

# SISTEM LAYANAN INFORMASI PELANGGAN TELEPON PT TELKOM BERBASIS WIRELESS APPLICATION PROTOCOL (WAP)

**Eko Aribowo**

Program Studi Teknik Informatika Universitas Ahmad Dahlan  
Kampus III Jl. Prof. Soepomo Janturan Yogyakarta  
Telp (0274) 379418 Fax. (0274) 381523  
[ekoab@uad.ac.id](mailto:ekoab@uad.ac.id), [ekoab@telkom.net](mailto:ekoab@telkom.net)

## **Abstrak**

Teknologi wireless internet atau akses internet tanpa kabel memungkinkan berbagai perangkat komunikasi bergerak (*mobile communication*) seperti *PocketPC*, *PDA*, dan *handphone* dapat melakukan akses ke jaringan internet. Hal ini memberikan alternatif solusi yang lebih praktis dan efisien terhadap kebutuhan akses informasi. Layanan informasi terhadap pelanggan telepon PT Telkom saat ini hanya dapat diakses secara langsung dengan cara melakukan panggilan. Apabila hal itu dilakukan melalui *mobile phone* maka biaya cukup tinggi. Makalah ini menguraikan desain dan implementasi sistem layanan informasi pelanggan Telepon PT Telkom berbasis WAP.

**Kata kunci** : sistem informasi, pelanggan PT Telkom, WAP, MySQL

## **1. PENDAHULUAN**

Saat ini para pemakai perangkat komunikasi bergerak, baik *handphose*, *Pocket PC*, maupun *PDA* (*Personal Digital Assistance*) apabila membutuhkan informasi mengenai pelanggan telepon PT Telkom maka harus melakukan pemanggilan terhadap layanan telkom, misalnya 108 atau 109. Hal tersebut tentu membutuhkan biaya pemanggilan yang jauh lebih tinggi jika dilakukan melalui *mobile phone* tersebut dibanding melalui *fixed phone*. Sebenarnya hal ini dapat diberikan layanan dengan biaya yang lebih efisien menggunakan teknologi yang sudah ada, salah satu alternatif tersebut adalah teknologi WAP (*Wireless Aplication Protocol*).

WAP merupakan teknologi yang memungkinkan berbagai perangkat bergerak (*mobile device*) dapat melakukan akses internet secara *wireless*. WAP menyediakan pilihan teknologi *browser* yang dapat dimanfaatkan secara maksimal untuk aplikasi berbasis *server* hingga dapat diakses melalui *handphone*. Makalah ini menguraikan desain dan implementasi hal tersebut.

## **2. LANDASAN TEORI**

### **2.1 JSP & MYSQL**

MySQL adalah *multiuser database* menggunakan bahasa *Structured Query Language* (*SQL*). *SQL* adalah bahasa standar yang digunakan untuk mengakses *database server*. Dalam konteks bahasa *SQL*, pada umumnya informasi tersimpan dalam tabel-tabel yang secara logika merupakan struktur dua dimensi yang terdiri atas baris-baris data yang berada dalam satu kolom atau lebih. Baris pada tabel sering disebut sebagai *instance* dari data, sedangkan kolom sering disebut sebagai *attribute* atau *field*. Keseluruhan tabel ini dihimpun dalam satu kesatuan yang disebut *database*.

Database *MySQL* memiliki kemampuan sebagai berikut :

1. Dapat dimanfaatkan dan digunakan oleh banyak *CPU* sekaligus
2. Memiliki konektivitas dengan Bahasa *C*, *C++*, *Java*, *Perl*, *PHP*, *Phyton*, dan *TCL*
3. Dapat berjalan dalam berbagai Sistem Operasi
4. Sangat cepat

5. Mendukung fungsi SQL
6. Mendukung ODBC
7. Mampu menangani database yang besar

*JSP (Java Server Pages)* adalah suatu teknologi *web* berbasis bahasa pemrograman *Java* dan berjalan di *Platform Java*, serta merupakan bagian teknologi *J2EE (Java 2 Enterprise Edition)* yang khusus menangani komponen *web*. *JSP* sangat sesuai dan tangguh untuk menangani presentasi di *web* dengan menyediakan cara yang lebih mudah dan cepat untuk membuat halaman-halaman *web* yang menampilkan isi secara dinamik.

