

**RANCANG BANGUN SISTEM INFORMASI APOTEK  
TERINTEGRASI MENGGUNAKAN  
*SERVICE ORIENTED ARCHITECTURE***



**SKRIPSI**

Diajukan kepada Fakultas Sains dan Teknologi  
Universitas Islam Negeri Sunan Kalijaga  
Untuk Memenuhi Sebagian Syarat Memperoleh Gelar Sarjana  
Strata Satu Teknik Informatika

**Disusun Oleh:**

**Azizi Khoirul Haq**  
**NIM. 06650084**

**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI SUNAN KALIJAGA  
YOGYAKARTA**

**2012**



**PENGESAHAN SKRIPSI/TUGAS AKHIR**

Nomor : UIN.02/D.ST/PP.01.1/2223/2012

Skripsi/Tugas Akhir dengan judul : Rancang Bangun Sistem Informasi Apotek Terintegrasi Menggunakan Service Oriented Architecture

Yang dipersiapkan dan disusun oleh :  
Nama : Azizi Khcirul Haq  
NIM : 06650084  
Telah dimunaqasyahkan pada : Senin, 2 Juli 2012  
Nilai Munaqasyah : A / B  
Dan dinyatakan telah diterima oleh Fakultas Sains dan Teknologi UIN Sunan Kalijaga

**TIM MUNAQASYAH :**

Ketua Sidang

M. Mustakim, M.T

NIP. 19790331 200501 1 004

Penguji I

Agung Fatwanto, Ph.D  
NIP.19770103 200501 1 003

Penguji II

Agus Mulyanto, M.Kom  
NIP.19710823 199903 1 003

Yogyakarta, 17 Juli 2012  
UIN Sunan Kalijaga  
Fakultas Sains dan Teknologi  
Dekan



Prof. Drs. H. Akh. Minhaji, M.A, Ph.D  
NIP. 195809 19 198603 1 002



## **SURAT PERSETUJUAN SKRIPSI/TUGAS AKHIR**

Hal : Persetujuan Skripsi / Tugas Akhir

Lamp : -

Kepada

Yth. Dekan Fakultas Sains dan Teknologi

UIN Sunan Kalijaga Yogyakarta

di Yogyakarta

*Assalamu'alaikum wr. wb.*

Setelah membaca, meneliti, memberikan petunjuk dan mengoreksi serta mengadakan perbaikan seperlunya, maka kami selaku pembimbing berpendapat bahwa skripsi Saudara:

Nama : Azizi Khoirul Haq

NIM : 06650084

Judul Skripsi : Rancang Bangun Sistem Informasi Apotek Terintegrasi Menggunakan  
*Service Oriented Architecture*

sudah dapat diajukan kembali kepada Program Studi Teknik Informatika Fakultas Sains dan Teknologi UIN Sunan Kalijaga Yogyakarta sebagai salah satu syarat untuk memperoleh gelar Sarjana Strata Satu dalam Teknik Informatika

Dengan ini kami berharap agar skripsi/tugas akhir Saudara tersebut di atas dapat segera dimunaqsyahkan. Atas perhatiannya kami ucapkan terima kasih.

Yogyakarta, 22 Juni 2012

Pembimbing

M. Mustakim, MT

NIP.19790331 200501 1 004

## PERNYATAAN KEASLIAN SKRIPSI

Yang bertanda tangan dibawah ini:

Nama : Azizi Khoirul Haq

NIM : 06650084

Program Studi : Teknik Informatika

Fakultas : Sains dan Teknologi

Menyatakan bahwa skripsi dengan judul "**RANCANG BANGUN SISTEM INFORMASI APOTEK TERINTEGRASI MENGGUNAKAN *SERVICE ORIENTED ARCHITECTURE***" tidak terdapat karya yang pernah diajukan untuk memperoleh gelar kesarjanaan disuatu Perguruan Tinggi, dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka.

Yogyakarta, 22 Juni 2012

Yang menyatakan



  
Azizi Khoirul Haq  
NIM. 06650084

## KATA PENGANTAR

Puji syukur kepada Allah SWT yang telah melimpahkan rahmat dan anugerah-Nya kepada penulis sehingga penulis dapat menyelesaikan skripsi dengan judul “Rancang Bangun Sistem Informasi Apotek Terintegrasi Menggunakan *Service Oriented Architecture*” ini dengan lancar dan tidak ada halangan suatu apapun. Sholawat serta Salam selalu penulis haturkan kepada Nabi Besar Muhammad SAW.

