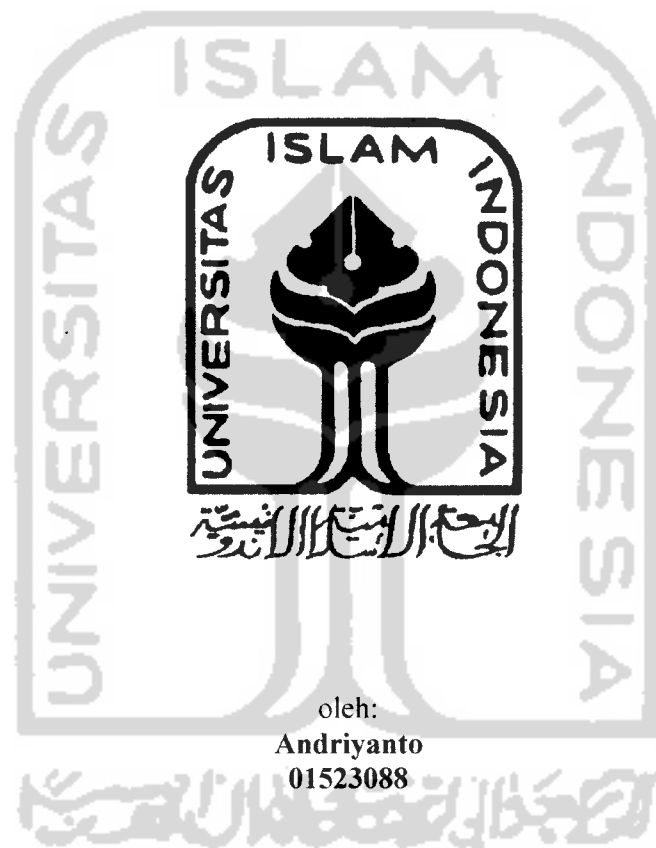


**APLIKASI MOBILE CHEF MENGGUNAKAN J2ME
DAN TEKNOLOGI WEB SERVICE**

TUGAS AKHIR

Diajukan sebagai Salah Satu Syarat
Untuk Memperoleh Gelar Sarjana
Jurusan Teknik Informatika



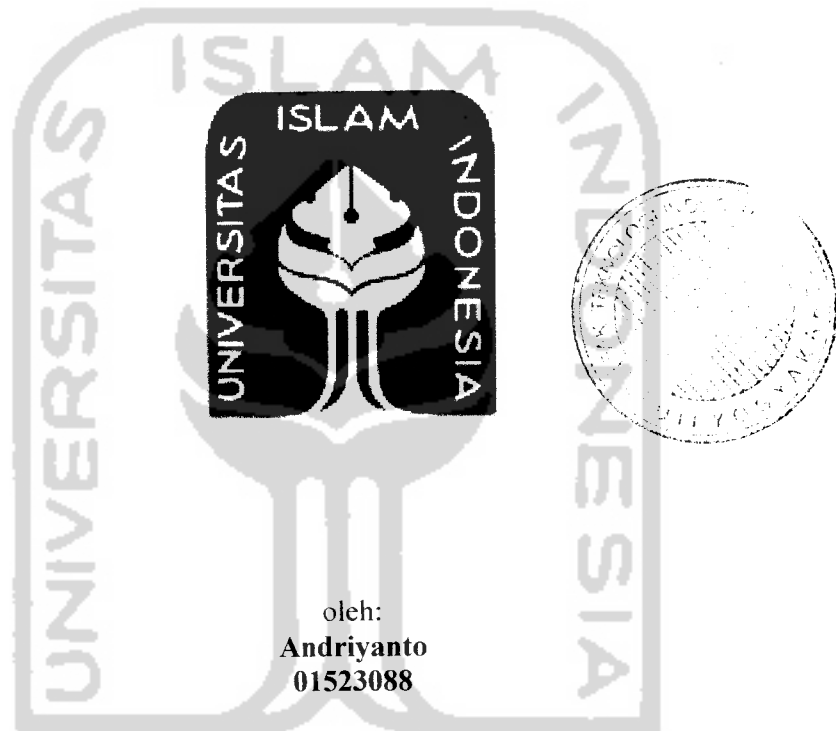
**JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA
YOGYAKARTA**

2007

APLIKASI MOBILE CHEF MENGGUNAKAN J2ME DAN TEKNOLOGI WEB SERVICE

TUGAS AKHIR

Diajukan sebagai Salah Satu Syarat
Untuk Memperoleh Gelar Sarjana
Jurusan Teknik Informatika



oleh:
Andriyanto
01523088

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA
YOGYAKARTA
2007**

LEMBAR PENGESAHAN PEMBIMBING

APLIKASI MOBILE CHEF MENGGUNAKAN J2ME DAN
TEKNOLOGI WEB SERVICE

TUGAS AKHIR



oleh:
Andriyanto
01523088

Yogyakarta, 13 Februari 2007

Pembimbing,

A handwritten signature in black ink, appearing to read 'Taufiq Hidayat', is written over a faint background of Arabic calligraphy.

Taufiq Hidayat, ST., MCS.

LEMBAR PENGESAHAN PENGUJI

APLIKASI MOBILE CHEF MENGGUNAKAN J2ME DAN TEKNOLOGI WEB SERVICE

TUGAS AKHIR

oleh:

Nama : Andriyanto
NIM : 01523088

Telah Dipertahankan di Depan Sidang Penguji sebagai Salah Satu Syarat untuk
Memperoleh Gelar Sarjana Teknik Informatika Fakultas Teknologi Industri
Universitas Islam Indonesia

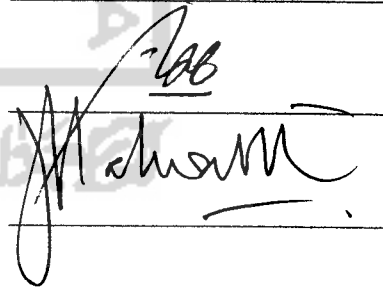
Yogyakarta, 6 Maret 2007

Tim Penguji

Taufiq Hidayat, ST., MCS
Ketua

Yudi Prayudi, S.Si., M.Kom
Anggota I

Affan Mahtarami, S.Kom
Anggota II



Mengetahui,
Ketua Jurusan Teknik Informatika
Fakultas Teknologi Industri
Universitas Islam Indonesia



Yudi Prayudi, S.Si., M.Kom

KATA PENGANTAR

Assalamu 'alaikum warahmatullahi wabarakatuh.

Alhamdulillah rabbil'alam, segala puji hanya bagi Allah, semoga cahayanya selalu menerangi siapapun yang senantiasa meninggikan dan memuliakan asma-Nya. Shalawat serta salam semoga selalu tercurah kepada pembawa risalah mulia, Rasulullah saw, beserta keluarga, dan penerus risalahnya yang setia hingga akhir zaman.

Adalah suatu kelegaan ketika akhirnya saya dapat menyelesaikan bagian akhir dari amanah yang panjang ini dengan begitu banyak kemudahan dan kekuatan yang Allah berikan melalui banyak pihak dengan bantuan, dukungan dan do'a. Untuk itulah, saya sangat ingin menghaturkan terima kasih kepada :

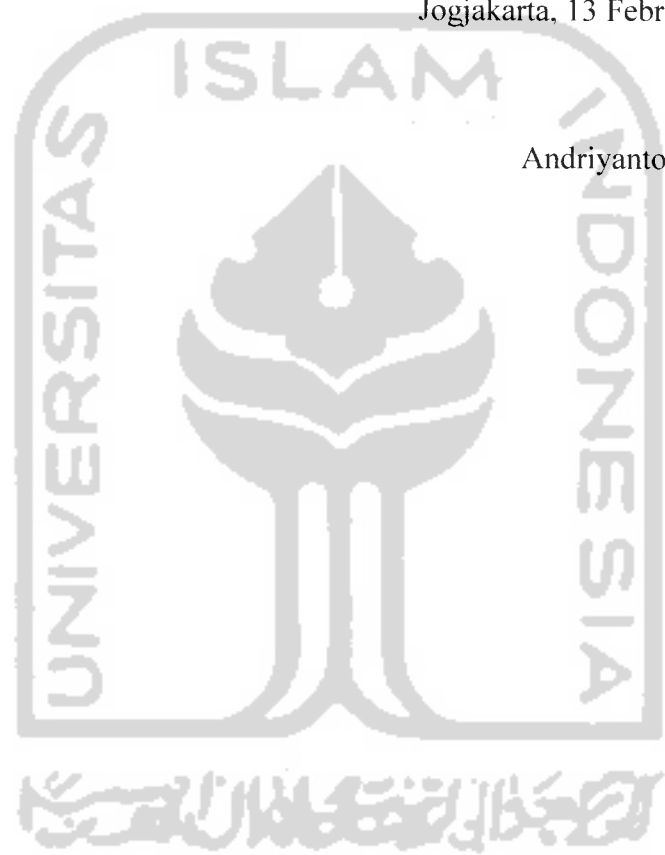
1. Ibu, Bapak dan Adikku, terima kasih untuk seluruh doa dan dukungannya.
2. Bapak Taufiq Hidayat ST, MCS., selaku dosen pembimbing yang telah meluangkan waktu untuk memberi arahan selama penulisan tugas akhir ini.
3. Teman-teman Biner yang selalu memberi dorongan agar saya cepat menyelesaikan laporan.
4. Teman dan saudara-saudaraku yang telah membantu dengan ikhlas.

Saya menyadari sepenuhnya bahwa tugas akhir ini memiliki banyak kekurangan dan masih jauh dari idealisme seorang peneliti maupun karya ilmiah. Untuk itulah saya meminta maaf yang setulus-tulusnya dan menunggu masukan yang berguna untuk perbaikan selanjutnya.

Akhir kata, semoga tugas akhir ini dapat memberi manfaat bagi para pembacanya.

Wassalamu 'alaikum warahmatullahi wabarakatuh.

Jogjakarta, 13 Februari 2006



Andriyanto

HALAMAN PERSEMBAHAN

Bismillahirrahmanirrahiim

Setulusnya,

Karya kecil ini dipersembahkan

Untuk Ibu, Ayah, dan Adikku



HALAMAN MOTTO

Dengan Nama Allah yang Maha Pengasih lagi Maha Penyayang.

(QS.Al-Fatihah [01]: 01)



ABSTRAKSI

Ide untuk membuat Aplikasi Mobile Chef muncul setelah semakin maraknya layanan-layanan *content provider* berbasis SMS yang notabene memberikan manfaat bagi masyarakat. Bahkan dengan harga yang tidak murah untuk ukuran SMS.

Aplikasi Mobile Chef dibuat untuk melestarikan budaya kuliner di Indonesia, yang sangat kaya. Dengan adanya layanan ini, masyarakat Indonesia dapat saling mengenal masakan-masakan dari daerah lain, sebagai awal untuk mengenal budaya masing-masing daerah.

Selain itu, aplikasi ini juga merupakan alternatif lain dalam memberikan layanan pada pengguna *mobile devices* yang murah, dan bermanfaat.



DAFTAR ISI

HALAMAN JUDUL.....	i
LEMBAR PENGESAHAN PEMBIMBING.....	ii
LEMBAR PERNYATAAN KEASLIAN TUGAS AKHIR.....	iii
LEMBAR PENGESAHAN PENGUJI.....	iv
HALAMAN PERSEMBAHAN.....	v
HALAMAN MOTTO.....	vi
KATA PENGANTAR.....	vii
ABSTRAKSI.....	ix
DAFTAR ISI.....	x
DAFTAR TABEL.....	xiii
DAFTAR GAMBAR.....	xiv
BAB I. PENDAHULUAN	
1.1. Latar Belakang.....	1
1.2. Rumusan Masalah.....	2
1.3. Batasan Masalah.....	2
1.4. Tujuan Penelitian.....	3
1.5. Manfaat Penelitian.....	3
1.6. Metodologi Penelitian.....	3
1.7. Sistematika Penulisan.....	3
BAB II. LANDASAN TEORI	
2.1. Gambaran Umum Java.....	6
2.2. Arsitektur Java.....	7
2.3. J2ME (Java 2 Micro Edition).....	8
2.3.1. Konfigurasi J2ME.....	8
2.3.2. J2ME Profile.....	10
2.4. Application Programming Interface.....	11
2.5. MIDlet.....	12

2.5.1.	Arsitektur MIDlet.....	12
2.4.2.	Eksekusi MIDlet	14
2.6.	Web Service.....	15
2.7.	MySQL Database Server.....	17
2.8.	Server Scripting PHP.....	18
2.9.	Unified Modelling Language.....	18
BAB III. ANALISIS KEBUTUHAN PERANGKAT LUNAK		
3.1.	Gambaran Umum Sistem.....	21
3.2.	Metode Analisis.....	21
3.3.	Kebutuhan Perangkat Lunak.....	21
3.4.	Analisis Kebutuhan Masukan pada Client.....	22
3.5.	Kebutuhan Proses pada Client.....	22
3.6.	Analisis Kebutuhan Keluaran pada Client.....	23
3.7.	Analisis Kebutuhan Masukan pada Server.....	23
3.8.	Analisis Kebutuhan Proses pada Server.....	23
3.9.	Kinerja yang Diharapkan.....	24
3.10.	Analisis Awal Fungsional Sistem.....	24
BAB IV. PERANCANGAN PERANGKAT LUNAK		
4.1.	Metode Perancangan.....	37
4.2.	Hasil Perancangan.....	41
4.2.1	Perancangan Aplikasi.....	41
4.2.1.1	Use Case Diagram.....	26
4.2.1.2	Activity Diagram.....	29
4.2.1.3	Class Diagram.....	30
4.2.1.4	Sequence Diagram.....	32
4.3.	Desain Basis Data.....	35
BAB V. IMPLEMENTASI PERANGKAT LUNAK		
5.1.	Pendahuluan.....	36
5.2.	Batasan Impelementasi.....	64
5.3.	Bahasa Pemrograman.....	35

5.4.	Alasan Pemilihan Perangkat Lunak.....	36
5.5.	Implementasi Antar Muka	
5.5.1	Tampilan Menu Utama pada Client.....	37
5.5.2	Tampilan Menu Mencari Resep pada Client.....	38
5.5.3	Tampilan Hasil Cari pada Client.....	39
5.5.4	Tampilan Detail Masakakn pada Client.....	40
5.5.5	Tampilan Menu Menambah Resep pada Client.....	40
5.5.6	Tampilan Menu Utama pada Web Admin.....	41
5.5.7	Tampilan Lihat Semua Data pada Web Admin.....	42
5.5.8	Tampilan Menu Pengisian Data Resep Masakan pada Web Admin.....	42
5.6.	Impementasi Prosedural.....	43
BAB VI. ANALISIS KINERJA PERANGKAT LUNAK		
6.1.	Analisis Proses.....	44
6.2.	Proses Pencarian Data Masakan Melalui Client.....	72
6.3.	Pengujian pada Proses Pengisian Data Resep Masakan.....	45
6.4.	Kekurangan dan Kelebihan Program.....	48

