

Aplikasi Pelaporan Berita Emergensi Secara Visual dan Tekstual Lewat Telepon Selular

Leo Willyanto Santoso, Sukanto Tedjokusuma, Marcel Renaldy Soetanto

Jurusan Teknik Informatika Fakultas Teknologi Industri

Universitas Kristen Petra

Jl. Siwalankerto 121-131 – Surabaya 60236

Telp +62 31 8439040, Fax + 62 31 8417658

Email :leow@petra.ac.id

Abstrak

Saat ini banyak situasi emergensi yang terjadi di sekitar kita. Pelaporan berita mengenai situasi emergensi tersebut terhadap sebuah media massa telah terbukti efektif. Saat ini kita dapat melaporkan situasi emergensi tersebut dengan menggunakan *Short Message Service* (SMS), *E-mail*, dan telepon. Akan tetapi *Email* dan telepon relatif sulit dijangkau pada saat situasi emergensi tersebut terjadi. Sedangkan SMS hanya dapat melaporkan situasi emergensi tersebut hanya berupa teks saja. Aplikasi ini memberikan alternatif lain untuk mempermudah dan mempercepat proses pelaporan situasi emergensi dengan menggunakan telepon selular ke sebuah *website* secara visual dan tekstual.

Dalam perancangan dan pembuatan aplikasi, terdiri atas : *database* untuk menyimpan berita yang dikirim; *website* untuk menampilkan berita yang dikirim; aplikasi *servlet* sebagai penghubung antara *database* dan *client* dengan menggunakan protokol *Hyper Text Transfer Protocol* (HTTP) pada *platform Java 2 Enterprise Edition* (J2EE); MIDlet pada telepon selular untuk mengirim berita emergensi melalui *General Packet Radio Service* (GPRS) dengan menggunakan *platform Java 2 Micro Edition* (J2ME).

Berdasarkan implementasi dan pengujian, disimpulkan bahwa J2ME didesain memiliki portabilitas yang tinggi. Akan tetapi adanya perbedaan implementasi dari pihak *vendor* telepon selular, maka menyebabkan tingkat portabilitas J2ME sedikit berkurang. Dengan adanya perbedaan tersebut, MIDlet pada aplikasi ini didesain dapat mengatasi perbedaan implementasi J2ME tersebut.

Kata kunci: J2ME, MIDlet, *servlet*, *mobile phone*, dan situasi emergensi.

1. Pendahuluan

Pada saat ini, kemajuan teknologi telepon selular yang berbasis *Global System for Mobile Communications* (GSM) sangatlah pesat. Dengan fitur-fitur baru yang ditawarkan oleh para *vendor* telah merubah telepon selular lebih dari sekedar alat komunikasi saja. Telepon selular telah berubah menjadi suatu perangkat multifungsi, dimana selain sebagai alat komunikasi, telepon selular juga dapat berfungsi untuk mengambil gambar dengan kamera *built-in*, bertukar file dengan menggunakan koneksi *wireless* seperti infra merah atau *bluetooth*, *browsing internet*, mengirimkan *Multimedia Messaging Services* (MMS), mendengarkan radio, mendengarkan musik dengan format MP3, dan lain sebagainya. Selain itu, fitur koneksi *General Packet Radio Services* (GPRS) telah menjadi standar pada telepon selular yang dirilis akhir-akhir ini. Hal ini ditunjang dengan semakin banyaknya operator telepon selular GSM yang juga mendukung koneksi GPRS.

Selain kemajuan telepon selular diatas, harga telepon selular semakin lama menunjukkan kecenderungan untuk semakin turun. Sehingga

telepon selular bukanlah suatu barang yang mewah, meskipun ada beberapa jenis yang ditujukan untuk kelas *high-end*. Hal diatas menyebabkan hampir semua orang memiliki telepon selular.

Dewasa ini setiap orang semakin membutuhkan informasi terhadap suatu peristiwa penting yang terjadi di sekitarnya secara cepat dan akurat. Seringkali suatu peristiwa penting terlambat untuk dilaporkan karena keterbatasan peralatan untuk melaporkan suatu kejadian seperti kamera untuk mengabadikan peristiwa tersebut, serta cukup rumitnya prosedur untuk mengirimkan laporan kejadian peristiwa ke suatu media massa.

Melihat permasalahan diatas, maka diperlukan sebuah aplikasi yang dapat memberikan solusi dengan menggunakan *device* yang hampir dimiliki setiap orang, yaitu telepon selular.

