

**RANCANGAN APLIKASI MOBILE UNTUK PENCARIAN
LOKASI BENGKEL TAMBAL BAN DI JL. KALIURANG
MENGUNAKAN TEKNOLOGI J2ME**

NASKAH PUBLIKASI



FERY ARDHIANSYAH

06.11.1045

**JURUSAN TEKNIK INFORMATIKA
SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER
“AMIKOM” YOGYAKARTA**

2010

NASKAH PUBLIKASI

**RANCANGAN APLIKASI MOBILE UNTUK PENCARIAN LOKASI
BENGKEL TAMBAL BAN DI JL. KALIURANG MENGGUNAKAN
TEKNOLOGI J2ME**

Disusun oleh :

FERY ARDHANSYAH
06.11.1045

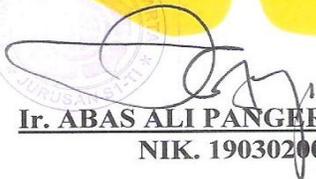
Dosen Pembimbing



HERI SISMORO, S.KOM, M.KOM
NIK. 190302057

Tanggal, 28 Juli 2010

Ketua Jurusan Teknik Informatika



Ir. ABAS ALI PANGERA, M.Kom
NIK. 190302008

**MOBILE APPLICATION DESIGN FOR LOCATION SEARCH TIRE REPAIR SHOP IN JL.
KALIURANG USING J2ME TECHNOLOGY**

**RANCANGAN APLIKASI MOBILE UNTUK PENCARIAN LOKASI BENGKEL TAMBAL BAN
DI JL. KALIURANG MENGGUNAKAN TEKNOLOGI J2ME**

Fery Ardhiansyah
Jurusan Teknik Informatika
STMIK AMIKOM

ABSTRACT

During this time, when we're on the road can happen at any time leaking tire. Which is it going anywhere without bias we expected. That of course we complicate matters when there is an urgent / important. We'll be bothered when I have to find the location nearest tire.

Given the current technological developments, especially in the areas of mobile, then the information can we access when we need it. To help solve that problem, hence made the program "Mobile Application Design for Location Search Tire Repair Shop in Jl. Kaliurang Using J2ME Technology" and created using NetBeans software 6.7.1.

This application will be very helping once in locating the nearest tire repair shop in accordance with the current location we are.

Keyword: Java, J2ME, Mobile Application.

1. PENDAHULUAN

1.1 Latar Belakang Masalah

Teknologi informasi sekarang ini berkembang semakin pesat. Media komunikasi juga telah banyak mengalami kemajuan karena dukungan teknologi informasi yang baik. Sistem informasi tidak hanya dikembangkan pada perangkat keras (*Hardware*), tetapi juga pada perangkat lunak (*software*).

2. LANDASAN TEORI

2.1 Konsep Dasar Pemrograman Java

Java adalah bahasa pemrograman berorientasi objek yang dapat dijalankan pada beberapa *Platform* Sistem Operasi. Java tidak hanya terfokus pada satu Sistem Operasi, tetapi juga banyak dikembangkan pada berbagai sistem operasi yang bersifat *open source*. Java dapat digunakan untuk membuat semua aplikasi web, desktop, ponsel, dan lainnya, sebagaimana dibuat dengan menggunakan bahasa pemrograman lainnya.

2.2 Pengenalan Java

File program sumber java, resminya disebut unit kompilasi (*compilation unit*), adalah suatu file teks yang berisi satu atau lebih definisi class. Compiler java memerlukan file-file ini untuk disimpan dalam file dengan akhiran atau ekstensi .java. Ketika program sumber java di-*compile*, setiap class diletakkan pada file keluaran masing-masing dengan akhiran atau ekstensi .class. Karena tidak adanya fungsi pada java, maka yang ada pada file sumber java hanyalah satu atau lebih definisi class.