DAFTAR PUSTAKA

LAMPIRAN

Lampiran A :	Prosedur Pencarian Data Resep Masakan pada Client.....	A-1
Lampiran B :	Prosedur Menampilkan Hasil Pencarian pada Client.....	B-1
Lampiran C :	Prosedur Menampilkan Detail Masakan pada Client.....	C-1
Lampiran D :	Prosedur Pengisian Data Resep Masakan pada Client.....	D-1
Lampiran E :	Prosedur Pengisian Data Masakan pada Web Admin.....	E-1
Lampiran F :	Prosedur Penghapusan Data Resep Masakan pada Web Admin.....	F-1

DAFTAR TABEL

Tabel 2.5.1. Status daur hidup MIDlet.....
Tabel 4.13. Tabel Masakan.....35



DAFTAR GAMBAR

Gambar 2.1.	Lingkungan Kerja Teknologi Java.....	8
Gambar 2.2.	Perbedaan CDC dan CLDC.....	10
Gambar 2.3.	Daur Hidup MIDlet.....	14
Gambar 2.4.	Contoh Skema WSDL.....	17
Gambar 2.5.	Arsitektur Sistem UML.....	19
Gambar 4.1.	Elemen Diagram Use Case.....	27
Gambar 4.2.	Use Case Diagram pada Client.....	28
Gambar 4.3.	Use Case Diagram pada Admin.....	29
Gambar 4.4.	Activity Diagram pada Client.....	29
Gambar 4.5.	Activity Diagram pada Admin.....	30
Gambar 4.6.	Class Diagram pada Client.....	31
Gambar 4.7.	Class Diagram pada Admin.....	32
Gambar 4.8.	Sequence Diagram Mencari Resep pada Client.....	33
Gambar 4.9.	Sequence Diagram Mencari Resep pada Admin.....	33
Gambar 4.10.	Sequence Diagram Menambah Data Resep pada Client.....	34
Gambar 4.11.	Sequence Diagram Menambah Data Resep pada Admin.....	34
Gambar 4.12.	Diagram Navigasi Program pada Client.....	35
Gambar 5.1.	Tampilan Menu Utama pada Client.....	37
Gambar 5.2.	Tampilan Mencari Resep.....	38
Gambar 5.3.	Tampilan Hasil Cari Resep.....	38
Gambar 5.4.	Tampilan Detail Masakan pada Client.....	39
Gambar 5.5.	Tampilan Menu Menambah Resep pada Mobile Chef Client.....	40
Gambar 5.6.	Tampilan Menu Utama pada Web Admin.....	40
Gambar 5.7.	Tampilan Lihat Semua Data.....	41
Gambar 5.8.	Tampilan Menu Isi Data Resep Masakan.....	42
Gambar 6.1.	Pengujian pada Proses Pencarian.....	44

Gambar 6.2.	Form List Hasil Cari.....	45
Gambar 6.3.	Form Pengisian Resep Masakan.....	46
Gambar 6.4.	Konfirmasi Pengisian Berhasil Dilakukan.....	46
Gambar 6.5.	Pesan Kesalahan Apabila Konkesi Gagal.....	47
Gambar 6.6.	Tampilan Hasil Pengisian pada Web Admin.....	47



BAB I

PENDAHULUAN

1.1. Latar Belakang Masalah

Mobile Devices khususnya handphone sekarang sudah merupakan barang yang jamak dimiliki oleh setiap orang. Rata-rata *mobile devices* tersebut sudah dapat mendukung aplikasi yang ditulis dengan bahasa pemrograman java. Bahkan sekarang ini, perusahaan-perusahaan penyedia layanan selular dengan bekerja sama dengan bidang usaha lainnya, berlomba-lomba untuk memberikan layanan mobile (content provider), agar para pelanggannya dapat mengakses informasi yang mereka butuhkan.

Khusus untuk aplikasi *mobile devices*, Sun Microsystem mengeluarkan edisi Java yang dinamakan J2ME atau Java 2 Micro Edition. J2ME dirancang untuk dapat menjalankan program Java pada perangkat-perangkat *mobile devices*, seperti handphone. Seperti teknologi Java lainnya, J2ME juga memungkinkan telepon selular mengakses sebuah database server untuk mengambil informasi yang dibutuhkan dari database server tersebut.

Aplikasi Mobile Chef dengan memanfaatkan teknologi Java 2 Micro Edition merupakan suatu aplikasi untuk *mobile devices* yang dapat terhubung dengan database resep-resep masakan. Ketika pengguna memasukkan sebuah bahan masakan, maka aplikasi akan memberikan informasi masakan yang menggunakan bahan dasar seperti yang dimasukkan tersebut, beserta bahan-bahan lain yang dibutuhkan, dan cara membuatnya.

Aplikasi ini umumnya ditujukan kepada pengguna *mobile devices* yang ingin membuat masakan yang lebih variatif kapanpun dan dimanapun juga, selama *mobile devices* dapat memperoleh sinyal untuk mengakses database. Lebih

lanjut lagi, aplikasi ini dapat digunakan oleh ibu-ibu yang seharian sibuk bekerja di luar rumah. Tanpa disadari, seringkali masakan dibuat untuk keluarga ternyata tidak berganti setiap harinya. Hal tersebut melatar belakangi pembuatan aplikasi ini untuk membantu memberikan informasi resep masakan sederhana yang lebih variatif, yang diklasifikasikan berdasarkan bahan dasar yang diperlukan.

1.2. Rumusan Masalah

Permasalahan yang timbul dari tugas akhir ini adalah bagaimana membangun aplikasi Mobile Chef dengan menggunakan teknologi J2ME, yang memungkinkan *mobile devices* dapat mengakses database masakan, yang diklasifikasikan dengan bahan dasar yang diperlukan.

1.3. Batasan Masalah

Untuk menghindari meluasnya materi pembahasan tugas akhir ini, maka penulis membatasi perangkat lunak hanya mencakup hal-hal berikut:

- a. Perancangan berbasis Java untuk mengakses informasi resep masakan dengan menggunakan teknologi J2ME dengan aplikasi web service.
- b. Layanan ini dapat dimanfaatkan oleh seluruh pengguna telepon seluler, sehingga untuk mengaksesnya tidak perlu menjadi anggota.
- c. Pengguna telepon seluler dapat menambahkan resep masakan darinya untuk disimpan dalam database, sehingga dapat dilihat pengguna lainnya.
- d. Disediakan website khusus untuk admin untuk melakukan input, edit, dan hapus data, untuk menjaga agar data tetap dinamis.
- e. Resep-resep masakan merupakan resep-resep sederhana yang diasumsikan memiliki satu bahan dasar.

1.4. Tujuan penelitian

Membangun sebuah aplikasi Mobile Chef dengan teknologi J2ME yang menyediakan informasi mengenai resep masakan, sebagai pilihan lain dari layanan jasa mobile.

1.5. Manfaat Penelitian

Dengan menggunakan layanan dari aplikasi ini, diharapkan pengguna dapat memperkaya pengetahuan kuliner dengan adanya resep-resep masakan, baik yang disediakan oleh layanan melalui admin, maupun resep-resep masakan yang dikirim oleh pengguna lainnya.

1.6. Metodologi Penelitian

Metode yang akan digunakan untuk menyelesaikan tugas akhir ini adalah:

- a. Studi literatur, dengan mempelajari literatur yang mendukung
- b. Pengumpulan data, mengumpulkan data tentang resep-resep masakan, dan mengklasifikkannya.
- c. Analisis kebutuhan masukan dan keluaran sistem.
- d. Perancangan sistem dengan mencari bentuk sistem yang tepat, dan interface yang akan ditampilkan.
- e. Pembangunan sistem dengan bantuan software dan hardware yang mendukung.
- f. Pengujian terhadap sistem yang telah dibangun

1.7. Sistematika Penulisan

Untuk mempermudah pembahasan tugas akhir ini, maka dalam penyusunan, penulis membagi pokok-pokok permasalahan ke dalam tujuh bab sebagai berikut:

BAB I Pendahuluan

Berisi tentang latar belakang diambilnya tema ini sebagai judul tugas akhir, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metodologi penelitian, dan sistematika penulisan laporan.

BAB II Landasan Teori

Bab ini memuat gambaran umum tentang Java dan teknologi J2ME, Web Service, WSDL, SOAP, PHP dan MySQL.

BAB III Analisis Kebutuhan Perangkat Lunak

Memuat tentang analisis permasalahan dan hasil analisis kebutuhan meliputi kebutuhan masukan dan keluaran sistem, antarmuka sistem, dan kinerja yang diharapkan.

BAB IV Perancangan Perangkat Lunak

Memuat tentang metode perancangan perangkat lunak yang dipakai dan hasil perancangan perangkat lunak

BAB V Implementasi Perangkat Lunak

Bagian ini memuat tentang implementasi aplikasi Mobile Chef sesuai dengan analisis kebutuhan dan desain berdasarkan perancangan sistem, serta batasan-batasan yang timbul.

BAB VI Analisis Implementasi Perangkat Lunak

Memuat dokumentasi hasil pengujian terhadap perangkat lunak yang dihasilkan, berkaitan dengan kesesuaian hasil perangkat lunak dengan kebutuhan perangkat lunak tersebut seperti yang telah diuraikan pada bab sebelumnya

BAB VII Penutup

Bab ini menguraikan kesimpulan yang diperoleh dari bab-bab sebelumnya yang telah dilakukan, serta saran-saran yang perlu diperhatikan berdasarkan

keterbatasan-keterbatasan yang ditemukan dan asumsi-asumsi yang dibuat selama melakukan tugas akhir.



LANDASAN TEORI

2.1. Gambaran umum Java

Bahasa Java merupakan karya dari Sun Microsystem Inc. awalnya bernama Oak, yang merupakan bagian dari proyek *Green*, yang dikembangkan khusus untuk memprogram perangkat-perangkat elektronik rumah tangga semacam televisi. Namun pada perkembangannya, bahasa pemrograman Oak ini menjadi bahasa yang bisa digunakan untuk pemrogramana secara umum dan dikenal sebagai bahasa Java saat ini.

Java merupakan bahasa pemrograman yang berorientasi pada objek. Pemrograman Berorientasi Objek, atau sering disebut juga Object Oriented Programming (OOP) merupakan suatu metode implementasi dimana program-program diorganisasikan sebagai kumpulan objek yang saling bekerja sama. java telah berkembang dari tujuan semula menjadi bahasa pemrograman kelas dunia untuk pengembangan aneka ragam aplikasi komputer yang berjalan di beragam perangkat mulai dari handled devices seperti handphone, PDA, sampai aplikasi tersebar skala enterprise di beragam komputer server. Java merupakan bahasa berorientasi objek yang dapat digunakan untuk pengembangan aplikasi mandiri, aplikasi berbasis internet maupun intranet, serta aplikasi untuk perangkat-perangkat cerdas yang dapat berkomunikasi lewat internet/ jaringan komunikasi.

Java adalah bahasa yang dapat dijalankan di manapun dan di sembarang platform apapun, di beragam lingkungan, internet, intranet, costumer electronic products, dan computer application [HAR03A].

2.2. Arsitektur Java

Dari rilis pertamanya, java melakukan hal yang sama, yakni kompiler Java akan mengkompilasi kode program Java menjadi Java *bytecode*. Java *bytecode* ini merupakan instruksi mesin yang dikenali oleh JVM (*Java Virtual Machine*) yang kemudian akan dieksekusi sesuai platform yang bersangkutan.

JVM dalam hal ini posisinya sejajar dengan *interpreter* selayaknya Perl, Python, dan TCL. *Bytecode* merupakan hasil kompilasi kompiler Java (*Java bytecode*), senantiasa mengikuti spesifikasi dan semantik yang didefinisikan dalam JVMMS (*Java Virtual Machine Specification*). Kode Java yang digunakan untuk memprogram akan senantiasa mengikuti spesifikasi dan semantik yang didefinisikan dalam JLS (*Java Language Specification*).

Menyadari akan hal ini, Sun Microsystem membagi arsitektur Java menjadi tiga bagian, yaitu:

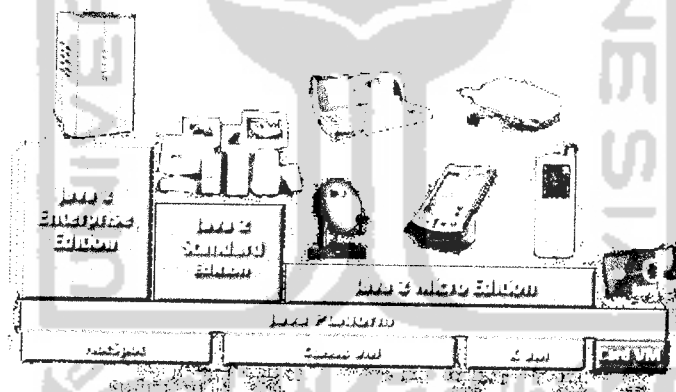
1. Enterprise Java (J2EE) untuk aplikasi berbasis web, aplikasi sistem tersebar dengan beraneka ragam klien dengan kompleksitas yang tinggi. Merupakan superset dari standar Java.
2. Standar Java (J2SE), bagian pertama adalah Java 2 Standard Edition yang berisi Java Runtime Environment (JRE) standar dan Java Development Kit (JDK) standar. Platform ini ditujukan untuk pengembangan program aplikasi Java pada komputer desktop dan aplikasi web.
3. Micro Java (J2ME) merupakan subset dari J2SE.

Dengan semboyannya “write one run everywhere”, portabilitas Java merupakan salah satu kekuatan yang dimiliki oleh Java, yang memungkinkan Java dijalankan pada sistem operasi apapun tanpa perlu melakukan kompilasi ulang program Java yang dibuat [HAR04].

2.3. J2ME (Java 2 Micro Edition)

J2ME dirancang agar dapat dijalankan pada peralatan embeded sistem dan *handheld* yang tidak mendukung secara penuh implementasi menggunakan J2SE, misalnya perangkat dengan kemampuan kecil, seperti handphone, PDA, dan lain-lain. Seperti aplikasi Java umumnya yang menggunakan JVM, dalam J2ME digunakan pula virtual machine yang disebut K virtual machine. K virtual machine adalah virtual machine yang sangat kecil dalam kebutuhan memorinya. Huruf K pada K virtual machine adalah singkatan dari kilobyte, untuk menggambarkan betapa virtual machine ini bekerja pada total memori yang sedemikian kecil mulai dari 128 kilobyte hingga maksimal rata-rata sekitar 512 kilobyte.