Dengan aplikasi yang dibuat maka setiap orang yang mempunyai telepon selular yang dilengkapi dengan kamera *built-in* dan koneksi GPRS, dapat melaporkan peristiwa tersebut dengan cepat ke media massa yang berbasis *website* serta dapat diakses dengan mudah, dimana saja kapan saja.

2. Java 2 Micro Edition

Java 2 Micro Edition (J2ME) adalah salah satu bagian dari *Java platform* yang dirancang khusus untuk dijalankan pada perangkat elektronik yang mempunyai RAM kecil dan prosesor yang mempunyai kemampuan terbatas. J2ME merupakan kumpulan dari spesifikasi yang mendefinisikan sekumpulan *platform*, setiap spesifikasi akan dapat mengakomodasi kebutuhan sejumlah *platform* dalam *scope* tertentu yang sama. Setiap *subset* dari *Java programming environment* untuk beberapa perangkat tertentu didefinisikan ke dalam satu atau lebih *profiles*, dimana setiap *profiles* merupakan pengembangan kemampuan lebih lanjut dari *configuration*. Penentuan *configuration* dan *profiles* suatu *device* bergantung pada lingkungan kerja *device* tersebut dan tujuan pemasarannya.

2.1. Configuration

Configuration adalah spesifikasi yang mendefinisikan *software environment* untuk beragam *device* yang didefinisikan berdasarkan sejumlah kesamaan karakteristiknya, yaitu tipe dan jumlah *memory* yang tersedia, kecepatan prosesor berikut tipenya, serta tipe koneksi jaringan yang tersedia untuk *device* tersebut. Saat ini dalam J2ME terdapat dua *configuration*, yakni:

1. *Connected Limited Device Configuration* (CLDC).
CLDC ditujukan untuk perangkat elektronik konsumsi *low end*. Platform CLDC yang paling umum adalah telepon selular atau PDA dengan ketersediaan *memory* sekitar 512KB.
2. *Connected Device Configuration* (CDC)
CDC ditujukan untuk perangkat elektronik yang masuk ke dalam kategori antara CLDC dan sistem *desktop* umumnya yang menjalankan J2SE. Perangkat *device* dalam kategori ini mempunyai prosesor yang lebih kapabel sehingga dapat mendukung lebih banyak kemampuan *Java software environment*.

2.2. Profiles

Profiles merupakan komplemen dari *configuration* dengan menambahkan sejumlah *class* tambahan yang memberikan fitur yang dibutuhkan bagi *device* tertentu atau *device* dengan *segment* pasar yang spesifik. Dalam J2ME terdapat enam *profiles*, yaitu:

1. *Mobile Information Device Profile* (MIDP)
Profile ini memberikan fitur tambahan berupa *networking*, *user interface components*, dan *local storage* pada CLDC. *Profile* ini ditujukan pada telepon selular dengan *display* dan fasilitas penyimpanan yang terbatas yang menyediakan *user interface* yang sederhana dan layanan jaringan dasar berbasis HTTP 1.1.
2. *PDA Profile* (PDAP)
PDA Profile hampir sama dengan MIDP tetapi ditujukan untuk PDA yang mempunyai *screen*

yang lebih baik dan *memory* yang lebih besar dari telepon selular.

3. *Foundation Profile*
Foundation Profile merupakan pengembangan dari CDC untuk mendukung hampir semua *core Java 2 Version 1.3 core libraries*.
4. *Personal Basis and Personal Profiles*
Personal Basis and Personal Profiles memberikan dukungan tambahan bagi *user interface functionality* dasar pada *Foundation Profile*.
5. *RMI Profile*
RMI Profile memberikan dukungan terhadap *library J2SE Remote Method Invocation libraries* pada *Foundation Profile*.
6. *Game Profile*
Game Profile akan memberikan *platform* untuk mengembangkan *software game* pada CDC

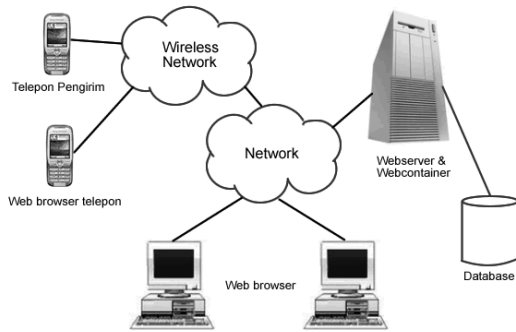
3. Mobile Media API (MMAPI)

Mobile Media API (MMAPI) merupakan *optional package* dalam MIDP [3]. *Package* ini merupakan *package* yang menangani pemrosesan multimedia pada MIDP. Dewasa ini, format dan tipe multimedia yang ada di pasar sangatlah beragam dan perkembangannya sangat pesat. Untuk mengatasi hal tersebut *Java Community Process* (JCP) mengeluarkan sebuah *package* untuk menangani pemrosesan multimedia yang bermacam-macam pada *devices* yang tipe dan kemampuan prosesoranya juga bermacam-macam.