2.2.1 Sejarah Perkembangan Java

Java dikembangkan oleh Sun Microsystem pada agustus 1991, dengan nama semula **Oak**. Katanya Oak adalah pohon semacam jait yang terlihat dari jendela tempat pembuatnya,

James Gosling, bekerja. Ada juga yang mengatakan bahwa Oak adalah singkatan dari “*Object Application Kernel*”, tetapi ada yang menyatakan hal itu muncul setelah nama Oak diberikan. Pada Januari 1995, karena nama Oak dianggap kurang komersial, maka nama Oak diganti menjadi Java.

2.2.2 Kelebihan Java

Kelebihan Java antara lain adalah:

- 1) Sederhana
- 2) OOP (*Object Oriented Programming*)
- 3) Multithreading
- 4) Dinamis
- 5) Aman
- 6) Multiplatform

2.2.3 Edisi Pada Java

Java 2 *platform* tersedia dalam tiga edisi untuk keperluan berbeda berikut :

- 1). Java 2 Standard Edition (J2SE)
- 2). Java 2 Enterprise Edition (J2EE)

- 3). Java 2 Micro Edition (J2ME)

2.3 Aplikasi Mobile

Aplikasi mobile adalah sebuah aplikasi yang memungkinkan anda melakukan mobilitas dengan menggunakan perlengkapan atau peralatan elektronik seperti *PDA* atau *Smartphone*. Dengan menggunakan aplikasi mobile, anda dapat dengan mudah melakukan berbagai macam aktifitas seperti hiburan, belajar, browsing, chatting, dan lain sebagainya.

2.4 Perangkat Lunak yang digunakan

2.4.1 Java

Java menurut definisi dari Sun adalah sekumpulan teknologi untuk membuat dan menjalankan perangkat lunak pada computer ataupun pada lingkungan jaringan. Java2 adalah generasi kedua dari Java *platform* (generasi awalnya adalah *Java Development Kit*). Java berdiri diatas sebuah mesin yang diberi nama *Java Virtual Machine* (JVM). Oleh karena itu, bahasa Java disebut sebagai bahasa pemrograman yang *portable* karena dapat dijalankan pada berbagai system operasi, asalkan pada system operasi tersebut terdapat JVM.

2.4.2 Java2 Micro Edition (J2ME)

J2ME (*java to micro edition*) adalah suatu set spesifikasi dan teknologi yang focus kepada perangkat konsumen. Perangkat ini memiliki jumlah memori yang sangat terbatas, menghabiskan sedikit daya dari baterai, layar yang kecil dan bandwidth yang rendah.

2.4.2.1 Connected Limited Device Configuration (CLDC)

CLDC adalah perangkat dasar dari J2ME, spesifikasi dasar berupa *library* dan API yang diimplementasikan pada J2ME. Perangkat tersebut dibatasi dengan keterbatasan memori, sumber daya, dan kemampuan memproses. Spesifikasi CLDC pada J2ME adalah spesifikasi minimal dari *package*, kelas, dan sebagian fungsi dari *Java Virtual Machine*.

2.4.2.2 Connected Device Configuration (CDC)

CDC adalah spesifikasi dari konfigurasi J2ME. CDC merupakan komunitas proses pada java yang memiliki standarisasi. CDC terdiri dari *virtual machine* dan kumpulan *library* dasar untuk dipergunakan pada *profile* industry. Implementasi CDC pada J2ME adalah *source code* yang menyediakan sambungan dengan macam-macam *platform*.

2.4.2.3 Mobile Information Device Profile (MIDP)

MIDP adalah spesifikasi untuk sebuah profile J2ME. MIDP memiliki lapisan diatas CLDC, API tambahan untuk mempertahankan aplikasi, *interface*, jaringan, dan penyimpanan. Pada saat ini terdapat MIDP 1.0 dan MIDP 2.0. Fitur tambahan MIDP 2.0 dibandingkan MIDP 1.0 adalah API untuk multimedia. Pada MIDP 2.0 terdapat dukungan memainkan *tone*, *tone sequence*, dan *file WAV*.

