

Penerapan Teknologi J2EE untuk Aplikasi Pencarian dan Pemesanan Mobil Baru dan Bekas pada Dealer Mobil via Handphone

Tjatur Kandaga, Teddy Hartandy

Jurusan Teknik Informatika
Fakultas Teknologi Informasi, Universitas Kristen Maranatha
Jl. Prof. Drg. Suria Sumantri no. 65 Bandung 40164
Email: tjatur.kandaga@eng.maranatha.edu , t3ddy_h@yahoo.com

Abstract

When someone want to buy a new or used car, he can make better decision if he can find out cars data and which dealers sell the cars. It takes a lot of time if someone going around visiting car dealers and seeking cars data.

Using Java enterprise (J2EE), desktop (J2SE) & mobile (J2ME) technology a software for searching and ordering car was built. The Java enterprise application part implemented using Enterprise Java Bean (EJB) and Web Service resides on a server which serve as central database for desktop and mobile application. Java desktop application part installed at car dealers. Car dealers use this application to maintain their cars data and calculating car selling forecast using lagrange method. Java mobile application part installed on user cell phone or PDA as MIDlet application used to search cars & dealers data and order cars. The software leveraging internet and GPRS/3G technology.

Keywords : J2EE, Java, forecast, car, internet, handphone

1. Pendahuluan

Produk ini mencakup ruang lingkup teknologi J2EE dan akan dinamakan “Aplikasi Pencarian dan Pemesanan Mobil”. Produk ini dapat dijalankan pada *handphone* merk / jenis apa saja asalkan *handphone*-nya *support* Java (J2ME™) dan dapat terkoneksi dengan internet melalui GPRS/3G atau koneksi lainnya.

Beberapa keuntungan yang diperoleh masyarakat atau pengguna dalam menggunakan layanan ini, antara lain :

- Memudahkan mencari informasi tentang mobil pada dealer – dealer mobil di Bandung, dimanapun dan kapanpun asal ada *handphone* yang mendukung Java (J2ME™)
- Memudahkan melakukan pemesanan ke dealer yang dituju pada *handphone*

Dari dua keuntungan di atas, pengguna akan merasakan perbedaannya jika melakukan pencarian data mobil secara manual misalnya dengan mencari di buku – buku otomotif atau langsung menanyakan ke dealer. Dengan pemesanan melalui *handphone*, hal ini bisa dilakukan kapan saja dan di mana saja.

2. Penggunaan Teknologi J2EE

J2EE adalah kumpulan teknologi yang cukup kuat dan berada di atas lingkungan Java 2 Standart Edition (J2SE). J2EE berbasis pada Java 2 yang berusaha untuk menyediakan sebuah lingkungan aplikasi yang bersifat *reliable* dan stabil serta

dapat dijalankan pada beberapa lingkungan sistem operasi. Teknologi enterprise sebagai perkembangan dari lingkungan Java 2 difokuskan pada pemenuhan antarmuka yang standar dimana aplikasi J2EE dapat menghasilkan sebuah aplikasi berbasis *server* yang tangguh (*robust*) dan tidak bergantung pada lingkungan sistem operasi yang digunakan. Produk ini nantinya akan mencakup ruang lingkup teknologi J2EE untuk *server*-nya, J2SE untuk aplikasi desktop dan J2ME untuk aplikasi mobile serta MySQL untuk database. Aplikasi ini akan dinamakan Aplikasi Pencarian dan Pemesanan Mobil atau disingkat *AppM*.

3. Perancangan Aplikasi AppM

Aplikasi *AppM* ini dibuat untuk dapat diakses oleh masyarakat umum (*client*) melalui *handphone* atau *PDA*, *dealer* dari *dealer-dealer* mobil yang telah terdaftar, dan *admin (server)* dari perusahaan pengembang aplikasi *AppM* ini nantinya.