Pada dasarnya, pemrosesan multimedia dapat dibagi menjadi dua bagian besar, yaitu menangani protokol pengiriman data (*protocol handling*) dan menangani isi datanya (*content handling*). Pada *protocol handling* yang dilakukan adalah membaca data dari sumbernya (seperti *file*, *server streaming*, *captured device*) lalu memprosesnya dalam *media processing system*. Sedangkan pada *content handling* yang dilakukan adalah memproses data media tersebut (seperti melakukan *parsing* atau *decoding*) kemudian melakukan *rendering* pada sebuah *output devices* seperti *audio speaker* atau *video display*. Pada API ini, terdapat dua objek *high level* yang digunakan yaitu *DataSource* dan *Player*. Masing-masing objek tersebut mewakili satu dari *multimedia processing*. Objek *DataSource* untuk mewakili *protocol handling*, sedangkan objek *Player* mewakili *content handling*.

4. Desain dan Implementasi Perangkat Lunak

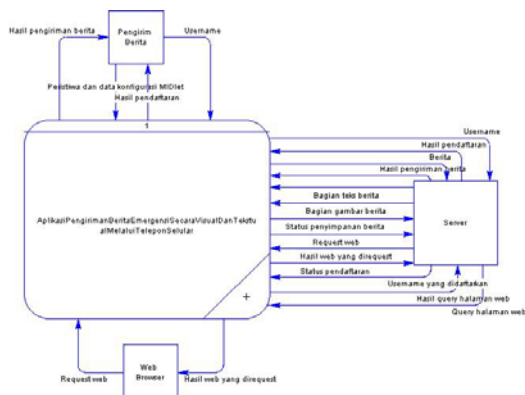
Arsitektur sistem pengiriman dan penerimaan berita secara lengkap diilustrasikan pada Gambar 1.



Gambar 1. Arsitektur Sistem Pelaporan Berita

Sistem terdiri atas sebuah *server* yang berfungsi sebagai *webserver* dan *webcontainer*, *servlet* yang menggunakan *platform* J2EE, dan telepon selular pengirim yang menggunakan *platform* J2ME, *web browser* pada telepon selular, *web browser* pada komputer *desktop*, dan sebuah *database server*.

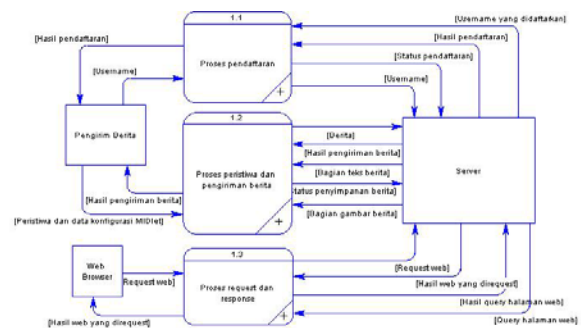
Secara garis besar, cara kerja sistem ini adalah pengirim harus mendaftarkan diri pada saat MIDlet digunakan untuk pertama kalinya. Setelah terdaftar, pengirim dapat mengirimkan berita melalui MIDlet yang sudah diinstall pada telepon selularnya ke *server*. Berita yang dikirim akan berupa gambar dan teks. Bagian berita yang berupa gambar akan disimpan di dalam *harddisk server*, sedangkan bagian berita yang berupa teks akan disimpan di dalam *database*. Setelah berita disimpan, maka berita hasil kiriman tersebut dapat dilihat melalui *web browser* yang ada pada telepon selular tersebut serta *web browser* pada komputer *desktop*. Agar dapat lebih mudah dipahami, disertakan *data flow diagram* dari sistem yang diilustrasikan pada Gambar 2.