2.4.3 Netbeans IDE 6.7.1

Pada tahun 1997 di bawah bimbingan Fakultas Matematika dan Fisika *Charles Univesrity, Praha*, diadakan sebuah proyek mahasiswa yang diberi nama *Xelfi*. Proyek inilah yang menjadi awal perkembangan *Netbaens*. Sebuah perusahaan kemudian dibentuk untuk proyek tersebut dan menghasilkan versi komersial *IDE (Integrated Development Enterprise) Netbeans*. Namun pada tahun 1999 *Sun Microsystem* membeli proyek ini kemudian menjadi *Netbeans Open Source* pada bulan juni tahun 2000.

2.5 Elemen-Elemen Sistem

Hubungan antara elemen-elemen dalam sistem dapat dilihat pada gambar 2.1 dibawah.

2.6 UML (Unified Modelling Language)

Metode perancangan yang dikembangkan untuk membangun aplikasi pencarian lokasi bengkel tambal ban Berorientasi Objek ini adalah dengan menggunakan *UML (Unified Modelling Language)* yang pada skripsi ini penulis menggunakan beberapa diagram, yaitu *Use Case Diagram*, *Class Diagram* dan *Sequence Diagram*.

2.6.1 Use Case

Use case diagram merupakan diagram yang berkerja dengan cara mendeskripsikan tipikal interaksi antara *user* (pengguna) sebuah sistem dengan suatu sistem tersendiri melalui sebuah cerita bagaimana sebuah sistem dipakai. *Use case diagram* terdiri dari sebuah aktor dan interaksi yang dilakukannya, aktor tersebut dapat berupa manusia, perangkat keras ataupun sistem lain yang berinteraksi dengan sistem.

2.6.2 Class Diagram

Class diagram digunakan untuk melakukan visualisasi struktur kelas- kelas dari suatu sistem dan merupakan tipe diagram yang paling banyak digunakan. *Class diagram* juga dapat memperlihatkan hubungan antar kelas dan penjelasan detail tiap-tiap kelas di dalam model.desain (*logical view*) dari suatu sistem. Selama proses desain, *class diagram* berperan dalam menangkap strukturdari semua kelas yg membentuk arsitektur sistem yg dibuat.

2.6.3 Sequence Diagram

Sequence diagram digunakan untuk menjelaskan interaksi objek yang disusun dalam suatu urutan waktu. Diagram ini secara khusus berasosiasi dengan *use case*. *Sequence diagram* memperlihatkan tahap demi tahap apa yang seharusnya terjadi untuk menghasilkan sesuatu di dalam *use case*. Tipe diagram ini sebaiknya digunakan di awal tahap desain atau analisis karena kesederhanaannya dan mudah untuk dimengerti.

3. ANALISIS DAN PERANCANGAN SISTEM

3.1 Analisis Permasalahan

Saat ini banyak sekali bengkel-bengkel yang ada di jalan kaliurang, namun tidak semua bengkel menyediakan jasa tambal ban. Mereka saat ini lebih merekomendasikan untuk mengganti ban dari pada menambal ban itu, karena proses penggantian jauh lebih cepat daripada menambal ban dan hasil yang mereka dapat juga akan lebih besar.

3.1.1 Identifikasi Masalah

Untuk mendapatkan pemecahan masalah yang akan terjadi diperlukan kegiatan identifikasi terhadap sistem yang saat ini sering dipakai untuk membuat dan menjalankan program/aplikasi pencarian lokasi bengkel tambal ban yang ada di jalan kaliurang yakni tidak tahunya seseorang pada lokasi bengkel tambal ban terdekat dengan kita, sehingga orang bingung untuk mencarinya.