3.1 Aplikasi AppM pada Desktop (admin)

Hal – hal yang bisa dilakukan oleh *admin* adalah

1. *Admin* memiliki *Username* dan *password* masing-masing.
2. *Admin* dapat mengganti *password* masing-masing.
3. *Admin* dapat melihat dan mengubah data pribadi atau profilnya masing-masing
4. *Admin* dapat melihat, menambah dan mengubah data pribadi atau profil dari *User* lain.
5. *Admin* dapat melihat, menambah dan mengubah data dealer yang ada.

3.2 Aplikasi AppM pada Desktop (dealer)

Hal – hal yang bisa dilakukan oleh *dealer* adalah

1. *Dealer* memiliki *Username* dan *password* masing-masing.
2. *Dealer* dapat mengganti *password* masing-masing.
3. *Dealer* dapat melihat dan mengubah data pribadi atau profilnya masing-masing
4. *Dealer* dapat melihat, menambah dan mengubah data mobil yang dimilikinya masing-masing.
5. *Dealer* dapat membatalkan atau menyetujui *request* pemesanan mobil yang masuk.
6. *Dealer* dapat melihat statistic data mobil yang requestnya diterima dilengkapi dengan forecast / perkiraan bulan selanjutnya disertai dengan grafik.

3.3 Aplikasi AppM pada Handphone

Hal – hal yang bisa dilakukan oleh *user* adalah

1. *User* tidak punya hak akses sama sekali di *desktop*.
2. *User* memiliki *Username* dan *password* masing-masing.
3. *User* dapat melakukan pencarian mobil berdasarkan 5 jenis data yaitu merk, jenis, type, harga, dan keterangan. Keterangan disini ditulis dengan semua hal mengenai mobil.
4. *User* dapat melakukan registrasi.

5. *User* yang telah melakukan registrasi dapat melakukan pencarian dan pemesanan mobil ke dealer yang telah dicari data mobilnya.

3.4 Forecast dengan menggunakan Lagrange Method

Untuk proses pembuatan *forecast* diambil dengan menggunakan method *lagrange*. Metode *Lagrange* merupakan salah satu metode *interpolasi-extrapolasi* yang digunakan untuk mencari titik-titik dari antara n buah titik $P_1(x_1, y_1)$, $P_2(x_2, y_2)$, $P_3(x_3, y_3)$, ..., $P_n(x_n, y_n)$ dengan menggunakan pendekatan fungsi *polynomial* yang disusun dalam kombinasi deret. Metode *Lagrange* pertama kali ditemukan oleh Joseph Louis Lagrange, seorang ahli fisika matematika asal Perancis, dimana rumus dasarnya diketemukan oleh Leonhard Euler, seorang matematikus dan ahli fisika yang teramat brilian dan ilmuwan dari Swiss yang terkemuka sepanjang masa di abad ke-17 sehingga metode Lagrange sering juga disebut "Euler-Lagrange".

Rumus dari metode Lagrange ini adalah :

$$P(x) = \frac{(X - X_2)(X - X_3) \dots (X - X_n)}{(X_1 - X_2)(X_1 - X_3) \dots (X_1 - X_n)} Y_1 + \frac{(X - X_1)(X - X_3) \dots (X - X_n)}{(X_2 - X_1)(X_2 - X_3) \dots (X_2 - X_n)} Y_2 + \dots + \frac{(X - X_1)(X - X_2) \dots (X - X_{n-1})}{(X_n - X_1)(X_n - X_2) \dots (X_n - X_{n-1})} Y_n$$

dimana :

- n adalah jumlah titik yang diketahui
- $P_i(x_i, y_i)$ adalah titik – titik yang diketahui dimana nilai $i = 1, 2, 3, \dots, n$
- x adalah titik yang akan dicari nilainya

Contoh dari perhitungan metode *Lagrange*, yang dikaitkan dengan aplikasi *AppM* ini adalah:

Misalkan diketahui penjualan mobil pada bulan Maret ada 7 buah, bulan April 3 buah dan Mei ada 5 buah. Maka yang ditanya adalah berapa perkiraan untuk bulan Juni?