Lingkungan Kerja Teknologi Java ditunjukkan pada gambar 2.1



Gambar 2.1 Lingkungan Kerja Teknologi Java

2.3.1. Konfigurasi J2ME

J2ME configuration mendefinisikan lingkungan kerja J2ME runtime. Dikarenakan setiap *mobile devices* memiliki fitur-fitur yang berbeda-beda, J2ME ini dirancang, yakni menyediakan librari standar yang mengimplementasikan fitur standar dari sebuah *mobile devices*. J2ME configuration merupakan JRE (*java Runtime Environtment*) pada *mobile devices*. Java Vritual Machine pada J2ME

berbeda dengan J2ME pada JDK, karena hanya fitur-fitur penting yang berkaitan dengan perangkat *wireless* saja yang diimplementasikan.

Ada dua kategori J2ME configuration, yaitu:

a. Configuration Limited Devices Configuration (CLDC)

Kategori ini umum digunakan untuk aplikasi Java pada handphone semacam Nokia, Samsung Java Phone, Motorola i85s, PDA semacam Palm, Pocket PC, dan Smartphone.

CLDC digunakan untuk mengimplementasikan program Java pada perangkat-perangkat keras dengan ukuran memori yang sangat terbatas, yakni 160-512 Kbytes. Akibatnya, fitur-fitur yang kurang penting untuk diimplementasikan dalam *handheld devices* harus dibuang.

Fitur-fitur yang dibuang tersebut antara lain:

1. Tidak ada dukungan floating point

Kelas-kelas untuk perhitungan *floating point* yaitu **java.lang.Float** dan **java.lang.Double** dibuang dari CLDC

2. Tidak ada dukungan untuk finalisasi objek

Garbage collector yang secara sederhananya digunakan untuk “membersihkan memori” membuang fungsi *finalize* pada kelas **java.lang.Object**, sekalipun fungsi ini sangat penting untuk Java 2.

3. Tidak ada dukungan untuk JNI

Kelas JNI memungkinkan Java mengakses librari yang dibuat dengan bahasa selain Java, namun tidak didukung oleh CLDC

4. Penanganan kesalahan/exception yang terbatas

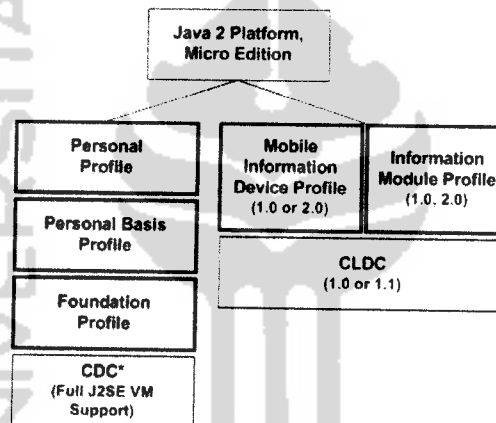
CLDC hanya mendefinisikan tiga kelas berikut untuk penanganan kesalahan/ exception

- a. Kelas `java.lang.Error`
- b. Kelas `java.lang.OutOfMemory`
- c. Kelas `java.lang.VirtualMachineError`

b. Connected Device Configuraton (CDC)

Kategori ini umumnya digunakan untuk aplikasi Java pada perangkat-perangkat *handked devices* dengan ukuran memori paling tidak 2 Mbytes.

Perbedaan CDC dan CLDC dinyatakan oleh gambar sbb:



*CDC 1.0 is based on J2SE 1.3
*CDC 1.1 is based on J2SE 1.4

Gambar 2.2 Perbedaan CDC dan CLDC

2.3.2. J2ME Profile

J2ME Profile menyediakan implementasi-implementasi tambahan yang sangat spesifik dari sebuah *handheld devices*. Seagai analogi, sebuah handphone Nokia/Siemens/Motorola memiliki kemampuan untuk menelpon ke suatu nomor handphone lain karena ini merupakan kemampuan standar dari sebuah handphone. Namun, masing-masing handphone tentulah ada fitur-fitur tersendiri, misalnya Siemens bisa menyimpan file MP3, Nokia memiliki game tersendiri dan lain sebagainya. Kemampuan standar itulah yang diimplementasikan oleh J2ME

Configuration, sedangkan kemampuan/ fitur lain yang sangat bergantung pada jenis perangkat *handheld* yang digunakan akan diimplementasikan oleh J2ME Profile.

Ada lima kategori J2ME Profile saat ini [WIC02], yakni:

a. Mobile Information Device Profile (MIDP)

MIDP menyediakan librari-librari Java untuk implementasi dasar antarmuka (GUI), implementasi jaringan (networking), database, dan timer. MIDP dirancang khusus untuk wireless phone dan pager.

b. Foundation Profile

Profile dasar untuk non-GUI network devices pada CDC.

c. Personal Profile

d. Personal Digital Assistance Profile

e. RMI Profile

2.4. Application Programming Interface (API)

API merupakan bagian yang menentukan dari *mobile devices*. Sebagai tambahan untuk API CLDC, terdapat dua komponen utama API yang berpotensi pada platform J2ME:

a. Mobile Information Device Profile (MIDP) APIs

Sebuah standarisasi dari sejumlah API yang ditentukan oleh Mobile Information Device Profile Expert Group (MIDPG).

b. OEM-specific APIs

Disediakan oleh OEM untuk mengakses fungsi tertentu yang dapat diberikan pada *handheld devices*. Aplikasi yang menggunakan APIs ini mungkin tidak akan cocok bila digunakan oleh *handheld* lain.

2.5. MIDlet

MIDlet adalah aplikasi yang dibuat menggunakan J2ME dan berjalan di perangkat yang mendukung MDP.

MIDlets terdiri atas :

- a. Descriptor (atribut aplikasi)
- b. Java Class
- c. Resources

Penanganan eksekusi aplikasi dilakukan oleh Java Application Manager (JAM). JAM akan mengatur instalasi, eksekusi, manajemen versi, dan perhitungan MIDlets.

Kendali yang diperbolehkan antara lain: *user interface (input/output)*, *persistance storage (simple record-oriented database)*, *networking (HTTP/WAP)*, *timers (Personal Information Management)*.

2.5.1. Arsitektur MIDlet

Paket `javax.microedition.midlet` memiliki kelas MIDlet yang mendefinisikan MIDP dan interaksinya dengan perangkat *handheld* yang bersangkutan. Kelas ini merupakan kelas utama dalam sebuah MIDlet, yaitu semua aplikasi MIDlet harus merupakan turunan dari kelas ini. Kelas ini merupakan kelas abstrak, sehingga kelas turunannya harus mengimplementasikan fungsi-fungsi abstrak di dalamnya. Ada tiga fungsi yang bersifat abstrak, yaitu:

- a. Protected abstract void `startApp()`
- b. Protected abstract void `pauseApp()`
- c. Protected abstract void `destroyApp()`

Fungsi-fungsi yang tidak abstrak ada pada kelas MIDlet bisa digunakan karena otomatis diturunkan dari aplikasi MIDlet, yaitu:

a. Public final void notifyDestroyed()

Digunakan oleh MIDlet untuk memberitahukan ke AMS (Application Management Software) bahwa MIDlet telah dihancurkan dan dibersihkan dari memori.

b. Public final void notifyPaused()

Digunakan oleh MIDlet untuk memberitahukan ke AMS bahwa MIDlet telah dinonaktifkan sementara (paused)

c. Public final String getAppProperty(string key)

Digunakan oleh MIDlet untuk mengambil informasi properti dari AMS sesuai parameter key pada fungsi getAppProperty()

d. Public final void resumeRequest

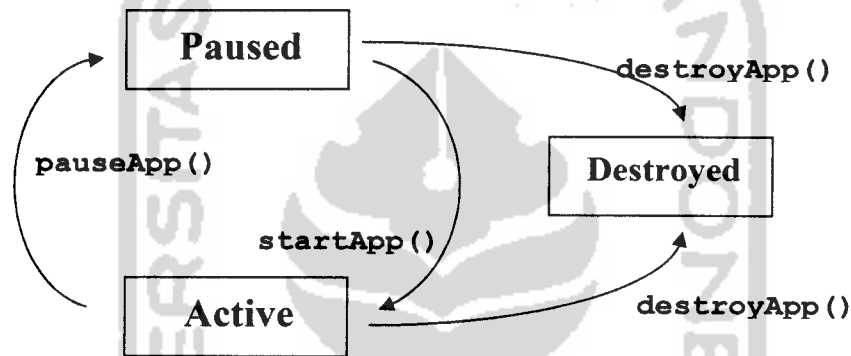
Digunakan oleh MIDlet untuk meminta ke AMS agar diaktifkan kembali

Tabel 2.5.1 Status daur hidup MIDlet

Status	Keterangan
Paused	<p>Status ini terjadi ketika MIDlet selesai diinisialisasi dan tidak melakukan aksi apapun.</p> <p>Pada status ini MIDlet tidak boleh sedang mengunci suatu sumber daya, misalnya sedang melakukan <i>locking</i> atau mengunci suatu file yang menyebabkan proses lain tidak bisa menulis ke file tersebut.</p> <p>Secara garis besar, status MIDlet Paused pada saat:</p> <p>Setelah MIDlet dibuat dengan konstruktor new()</p> <p>Dari status Active, kemudian berhasil memanggil fungsi MIDlet.pauseApp()</p> <p>Dari status Active, kemudian berhasil memanggil fungsi MIDlet.notifyPaused()</p> <p>Dari status Active, namun ketika akan start() terjadi kesalahan berupa exception: MIDlet.StateChangeException()</p>

Active	Status ini terjadi ketika MIDlet sedang aktif/ berjalan dengan normal, yakni setelah memanggil fungsi MIDlet.startApp()
Destroyed	Status ini terjadi ketika MIDlet berhenti berjalan (identik dengan Exit), sehingga seluruh sumber daya yang digunakan akan dibebaskan. Status ini terjadi ketika berhasil dilakukan pemanggilan fungsi MIDlet.destroyApp() atau MIDlet.notifyDestroyed()

Jika digambarkan status daur hidup MIDlet dapat dilihat pada gambar



Gambar 2.3 Daur Hidup MIDlet

2.5.2. Eksekusi MIDlet

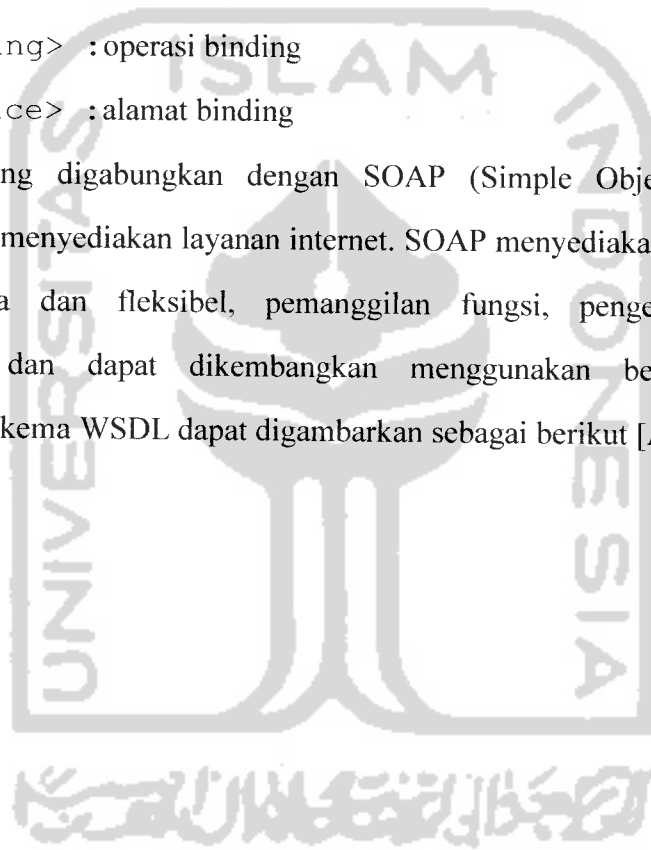
Umumnya urutan eksekusi sebuah MIDlet adalah sebagai berikut

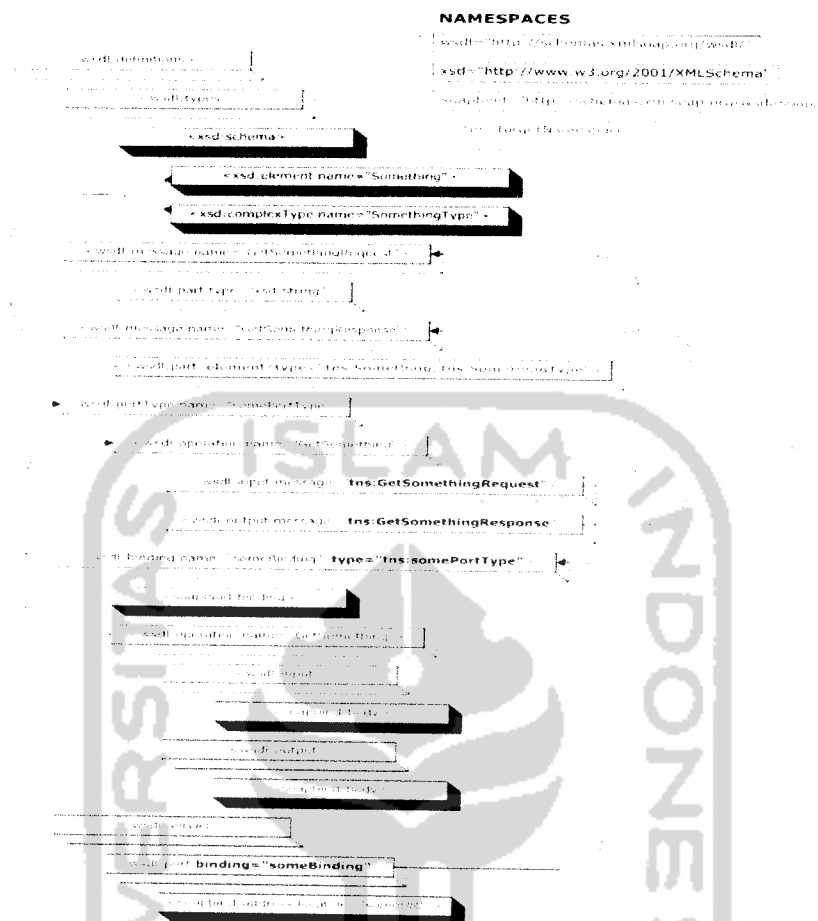
1. AMS menginisialisasi sebuah objek MIDlet, sehingga MIDlet berada pada status *paused*
2. Ketika AMS memutuskan bahwa MIDlet sudah waktunya dijalankan, maka AMS memanggil fungsi MIDlet.start.App() sehingga MIDlet mulai aktif berfungsi

IBM research dan Microsoft. Nilai lebih WSDL adalah mudah dibaca oleh mesin dan manusia karena ditulis dalam bentuk XML. Elemen-elemen yang terdapat dalam WSDL terdiri:

- o <types> : definisi tipe data
- o <message> : parameter operasi
- o <portType>: definisi operasi
- o <binding> : operasi binding
- o <service> : alamat binding

WSDL sering digabungkan dengan SOAP (Simple Object Application Protocol), untuk menyediakan layanan internet. SOAP menyediakan transmisi data yang sederhana dan fleksibel, pemanggilan fungsi, pengembalian nilai, multiplatform, dan dapat dikembangkan menggunakan berbagai bahasa pemrograman. Skema WSDL dapat digambarkan sebagai berikut [AND06]:





Gambar 2.4 Contoh Skema WSDL

File WSDL akan disimpan di web server. Dalam kasus ini WSDL akan diakses oleh *mobile devices*. Di dalam WSDL sudah ditentukan parameter yang diperlukan untuk melakukan request dan response.