Selanjutnya penulis mengucapkan banyak-banyak terima kasih yang tak terkira kepada:

1. Bapak Prof. Dr. Musa Asy'ari, M.A, Ph.D, selaku Rektor Universitas Islam Negeri Sunan Kalijaga.
2. Bapak Prof. Drs. H. Akhmad Minhaji, M.A, Ph.D, selaku Dekan Fakultas Sains dan Teknologi UIN Sunan Kalijaga.
3. Bapak Agus Mulyanto, M.Kom, selaku Ketua Program Studi Teknik Informatika UIN Sunan Kalijaga yang telah memberikan kesempatan pada penulis untuk membuat tugas akhir ini.
4. Bapak Mustakim, MT, selaku pembimbing skripsi yang dengan penuh kesabaran dan ketelitian telah mengarahkan dan membimbing penulis selama ini.
5. Para Dosen Program Studi Teknik Informatika UIN Sunan Kalijaga yang telah memberi bekal ilmu pengetahuan yang banyak kepada penulis, semoga ilmunya menjadi amal *jariyah* di dunia hingga akhirat.

6. Kedua orang tuaku Drs. Mohammad Isa Ashari, S.Pd.I dan Winarni, S.Pd serta adikku semata wayang Muhammad Saiful Amri yang telah memberikan dorongan, semangat, do'a dan kasih sayang yang tiada hentinya.
7. Teman-teman Asisten Laboratorium Sistem Informasi UIN Sunan Kalijaga yang banyak menginspirasi : Sigit, Irvan, Ayu, Sunu, Fahmi, Habibi.
8. Laboran Teknik Informatika UIN Sunan Kalijaga : Kang Munawwir, Mas Yusuf Murdani, Mas Iqbal AT, Mas Rian, Kang Iwan. Terima kasih semuanya.
9. Sahabat-sahabatku Agung, Haryo, Alex, Imam, Mas Antok, Didik, Nurdin, Rian, Ismail, Jusmail, Irvan, Sunu, Sigit, Fathan, Ayu, Habibi, Fahmi, Arin, Nur Anisha, Rischan, Fadli, Hafidz dan semua teman-teman teknik informatika angkatan 2006 yang tidak dapat penulis sebutkan satu persatu.

Penulis merasa masih banyak sekali kekurangan dan kelemahan dalam penelitian ini, oleh karena itu kritik dan saran senantiasa penulis harapkan dari para pembaca. Akhirnya semoga penelitian ini dapat dimanfaatkan sebaik-baiknya.

Yogyakarta, 22 Juni 2012

Azizi Khoirul Haq  
NIM. 06650084

## HALAMAN PERSEMBAHAN

*Skripsi ini penulis persembahkan kepada:*

- ❖ *Alloh Subhanahu wa Ta'ala, yang telah melimpahkan banyak karunia kepada penulis, sehingga skripsi ini dapat diselesaikan dengan baik dan lancar.*
- ❖ *Sholawat serta Salam kepada Nabi Besar Muhammad Shallallahu 'alaihi wa Sallam.*
- ❖ *Ayahanda dan Ibunda tercinta, yang telah membesarkan, membimbing dan mendo'akanku dengan penuh kesabaran dan kasih sayang.*
- ❖ *Adikku tersayang yang selalu memberikan dorongan semangat dan inspirasi.*
- ❖ *Keluargaku di Yogyakarta : Mbah Jarni, Mbah Juminten, Alm. Mbah Tumirin semoga beliau diberikan tempat di surga-Nya, Mbah Asih, Mbak Iput, Mas Purwo Sutedjo, Mas Yono, Mas Sukoco, Mbak Bening dan keponakanku Yudha. Terima kasih atas semuanya.*
- ❖ *Keluargaku di Pacitan : Alm. Mbah Katiyo, Mbah Suratini yang telah mengasuhku dari kecil, Bulik Harti, Om Rahmad, Mbah Abdulhadi, Alm. Mbah putri Suyati, Om Supriyadi, Om Muhandi, Om Ansori, Bulik Dwi, Bulik Nur, Bulik Umi, Dik Tofa, Dik Andri, Dik Wikan, Dik Ika, Dik Riza, Dik Azza, Dik Mirza dan semua keponakanku yang masih kecil. Terima kasih atas segala kasih sayang, keceriaan, dan kehangatan dalam keluarga besar Trah Ahmad Tarmin dan keluarga besar Trah Mbah Sanawi.*