- Pertama : Misalkan index bulan Maret = 1 ($x_1 = 1$), bulan April = 2 ($x_2 = 2$), bulan Mei = 3 ($x_3 = 3$) maka untuk bulan Juni mempunyai index 4 ($x = 4$).
- Kedua : Menentukan titik, yaitu bulan Maret mempunyai index = 1 dan nilai penjualan mobilnya = 7 sehingga didapat titik $(x_1, y_1) = (1, 7)$. Untuk bulan April didapat titik $(x_2, y_2) = (2, 3)$ dan bulan Mei $(x_3, y_3) = (3, 5)$.
- Ketiga : Masukkan dalam rumus :

$$P(x) = \frac{(X - X_2)(X - X_3) \dots (X - X_n)}{(X_1 - X_2)(X_1 - X_3) \dots (X_1 - X_n)} Y_1 + \frac{(X - X_1)(X - X_3) \dots (X - X_n)}{(X_2 - X_1)(X_2 - X_3) \dots (X_2 - X_n)} Y_2 + \dots + \frac{(X - X_1)(X - X_2) \dots (X - X_{n-1})}{(X_n - X_1)(X_n - X_2) \dots (X_n - X_{n-1})} Y_n$$

$$P(4) = \frac{(4-2)(4-3)}{(1-2)(1-3)} 7 + \frac{(4-1)(4-3)}{(2-1)(2-3)} 3 + \frac{(4-1)(4-2)}{(3-1)(3-2)} 5$$

$$P(4) = \frac{(2) \cdot (1)}{(-1) \cdot (-2)} 7 + \frac{(3) \cdot (1)}{(1) \cdot (-1)} 3 + \frac{(3) \cdot (2)}{(2) \cdot (1)} 5$$