3.1.2 Penyelesaian masalah

Berdasarkan permasalahan dan penyebab masalah diatas, dapat dicarikan beberapa pendekatan untuk menyelesaikan permasalahan diatas. Solusi untuk menyelesaikan permasalahan tersebut diuraikan dalam pendekatan-pendekatan berikut ini:

3.2 Metode Analisis

Metode analisis yang digunakan adalah dengan cara survei dan wawancara langsung dengan bengkel yang bersangkutan. Selama proses penelitian, yang saya tanyakan adalah jam buka bengkel tersebut dan yang terakhir yaitu nomor ponsel bengkel.

3.2.1 Hasil Analisis

Berikut ini adalah daftar bengkel-bengkel yang ada di jl. Kaliurang lengkap dengan posisi bengkel tersebut dalam (km):

1. JMA Motor “km. 4,5”
2. Bengkel Raihan “km. 4,5”
3. MK Motor “km. 5”

3.3 Analisis Perangkat Keras (*Hardware*)

Bagian ini menjelaskan hardware yang digunakan dalam proses perancangan aplikasi pencarian lokasi bengkel tambal ban dan digunakan pada saat kita membutuhkan suatu informasi lokasi terdekat bengkel tambal ban.

3.4 Analisis Perangkat Lunak (*Software*)

Perangkat lunak juga merupakan perangkat yang sangat penting dalam proses pengolahan data, karena perangkat lunak berisikan program yang perintahnya digunakan untuk menjalankan sistem komputer.

3.5 Analisis Kebutuhan

Aplikasi pencarian lokasi bengkel tambal ban adalah sebuah sistem yang akan dibangun dengan beberapa komponen sistem, beberapa kebutuhan tersebut seperti kebutuhan akan data masukan, fungsi-fungsi, keluaran sistem, dan antar muka sistem dan semua komponen akan saling berhubungan sehingga membentuk sebuah aplikasi.

3.6 Kebutuhan Fungsional

Adapun kebutuhan yang akan diimplementasikan dalam pembuatan aplikasi MpitStop adalah sebagai berikut:

Kebutuhan	Keterangan
Form Utama	Pada form utama terdapat text field yang nantinya akan kita masukkan lokasi pengguna saat itu.
Menu OK pada Form Utama	Sistem bisa mengecek data yang dimasukkan pengguna pada text field dan jika data valid maka akan mencari lokasi bengkel tambal ban terdekat.
Detail Form	Menampilkan hasil pencarian lokasi bengkel.
Edit SMS	Menuju form edit sms.

Form Edit SMS	Sistem memberikan pilihan kepada pengguna untuk mengirim sms pada bengkel yang terdekat dan menyesuaikan pesan yang akan dikirimnya.
Kirim SMS	Tombol menu untuk mengirim pesan.
Exit	Untuk keluar aplikasi

Tabel 3.1 tabel kebutuhan fungsional

3.7 Perancangan Sistem

3.7.1 Alur Kerja Sistem

Berikut ini adalah alur kerja sistem yang menggunakan aplikasi *mobile* pencarian lokasi bengkel tambal ban:

Pengguna menentukan lokasi sekarang pada waktu ban motor/mobil anda bocor (dengan melihat baliho atau papan reklame)

Pengguna memasukkan data dalam (km) dan dikalikan dengan (x10)

Pengguna mendapatkan hasil lokasi bengkel dan jarak dari posisi orang itu berada dalam (m).

Pengguna mengedit dan mengirim sms (pesan) selanjutnya memilih satu dari dua bengkel hasil pencarian program tersebut untuk dihubungi.

Pengguna akan menerima pesan balasan dari bengkel yang isinya bengkel tersebut buka atau tidak.

3.7.2 Metode Perancangan

Metode perancangan yang dikembangkan untuk membangun aplikasi pencarian lokasi bengkel tambal ban Berorientasi Objek ini adalah dengan menggunakan *UML (Unified Modelling Language)* yang pada skripsi ini penulis menggunakan beberapa diagram, yaitu *Use Case Diagram*, *Class Diagram* dan *Sequence Diagram*.