- ❖ *Guru-guruku di SDN Sugihwaras I, MTsN Pacitan, dan MAN Pacitan. Terima kasih atas semua ilmu dan pelajaran yang banyak diberikan kepadaku. Hanya Alloh SWT yang bisa membalas kebaikan bapak dan ibu guru semua.*
- ❖ *Teman-temanku di Alumni MAN Pacitan : Ali Sujiwo, M.Irfan Fathoni, M.Fatoni, Ahmad Badaruzzaman, Hadi "Gembul" Winarno, Kang Hanif, Fitri Suryaningsih, Mulyani, Tupani, Khoirul Anam, Saifuddin Zuhri, Arif Mustofa, Ras Rekker, Emmy Zulaichah, Hendro Putranto, dan semuanya yang tak bisa kusebutkan satu-persatu. Terima kasih sobatku dan sahabatku.*
- ❖ *Teman-temanku di Alumni SDN Sugihwaras I : Hardi Risdiyanto, Kristinawati Trianingsih, Roni B.A, Oni Prasetyawan, Riki Yidha P., Maryono dan semuanya. Terima kasih sobat atas keceriaan dan persahabatan ini.*
- ❖ *Teman temanku Teknik Informatika : Agoenk, Haryo, Mas Antok, Alex, Imam, Sigit, Irvan, Sidiq, Sunu, Didik, Nurdin, Rian, Joko, Wachid, Jusmail, Fikri, Mas A'an, Mas Ayun, Arfan, Adit, Triawan, Sepran, Rifqi, Habibi, Fahmi, Budi, Fendi, Mas Ganjar, Mas Luluk, Alfan, Ayu, Tami, Anggi, Hadiyatun, Syifa, Rizki dan yang lain yang tidak bisa kusebutkan satu persatu.*
- ❖ *Seluruh pecinta teknologi informasi dimanapun berada. Kita selalu berbagi dalam keilmuan untuk kemajuan bangsa dan negara.*



## HALAMAN MOTTO

قُلْ لَوْ كَانَ الْبَحْرُ مِدَادًا لِكَلِمَاتِ رَبِّي لَنَفِدَ الْبَحْرُ قَبْلَ أَنْ تَنْفَدَ  
كَلِمَاتُ رَبِّي وَلَوْ جِئْنَا بِمِثْلِهِ مَدَدًا ﴿١٠٩﴾

*"Katakanlah: Sekiranya lautan menjadi tinta untuk (menulis) kalimat-kalimat  
Tuhanku, sungguh habislah lautan itu sebelum habis (ditulis) kalimat-kalimat  
Tuhanku, meskipun Kami datangkan tambahan sebanyak itu (pula)."*

--QS. Al-Kahfi : 109--

*"Barangsiapa yang menempuh jalan untuk mencari ilmu, niscaya Allah akan  
memudahkannya (ilmu) jalan ke surga."*

--HR. At-Turmudzi--

*"Sekadar mengetahui tidak cukup, kita harus menginginkannya. Sekadar berkeinginan  
tidaklah cukup, kita harus bertindak."*

--Johann von Goethe--

*"Kesuksesan mendatangi mereka yang tidak takut gagal"*

--Winston Churchill--

## DAFTAR ISI

Halaman Judul .....	i
Halaman Pengesahan Skripsi/Tugas Akhir .....	ii
Halaman Persetujuan Skripsi/Tugas Akhir .....	iii
Halaman Pernyataan Keaslian Skripsi .....	iv
Kata Pengantar.....	v
Halaman Persembahan .....	vii
Halaman Motto .....	ix
Daftar Isi.....	x
Daftar Gambar .....	xiv
Daftar Tabel.....	xix
Daftar Lampiran.....	xx
Intisari .....	xxi
Abstract .....	xxii
<b>BAB I PENDAHULUAN.....</b>	<b>1</b>
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah .....	2
1.3 Batasan Masalah .....	3
1.4 Tujuan Penelitian .....	3
1.5 Manfaat Penelitian .....	4
1.6 Keaslian Penelitian.....	4