Kelebihan-kelebihan *JSP* yaitu:

1. Portable karena dibuat dengan teknologi *Java*. Portable disini berarti suatu aplikasi yang dibuat dengan *Java* dapat berjalan di berbagai sistem operasi dengan sedikit modifikasi, bahkan mungkin tanpa modifikasi sedikit pun
2. Memudahkan pembuatan aplikasi dengan *tag*. *JSP* memiliki kumpulan *tag* yang memiliki berbagai fungsi yang mudah digunakan.]
3. Manajemen *memory* yang baik. *JSP* memiliki manajemen *memory* dan sekuritas yang baik seperti halnya *Java*
4. Memiliki performansi tinggi terhadap banyak *request* sekaligus dalam waktu yang sama.
5. Memiliki akses ke *API Java* yang lengkap, seperti *JDBC* dan *Java Mail*
6. Dapat menggunakan komponen yang *portable* dan *reusable* seperti *JavaBean*
7. Mudah dalam *deployment* dan *maintenance*. Aplikasi *JSP* memungkinkan untuk dikembangkan ke aplikasi bersekala interface seperti halnya aplikasi *J2EE*.

## 2.2 WAP

WAP merupakan langkah awal menuju *Mobile Internet* yang memungkinkan telepon seluler (ponsel) bisa mengakses internet. Tujuan spesifik WAP adalah untuk memberikan metode standard yang sederhana dan hanya terbatas bagi ponsel untuk mengakses internet. Untuk menampilkan halaman WAP pada ponsel maka pada tersebut harus terdapat *micro-browser*, seperti halnya browser pada internet.

Ponsel tidaklah secanggih *PC* yang mampu mengolah teknologi multimedia. Kecepatan koneksi internet pada *handphone WAP* menuju *server* melalui *WAP Gateway* yang memanfaatkan gelombang elektromagnetik tidak sebanding dengan kecepatan koneksi internet menggunakan sistem kabel (*fixed line*). Sebagai perbandingan, kecepatan koneksi ponsel WAP masih terbatas pada 9600 bps sedangkan koneksi *fixed line* bisa mencapai 256 kbps, bahkan bisa lebih tinggi lagi.

## 2.3 Deskripsi Sistem

Terdapat beberapa peralatan yang digunakan untuk mendukung aplikasi sistem pelayanan pelanggan telepon berbasis WAP dengan pembagian secara garis besarnya seperti berikut.

### a. RAS Server

RAS (*Remote Access Service*) adalah server yang memberikan layanan (*content*) ke *client* secara *remote* dari jarak jauh menggunakan protokol *PPP (Point to Point Protocol)* untuk koneksi antara *client* dengan *server* melalui koneksi *dial in*. Metode yang digunakan adalah dengan cara mengkonfigurasi *software (mgetty)* pada sistem operasi *Linux*. Dengan memasang *modem* dan mengkonfigurasi sedemikian rupa pada sistem operasi *Linux* memungkinkan PC dapat bekerja sebagai RAS server.

### b. WAP Gateway

*Gateway* merupakan peralatan yang menghubungkan antara dua elemen yang berbeda protokolnya atau sebagai jalur keluar-masuknya data. *Software* yang digunakan sebagai *gateway* adalah *kannel WAP gateway*. Koneksi antara *client* dan *gateway* menggunakan protokol WAP. WAP gateway menerima *request* kemudian memprosesnya sehingga dapat diterima oleh *web server* menggunakan protokol *HTTP*.

### c. Web Server

*Web server* merupakan *server internet* yang mampu melayani koneksi transfer data dalam protokol *HTTP*. Agar dapat diakses oleh *WAP browser*, *content (data)* pada *web server* dibuat menggunakan bahasa pemrograman *web* yang dipadukan dengan bahasa *WML*

(*Wireless Markup Language*), sebuah varian dari *HTML* yang bisa digunakan untuk menampilkan isi *web* di monitor ponsel. Program untuk *content* tersebut bersifat dinamis dan berhubungan erat dengan *database*. Dalam hal ini *Web server* yang digunakan adalah *Apache Tomcat* yang mendukung *programming Web Java Server Pages* dan terkoneksi dengan *database MySQL*.