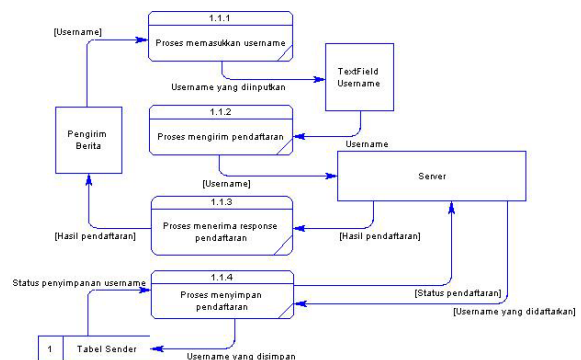
Gambar 2. Context Diagram Sistem Pelaporan Berita

Dalam *context diagram* pada Gambar 2. terdapat tiga *external entity* yaitu pengirim berita, *web browser* dan *server*. Tugas dari pengirim berita dalam *context diagram* tersebut adalah memasukkan

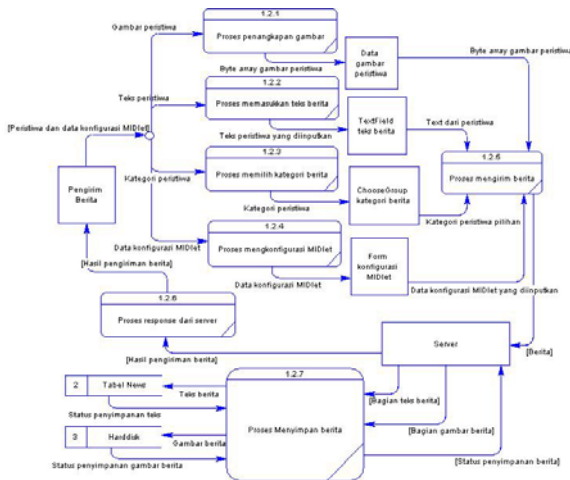
username untuk mendaftarkan diri, memasukkan berita dari peristiwa yang terjadi serta memasukkan data-data konfigurasi MIDlet. Tugas dari *web browser* pada *context diagram* tersebut adalah mengirimkan *request* terhadap suatu halaman *web*, kemudian menerima *response* dari *server* serta menampilkannya kepada pengguna. Sedangkan tugas dari *server* adalah memproses pendaftaran yang dilakukan pengirim berita, memproses berita yang dikirim pengirim sehingga dapat disimpan di dalam *database* yang akhirnya dapat ditampilkan pada halaman *web* yang di-*request* oleh *web browser*. Penjelasan lebih detail akan didapat setelah *context diagram* sudah di dekomposisi seperti yang tergambar pada Gambar 3, Gambar 4, Gambar 5 dan Gambar 6.



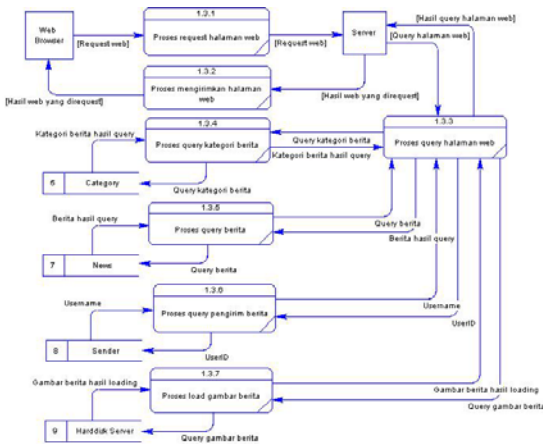
Gambar 3. DFD Level 1 Sistem Pelaporan Berita



Gambar 4. DFD Level 2 Proses Pendaftaran Sistem Pelaporan Berita



Gambar 5. DFD Level 2 Proses Peristiwa dan Pengiriman Berita Sistem Pelaporan Berita



Gambar 6. DFD Level 2 Proses Request dan Response Sistem Pelaporan Berita

5. Uji Coba

Proses pengujian dilakukan pada keempat elemen sistem, yaitu *website* untuk komputer *desktop*, *website* untuk telepon selular, *servlet*, dan MIDlet. Pengujian yang dilakukan bertujuan untuk menjelaskan apakah aplikasi pelaporan berita emergensi secara visual dan tekstual lewat telepon selular yang dibuat telah sesuai dengan targetnya, yaitu melaporkan atau mengirimkan berita secara visual dan tekstual pada sebuah *website* yang dapat diakses melalui komputer *desktop* maupun telepon selular serta untuk mengetahui tingkat portabilitas aplikasi pada beberapa *vendor* telepon selular yang terkemuka di Indonesia pada khususnya.