<b>BAB II TINJAUAN PUSTAKA DAN LANDASAN TEORI .....</b>	<b>5</b>
2.1 Tinjauan Pustaka.....	5
2.2 Landasan Teori.....	7
2.2.1 Konsep Sistem Informasi .....	7
2.2.2 <i>Service Oriented Architecture</i> .....	9
2.2.3 <i>Web Services</i> .....	13
2.2.4 Konsep <i>Collaborative E-Commerce</i> .....	15
2.2.5 Pemrograman Berorientasi Objek.....	17
2.2.6 <i>Unified Modeling Language (UML)</i> .....	19
2.2.7 <i>Extensible Markup Language (XML)</i> .....	20
2.2.8 <i>PHP Hypertext Preprocessor (PHP)</i> .....	22
2.2.9 <i>Java 2 Micro Edition (J2ME)</i> .....	23
2.2.10 <i>Visual Basic. NET (VB.NET)</i> .....	25
2.2.11 <i>MySQL Database</i> .....	27
2.2.12 <i>PostgreSQL Database</i> .....	28
2.2.13 <i>Model-View-Controller (MVC)</i> .....	29
2.2.14 <i>Code Igniter Framework</i> .....	31
 <b>BAB III METODE PENGEMBANGAN SISTEM.....</b>	 <b>36</b>
3.1 Studi Pendahuluan.....	36
3.2 Pengumpulan Data .....	36
3.3 Perancangan Model.....	36
3.4 Kebutuhan Pengembangan Sistem.....	37
3.5 Metode Pengembangan Sistem.....	38

<b>BAB IV ANALISIS DAN PERANCANGAN SISTEM.....</b>	<b>40</b>
4.1 Analisis Kebutuhan Sistem.....	40
4.2 Perancangan Sistem .....	41
4.2.1 Arsitektur Sistem .....	41
4.2.2 <i>Use Case Diagram</i> .....	42
4.2.3 <i>Sequence Diagram</i> .....	44
4.2.4 <i>Activity Diagram</i> .....	51
4.2.5 <i>Class Diagram</i> .....	57
4.2.6 Perancangan Antar Muka.....	58
4.2.7 Perancangan Basis Data .....	60
<b>BAB V IMPLEMENTASI DAN PENGUJIAN SISTEM.....</b>	<b>64</b>
5.1 Implementasi Sistem .....	64
5.1.1 Implementasi Basis Data.....	66
5.1.2 Implementasi <i>Web Service</i> .....	71
5.1.3 Implementasi Sistem Berbasis <i>Web</i> .....	74
5.1.3.1 Implementasi Login .....	74
5.1.3.2 Implementasi Pemesanan .....	76
5.1.3.3 Implementasi Konfirmasi Pemesanan.....	79
5.1.3.4 Implementasi Pembayaran .....	82
5.1.3.5 Implementasi Konfirmasi Pembayaran .....	83
5.1.3.6 Implementasi Pengiriman.....	85
5.1.3.7 Implementasi Laporan.....	86
5.1.3.8 Implementasi Menampilkan Stok.....	90

5.1.3.9 Implementasi Registrasi .....	92
5.1.3.10 Implementasi Manajemen Member .....	94
5.1.4 Implementasi Sistem Berbasis <i>Desktop</i> .....	97
5.1.5 Implementasi Sistem Berbasis <i>Mobile</i> .....	98
5.2 Pengujian Sistem.....	99
<b>BAB VI PENUTUP .....</b>	<b>104</b>
6.1 Kesimpulan.....	104
6.2 Saran.....	104
<b>DAFTAR PUSTAKA .....</b>	<b>106</b>
<b>LAMPIRAN .....</b>	<b>109</b>