#### d. WAP Browser

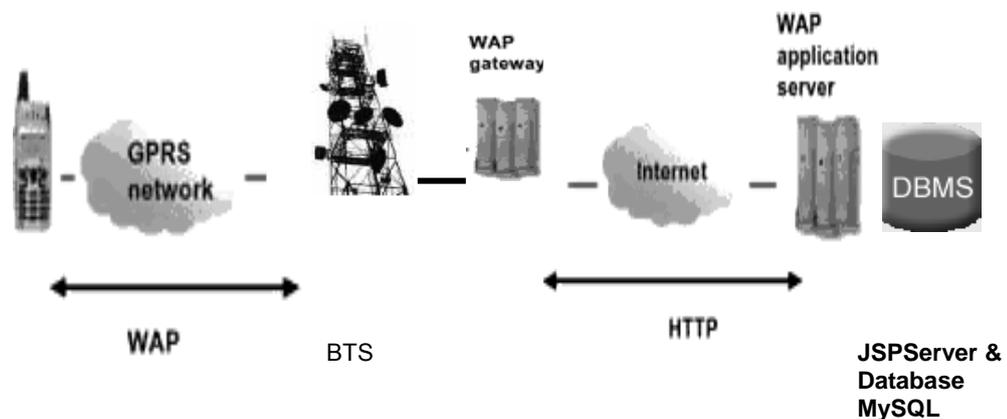
Pada ponsel yang *WAP enable* terdapat *software* yang disebut *microbrowser* untuk melayani aplikasi *WAP*. Selain itu layanan dari operator selular juga memegang peranan penting. Dengan adanya layanan *GPRS* yang sekarang sudah diberikan oleh semua operator selular di Indonesia memungkinkan untuk mengakses *mobile internet* dengan kecepatan yang cukup tinggi melalui ponsel. Jika ponsel dan *SIM card* dari operator selular menyediakan fasilitas tersebut maka aplikasi *WAP* dapat diakses.

### 3. KONFIGURASI SISTEM

Perincian secara garis besar untuk masing-masing konfigurasi diuraikan seperti berikut.

#### a. Konfigurasi Perangkat Keras

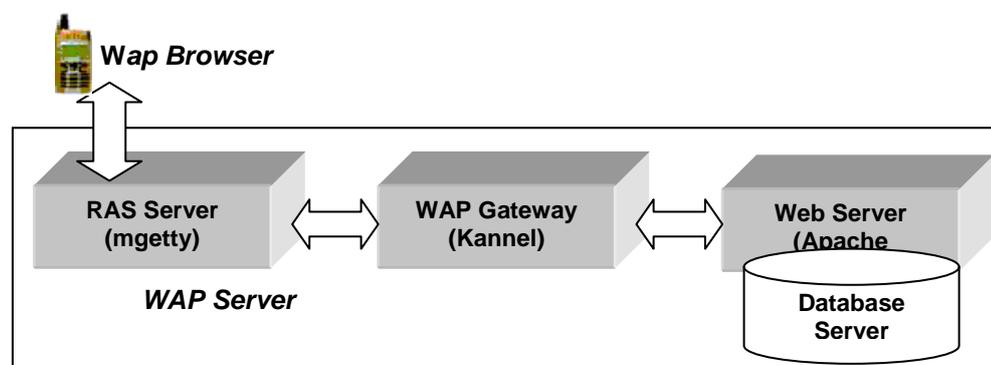
Konfigurasi perangkat keras informasi pelayanan pelanggan telepon berbasis *WAP* secara garis besar terlihat pada Gambar 1.



Gambar 1. Konfigurasi perangkat keras

#### b. Konfigurasi Perangkat Lunak

Perangkat lunak yang digunakan menggunakan *software-software* berbasis *open source*. Konfigurasi perangkat lunak informasi pelayanan pelanggan telepon berbasis *WAP* yang digunakan mengacu pada Gambar 2.



Gambar 2. Konfigurasi *software* aplikasi

#### 4. IMPLEMENTASI SISTEM

Konfigurasi model *system WAP* dan bentuk rancangan basis data yang telah ada digunakan untuk diimplementasikan pada tahap perancangan menu tampilan input data dan menu tampilan output serta teknologi pendukung yang diperlukan dalam pembuatan aplikasi perangkat lunak. Tahapan implementasi untuk masing-masing langkah dapat diuraikan seperti berikut.