2.7. MySQL Database Server

MySQL adalah *Relational Database Management System* (RDBMS) yang didistribusikan secara gratis di bawah lisensi GPL (General Public License), dimana setiap orang bebas untuk menggunakan MySQL, namun tidak boleh dijadikan produk turunan yang bersifat *closed source* atau komersial.

MySQL mengimplementasikan tiga sub bahasa yang ada pada SQL, yaitu *Data Definition Language* (DDL), *Data Manipulation Language* (DML), dan *Data*

Control Language (DCL). DDL berfungsi untuk membuat tabel, mengubah tabel, dan menghapus tabel. DML berfungsi untuk melihat, menambah, menghapus, dan mengubah isi tabel. Sedangkan DCL untuk kepentingan keamanan database, seperti memberikan hak akses ke database dan menghapus hak akses tersebut.

2.8. Server Side Scripting PHP

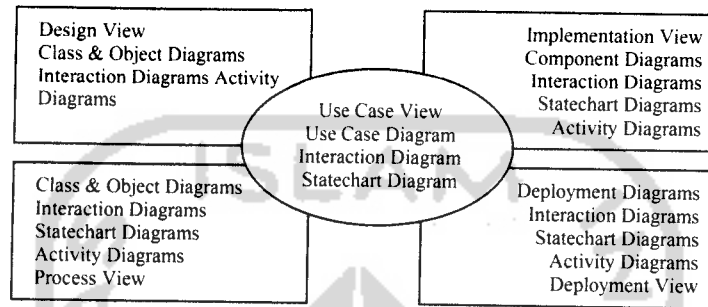
PHP (*Hypertext PreProcessor*) adalah skrip yang bersifat *server-side* yang berarti skrip ini dijalankan di komputer server. Dengan menggunakan PHP, maka maintenance suatu situs web menjadi lebih mudah. Proses update data dapat dilakukan dengan menggunakan aplikasi yang dibuat dengan menggunakan skrip PHP. Kemampuan PHP yang paling diandalkan dan signifikan adalah dukungan kepada banyak database. PHP juga mendukung komunikasi dengan layanan lain yang menggunakan protokol IMAP, SNMP, POP3, HTTP, SOAP dan lain sebagainya.

2.9. Unified Modelling Language

UML merupakan bahasa standar untuk mengekspresikan model berorientasi objek di beragam level abstraksi dan beragam pandangan, mulai level konseptual, sampai level implementasi. Pemoelan sangat memperjelas hal yang diperlukan, dan telah dilakukan pengembang. UML juga dapat menjadi sarana dokumentasi seluruh tahap pengembangan perangkat lunak.

Sebagai salah satu bagian dari metode pembangunan perangkat lunak, UML adalah bahasa yang *process-independent*, yaitu bahasa yang saling bebas terhadap proses yang ada di dalamnya. Oleh sebab itu, UML dapat digunakan dalam semua metode perancangan berorientasi objek.

Arsitektur sistem menurut spesifikasi UML, dapat dideskripsikan dalam lima sudut pandang (view) yang dapat digambarkan sebagai berikut.

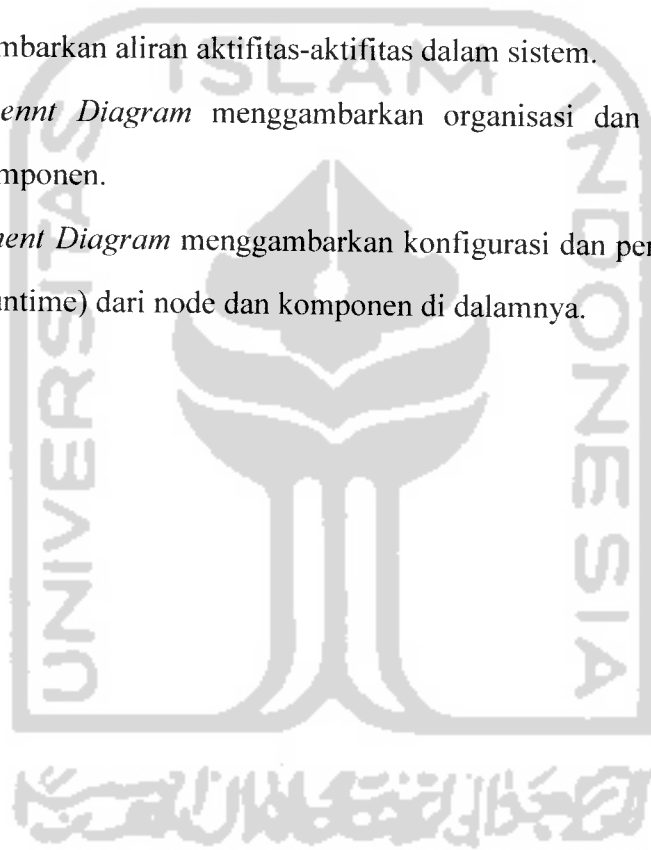


Gambar 2.5 Arsitektur Sistem UML

Kelima sudut pandang diatas dapat dideskripsikan sebagai berikut:

1. *Class & Object Diagram* menggambarkan hubungan antar kelas, antar muka, dan hubungan antar objeknya.
2. *Use Case Diagram* menggambarkan hubungan antar kasus dan aktor (jenis dari kelas)
3. *Interaction Diagram* menggambarkan hubungan antar interaksi dari ekelompok objek. UML menggambarkan interaction diagram dalam diagram-diagram :
 - a. *Sequence Diagram* menggambarkan hubungan antar interaksi dan objek-objek yang ada.
 - b. *Colaboration Diagram* imenggambarkan interaksi diagram yang lebih menekankan pada organisasi struktural dari objek yang menerima dan mengirim pesan

4. *Statechart Diagram* menggambarkan *state machine*, *state machine*, yang terdiri dari keadaan (state), transisi, kejadian (event) dan aktifitas. *Statechart Diagram* memodelkan sifat-sifat dari antarmuka, kelas, atau kolaborasi dengan penekanan pada keteraturan kejadian dari sifat suatu objek.
5. *Activity Diagram* adalah bentuk khusus dari *statechart diagram* yang menggambarkan aliran aktifitas-aktifitas dalam sistem.
6. *Component Diagram* menggambarkan organisasi dan ketergantungan antar komponen.
7. *Deployment Diagram* menggambarkan konfigurasi dan pemrosesan waktu pakai (runtime) dari node dan komponen di dalamnya.



BAB III

ANALISIS KEBUTUHAN PERANGKAT LUNAK

3.1. Gambaran Umum Sistem

Aplikasi MobileChef terdiri dari 2 antarmuka, yaitu:

1. *Mobile devices*, sebagai client untuk memperoleh data masakan berdasarkan bahan dasar yang dimasukkan. Di sini pengguna juga dapat melakukan penambahan resep.
2. *Web*, digunakan oleh administrator untuk melakukan penambahan, pengubahan, atau penghapusan data pada database masakan.

Kedua sistem akan terhubung pada database yang sama.

3.2. Metode Analisis

Tahap analisis digunakan untuk mengetahui dan menterjemahkan semua permasalahan, kebutuhan perangkat lunak, dan kebutuhan sistem yang akan dibangun.

Dengan menggunakan notasi UML dalam visualisasi, merancang dan mendokumentasikan sistem perangkat lunak untuk menggambarkan sebuah sistem perangkat lunak yang terkait dengan objek, sehingga dapat membantu dalam perancangan dan analisis sistem.

3.3. Kebutuhan Perangkat Lunak

Kebutuhan perangkat lunak yang dirancang meliputi:

- Aplikasi dapat dijalankan pada emulator *mobile devices*
- User dapat memperoleh data resep masakan berdasarkan bahan dasar yang dimasukkan.

- User dapat melakukan pengisian data resep masakan melalui emulator *mobile devices*
- Admin dapat melakukan penambahan, pengubahan, dan penghapusan data resep masakan yang ada pada database.

3.4. Analisis Kebutuhan Masukan pada Client

Kebutuhan masukan dari user untuk proses-proses yang ada pada sistem

1. Proses penambahan resep masakan, kebutuhan masukannya:
 - Nama Masakan : nama masakan
 - Bahan Dasar : bahan dasar yang diperlukan oleh masakan
 - Bahan : bahan-bahan lain, yang merupakan pelengkap
 - Bumbu : bumbu yang diperlukan, yang semuanya sudah umum ada di dapur
 - Cara Memasak : cara memasak masakan
2. Proses pencarian data resep masakan, kebutuhan masukannya berupa bahan dasar yang dibutuhkan.

3.5. Kebutuhan Proses pada Client

Untuk kebutuhan proses dapat dijabarkan menjadi beberapa proses, yaitu:

1. Penambahan resep masakan
Pegguna memasukkan data resep masakan yang berupa nama masakan, bahan dasar, bahan, bumbu, cara memasak
2. Pencarian resep masakan
Pegguna dapat mengakses resep masakan yang dibutuhkan dengan memasukkan sebuah bahan dasar.

3.6. Analisis Kebutuhan Keluaran pada Client

Sedangkan keluaran yang dihasilkan pada rancang bangun aplikasi ini adalah:

- Daftar resep masakan yang sesuai dengan bahan dasar yang dimasukkan user pada proses pencarian
- Data lengkap resep masakan, meliputi nomor masakan, nama masakan, bahan dasar, bumbu, dan cara memasaknya.

3.7. Analisis Kebutuhan Masukan pada Server

1. Proses penambahan resep masakan, kebutuhan masukannya:
 - Nama Masakan : nama masakan
 - Bahan Dasar : bahan dasar yang diperlukan oleh masakan
 - Bahan : bahan-bahan lain, yang merupakan pelengkap
 - Bumbu : bumbu yang diperlukan, yang semuanya sudah umum ada di dapur
 - Cara Memasak : cara memasak masakan
3. Proses pencarian data resep masakan, kebutuhan masukannya berupa bahan dasar yang dibutuhkan.

3.8. Analisis Kebutuhan Proses pada Server

1. Penambahan resep masakan

Admin memasukkan data resep masakan yang berupa nama masakan, bahan dasar, bahan, bumbu, cara memasak

2. Pencarian resep masakan

Admin dapat mengakses resep masakan yang dibutuhkan dengan memasukkan sebuah bahan dasar.

3. Proses Penghapusan data masakan

Admin dapat menghapus data resep masakan.

3.9. Kinerja yang diharapkan

Kinerja yang diharapkan dari hasil analisis diatas adalah:

1. Client dapat memberikan layanan berupa pencarian dan penambahan resep masakan melalui *mobile devices*.
2. Admin dapat menjaga data tetap dinamis dengan melakukan penambahan dan penghapusan data resep masakan melalui server.

3.10. Analisis Awal Fungsional Sistem

Ada dua kategori dalam penerapan aplikasi J2ME, yaitu Local Application dan Network Application. Untuk Local Application, aplikasi J2ME menggunakan database RMS yang merupakan media penyimpanan yang menyatu pada *handheld devices*. Local Application tidak perlu terhubung ke server setiap kali akan mengambil data. Pada Network Application, *mobile device* harus terhubung dengan server setiap ingin mengakses data. Sistem ini termasuk Network Application. *Mobile device* adalah sebagai WSDL client, sedangkan server adalah PC (*personal computer*). Client dan server tersebut berkomunikasi melalui *web services*. Setiap akan melakukan request, *mobile devices* membutuhkan hubungan ke server.

BAB IV

PERANCANGAN PERANGKAT LUNAK

4.1 Metode Perancangan

Metode perancangan yang digunakan dalam perancangan aplikasi perangkat lunak adalah menggunakan bahasa UML (Unified Modelling Language) [SCH99]. Tahapan perancangan yang dibahas merupakan perancangan yang akan menghasilkan kebutuhan sistem aplikasi dan pemilihan teknologi, hal ini mencakup iterasi dan perancangan desain.

4.2 Hasil Perancangan

Rancang bangun aplikasi Mobile Chef dengan J2ME meliputi tiga tahap perancangan, yaitu:

1. Perancangan Aplikasi
2. Perancangan Database

4.2.1. Perancangan Aplikasi

UML adalah bahasa grafis yang mendokumentasikan, menspesifikasikan, dan membangun sistem perangkat lunak. UML berorientasi objek, menerapkan banyak level abstraksi, tidak tergantung pada proses pengembangan, tidak tergantung bahasa pemrograman dan teknologi.

UML telah menjadi bahasa standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem piranti perangkat lunak. Karena UML juga menggunakan Class dan Operation dalam konsep dasarnya, maka UML lebih cocok untuk penulisan piranti lunak dalam bahasa pemrograman berorientasi objek, dan telah menjadi standar pemodelan untuk aplikasi berorientasi objek.

Rancang bangun aplikasi Mobile Chef ini terbagi dalam berbagai sudut pandang yang dapat dikategorikan ke dalam diagram berikut:

1. *Use Case Diagram*
2. *Class dan Object Diagram*
3. *Activity Diagram*
4. *Sequence Diagram*

4.2.1.1 Use Case Diagram

Diagram Use Case digunakan untuk mendeskripsikan apa yang seharusnya dilakukan oleh sistem. Diagram use case menyediakan cara mendeskripsikan pandangan eksternal terhadap sistem dan interaksi-interaksinya dengan dunia luar. Dengan cara ini, diagram use case menggantikan diagram konteks pada pendekatan konvensional.