$$P(4) = \frac{2}{2} 7 + \frac{3}{-1} 3 + \frac{6}{2} 5$$

$$P(4) = 1 \cdot 7 + (-3) \cdot 3 + 3 \cdot 5$$

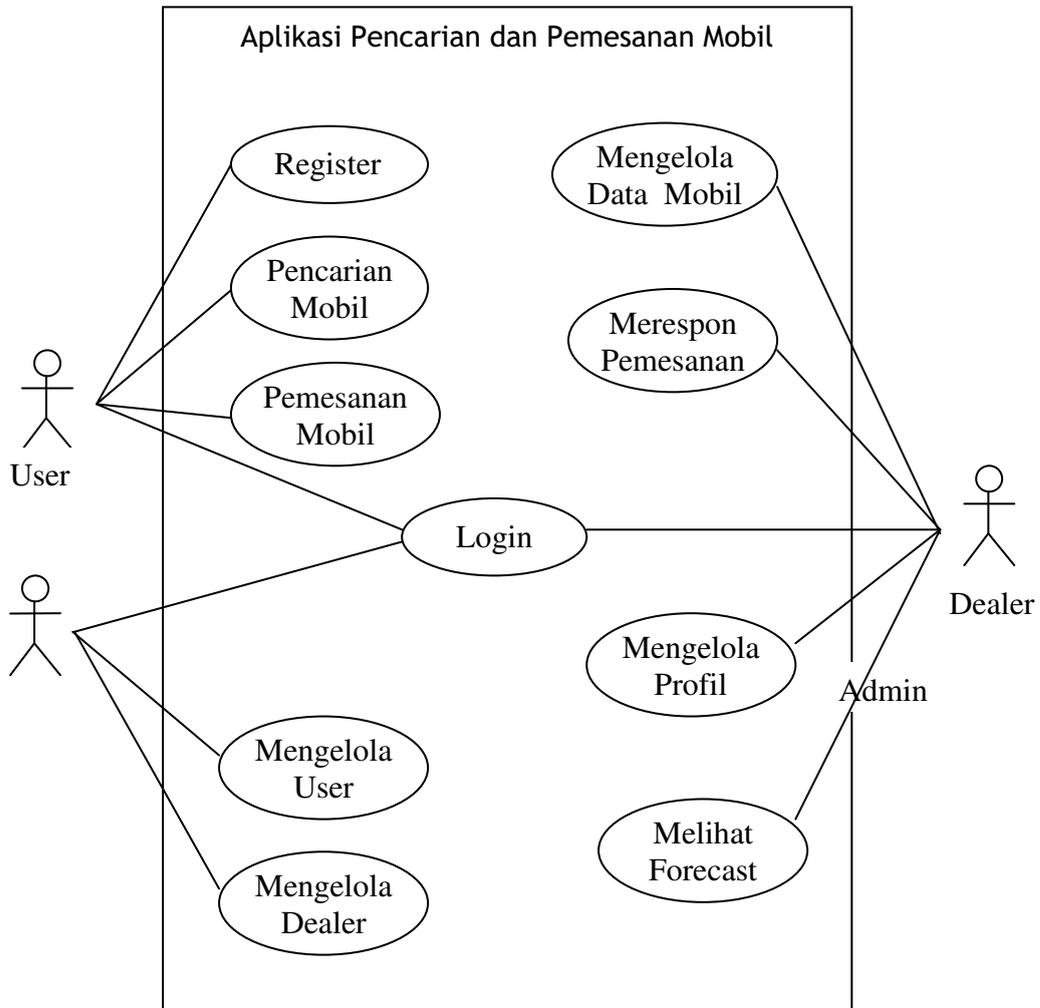
$$P(4) = 7 - 9 + 15$$

$$P(4) = 13$$

Maka perkiraan penjualan mobil di bulan Juni jika menggunakan metode *Lagrange* adalah 13 buah mobil.

Pada aplikasi *AppM* ini, proses pembuatan algoritma *Lagrange* menggunakan *library* yang sudah dibuat. *Library* ini dinamakan *WebCab Functions v2.0 J2SE Component Demo* yang dapat di-*download* dari situsnya yaitu www.webcabcomponents.com. *Library* ini tidak *open-source* sehingga tidak disediakan *source-code*-nya. Walaupun tidak disediakan *source-code* tetapi diberikan *documentation* dan contoh yang ada sehingga bisa mencoba sendiri *library* ini.

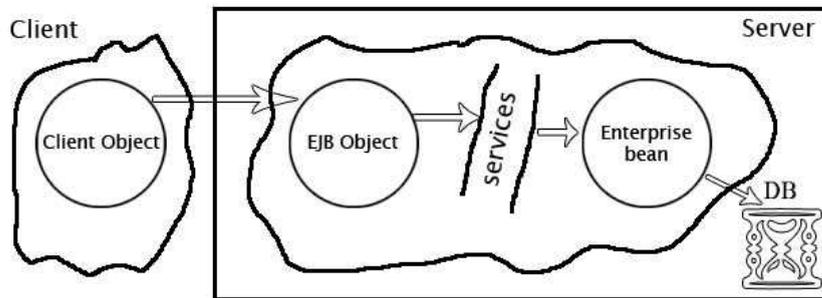
Fungsionalitas aplikasi dapat digambarkan melalui diagram *use case* sebagai berikut :