Dalam pengujian tingkat portabilitas MIDlet, penulis akan mengujinya pada telepon selular secara langsung dan melalui beberapa *emulator* Java. Dalam pengujian pada telepon selular secara langsung, pengujian akan menggunakan

telepon selular Sony Ericsson K500i dan Nokia 6600 serta Indosat IM3 sebagai operator telepon selularnya. Sedangkan untuk pengujian menggunakan *emulator*, pengujian menggunakan *emulator* dari Sony Ericsson, Nokia, Motorola, Siemens, serta Samsung.

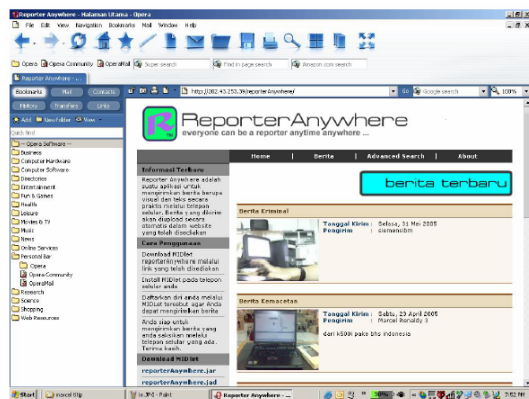
5.1 Pengujian Website

Proses pengujian yang dilakukan pada *website* versi komputer *desktop* adalah dengan cara mengakses *website* dengan resolusi dan *browser* yang berbeda-beda. Resolusi yang digunakan dalam menguji adalah 800x600 *pixel* dan 1024x768 *pixel*. Sedangkan *web browser* yang digunakan dalam menguji adalah Internet Explorer versi 5 keatas, versi 6 keatas, Mozilla Firefox 1.0, dan Opera 7.0. Hasil pengujian *website* versi komputer *desktop* dapat dilihat pada Tabel 1 dan Tabel 2.

Tabel 1. Hasil Pengujian Website versi Komputer Desktop

	Browser					
	Internet Explorer 5.x, 6.x		Mozilla Firefox 1.x		Opera 7.0	
Resolusi	800x600	1024x768	800x600	1024x768	800x600	1024x768
index.jsp	Baik	Baik	Baik	Baik	Baik	Baik
news.jsp	Baik	Baik	Baik	Baik	Baik	Baik
advanced search.jsp	Baik	Baik	Baik	Baik	Baik	Baik
about.jsp	Baik	Baik	Baik	Baik	Baik	Baik

Kondisi Baik yang dimaksud penulis dalam Tabel 1. adalah *website* dapat ditampilkan dengan sempurna dan tidak terdapat gambar yang *miss link* serta seluruh *link* dalam *website* dapat berfungsi sesuai dengan harapan.



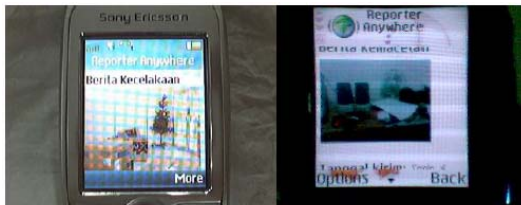
Gambar 7. Hasil Pengujian Website pada Opera

Tabel 2. Hasil Pengujian Halaman Administrator

Gambar	Keterangan
	Bagian dari halaman administrator untuk melakukan login.
	Bagian dari halaman administrator untuk memilih menu yang diinginkan administrator.
	Bagian dari halaman administrator untuk menghapus berita yang tidak pantas.
	Bagian dari halaman administrator untuk menghapus gambar yang tidak selesai terkirim.

Tabel 3. Hasil Pengujian Website versi Telepon Seluler

Halaman web	Browser		
	Openwave	Sony Ericsson K500i	Nokia 6600
Halaman utama	Baik	Baik	Baik
Halaman berita	Baik	Baik	Baik
Halaman pencarian	Baik	Baik	Baik
Halaman hasil pencarian	Baik	Baik	Baik



Gambar 8. Hasil Pengujian Website pada Sony Ericsson K500i dan Nokia 6600

5.2 Pengujian Servlet

Pengujian yang dilakukan pada *servlet* yang dibuat adalah dengan mengaksesnya baik untuk proses mendaftarkan diri maupun untuk mengirimkan berita. Proses pengujian dilakukan dari *emulator* dan dari telepon selular secara langsung. Hasil pengujian pada *servlet* dapat dilihat pada Tabel 4.