Elemen diagram use case adalah:

1. Aktor

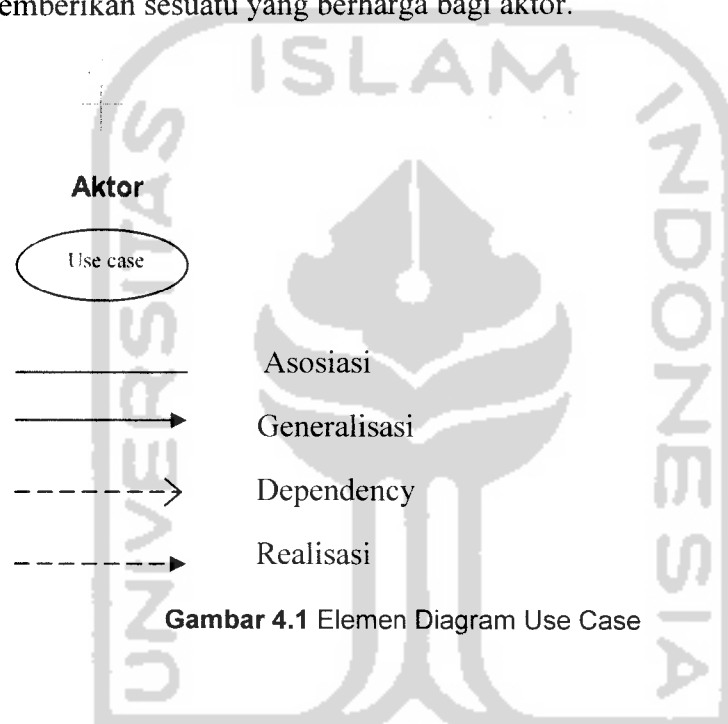
Aktor adalah pemakai sistem, dapat berupa manusia atau sistem terotomatisasi lain. Aktor adalah sesuatu atau seseorang yang berinteraksi dengan sistem, yaitu siapa atau apa yang menggunakan sistem. Yang dimaksud dengan berinteraksi adalah aktor mengirim atau menerima pesan ke atau dari sistem, atau mempertukarkan informasi dengan sistem.

Aktor adalah tipe (kelas). Aktor mempersentasikan peran bukan pemakai individu dari sistem. Use case selaludiawali oleh aktor yang mengirim pesan, disebut dengan stimulus. Ketika satu use case dilakukan, use case dapat mengirim pesan ke satu aktor atau lebih.

2. Use case

Use case adalah cara spesifik penggunaan sistem oleh aktor. Ciri-ciri dari use case adalah:

- Pola perilaku yang harus dipenuhi oleh sistem.
- Sekuen transaksi terhubung yang dilakukan aktor dan sistem.
- Memberikan sesuatu yang berharga bagi aktor.



Gambar 4.1 Elemen Diagram Use Case

Use case menspesifikasikan perilaku sistem atau bagian sistem dan merupakan deskripsi sekumpulan sekuen aksi termasuk varian-varian yang dilakukan sistem untuk memproduksi hasil atau nilai ke aktor. Use case melibatkan interaksi antara aktor-aktor dan sistem. Use case mengemukakan suatu kerja yang tampak. Dari perspektif aktor, use case melakukan sesuatu yang berharga bagi aktor seperti menghitung sesuatu, menghasilkan objek baru, atau mengubah state objek lain.

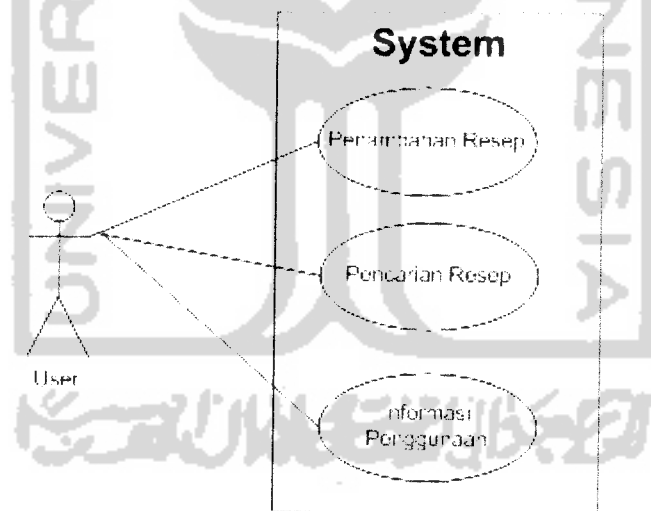
Hasil perancangan diagram Use Case dijelaskan sebagai berikut:

1. Aktor atau pelaku yang terlibat dalam program diagram use case ini dibagi menjadi pengguna dan admin. Dalam diagram ini dijelaskan

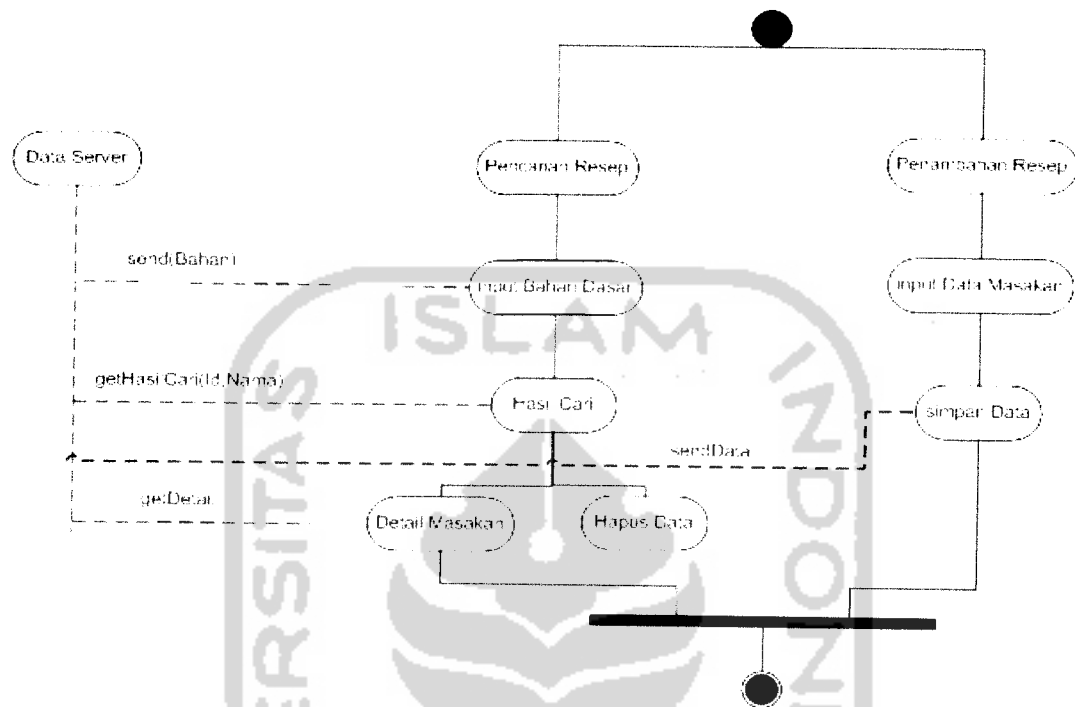
bahwa pengguna dapat melakukan penambahan resep, pencarian resep. Sedangkan pengeditan dan penghapusan data resep hanya dapat dilakukan oleh admin melalui website yang khusus disediakan.

2. Use case pencarian resep akan melakukan proses pencarian data sesuai dengan bahan dasar yang dimasukkan oleh pengguna.
3. Use case penambahan resep akan melakukan penambahan data resep ke database.
4. Use case informasi menyediakan informasi mengenai cara penggunaan aplikasi ini.

Hubungan antara kasus dan aktor didalamnya ditunjukkan oleh gambar 4.2



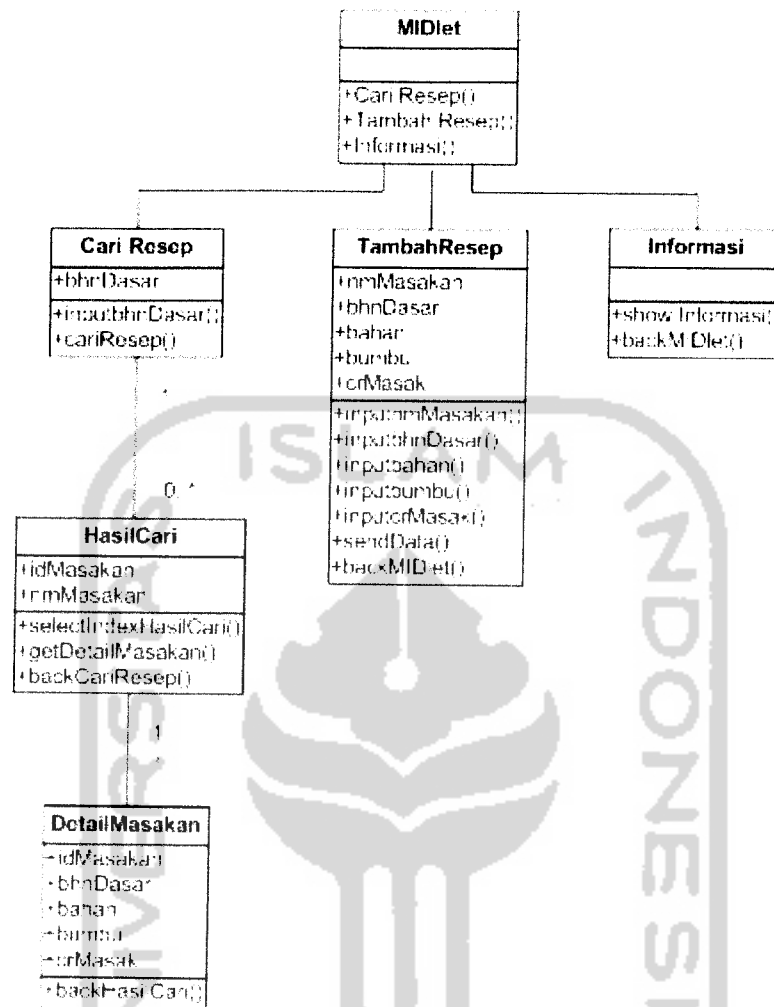
Gambar 4.2 Use Case Diagram pada Client



Gambar 4.5 Activity Diagram pada Admin

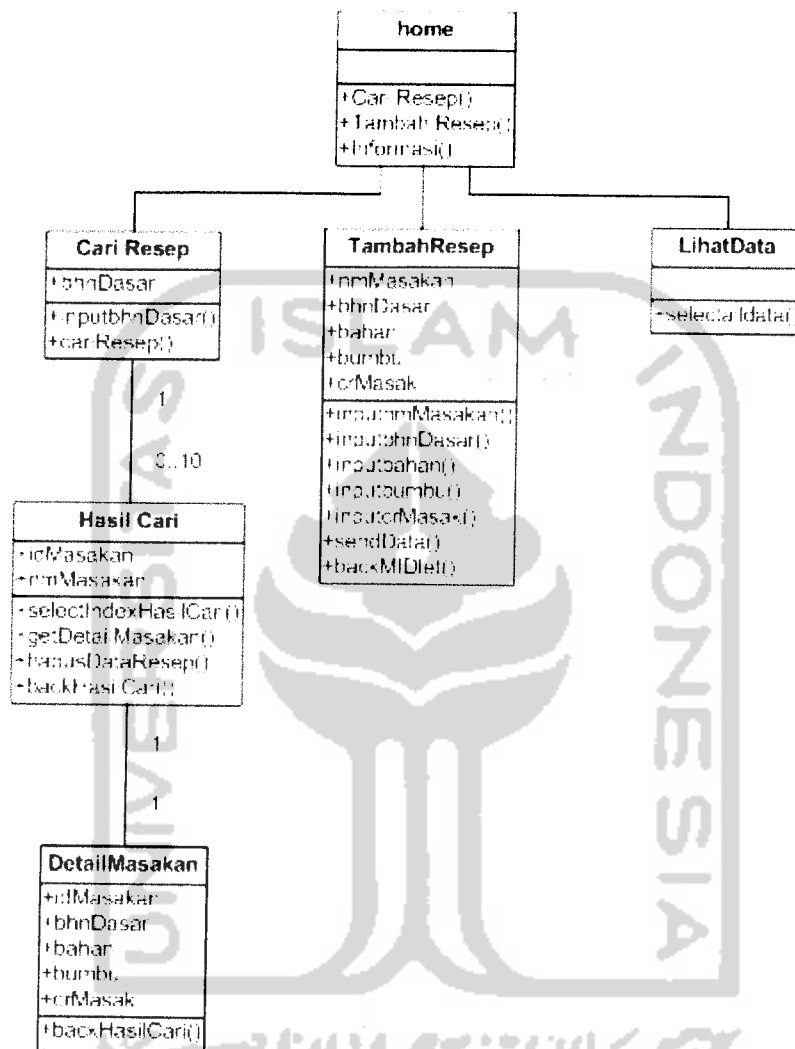
4.3.1.3 Class Diagram

Class diagram yang dibangun dalam perangkat lunak akan dikelompokkan menjadi paket-paket. Diagram class ditunjukkan seperti pada gambar 4.6.



Gambar 4.6 Class Diagram pada Client

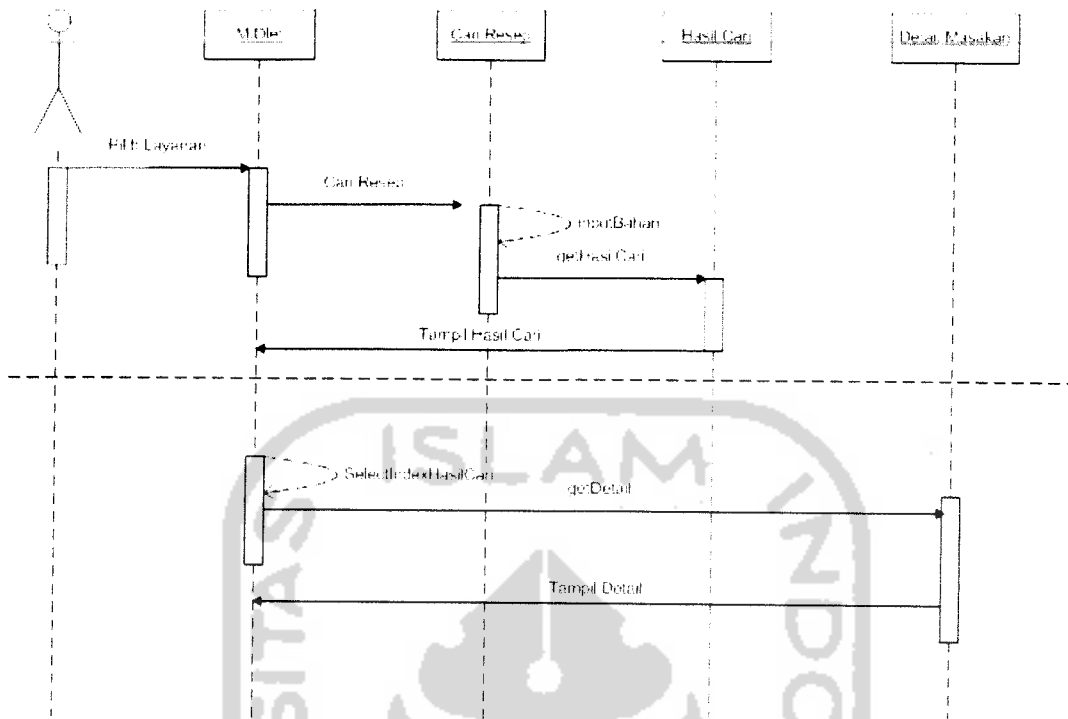
Class Diagram pada admin ditunjukkan pada gambar 4.7