4. Implementasi aplikasi AppM

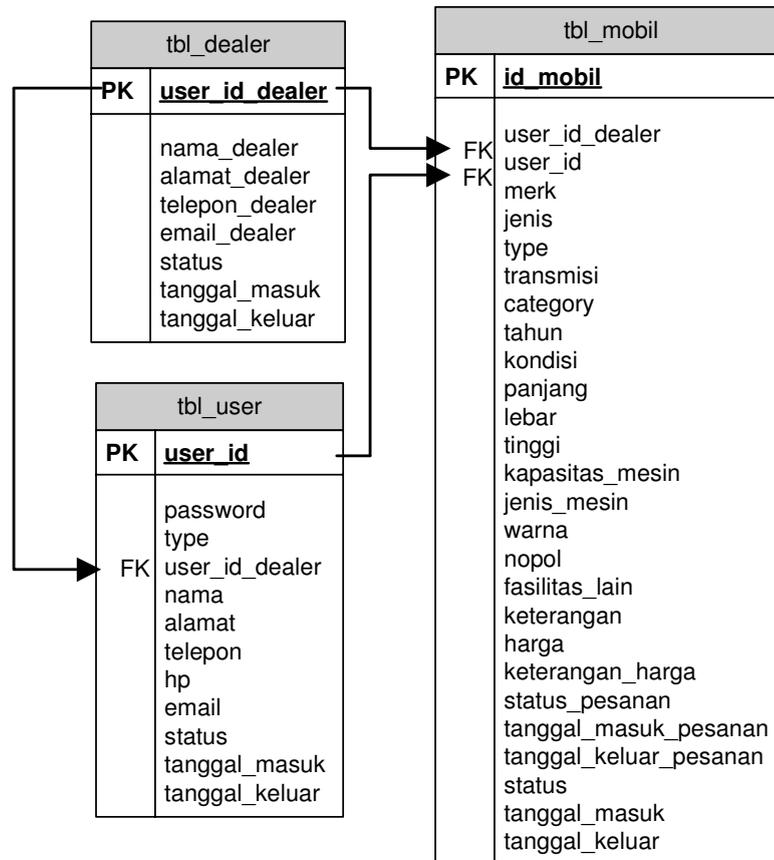
4.1 Komponen EJB pada Aplikasi AppM

Salah satu komponen J2EE yang digunakan adalah EJB singkatan dari *Enterprise Java Bean* yaitu model komponen berbasis kontainer yang dapat didistribusikan dan menangani pengelolaan kontainer pada komponen Java. Komponen Java yang berada di lingkungan EJB disebut bean, ditempatkan pada *server* yang menyediakan daur hidup, pengelolaan dan pelayanan pada dirinya sendiri seperti keamanan, transaksi dan penanganan objek. EJB dapat menjadi salah satu teknologi yang sempurna untuk membuat komponen bisnis yang mendukung sebuah aplikasi enterprise. EJB merupakan komponen *server-side* yang membungkus proses bisnis dari aplikasi.



Gambar 1 Arsitektur EJB

Komponen EJB ini pun masih terbagi menjadi 3 yaitu *Session Bean*, *Entity Bean*, dan *Message Driven Bean*. Masing – masing jenis bean ini mempunyai kegunaanya masing – masing. *Session Bean* digunakan untuk melakukan pengecekan atau perhitungan data yang masuk tetapi tidak berhubungan dengan database. *Entity Bean* digunakan untuk melakukan pengecekan atau perhitungan data yang masuk dan keluar pada database. Sedangkan *Message Driven Bean* digunakan untuk mengirimkan pesan yang tidak membutuhkan balasan secara langsung (*asynchronous*). Pada aplikasi AppM ini, bean yang digunakan adalah *Entity Bean* yang secara langsung berhubungan dengan database yang dibuat pada AppM ini. Database yang dibuat adalah seperti dibawah ini.



Gambar 2 Database Diagram Aplikasi AppM

Dari database diatas dapat diketahui bahwa database pada aplikasi AppM ini terdiri dari 3 table yaitu tbl_dealer, tbl_User, dan tbl_mobil. Tabel dealer digunakan untuk memasukkan data dealer yang ada di Bandung yang terdaftar pada aplikasi ini. Tabel mobil digunakan untuk memasukkan semua data mobil yang ada pada dealer. Field *User_id_dealer* pada tbl_mobil digunakan untuk membedakan mobil dari masing – masing dealer. Sedangkan *User_id* digunakan untuk mengetahui data pesanan / request mobil yang masuk, *User_id* akan bernilai 0 jika tidak terdapat *request* pada mobil yang ditunjuk. Sedangkan tbl_User digunakan untuk memasukkan semua data User yang ada baik itu User sebagai dealer, User sebagai *admin*, dan User sebagai *requester*. Masing – masing User akan dibedakan pada field type. Field *User_id_dealer* digunakan untuk membedakan User dealer dengan User sebagai admin dan requester, *User_id_dealer* akan bernilai 0 jika User bukan dealer sedangkan jika User sebagai dealer maka *User_id_dealer* akan bernilai sama dengan *User_id_dealer* pada tbl_dealer.