Tabel 4. Hasil Pengujian Servlet

		Servlet			
		Proses pendaftaran		Proses pengiriman berita	
Jumlah Client		1	Lebih dari 1	1	Lebih dari 1
Client	Emulator	Baik	Baik	Baik	Baik
	Telepon selular	Baik	Baik	Baik	Baik

5.3 Pengujian MIDlet

Pada pengujian MIDlet pada telepon selular, pengujian menggunakan telepon selular Sony Ericsson K500i dan Nokia 6600 dengan Indosat IM3 sebagai operatornya. Hasil dari pengujian ini dapat dilihat pada Tabel 5.

Tabel 5. Hasil Pengujian Midlet

Parameter pengujian	Nokia 6600	Sony Ericsson K500i
Proses menampilkan <i>splash screen</i>	Sukses	Sukses
Proses pendaftaran	Sukses	Sukses
Proses penyimpanan <i>userid</i> pada RMS	Sukses	Sukses
Proses pengambilan gambar	Sukses	Sukses
Proses pengiriman berita	Sukses	Sukses
Proses penanganan <i>time out</i>	Sukses	Sukses
Ukuran file gambar	± 45 KB	± 5 KB
Format file gambar	PNG	PNG



Gambar 9. Hasil Pengujian Midlet pada Telepon Seluler

6. Kesimpulan

Beberapa kesimpulan yang dapat ditarik dari penelitian ini adalah sebagai berikut:

1. J2ME dirancang untuk dapat berjalan pada telepon selular dari *vendor* yang berbeda-beda, tetapi karena belum adanya persepsi yang sama diantara para *vendor* maka menyebabkan *package* dalam J2ME diimplementasikan dengan tingkat yang berbeda-beda. Hal tersebut menyebabkan tingkat portabilitas J2ME sedikit berkurang.
2. *Servlet* dapat digunakan untuk menjembatani koneksi antara *mobile device* dengan *server* khususnya dengan *web server* maupun *web container*. *Servlet* juga mampu menjembatani *mobile device* dengan *database server* melalui koneksi JDBC.
3. MIDlet berhasil dijalankan dengan baik untuk melakukan proses pengiriman berita berupa gambar dan teks pada sebuah *web server* atau *web container* dengan bantuan *servlet*. Berita tersebut dapat dilihat dan dibaca dengan baik

melalui *browser* pada komputer *desktop* maupun *browser* yang terdapat pada telepon selular.

Daftar Pustaka

- [1] Avedal, Karl, et al. (2000). *Professional Java Server Programming J2EE Edition*. Birmingham: Wrox Press.
- [2] IETF. (Juni 1999). *Hypertext Transfer Protocol -- HTTP/1.1*. Last viewed: 25 Mei 2005.
<http://www.ietf.org/rfc/rfc2616.txt>
- [3] Java Community Process. (2003). *JSR-000135 Mobile Media API Specification 1.1*. Last viewed: 25 Mei 2005.
<http://jcp.org/aboutJava/communityprocess/final/jsr135/index2.html>
- [4] Nokia. (2005). *Known Issues In The Nokia 6600 J2ME Implementation*.
- [5] Ortiz, C. Enrique. (Agustus 2003). *The Generic Connection Framework*. Last viewed: 25 Mei 2005.
<http://developers.sun.com/techtopics/mobility/midp/articles/genericframework/index.html>
- [6] Siemens. (2004). *Siemens Mobility Toolkit for Java™ Development Programmer's Reference*.
- [7] Sony Ericsson. (1 Desember 2004). *Knowledge Base Article # 6657*. Last viewed: 25 Mei 2005.
<http://developer.sonyericsson.com/viewSolution.do?id=6657>
- [8] Sony Ericsson. (2005). *Developer Guidelines Java J2ME for Sony Ericsson mobile phones* (6th ed). Swedia: Author
- [9] Sun Microsystems. (2003). *Java™ 2 Software Development Kit, Enterprise Edition Documentation*.
- [10] Sun Microsystems. (2005). *Java™ 2 Software Development Kit, Micro Edition Documentation*.
- [11] Sun Microsystems. (n.d.). *J2ME Technologies Overview*. Last viewed: 25 Mei 2005.
<http://java.sun.com/j2me/docs/j2me-ds.pdf>
- [12] Sun Microsystems. (n.d.). *JavaServer Pages (JSP) v2.0 Syntax Reference*. Last viewed: 25 Mei 2005.
<http://java.sun.com/products/jsp/syntax/2.0/syntaxref20.html>