3.7.3 Perancangan UML

Untuk lebih menjelaskan aplikasi yang dibangun, digunakan 3 (tiga) model diagram *UML*, yaitu : *use case diagram*, *sequence diagram* dan *class diagram*.

3.7.3.1 Use Case Diagram

Use case diagram merupakan diagram yang berkerja dengan cara mendeskripsikan tipikal interaksi antara *user* (pengguna) sebuah sistem dengan suatu sistem tersendiri melalui

sebuah cerita bagaimana sebuah sistem dipakai. *Use case diagram* terdiri dari sebuah aktor dan interaksi yang dilakukannya, aktor tersebut dapat berupa manusia, perangkat keras ataupun sistem lain yang berinteraksi dengan sistem.

3.7.3.2 Class Diagram

Class diagram digunakan untuk melakukan visualisasi struktur kelas-kelas dari suatu sistem dan merupakan tipe diagram yang paling banyak digunakan. *Class diagram* juga dapat memperlihatkan hubungan antar kelas dan penjelasan detail tiap-tiap kelas di dalam model desain (*logical view*) dari suatu sistem. Selama proses desain, *class diagram* berperan dalam menangkap struktur dari semua kelas yg membentuk arsitektur sistem yg dibuat.

3.7.3.3 Sequence Diagram

Sequence diagram digunakan untuk menjelaskan interaksi objek yang disusun dalam suatu urutan waktu. Diagram ini secara khusus berasosiasi dengan *use case*. *Sequence diagram* memperlihatkan tahap demi tahap apa yang seharusnya terjadi untuk menghasilkan sesuatu di dalam *use case*.

3.8 Kebutuhan Antarmuka (Interface)

Antarmuka pemakai atau *user interface* adalah bagian penghubung antara aplikasi dengan pemakai. Kebutuhan terhadap antarmuka yang diinginkan sebaik mungkin bersifat *user friendly*, artinya pengguna dapat menggunakan perangkat lunak yang dibuat dengan mudah dan nyaman mungkin untuk mendapatkan suatu informasi yang diinginkan oleh user.

4. IMPLEMENTASI DAN PEMBAHASAN

4.1 Implementasi Sistem

Implementasi sistem merupakan tahapan setelah analisis dan perancangan sistem pada rekayasa perangkat lunak yang mana aplikasi siap dioperasikan pada keadaan yang sebenarnya, sehingga dari sini akan diketahui apakah program atau aplikasi yang telah dibuat

benar-benar dapat menghasilkan keluaran yang sesuai dengan tujuan yang diinginkan. Program harus bebas dari kesalahan-kesalahan sebelum program diterapkan dan diimplementasikan. Kesalahan program yang mungkin terjadi adalah kesalahan bahasa (sintaks), kesalahan pada saat program sedang berjalan (runtime) atau kesalahan logika. Setelah program bebas dari kesalahan, program diuji dengan memasukkan data uji untuk diolah.

4.1.1 Pengenalan Netbeans

Tools yang dipakai dalam merancang aplikasi sistem pada “Rancangan Aplikasi Mobile Untuk Mencari Lokasi Bengkel Tambal Ban di Jalan Kaliurang” adalah Netbeans IDE versi 6.7.1 (<http://netbeans.org/>). Pada Netbeans IDE versi 6.7.1 disediakan cara untuk membuat aplikasi mobile, yaitu dengan menggunakan J2ME (Java 2 Micro Edition).

4.2 Implementasi Rancangan

Penulis menggunakan netbeans 6.7.1 karena merupakan salah satu ide java yang paling banyak digunakan oleh para developer. Selain itu netbeans merupakan *open source* dan gratis. Pada website netbeans.org juga didapatkan banyak tutorial untuk menggunakannya sehingga mudah dipahami oleh para *user*.