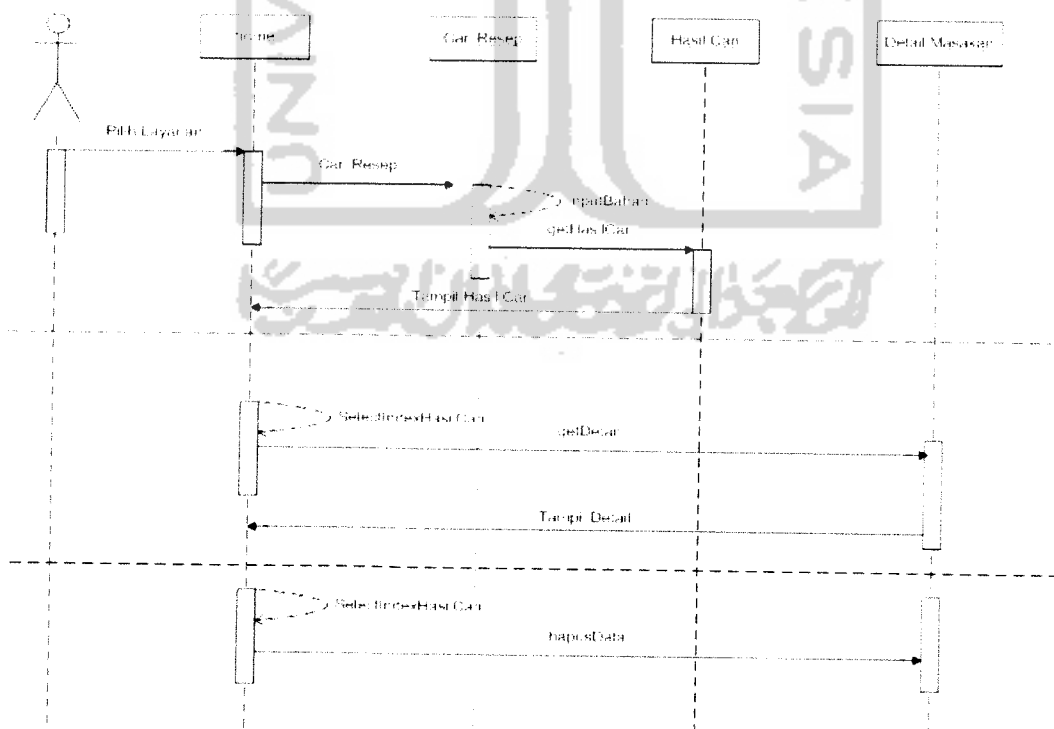
Gambar 4.7 Class Diagram pada Admin

4.2.1.4 Sequence Diagram

Sequence Diagram menggambarkan perilaku sistem secara dinamis dan memperlihatkan interaksi dari objek-objek yang ada pada Aplikasi Mobile Chef, yaitu Sequence Diagram Mencari Resep, dan Sequence Diagram Menambah Resep. Sequence diagram mencari resep digambarkan pada gambar 4.8

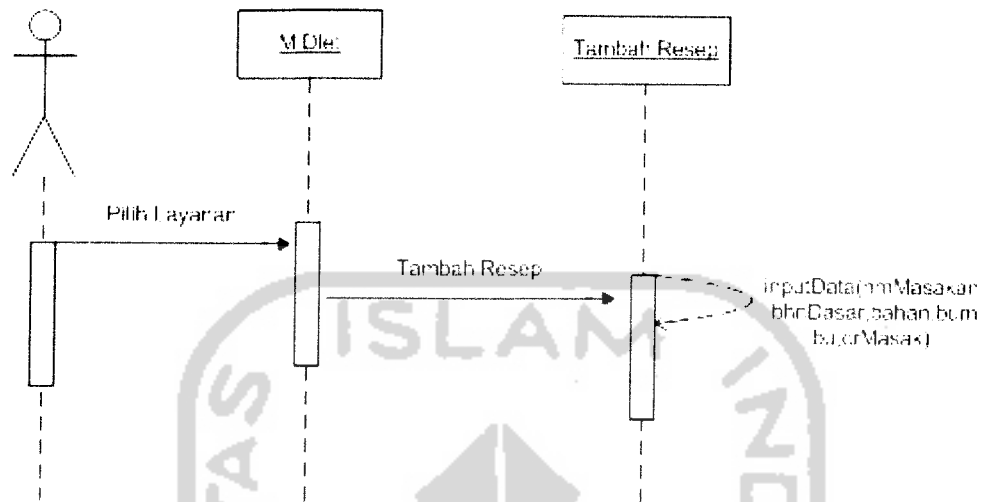


Gambar 4.8 Sequence Diagram Mencari Resep pada Client

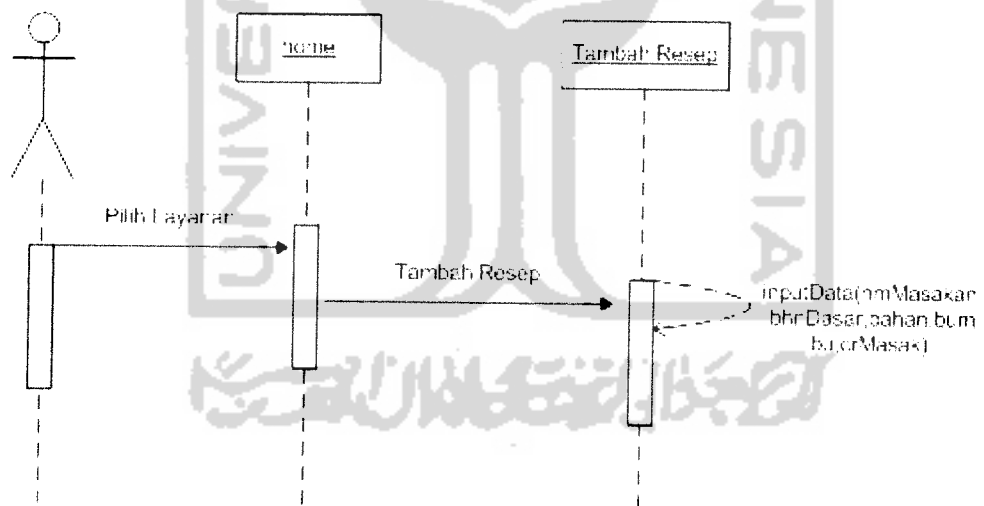


Gambar 4.9 Sequence Diagram Mencari Resep pada Admin

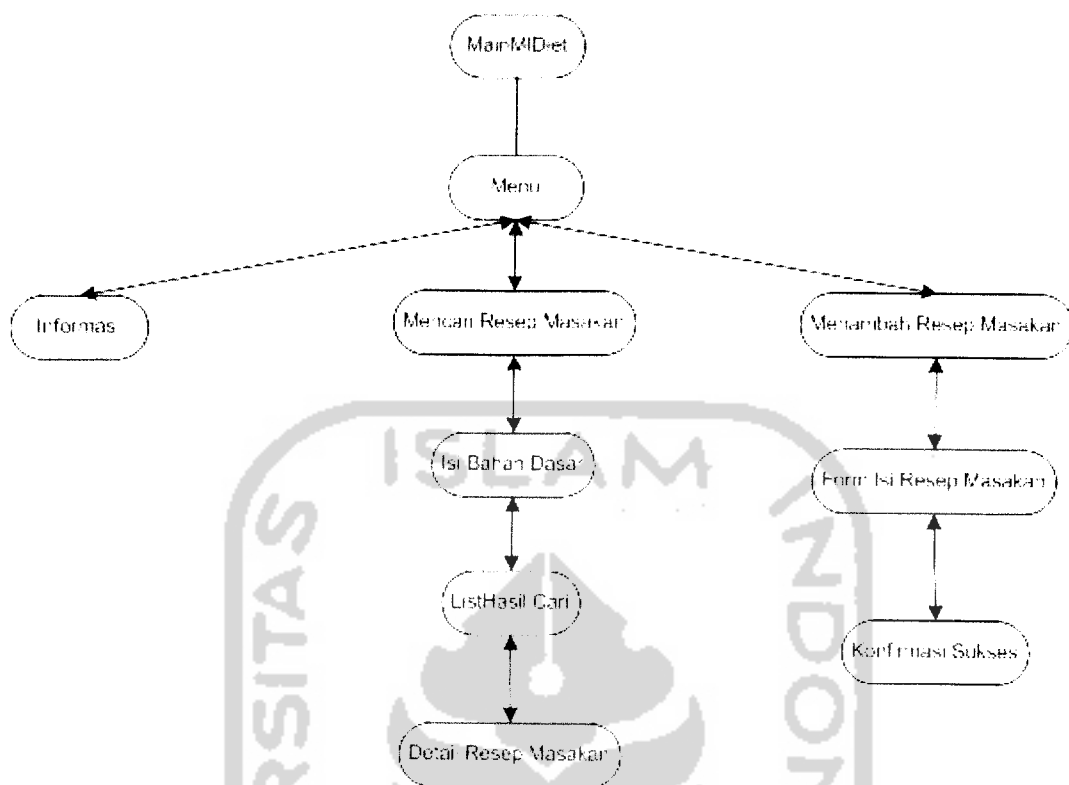
Sequence diagram menambah resep digambarkan pada gambar 4.5



Gambar 4.10 Sequence Diagram Menambah Data Resep Masakan pada Client



Gambar 4.11 Sequence Diagram Menambah Data Resep Masakan pada Admin



Gambar 4.12 Diagram Navigasi Program pada Client

4.3 Desain Basis Data

Aplikasi Mobile Chef menggunakan sebuah tabel basis data, yaitu *tmasakan*. Struktur tabel yang digunakan dapat dilihat pada tabel 4.10

Tabel 4.13 Tabel Masakan

No	Field Name	Type	Attribut	Keterangan
1	id_masakan	Integer (3)	Primary	Auto increment
2	nm_masakan	Text		Nama Resep Masakan
3	bhn_dasar	longtext		Bahan dasar yang digunakan dalam resep
4	bahan1	longtext		Bahan-bahan lain selain bumbu
5	bahan2	longtext		Bumbu yang digunakan dalam resep
6	cara	longtext		Cara memasak resep

BAB V

IMPLEMENTASI PERANGKAT LUNAK

5.1 Pendahuluan

Aplikasi Mobile chef diimplementasikan dengan bahasa pemrograman Java 2 Micro Edition (J2ME) sebagai client, sedangkan untuk aplikasi server digunakan bahasa pemrograman PHP5 dan MySQL. Sebagai perantara antara client dan service digunakan teknologi Web Service. Bab Implementasi Perangkat Lunak ini dibagi menjadi dua bagian, yaitu:

1. Implementasi antar muka yang menunjukkan antarmuka apa saja yang ada di sistem
2. Implementasi Prosedural yang berisi listing program dari Procedure-Procedure penting yang ada pada sistem

5.2 Batasan Implementasi

Batasan Implementasi meliputi batasan minimal untuk perangkat keras (hardware) dan perangkat lunak (software) yang diperlukan agar sistem yang dibuat dapat berjalan dengan baik.

5.3 Bahasa Pemrograman

Bahasa pemrograman yang digunakan untuk implementasi aplikasi Mobile Chef adalah:

1. Pada Client (mobile devices) : J2ME (Java 2 Micro Edition)
2. Pada Server : PHP, MySQL
3. Pada Web Service : XML

5.4 Alasan pemilihan perangkat lunak

Perangkat lunak yang digunakan adalah Java 2 Micro Edition. Alasan digunakannya perangkat lunak ini, karena :

1. Bebas digunakan di sembarang platform
2. Pengiriman dan aplikasi layannanya yang lebih dinamis
3. Lebih menguntungkan dari segi operational cost

Pertimbangan menggunakan web service adalah, standar web service memungkinkan client dan server berkomunikasi melalui HTTP dengan platform atau bahasa pemrograman yang berbeda-beda. Perangkat lunak penunjang yang dibutuhkan untuk pengembangan implementasi perangkat lunak aplikasi Mobile Chef adalah:

1. Sistem Operasi Menggunakan Windows XP
2. Software WAMP5, yang berisi modul-modul PHP5, Apache 2, dan MySQL.
3. Aplikasi J2ME dibuat dengan Netbeans IDE 5.5.
4. Aplikasi web menggunakan DreamWeaver MX
5. JDK versi 1.5
6. Emulator Wireless Toolkit 2.2

5.5 Implementasi antarmuka

5.5.1 Tampilan menu utama pada client

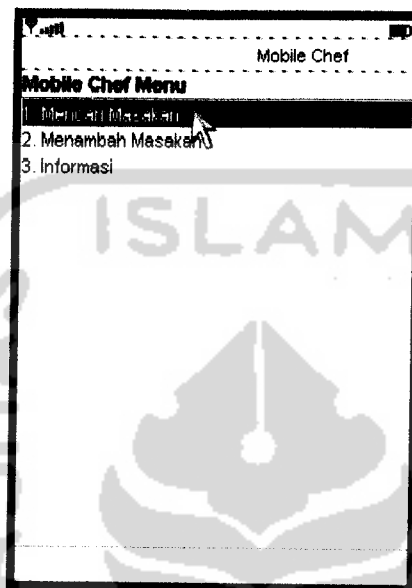
Tampilan ini muncul pertama kali setelah MIDlet dijalankan. Dalam form ini terdiri dari 3 submenu, yaitu:

1. Mencari resep
Menu ini menuju ke form pencarian resep.
2. Mengisi resep
Menu ini dipilih pengguna untuk mengisikan resep ke database

3. Informasi

Menu ini berisi cara penggunaan aplikasi Mobile Chef

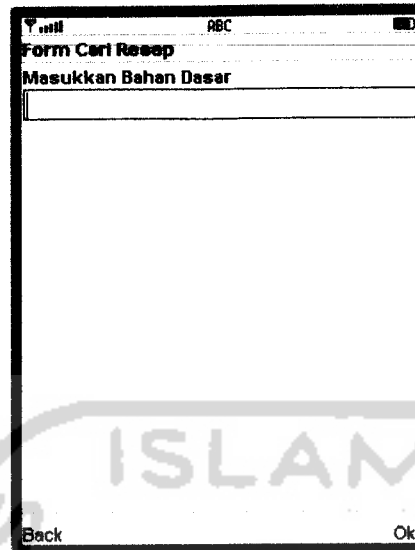
Tampilan menu utama ditunjukkan pada gambar 5.1