Pembuatan bean pada EJB akan disesuaikan dengan banyaknya table pada database dimana dalam kasus ini ada 3 tabel yaitu Dealer, Mobil dan User. Masing – masing bean akan di-package menjadi sebuah package EJB (*JAR Deployment Application*).

4.2 Komponen J2SE pada Aplikasi AppM

J2SE adalah inti dari bahasa pemrograman Java. JDK (Java Developer's Kit) adalah salah satu tool dari J2SE untuk melakukan kompilasi file java yang berekstension *.java. Setelah dikompilasi maka akan dihasilkan file biner dari Java yang berekstension *.class. Dan dengan tool JDK ini pula bisa menjalankan aplikasi berekstension class. Tool J2SE yang digunakan untuk membuat aplikasi AppM ini adalah JDK versi 1.6 yang dapat di-download dari situs Sun yaitu <http://java.sun.com/j2se/>, jdk merupakan tool open source dari Sun yang gratis.

Pada aplikasi AppM untuk desktop digunakan aplikasi Launch4J sehingga didapat file yang berekstension *.exe dengan nama TaMainClient.exe. Sedangkan file untuk di-deploy pada server tidak menggunakan file TaMainClient.exe namun menggunakan file yang berekstension *.jar yaitu TaMainClient.jar.

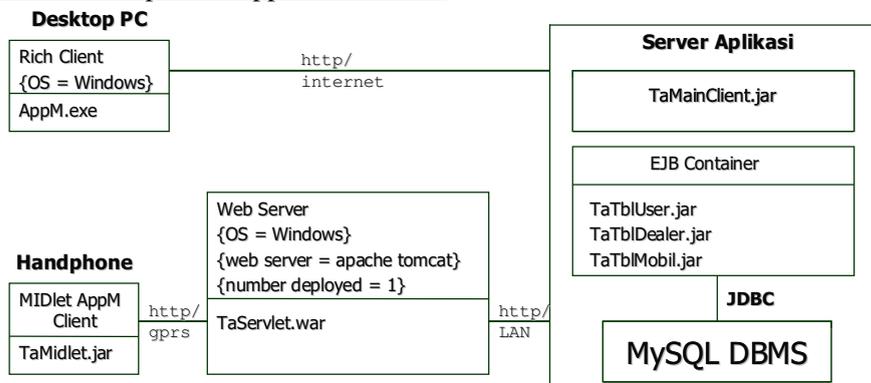
4.3 Komponen J2ME pada Aplikasi AppM di handphone

Jika pada J2SE (*Java 2 Standard Edition*), untuk menjalankan aplikasi Java harus menggunakan interpreter JVM (*Java Virtual Machine*), maka pada platform J2ME mempergunakan KVM (*Kilo Virtual Machine*) yang “ditanam” dalam *handheld devices*. Sebagai informasi bahwa J2EE merupakan *superset* dari J2SE, sebaliknya J2SE merupakan *superset* bagi J2ME. Apa artinya? API yang ada pada J2ME sebagian mengadopsi yang ada pada API J2SE, selain juga mengimplementasikan API spesifik untuk teknologi J2ME itu.

Pada aplikasi AppM ini, aplikasi J2ME atau yang biasa disebut MIDlet akan dibuat 1 class dan di-package menjadi 1 file *execution handphone (JAR application)*.

Sedangkan untuk menghubungkan aplikasi midlet ini dengan server tidak langsung dihubungkan, namun menggunakan *web services* yang merupakan salah satu komponen J2EE juga yaitu Servlet. File servlet pada aplikasi AppM ini menggunakan beberapa class dan di-package juga menjadi 1 file *web deployment (WAR web application)*.