## DAFTAR GAMBAR

Gambar 2.1	Ide dasar <i>service oriented architecture</i> .....	10
Gambar 2.2	Struktur hirarki SOA.....	12
Gambar 2.3	Susunan standar <i>web services</i> .....	14
Gambar 2.4	Skema integrasi B2B .....	16
Gambar 2.5	Teknis klasifikasi informasi dalam XML.....	22
Gambar 2.6	Variasi arsitektur teknologi Java.....	24
Gambar 2.7	Platform J2ME dan <i>market target</i> .....	24
Gambar 2.8	Konsep hubungan <i>model-view-controller</i> .....	30
Gambar 2.9	Aliran data pada <i>framework</i> CodeIgniter .....	34
Gambar 3.1	Model <i>waterfall</i> .....	38
Gambar 4.1	Interaksi antara pemakai dan aplikasi .....	41
Gambar 4.2	Diagram <i>use case</i> .....	42
Gambar 4.3	Diagram <i>sequence</i> pemesanan.....	44
Gambar 4.4	Diagram <i>sequence</i> konfirmasi pemesanan .....	45
Gambar 4.5	Diagram <i>sequence</i> pembayaran .....	46
Gambar 4.6	Diagram <i>sequence</i> konfirmasi pembayaran.....	47
Gambar 4.7	Diagram <i>sequence</i> pengiriman .....	47
Gambar 4.8	Diagram <i>sequence</i> laporan.....	48
Gambar 4.9	Diagram <i>sequence</i> menampilkan stok .....	49
Gambar 4.10	Diagram <i>sequence</i> registrasi .....	49
Gambar 4.11	Diagram <i>sequence</i> manajemen member.....	50

Gambar 4.12	Diagram <i>activity</i> pemesanan .....	52
Gambar 4.13	Diagram <i>activity</i> konfirmasi pemesanan .....	53
Gambar 4.14	Diagram <i>activity</i> pembayaran.....	53
Gambar 4.15	Diagram <i>activity</i> konfirmasi pembayaran .....	54
Gambar 4.16	Diagram <i>activity</i> pengiriman.....	55
Gambar 4.17	Diagram <i>activity</i> laporan .....	55
Gambar 4.18	Diagram <i>activity</i> menampilkan stok.....	56
Gambar 4.19	Diagram <i>activity</i> registrasi.....	56
Gambar 4.20	Diagram <i>activity</i> manajemen member .....	57
Gambar 4.21	<i>Class diagram</i> sistem informasi apotek terintegrasi.....	58
Gambar 4.22	Desain antarmuka aplikasi berbasis <i>web</i> .....	59
Gambar 4.23	Desain antarmuka aplikasi <i>desktop</i> untuk <i>member</i> .....	59
Gambar 4.24	Desain antarmuka aplikasi <i>mobile</i> untuk pengunjung.....	60
Gambar 5.1	Informasi <i>service</i> PostgreSQL dalam keadaan aktif .....	64
Gambar 5.2	Informasi <i>service</i> MySQL dalam keadaan aktif .....	64
Gambar 5.3	Informasi PHP 5 dalam keadaan aktif .....	65
Gambar 5.4	Informasi IDE Visual Studio 2010 dalam keadaan aktif .....	65
Gambar 5.5	Informasi <i>simulator</i> J2ME dalam keadaan aktif.....	66
Gambar 5.6	Layanan <i>web service</i> apotek terintegrasi.....	71
Gambar 5.7	<i>Listing code</i> koneksi basis data .....	74
Gambar 5.8	<i>Listing code</i> halaman login member .....	75
Gambar 5.9	Tampilan login member .....	75
Gambar 5.10	<i>Listing code</i> halaman login administrator .....	75