4.2.1 Perancangan Form

Setelah tahap perancangan aplikasi selesai, tahap selanjutnya adalah pembuatan interface aplikasi dengan menggunakan Netbeans IDE versi 6.7.1. Program aplikasi berfungsi sebagai interface antara user dengan data ataupun hasil yang diinginkan user (sesuai tujuan pembuatan aplikasi). Pada tahapan ini, sebuah software atau aplikasi dibuat berdasarkan analisis dan use case yang telah dibuat.

4.2.2 Pembuatan Form

Pada sub bab ini tidak akan membahas cara pembuatan keseluruhan aplikasi dengan form-form didalamnya, namun hanya akan membahas proses pembuatan Form Utama. Selain

itu, pembuatan form tidak akan dijelaskan secara detail langkah demi langkahnya, namun hanya akan membahas komponen objek, serta beberapa baris code program yang digunakan dalam form aplikasi ini. Proses pembuatan ini dimulai dengan memilih tools tampilan flow pada visual MIDlet yang telah terbuka, kemudian pilih pada komponen palette bagian displayables setelah itu pilih item Form kemudian *drag* pada lembar kerja.

4.3 Pengujian Sistem dan Pembangunan Sistem

Pengujian sistem dapat dilakukan dengan beberapa pendekatan. Pada Laporan ini penulis menggunakan metode pendekatan testing sistem, yaitu metode *Black-Box* dan *White-Box*.

4.3.1 Metode Black-Box

Pengujian black-box merupakan tahap yang berfokus pada pernyataan fungsional perangkat lunak. Test case ini bertujuan untuk menunjukkan fungsi perangkat lunak tentang cara beroperasinya. Apakah pemasukan data telah berjalan sebagai mana mestinya dan apakah informasi yang tersimpan dapat dijaga kemutahirannya.

4.3.2 Metode White-Box

White Box adalah metode pengujian desain test case yang menggunakan *struktur control* desain secara procedural untuk memperoleh *test case*. Dengan menggunakan metode pengujian *white box* pembuat sistem dapat melakukan test case yang memberikan jaminan bahwa semua jalur independent pada suatu modul telah digunakan paling tidak satu kali, menggunakan pada sisi *false* dan *true*, mengeksekusi semua *loop* pada batasan mereka dan pada batas operasional mereka, dan menggunakan struktural data internal untuk menjaga validitasnya.

Test ini ditujukan untuk meramalkan cara kerja perangkat lunak secara detail. Karenanya *jalur logikal* perangkat lunak akan ditest dengan menyediakann test case yang akan mengerjakan kumpulan kondisi dan atau pengulangan secara fisik.

Selain berfungsi sebagaimana dijabarkan di atas, pengujian white box juga dilakukan untuk mengetahui kesalahan-kesalahan yang tidak bisa diatasi oleh sistem ataupun keanehan-keanehan pada hasil/out put dari suatu proses dalam program.

Kesalahan atau keanehan tersebut bisa disebabkan oleh kesalahan dalam logika program, syntax, dan *code* program dimana kesalahan tersebut hanya pembuat program saja yang bisa memecahkan masalah tersebut. User hanya mengetahui out put yang berbeda dengan apa yang diharapkan.

4.3.3 Pembangunan Sistem

Jika aplikasi sudah dapat berjalan dengan baik, maka proses validasi berjalan pada kondisi yang diharapkan, error testing, dan debugging sudah bisa diperbaiki, serta program aplikasi sudah final, maka langkah selanjutnya adalah tahap compiling source code atau tahap pembuatan file .jar dan pembuatan file .jad.

4.4 Instalasi Sistem

Instalasi aplikasi merupakan langkah awal untuk melakukan pengujian sistem. Namun sebelum proses instalasi dilakukan, ada beberapa hal yang perlu dipersiapkan guna proses instalasi, yaitu :

- Handphone yang support platform java MIDP 2.0.