Gambar 5.1 Tampilan menu utama pada Mobile Chef client

5.5.2 Tampilan menu mencari resep pada client

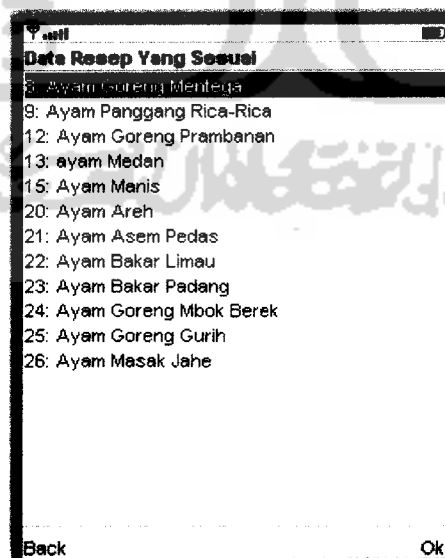
Pada tampilan ini disediakan form dimana pengguna dapat mencari resep dengan memasukkan bahan dasar yang dibutuhkan oleh masakan. Tampilan menu mencari resep ditunjukkan pada gambar 5.2



Gambar 5.2 Tampilan mencari resep

5.5.3 Tampilan hasil cari pada client

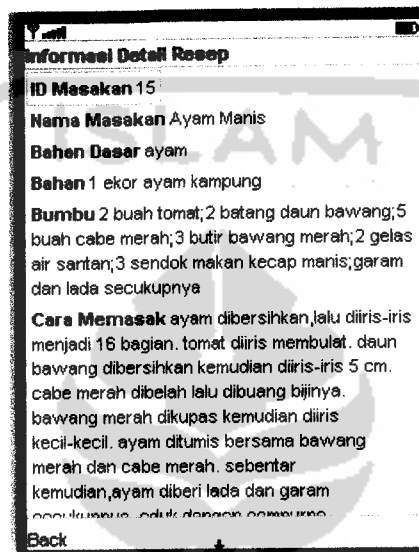
Pada tampilan ini berisi masakan-masakan yang menggunakan bahan dasar sesuai yang diisikan oleh pengguna pada form pencarian sebelumnya. Pengguna dapat memilih resep masakan yang diinginkan, dan melihat informasi detailnya. Tampilan hasil cari resep ditunjukkan pada gambar 5.3



Gambar 5.3 Tampilan hasil cari resep

5.5.4 Tampilan detail masakan

Tampilan ini berisi informasi detail mengenai resep masakan yang meliputi, nama masakan, bahan dasar, bahan, bumbu, dan cara memasaknya. Tampilan detail masakan ditunjukkan seperti pada gambar 5.4



Gambar 5.4 Tampilan detail masakan pada *client*

5.5.5 Tampilan menu menambah resep pada client

Tampilan ini digunakan untuk pengisian resep baru dari pengguna *mobile devices*. Tampilan menu menambah resep ditunjukkan pada gambar 5.5.

Form Menambah Resep

Nama Masakan

Bahan Dasar

Bahan

Bumbu

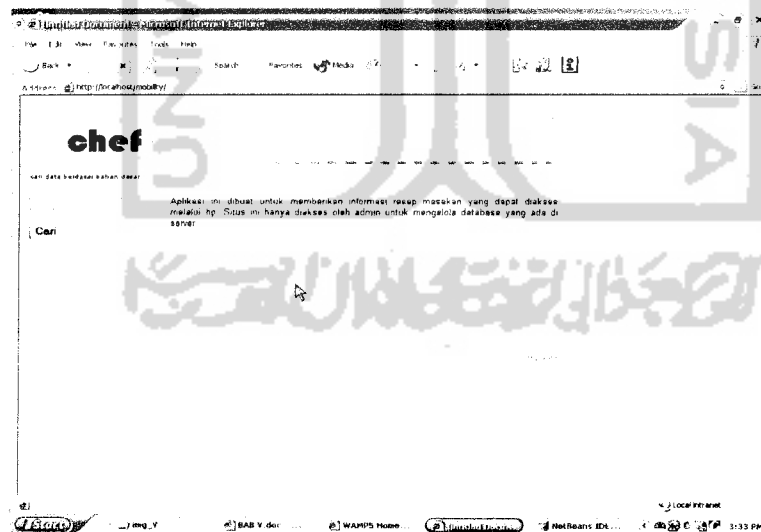
Cara Memasak

Back Ok

Gambar 5.5 Tampilan menu menambah resep pada Mobile Chef client

5.5.6 Tampilan menu utama pada web admin

Gambar 5.6 menunjukkan tampilan menu utama pada web admin.



Gambar 5.6 Tampilan menu utama pada web admin

5.5.7 Menu lihat semua data pada web admin

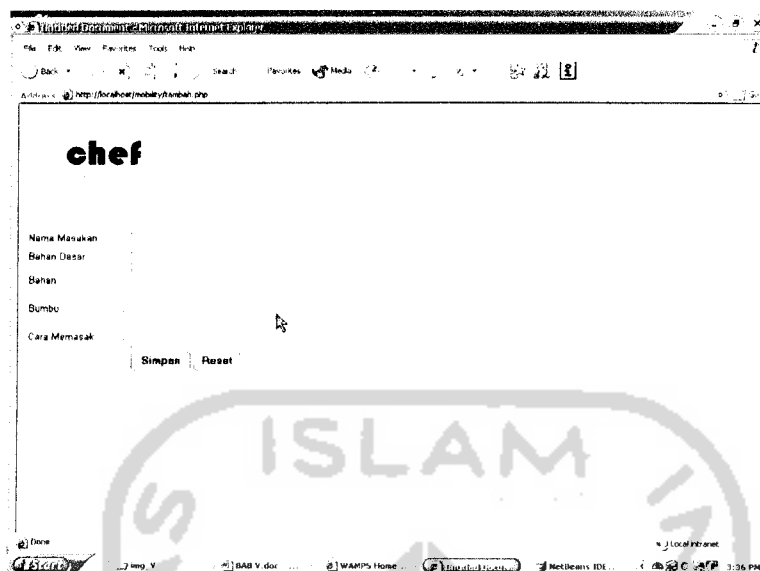
Pada tampilan menu lihat semua data berisi semua data resep yang ada pada database. Proses edit data resep dan penghapusan data resep dapat dilakukan melalui *link* yang tersedia. Tampilan menu lihat semua data seperti pada gambar 5.7

No ID	Nama Masakan	Bahan Dasar
5	Sup Bunga Kol	Kembang Kol
6	Kangkung Belacan	Kangkung
7	Sambal Goreng Buncis	Buncis
8	Ayam Goreng Mentega	Ayam
9	Ayam Panggang Rusa	Ayam
13	ayam Madan	Ayam
12	Ayam Goreng Prambanan	Ayam
14	Ayam Taliar	ayam
15	Ayam Manis	ayam
16	Ayam Jagung	jagung

Gambar 5.7 Tampilan menu lihat semua data

5.5.8 Menu pengisian data resep masakan pada web admin

Dalam tampilan ini disediakan form untuk mengisi data resep baru bagi admin. Tampilan mengisi resep masakan ditunjukkan seperti pada gambar 5.8



Gambar 5.8 Tampilan menu isi data resep masakan

5.6 Implementasi Prosedural

Pada bagian ini dijelaskan implementasi secara prosedural. Setiap prosedur yang ada dalam aplikasi akan dijelaskan beserta kebutuhan masukan dan keluarannya.

1. Prosedur pencarian data resep masakan pada *client* digunakan untuk mencari resep masakan dari mobile devices. Pencarian resep masakan dilakukan dengan keyword sebuah nama bahan dasar. Listing prosedur pencarian data resep masakan dari client ditunjukkan pada Lampiran A.
2. Prosedur menampilkan hasil pencarian pada client akan menampilkan hasil pencarian berupa nomor ID masakan dan nama resep masakan. Listing prosedur menampilkan hasil pencarian pada client ditunjukkan pada Lampiran B.
3. Prosedur menampilkan detail masakan pada *client* dieksekusi setelah pengguna memilih salah satu nama resep masakan yang tampil di hasil pencarian. Listing prosedur menampilkan detail masakan ditunjukkan pada Lampiran C.
4. Prosedur pengisian data masakan pada client terdiri dari input nama masakan, bahan dasar, bahan, bumbu, dan cara memasak. Prosedur pengisian resep masakan pada client ditunjukkan pada lampiran D.
5. Prosedur pengisian data resep masakan melalui web admin ditunjukkan pada Lampiran E.
6. Prosedur penghapusan data resep masakan melalui web admin ditunjukkan pada Lampiran F.

Y. 111 REC

lagikan resep masakan pilihan anda di MobileChef

Form Menambah Resep

Nama Masakan
TEMPE GORENG GARING

Bahan Dasar
TEMPE

Bahan
TEMPE, TEPUNG

Bumbu
BAWANG

Cara Memasak
DIGORENG

Back Ok

Gambar 6.3 Form pengisian resep masakan

Setelah dilakukan pengisian form, apabila resep berhasil ditambahkan ke database, akan muncul peringatan seperti pada gambar 6.4

Y. 111

Pendaftaran berhasil

Pendaftaran sukses

Done

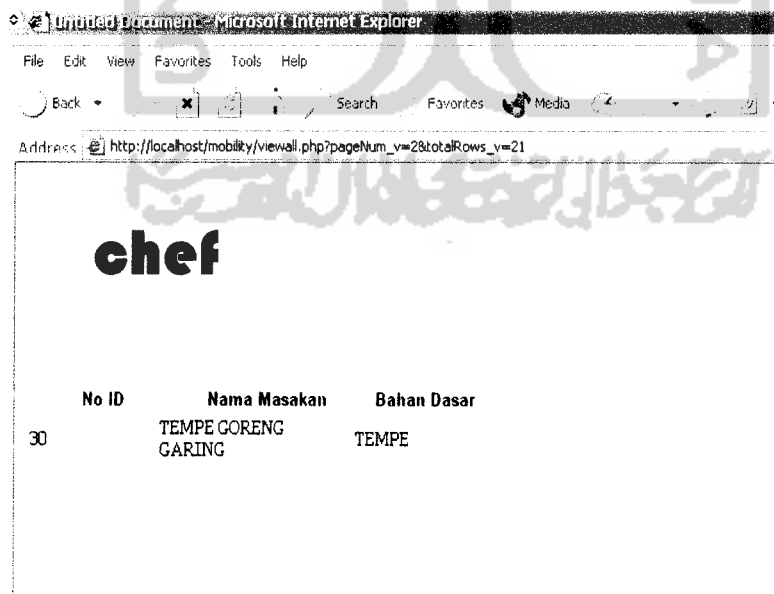
Gambar 6.4 Konfirmasi pengisian berhasil dilakukan

Apabila aplikasi gagal dalam melakukan koneksi ke server, akan muncul pesan kesalahan seperti gambar 6.5



Gambar 6.5 Pesan kesalahan apabila koneksi gagal

Hasil pengisian resep yang baru dapat dilihat pada web admin seperti pada gambar 6.5



Gambar 6.6 Tampilan database pada web admin

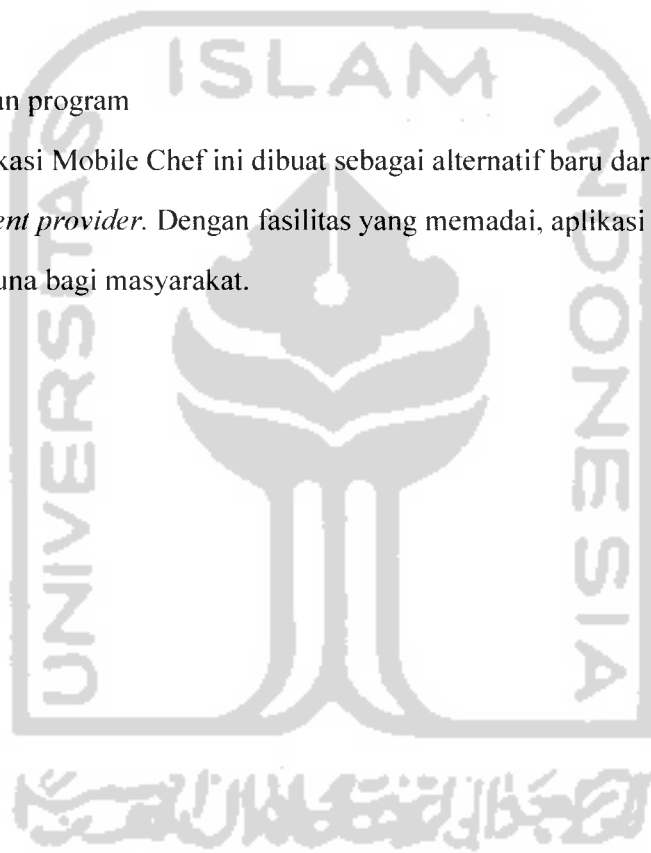
6.4 Kekurangan dan kelebihan program

Kekurangan aplikasi Mobile Chef ini adalah:

1. Sistem hanya dapat melakukan pencarian resep masakan dengan sebuah bahan dasar saja.
2. Aplikasi belum dapat menerapkan sistem keanggotaan, yang dapat mempermudah melakukan inventarisasi resep.

Kelebihan program

1. Aplikasi Mobile Chef ini dibuat sebagai alternatif baru dari layanan *mobile content provider*. Dengan fasilitas yang memadai, aplikasi ini tentu sangat berguna bagi masyarakat.



BAB VII SIMPULAN DAN SARAN

7.1. Simpulan

Setelah menyelesaikan pembuatan Aplikasi Mobile Chef menggunakan J2ME dan Teknologi Web Service, dapat diambil kesimpulan sebagai berikut:

1. Aplikasi Mobile Chef dibuat menggunakan database server lebih baik daripada menggunakan database RMS. Selain akan memakai memori yang lebih sedikit, juga data yang ditransfer lebih banyak.
2. Teknologi Web Service sangat dibutuhkan dalam aplikasi-aplikasi mobile yang akan datang, karena dapat menjembatani platform-platform yang berbeda.
3. Dalam aplikasinya Mobile Chef akan lebih murah dibandingkan dengan mengakses sebuah situs dengan GPRS, karena, dalam Mobile Chef, *mobile devices* dan database hanya mempertukarkan data yang berupa teks.

7.2 Saran

Saran yang dapat dikembangkan untuk pengembangan Aplikasi Mobile Chef Menggunakan J2ME dan Teknologi Web Service adalah:

1. Aplikasi perlu dikembangkan agar tidak hanya mencari resep dengan sebuah nama bahan dasar, tetapi dengan beberapa bahan dasar.
2. Penambahan data resep pada dilakukan secara teratur, sehingga layanan dapat dijaga agar tetap dinamis.
3. Perlu ditambahkan fasilitas keanggotaan pada web Mobile Chef, untuk mempermudah inventarisasi data resep masakan.
4. Pengubahan layout pada web sehingga lebih menarik.