Gambar 5.11	Tampilan login administrator .....	76
Gambar 5.12	<i>Listing code</i> fungsi pesan_obat.....	76
Gambar 5.13	Tampilan pesan obat .....	77
Gambar 5.14	<i>Listing code</i> fungsi cart_sementara.....	77
Gambar 5.15	Tampilan halaman keranjang belanja .....	78
Gambar 5.16	<i>Listing code</i> fungsi simpan_pesanan .....	78
Gambar 5.17	Tampilan halaman simpan pesanan .....	79
Gambar 5.18	<i>Listing code</i> fungsi manajemen_pemesanan .....	80
Gambar 5.19	Tampilan halaman manajemen pemesanan .....	80
Gambar 5.20	<i>Listing code</i> fungsi proses_pemesanan .....	81
Gambar 5.21	Tampilan konfirmasi pemesanan.....	81
Gambar 5.22	<i>Listing code</i> fungsi simpan_proses_pemesanan .....	82
Gambar 5.23	<i>Listing code</i> fungsi bayar_pesanan .....	82
Gambar 5.24	Tampilan bayar pesanan.....	82
Gambar 5.25	<i>Listing code</i> fungsi riwayat_pesanan_popup .....	83
Gambar 5.26	Tampilan <i>popup</i> riwayat pesanan .....	83
Gambar 5.27	<i>Listing code</i> fungsi manajemen_pembayaran .....	84
Gambar 5.28	Tampilan manajemen pembayaran .....	84
Gambar 5.29	<i>Listing code</i> fungsi ubah_status.....	85
Gambar 5.30	Tampilan konfirmasi pembayaran .....	85
Gambar 5.31	<i>Listing code</i> fungsi lacak_pesanan .....	86
Gambar 5.32	Tampilan lacak pesanan .....	86
Gambar 5.33	<i>Listing code</i> fungsi laporan_penjualan .....	87



Gambar 5.34	<i>Listing code</i> untuk <i>rendering</i> laporan penjualan .....	87
Gambar 5.35	Tampilan laporan penjualan .....	87
Gambar 5.36	Tampilan hasil laporan penjualan .....	88
Gambar 5.37	<i>Listing code</i> fungsi laporan pembelian .....	88
Gambar 5.38	<i>Listing code</i> untuk <i>rendering</i> laporan pembelian .....	89
Gambar 5.39	Tampilan laporan pembelian .....	89
Gambar 5.40	Tampilan hasil laporan pembelian .....	89
Gambar 5.41	<i>Listing code</i> fungsi pencarian .....	90
Gambar 5.42	Tampilan menampilkan stok .....	90
Gambar 5.43	<i>Listing code</i> fungsi pesanan terbanyak dan terbaru .....	91
Gambar 5.44	Tampilan pesanan terbanyak dan terbaru .....	91
Gambar 5.45	<i>Listing code</i> menampilkan lokasi apotek .....	92
Gambar 5.46	Tampilan pencarian lokasi apotek .....	92
Gambar 5.47	<i>Listing code</i> fungsi <code>add_member</code> .....	93
Gambar 5.48	Tampilan registrasi .....	93
Gambar 5.49	<i>Listing code</i> fungsi <code>manajemen_member</code> .....	94
Gambar 5.50	Tampilan manajemen member .....	94
Gambar 5.51	<i>Listing code</i> fungsi mengubah member .....	95
Gambar 5.52	Tampilan ubah member .....	95
Gambar 5.53	<i>Listing code</i> fungsi <code>detail_member</code> .....	96
Gambar 5.54	Tampilan detail member .....	96
Gambar 5.55	<i>Listing code</i> fungsi pencarian stok aplikasi <i>desktop</i> .....	97
Gambar 5.56	Tampilan pencarian stok aplikasi <i>desktop</i> .....	97

Gambar 5.57	<i>Listing code</i> fungsi pencarian stok aplikasi <i>mobile</i> .....	98
Gambar 5.58	Tampilan pencarian stok aplikasi <i>mobile</i> .....	98
Gambar 5.59	Tampilan <i>tool</i> pengujian <i>web service</i> .....	99

## DAFTAR TABEL

Tabel 4.1	Definisi aktor pada diagram <i>use case</i> .....	43
Tabel 4.2	Definisi <i>use case</i> .....	43
Tabel 4.3	Desain tabel member .....	61
Tabel 4.4	Desain tabel admin.....	61
Tabel 4.5	Desain tabel status_pemesanan .....	62
Tabel 4.6	Desain tabel status_pembayaran .....	62
Tabel 4.7	Desain tabel pemesanan.....	62
Tabel 4.8	Desain tabel detail_pemesanan .....	63
Tabel 4.9	Desain tabel pembayaran .....	63
Tabel 5.1	Daftar tabel pada basis data Apotek K24 Manyar.....	70
Tabel 5.2	Daftar tabel pada basis data Apotek 55 Bandung .....	70
Tabel 5.3	Operasi layanan <i>web service</i> .....	71
Tabel 5.4	Hasil pengujian layanan <i>web service</i> .....	100
Tabel 5.5	Skenario pengujian .....	101
Tabel 5.6	Hasil pengujian fungsional sistem.....	102
Tabel 5.7	Hasil pengujian antarmuka .....	102