Karena aplikasi ini menggunakan tipe data *float* (bilangan real) untuk perhitungan jarak dan lokasi. Platform MIDP 1.0 tidak mendukung tipe data *float* secara native.

Namun tersedia beberapa library tambahan untuk menghitung bilangan real pada MIDP 1.0. Sehingga masih bisa dikembangkan untuk platform MIDP 1.0.

- Kabel data atau koneksi internet pada Handphone tersebut.
- Pulsa apabila akan menggunakan internet.

Untuk menginstal aplikasi ini pada sebuah perangkat yang mendukung ada dua cara, yaitu:

Offline Installation, yaitu file yang berekstensi jar dari aplikasi tersebut, mPitStop.jar, sudah di copy pada komputer ataupun perangkat lainnya yang

mampu melakukan transfer data kepada perangkat yang akan di instal. Kemudian copy file tersebut pada perangkat yang akan di instal. Buka dan jalankan file tersebut dan proses instalasi telah selesai.

2. Online, yaitu instalasi transfer data melalui internet. Untuk melakukan instalasi dengan cara ini, perangkat yang akan digunakan harus mempunyai koneksi ke internet. Kemudian dengan menggunakan browser bawaan tersebut, kunjungi website yang menyediakan aplikasi mPitStop. Dalam hal ini penulis mencobanya pada URL <http://www.asalketik.com/skripsi> kemudian pilih mPitStop. Maka apabila koneksi ke internet pada perangkat tersebut normal, browser akan menanyakan apakah kita akan mendownload aplikasi tersebut. Kemudian pilih yes atau ok dan tunggu beberapa saat apabila proses download sampai selesai maka akan ada konfirmasi dari browser bahwa aplikasi telah selesai di download. Pada beberapa perangkat juga akan menanyakan di mana kita akan meletakkan aplikasi tersebut. Setelah itu proses instalasi selesai dan bisa di jalankan.

5. KESIMPULAN DAN SARAN

5.1 Kesimpulan

Dari hasil analisis dan perancangan aplikasi MPitStop, untuk memberikan hasil pencarian yang akurat, maka dapat diambil beberapa kesimpulan sebagai berikut:

1. Aplikasi MPitStop memberikan kemudahan disaat orang sedang mengalami kejadian ban bocor dan tidak tahu lokasi bengkel tambal ban terdekat dengan lokasi kita berada sekarang
2. Mempunyai rancangan yang mudah digunakan oleh pengguna
3. Cocok untuk digunakan pada ponsel yang mempunyai spesifikasi kecil
4. Memperoleh hasil pencarian dengan cepat

5.2 Saran

1. Jika mau mengembangkan sistem harus dengan menambahkan fungsi-fungsi tambahan
2. Sebelum menginputkan lokasi, anda sebelumnya harus mengetahui lokasi anda saat ini

DAFTAR PUSTAKA

Al Fatta, Hanif, 2007, *Analisis & Perancangan Sistem Informasi untuk keunggulan bersaing perusahaan dan oraganisasi modern*, Andi Offset, Yogyakarta

Gunadi, Hariman, S.Si., MT & Suhendar, A, S.Si., 2002, *Visual Modeling*

Jogiyanto. 1995. *Analisis dan Desain Sistem Informasi*. Yogyakarta: Andi Offset

Kadir, Abdul, 2003. *Dasar Pemrograman Java 2*. Yogyakarta: Andi Offset

Kristanto, Andri, 2008, *Perancangan Sistem Informasi dan aplikasinya*, Gava Media, Yogyakarta

Menggunakan UML dan Rational Rose, Informatika, Bandung

<http://netbeans.org/features/javame/> diakses tanggal 6 juli 2010

Wicaksono, Andrian, 2010, sejarah perkembangan java
<http://osum.sun.com/profiles/blogs/sejarah-perkembangan-java> diakses tanggal 6 juli 2010