##### a. Konfigurasi WAP Gateway

*WAP gateway* yang digunakan yaitu *kannel gateway* merupakan *software open source*. Arsitektur *Kannel gateway* secara umum terdiri atas tiga komponen, yaitu *bearerbox*, *smsbox*, dan *wapbox*. Untuk membangun *WAP server* yang dibutuhkan hanyalah *bearerbox* dan *wapbox*. *Bearerbox* akan mengaplikasikan tingkatan *bearer* dalam *protocol WAP*, sedangkan *wapbox* akan mengaplikasikan tingkatan *layer* di atas *layer datagram*. Untuk setting *WAP gateway*, */etc/kannel/kannel.conf* diubah seperti cara di bawah ini :

```
group = core
admin-port = 13000
wapbox-port = 9201
wdp-interface-name = "10.14.200.199"
log-file = "/tmp/kannel.log"
box-allow-ip = "*.*.*.*"
access-log = "/tmp/access.log"
group = wapbox
bearer-box-host = server
```

Setelah konfigurasi file tersebut selesai perlu disimpan kemudian mengetik perintah :

```
[root@domain]# /etc/rc.d/init.d/kannel start
```

Untuk melihat message/pesan dapat dilihat :

```
[root@domain]# tail -f /tmp/access.log
```

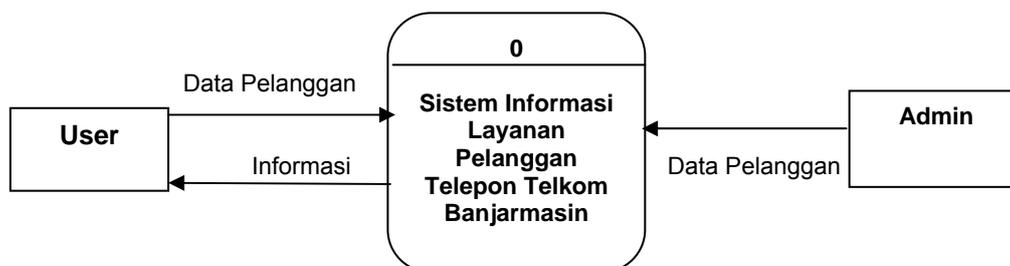
##### b. Konfigurasi Pengkoneksian JSP dan MySQL

Untuk pengkoneksian antara *JSP* dan *MySQL* perlu ditambahkan pada */etc/rc.d/rc.local* atau pada file */root/.bash\_profile* dengan perintah di bawah ini :

```
PATH=/usr/java/j2sdk1.4.2/bin:$PATH
JAVA_HOME=/usr/java/j2sdk1.4.2
CATALINA_HOME=/tomcat
CLASSPATH=/tomcat/lib/servlet.jar:/j2sdk1.4.2/lib/tools.jar
CLASSPATH=/m3mysql/mysql-connector-java-3.0.8-stable-bin.jar
```

##### c. Diagram Arus Data

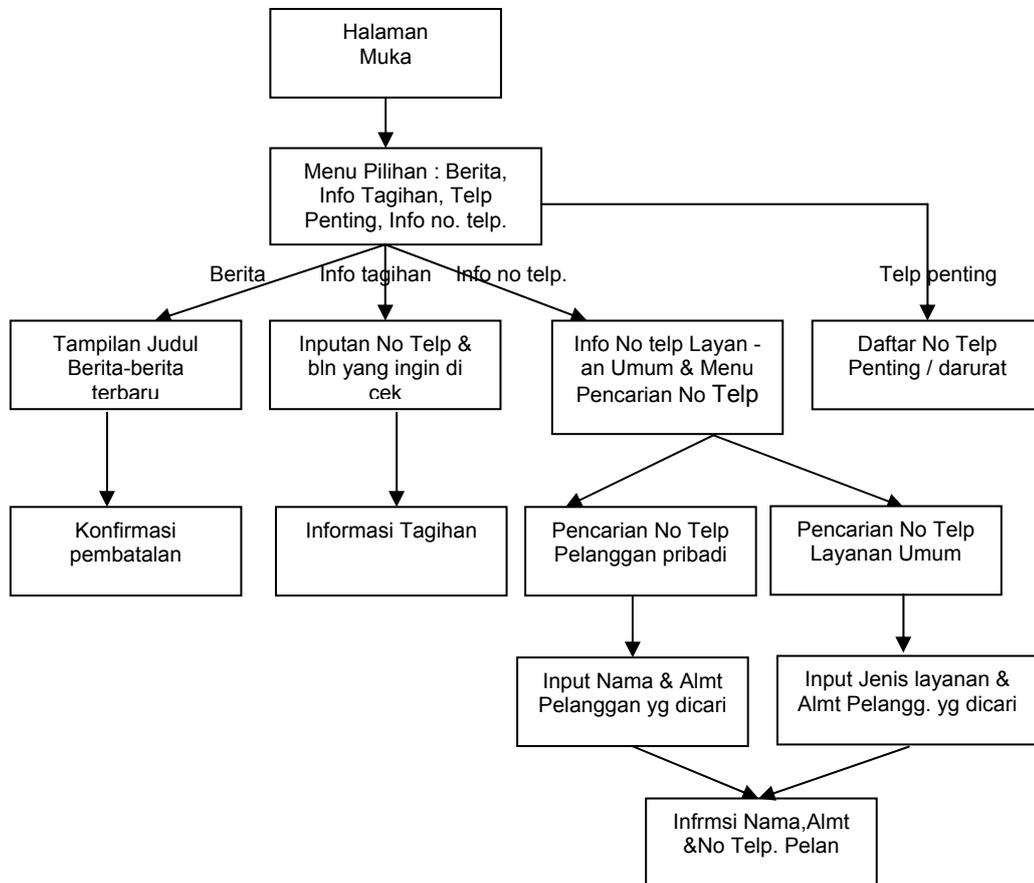
Diagram arus data menunjukkan aliran data dalam suatu proses. Dalam makalah ini *DAD* yang disampaikan level 0, artinya hanya digambarkan proses yang terjadi hubungan antar entitas yang ada dengan sistem, dalam sistem informasi pelayanan pelanggan telepon online, dapat dijelaskan seperti pada Gambar 3.