## DAFTAR LAMPIRAN

Lampiran A	Kode WSDL ( <i>Web Service Description Language</i> ) .....	109
Lampiran B	Kode Sumber <i>Web Service</i> Apotek .....	113
Lampiran C	Kode File Konfigurasi <i>Database</i> .....	120
Lampiran D	Kode File <i>Controller</i> .....	121
Lampiran E	Kode File <i>Model</i> .....	159
Lampiran F	Kode Sumber Aplikasi <i>Desktop</i> .....	170
Lampiran G	Kode Sumber Aplikasi <i>Mobile</i> .....	171
Lampiran H	<i>Form</i> Pengujian Rancang Bangun Sistem Informasi Apotek Terintegrasi Menggunakan <i>Service Oriented Architecture</i> .....	174
Lampiran I	Daftar Penguji.....	175
	<i>Curriculum Vitae</i> .....	176

# RANCANG BANGUN SISTEM INFORMASI APOTEK TERINTEGRASI MENGUNAKAN *SERVICE ORIENTED ARCHITECTURE*

**Azizi Khoirul Haq**

**NIM. 06650084**

## INTISARI

Sistem informasi apotek yang ada saat ini, kebanyakan dirancang dan dibangun hanya untuk mengelola satu buah apotek tertentu. Dalam hal pemesanan obat, pihak apotek kebanyakan masih menggunakan cara manual, seperti kirim surat pemesanan, maupun melakukan pemesanan melalui telepon. Basis data yang dibangun sebagai *back-end* dari sistem informasi apotek, juga kebanyakan menggunakan perangkat lunak basis data yang beragam. Saat ini, dari sisi pelanggan, untuk mendapatkan informasi keberadaan suatu produk obat, pelanggan apotek perlu mendatangi tiap apotek satu persatu sampai menemukan produk obat yang dicari. Hal ini tentu saja akan menambah biaya yang tentu sangat tidak efisien dari segi ekonomi.

Penelitian ini bertujuan untuk mengembangkan suatu sistem informasi apotek yang terintegrasi dengan memanfaatkan *Service Oriented Architecture*. Metode pengembangan sistem ini dibangun dengan pendekatan *Systems Development Life Cycle (SDLC)* dengan beberapa tahapan yaitu analisis, desain, pengkodean, pengujian, dan pemeliharaan. Rancangan diimplementasikan dengan *Unified Modeling Language (UML)*. Bahasa pemrograman yang digunakan dalam penelitian ini yaitu PHP dengan *framework* CodeIgniter, *Visual Basic .NET (VB.NET)*, dan *Java 2 Mobile Edition (J2ME)*.

Hasil penelitian ini yaitu dihasilkannya sistem informasi apotek yang terintegrasi satu sama lain dengan sistem informasi apotek maupun aplikasi lainnya, baik berbeda dari segi bahasa pengembangannya maupun berbeda dalam basis data yang digunakan. Dengan terintegrasinya sistem, maka dapat digunakan untuk kegunaan yang lebih luas, seperti pemesanan barang antar sistem apotek yang berbeda. Selain itu dihasilkan pula aplikasi berbasis *mobile* dan *desktop* untuk mengakses informasi stok obat. Diharapkan dengan aplikasi *mobile* ini mampu mendekatkan pihak apotek dengan pelanggan yang tentunya berdampak pada hubungan pelanggan dan pihak apotek yang semakin erat.

**Kata Kunci** : Sistem Informasi Apotek, UML, PHP, Visual Basic .NET, J2ME

# DESIGNING AND IMPLEMENTING INTEGRATED PHARMACY INFORMATION SYSTEM USING SERVICE ORIENTED ARCHITECTURE

Azizi Khoirul Haq

NIM. 06650084

## ABSTRACT

Pharmacy information systems that exist today, many designed and built just to manage a single specific pharmacy. In terms of ordering medications, the pharmacies most still use the manual method, such as send a letter of reservation, or make a reservation by phone. The database is built as a back-end of the pharmacy information system, also use the software most diverse database. Currently, from the customer side, to obtain the presence information of a drug product, customers need to go to every pharmacy pharmacy one by one to find the product you are looking for drugs. This of course will add to the cost of the course is very inefficient in economic terms.