DAFTAR PUSTAKA

- [AND06] Andriyanto, *eBDesk News Letter Building Community*. eBDesk, 2006
- [CHR01] Christensen, Eric., Curbera, Fransisco., Meredith, Greg., Weerawarana, Sanjiva., *Web Service Definition Language*. [http:// www.w3.org](http://www.w3.org), diakses tanggal 10 September 2006.
- [DAN03] Daniel, Roy. *Pengenalan Konsep XML Web Service*. <http://www.ilmukomputer.com>, diakses tanggal 10 September 2006.
- [FOW04] Fowler, Martin. *UML Distiled*. Terjemahan Tim Penerjemah Penerbit Andi. Jogjakarta. Penerbit ANDI, 2004.
- [HAR03] Hariyanto, Bambang. *Esensi-esensi pemrograman Java*. Bandung: Informatika Bandung, 2003.
- [HAR04] Hartanto, Antonius Aditya. *Modul Pelatihan J2ME dan Pengembangan Platform Aplikasi Mobile di Masa Depan*. Yogyakarta: Himpunan Mahasiswa Elektronika Universitas Negeri Yogyakarta, 2004.
- [MAH00] Qusay, Mahmoud. *MIDP Network Programming using HTTP and the Connection Framework*. <http://www.sun.org>, diakses tanggal 9 September 2006.
- [MAH00] Qusay, Mahmoud. *Wireless Application Programming: MIDP Programming and Packaging Basics*. <http://www.sun.org>, diakses tanggal 9 September 2006.
- [OQB00] Oqbuji, Uche. *Using WSDL in SOAP Applications*. <http://www.ibm.com>, diakses tanggal 9 September 2006.

LAMPIRAN A

PROSEDUR PENCARIAN DATA RESEP MASAKAN PADA CLIENT

```

public void panggilCariMasakan(){
    try {
        jawaban = chefClient.cariMasakan(bahan.getString());
        if (listHasilCari != null){
            listHasilCari.deleteAll();
            jawabanID = new int[jawaban.length];
            jawabanNama = new String[jawaban.length];
            for (int i=0; i<jawaban.length; i++){
                System.out.println(":" +jawaban[i].getNmMasakan());
                jawabanID[i] = jawaban[i].getIdMasakan();
                jawabanNama[i] = jawaban[i].getNmMasakan();

                listHasilCari.append(jawaban[i].getIdMasakan()+":
                "+jawaban[i].getNmMasakan(), null);
            }
        }
    } catch (Exception ex) {
        ex.printStackTrace();
        getDisplay().setCurrent(get_alert1());
    }
}

public Form get_formCari() {
    if (formCari == null) {
        formCari = new Form("Form Cari Resep", new Item[]
(get_bahan()));
        formCari.addCommand(get_okCommand1());
        formCari.addCommand(get_backCommand2());
        formCari.setCommandListener(this);
    }
    return formCari;
}

public TextField get_bahan() {
    if (bahan == null) {
        bahan = new TextField("Masukkan Bahan Dasar",
null, 120, TextField.ANY);
    }
    return bahan;
}

```

LAMPIRAN B

PROSEDUR MENAMPILKAN HASIL PENCARIAN PADA CLIENT

```
public List get_listHasilCari() {
    if (listHasilCari == null) {
        listHasilCari = new List("Data Resep Yang Sesuai",
Choice.IMPLICIT, new String[0], new Image[0]);
        listHasilCari.addCommand(get_okCommand2());
        listHasilCari.addCommand(get_backCommand3());
        listHasilCari.setCommandListener(this);
        listHasilCari.setSelectedFlags(new boolean[0]);
        jawabanID = new int[jawaban.length];
        jawabanNama = new String[jawaban.length];
        for (int i=0; i<jawaban.length; i++){
            System.out.println(":
"+jawaban[i].getNmMasakan());
            jawabanID[i] = jawaban[i].getIdMasakan();
            jawabanNama[i] = jawaban[i].getNmMasakan();
            listHasilCari.append(jawaban[i].getIdMasakan()+":
"+jawaban[i].getNmMasakan(), null);
        }
    }
    return listHasilCari;
}
```


LAMPIRAN C

PROSEDUR MENAMPILKAN DETAIL RESEP MASAKAN PADA CLIENT

```

public Form get_detail() {
    if (detail == null) {
        detail = new Form("Informasi Detail Resep", new
Item[] {
            get_detID(),
            get_detNama(),
            get_detBhnDasar(),
            get_detBahan(),
            get_detBumbu(),
            get_detCara()
        });
        detail.addCommand(get_backCommand5());
        detail.setCommandListener(this);

        get_detID().setText(""+jawabanDetail.idMasakan);
        get_detNama().setText(jawabanDetail.nmMasakan);
        get_detBhnDasar().setText(jawabanDetail.bhnDasar);
        get_detBahan().setText(jawabanDetail.bahan);
        get_detBumbu().setText(jawabanDetail.bumbu);
        get_detCara().setText(jawabanDetail.crMasak);
    }
    return detail;
}

public StringItem get_detID() {
    if (detID == null) {
        detID = new StringItem("ID Masakan", "<Enter Text>");
    }
    return detID;
}

public StringItem get_detNama() {
    if (detNama == null) {
        detNama = new StringItem("Nama Masakan", "<Enter
Text>");
    }
    return detNama;
}

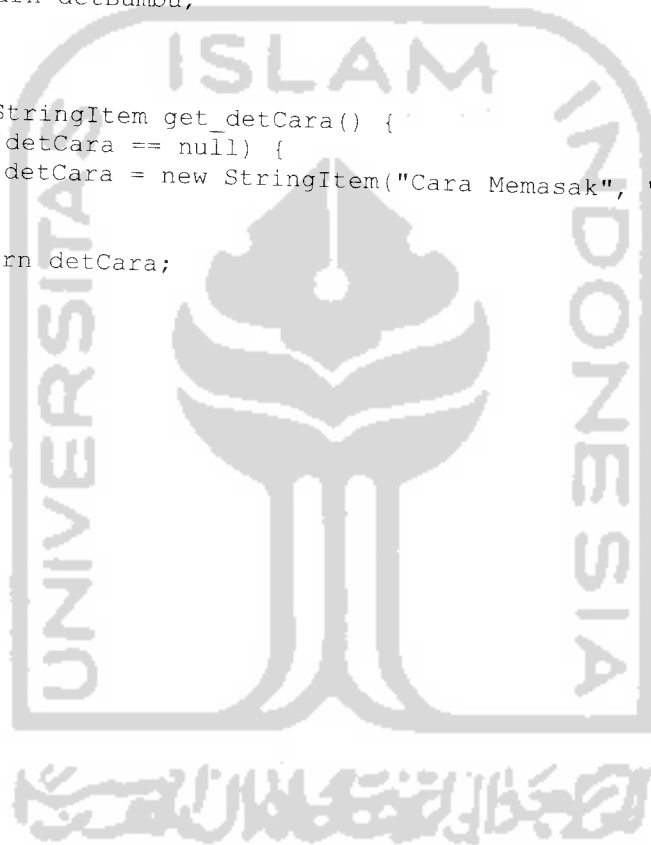
public StringItem get_detBhnDasar() {
    if (detBhnDasar == null) {
        detBhnDasar = new StringItem("Bahan Dasar", "<Enter
Text>");
    }
    return detBhnDasar;
}

```

```
public StringItem get_detBahan() {
    if (detBahan == null) {
        detBahan = new StringItem("Bahan", "<Enter Text>");
    }
    return detBahan;
}

public StringItem get_detBumbu() {
    if (detBumbu == null) {
        detBumbu = new StringItem("Bumbu", "<Enter Text>");
    }
    return detBumbu;
}

public StringItem get_detCara() {
    if (detCara == null) {
        detCara = new StringItem("Cara Memasak", "<Enter
Text>");
    }
    return detCara;
}
```



LAMPIRAN D

PROSEDUR PENGISIAN DATA RESEP MASAKAN PADA CLIENT

```

public Form get_formIsi() {
    if (formIsi == null) {

        formIsi = new Form("Form Menamah Resep", new Item[] {
            get_isiNamaMasakan(),
            get_isiBahanDasar(),
            get_isiBahan(),
            get_isiBumbu(),
            get_isiCara()
        });
        formIsi.addCommand(get_okCommand3());
        formIsi.addCommand(get_backCommand4());
        formIsi.setCommandListener(this);
        formIsi.setTicker(get_ticker2());
    } else {
        get_isiNamaMasakan().setString("");
        get_isiBahanDasar().setString("");
        get_isiBahan().setString("");
        get_isiBumbu().setString("");
        get_isiCara().setString("");
    }
    return formIsi;
}

public TextField get_isiNamaMasakan() {
    if (isiNamaMasakan == null) {
        isiNamaMasakan = new TextField("Nama Masakan", null,
120, TextField.ANY);
    }
    return isiNamaMasakan;
}

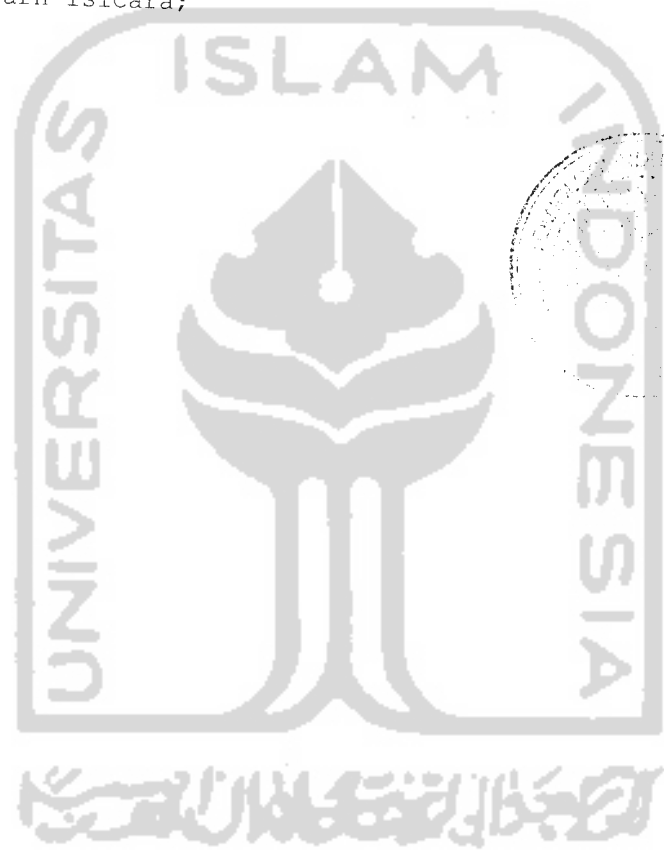
public TextField get_isiBahanDasar() {
    if (isiBahanDasar == null) {
        isiBahanDasar = new TextField("Bahan Dasar", null, 120,
TextField.ANY);
    }
    return isiBahanDasar;
}

public TextField get_isiBahan() {
    if (isiBahan == null) {

        isiBahan = new TextField("Bahan", null, 120,
TextField.ANY);
    }
    return isiBahan;
}

```

```
public TextField get_isiBumbu() {  
    if (isiBumbu == null) {  
        isiBumbu = new TextField("Bumbu", null, 120,  
TextField.ANY);  
    }  
    return isiBumbu;  
}  
public TextField get_isiCara() {  
    if (isiCara == null) {  
        isiCara = new TextField("Cara Memasak", null, 120,  
TextField.ANY);  
    }  
    return isiCara;  
}
```



LAMPIRAN E

PROSEDUR PENGISIAN DATA MASAKAN PADA WEB ADMIN

```

<?php require_once('Connections/conn5.php'); ?>
<?php
function GetSQLValueString($theValue, $theType, $theDefinedValue
= "", $theNotDefinedValue = "")
{
    $theValue = (!get_magic_quotes_gpc()) ? addslashes($theValue) :
    $theValue;

    switch ($theType) {
        case "text":
            $theValue = ($theValue != "") ? "'" . $theValue . "'" :
            "NULL";
            break;
        case "long":
        case "int":
            $theValue = ($theValue != "") ? intval($theValue) : "NULL";
            break;
        case "double":
            $theValue = ($theValue != "") ? "'" . doubleval($theValue)
            . "'" : "NULL";
            break;
        case "date":
            $theValue = ($theValue != "") ? "'" . $theValue . "'" :
            "NULL";
            break;
        case "defined":
            $theValue = ($theValue != "") ? $theDefinedValue :
            $theNotDefinedValue;
            break;
    }
    return $theValue;
}

$editFormAction = $_SERVER['PHP_SELF'];
if (isset($_SERVER['QUERY_STRING'])) {
    $editFormAction .= "?" .
    htmlentities($_SERVER['QUERY_STRING']);
}

if ((isset($_POST["MM_insert"])) && ($_POST["MM_insert"] ==
"form1")) {
    $insertSQL = sprintf("INSERT INTO tmasakan (nm_masakan,
bhn_dasar, bahan1, bahan2, cara) VALUES (%s, %s, %s, %s, %s)",
GetSQLValueString($_POST['nama'], "text"),
GetSQLValueString($_POST['bhndasar'], "text"),
GetSQLValueString($_POST['bahan1'], "text"),
GetSQLValueString($_POST['bahan2'], "text"),
GetSQLValueString($_POST['cara'], "text"));

    mysql_select_db($database_conn5, $conn5);
}

```

```
$Result1 = mysql_query($insertSQL, $conn5) or  
die(mysql_error());  
  
$insertGoTo = "viewall.php";  
if (isset($_SERVER['QUERY_STRING'])) {  
    $insertGoTo .= (strpos($insertGoTo, '?')) ? "&" : "?";  
    $insertGoTo .= $_SERVER['QUERY_STRING'];  
}  
header(sprintf("Location: %s", $insertGoTo));  
}  
  
mysql_select_db($database_conn5, $conn5);  
$query_rt = "SELECT * FROM tmasakan";  
$rt = mysql_query($query_rt, $conn5) or die(mysql_error());  
$row_rt = mysql_fetch_assoc($rt);  
$totalRows_rt = mysql_num_rows($rt);  
?>
```



```
    header(sprintf("Location: %s", $deleteGoTo));
}

$colname_rh = "1";
if (isset($_GET['id_masakan'])) {
    $colname_rh = (get_magic_quotes_gpc()) ? $_GET['id_masakan'] :
    addslashes($_GET['id_masakan']);
}
mysql_select_db($database_conn5, $conn5);
$query_rh = sprintf("SELECT id_masakan, nm_masakan FROM tmasakan
WHERE id_masakan = %s", $colname_rh);
$rh = mysql_query($query_rh, $conn5) or die(mysql_error());
$row_rh = mysql_fetch_assoc($rh);
$totalRows_rh = mysql_num_rows($rh);
?>
```