Gambar 3. Diagram konteks sistem

#### d. Hippo Menu Pelayanan

Diagram menggambarkan menu yang ada untuk mempermudah melakukan eksplorasi pada sistem yang dibangun



Gambar 4. Diagram hippo menu

## 5. UJI COBA KELAYAKAN PROGRAM

Pengujian program dilakukan dengan cara mengamati keluaran (*output*) sistem untuk berbagai masukan (*input*). Hasil pengujian program yang dilakukan ditunjukkan pada pemakai. Dari hasil uji program diperoleh persepsi pemakai (*user*) terhadap program, melalui tanggapan mengenai tingkat keramahan program, format tampilan, format masukan, dan keluaran. Jika sebagian *user* merasa semua hal tersebut baik maka sistem telah sesuai dengan rancangan untuk berbagai variasi data, dan sistem dinyatakan baik.

### a. Spesifikasi Sistem Pengujian

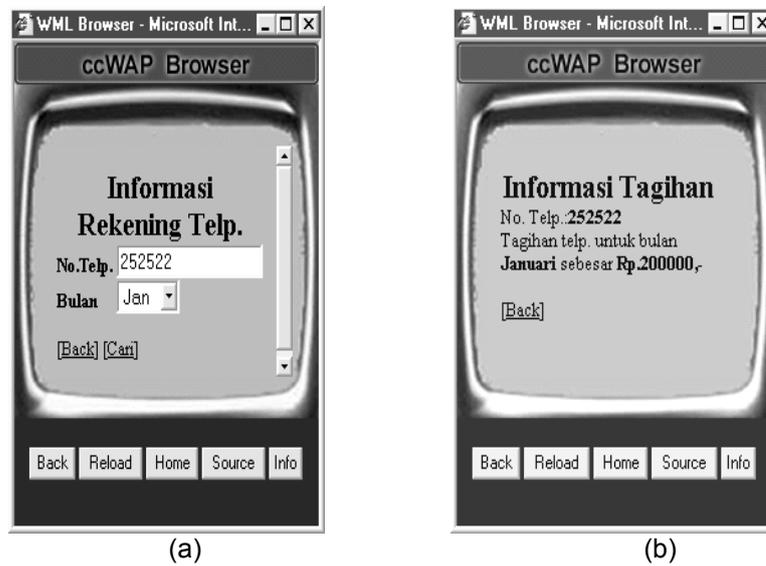
Pengujian dilakukan pada sebuah komputer dengan spesifikasi sebagai berikut :

Sistem Operasi Linux Red Hat 7.1  
 MySQL Server 3.21  
 RAM 128 Mb  
 Hardisk 40 Gb  
 VGA Card 64 Mb  
 Monitor dengan dukungan maksimal 1024x800 pixel, true color 24 bit  
 CheckCom WAP browser, emulator WAP untuk PC

