU string hanya dapat bekerja dengan modem SMS gateway tertentu (dalam hal ini adalah telepon seluler).
5. Dibutuhkan waktu yang lebih lama untuk mengirimkan SMS ke beberapa user, untuk itu muncul gagasan untuk menggunakan *multi gateway operator*
6. Dalam perancangan database sistem kalender, perlu menambah satu database lagi yang berisi jadwal harian (sebagai *weekly temporary table*) yang merupakan duplikasi dari jadwal kuliah permanen selama satu semester (sebagai *fix table*), sehingga jika terjadi perubahan, perubahan tersebut tidak akan berdampak pada minggu-minggu selanjutnya.

## DAFTAR REFERENSI

- Achmatim, "*Koneksi Java-MySQL*", Jurnal, [www.bl.ac.id](http://www.bl.ac.id).
- Dian Sano, Alb., "*HTML, JSP, dan MySQL*", Penerbit ANDI, Yogyakarta, 2005.
- Gogothebee, "*Patching Siemens C55*", Jurnal, [www.c55patch.hit.bg](http://www.c55patch.hit.bg), 2004.
- Hermawan, Ari, "*Pemrograman Java*", Jurnal, PT Masterweb Media, 2002.
- Ifeachor, E.C., & Jervis, B.W., "*Digital Signal Processing*", Prentice Hall
- Muhardin, Endy., "*PHP Programming Fundamental dan MySQL Fundamental*",  
Jurnal, [www.ikc.cbn.net.id](http://www.ikc.cbn.net.id), 2006.
- Prasetyo, Harmi, "*Membangun SMS Gateway dengan Gammu dan Mysql*", Jurnal,  
[www.wordpress.com](http://www.wordpress.com), 2006.
- Rozidi, Romzi Imron., "*Membuat Sendiri SMS Gateway (ESME) Berbasis Protokol SMPP*", Penerbit ANDI, Yogyakarta, 2004.
- Setia, Danny, "*SMS Gateway di Windows*", Jurnal, [www.wordpress.com](http://www.wordpress.com), 2006.
- Tim Penelitian dan Pengembangan Wahana Komputer, "*Pengembangan Aplikasi Sistem Informasi Akademik Berbasis SMS dengan Java*", Penerbit Salemba Infotek, Jakarta, 2005.