Proses detail aplikasi AppM ini adalah :



Gambar 3 Deployment Diagram AppM

4.4 Tampilan Aplikasi Pencarian Mobil pada *handphone*



Gambar 4 Gambar Pencarian data mobil

Misalkan dari gambar diatas akan mencari mobil dengan merk suzuki, maka ketikkan suzuki pada field merk lalu pilih 'Search' pada menu pilihan sehingga akan menghasilkan gambar seperti dibawah.



Gambar 5 Gambar Hasil Pencarian data mobil

Gambar diatas merupakan hasil pencarian dari data mobil dengan merk 'suzuki'. Pencarian diatas secara *default* melakukan ranking berdasarkan *index* pada

database. Dan misalkan memilih mobil Suzuki Grand Vitara maka akan keluar form seperti dibawah ini.



Search Result

ID Mobil: 6

Merk: suzuki

Jenis: grand vitara

Type: std

Transmisi: MANUAL

Kategori: City Car

Tahun: 2007

Kondisi: NEW

Panjang: 3512 mm

Lebar: 1235 mm

Tinggi: 1201 mm

Kapasitas Mesin: 1000 cc

Jenis Mesin: VVT-i

Warna: SILVER

Back ↑ Send Request

Gambar 6 Gambar Hasil data mobil

5. Kesimpulan

Beberapa kesimpulan yang dapat ditarik berdasarkan proses pembangunan aplikasi dan penerapannya di lapangan adalah :

- Aplikasi AppM ini ketika diterapkan dapat membantu mempermudah pencarian informasi dan pemesanan mobil pada dealer–dealer mobil karena dapat dilakukan dimanapun dan kapanpun.
- Dealer mobil mendapatkan jalur promosi baru yang *online* 24/7, tidak tergantung dari waktu buka tokonya.
- Aplikasi AppM ini diimplementasikan menggunakan teknologi J2EE yang didesain untuk memudahkan semua pihak baik *User* untuk mengakses aplikasi maupun programmer sendiri dalam melakukan *develop* dan *maintenance*.
- Aplikasi AppM juga mendapat keuntungan dari *server & container* J2EE seperti *availability*, *scalability*, *security*, dan *maintainability*.

Daftar Pustaka

- [Bar04] Barbagallo, Ralph. 2004. “Wireless Game Development in Java™ with MIDP 2.0”.
- [Har03] Hartanto, A.Aditya. 2003. *Java 2 Micro Edition Tingkat Lanjut*. Jakarta : PT. Elex Media Komputindo.
- [Har03] Hartanto, A.Aditya. 2003. *Tip dan Trik Java 2 Micro Edition*. Jakarta : PT. Elex Media Komputindo.

- [Jim06] Jim Farley and William Crawford with Prakash Malani, John G. Norman, and Justin Gehtland. 2006. *Java Enterprise in a Nutshell, 3rd Edition*. O'Reilly Media.
- [Lam04] Lam, Jason. 2004. "J2ME & Gaming".
- [Luk04] Luke Taylor and The JBoss Group. 2004. *Getting Started with JBoss*.
- [Sha06] Shalahuddin dan Rosa. 2006. *J2EE Dalam Aplikasi Enterprise*. Bandung : Penerbit Informatika.
- [Wic02] Wicaksono, Ady. 2002. *Dasar-dasar Pemrograman Java 2*. Jakarta : PT. Elex Media Komputindo.
- [Wic03] Wicaksono, Ady. 2003. *Pemrograman Internet dan XML pada Ponsel dengan MIDlet Java*. Jakarta : PT. Elex Media Komputindo.
- [Wic03] Wicaksono, Ady. 2003. *Pemrograman Wireless dengan Java*. Jakarta : PT. Elex Media Komputindo.
- [Yan04] Yan, Michael Juntao and Sharp, Kevin. 2004. *Developing Scalable Series 40 Applications: A Guide for Java Developers*.