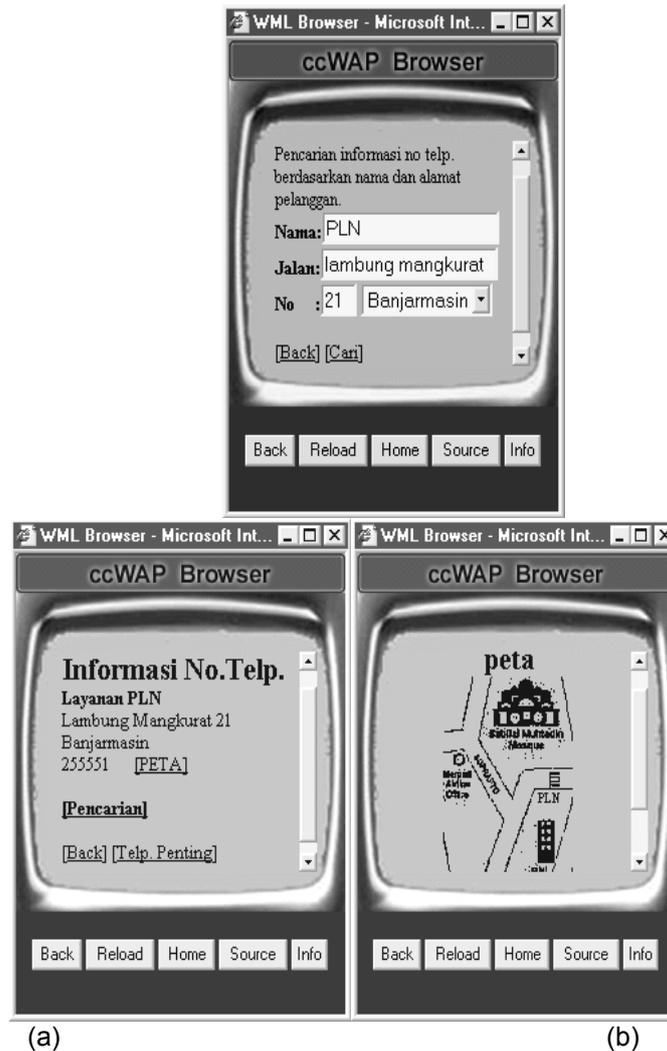
### b. Black Box Test

Pengujian program dilakukan dengan metode *Black Box Test*, yaitu metode pengujian yang dilakukan untuk memperoleh gambaran kesesuaian antara *input* dan *output*. Dalam uji ini

sistem dianggap sebagai kotak hitam yang memiliki pintu *input* dan *output*. Untuk keperluan unjuk kerja ini, sistem diberi masukan beberapa data dan diambil *outputnya*. Pengujian tersebut dapat dilihat seperti pada Gambar 5 dan 6.



Gambar 5. Pengujian *black box test* pencarian info tagihan (a) input (b) output



Gambar 6. Pengujian *black box test* pencarian nomor telepon pelanggan (a) input (b) output

Jika *output* sesuai dengan *input* yang diberikan maka kesimpulan pengujian *black box test* berjalan sesuai dengan yang diharapkan. *Output* data sesuai dengan *input* yang diberikan, data pelanggan dan input masukan sudah sesuai dan baik.

## 6. KESIMPULAN

Berdasarkan uraian di atas dapat dikatakan bahwa untuk memberikan layanan informasi pelanggan telepon PT Telkom dapat dibuat layanan berbasis WAP sehingga informasi mengenai pelanggan telepon PT Telkom dapat diakses melalui *mobile phone* dengan biaya yang lebih efisien.

## DAFTAR PUSTAKA

- [1] Cornell, Gary, "Core Java Edisi Indonesia", Penerbit Andi, Yogyakarta.
- [2] Mobile Communication Laboratory STT Telkom Bandung, "Membangun Wireless Application Protocol", Elex Media Komputindo, Jakarta.
- [3] Rickyanto Isak, "Java Server Pages", Elex Media Komputindo, 2003.
- [4] Saputro, H., "Manajemen Database MySQL Menggunakan MySQL Front", Elex Media Komputindo, Jakarta, 2003.

- [5] Tim, "**Pengembangan Program WAP**", Penerbit Andi, Yogyakarta, 2003.
- [6] Wicaksono, A., "**Dasar-Dasar Pemrograman Java 2**", Elex Media Komputindo, Jakarta, 2002.