This study aims to develop an integrated pharmacy information system by utilizing Service Oriented Architecture. Method of system development approach is built with Systems Development Life Cycle (SDLC) with several stages of the analysis, design, coding, testing, and maintenance. Plans are being implemented by the Unified Modeling Language (UML). The programming language used in this study is PHP with CodeIgniter framework, Visual Basic. NET (VB.NET), and Java 2 Mobile Edition (J2ME).

The results of this research is the production of integrated pharmacy information system to each other with the pharmacy information system and other applications, both in terms of different languages and different development in the data base used. With the integration of the system, it can be used for broader uses, such as ordering goods between different pharmacy systems. In addition it also produced mobile and desktop-based applications to access information drug stocks. These mobile applications are expected to be able to close the pharmacy with a customer who would have an impact on customer relationships and the pharmacies are getting close.

**Keywords:** Pharmacy Information Systems, UML, PHP, Visual Basic. NET, J2ME

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Sistem informasi apotek yang ada saat ini, kebanyakan dirancang dan dibangun hanya untuk mengelola satu buah apotek tertentu. Dalam hal pemesanan obat, pihak apotek kebanyakan masih menggunakan cara manual, seperti kirim surat pemesanan, maupun melakukan pemesanan melalui telepon. Basis data yang dibangun sebagai *back-end* dari sistem informasi apotek, juga kebanyakan menggunakan perangkat lunak basis data yang beragam.

Saat ini, dari sisi pelanggan, untuk mendapatkan informasi keberadaan suatu produk obat, pelanggan apotek perlu mendatangi tiap apotek satu persatu sampai menemukan produk obat yang dicari. Apalagi untuk ketersediaan obat yang langka dipasaran, pastilah akan sulit untuk mencari obat dengan cepat. Hal ini tentu saja akan menambah biaya yang tentu sangat tidak efisien dari segi ekonomi. Selain itu kecepatan pencarian obat juga sangat diperlukan oleh pelanggan. Hal ini tentu akan sangat menyulitkan pengguna yang ingin mendapatkan akses informasi obat yang cepat dan akurat serta efisien dari segi waktu dan biaya.

Dilihat dari segi efisiensi, tentu akan lebih efisien apabila dibuat suatu sistem yang menjembatani antar apotek untuk saling pesan obat dengan mudah, bisa melakukan pemesanan langsung atau *direct order* kepada apotek lain yang memiliki sistem informasi apotek yang berbeda baik dalam bahasa pemrograman

maupun basis data yang digunakan. Dengan menggunakan konsep pemesanan *business to business* (B2B) akan dapat dilakukan penghematan dari segi keuangan.

*Service Oriented Architecture* (SOA) merupakan model arsitektur untuk membuat rekayasa perangkat lunak menggunakan *service* yang terdapat pada jaringan. Komponen-komponen rekayasa perangkat lunak pada SOA bersifat *loose coupling*, sehingga bisa digunakan kembali. Aplikasi pada SOA dibangun berdasarkan *service*, yang mana merupakan implementasi dari fungsi bisnis yang terdefinisi jelas dan bisa dikonsumsi oleh klien pada berbagai aplikasi atau proses bisnis (Permana, 2011).

Salah satu solusi yang bisa dilakukan yaitu perlu adanya integrasi sistem informasi apotek. Arsitektur pengembangan yang telah banyak digunakan dan menurut peneliti sebagai solusi yang terbaik yaitu dengan menggunakan *Service Oriented Architecture* (SOA).

## **1.2 Rumusan Masalah**

Melihat latar belakang masalah diatas, serta melihat berbagai metode pengembangan perangkat lunak, maka peneliti mencoba untuk membuat rumusan masalah sebagai berikut:

1. Bagaimana merancang sistem informasi apotek terintegrasi menggunakan *Service Oriented Architecture* (SOA) dengan pendekatan berorientasi objek?
2. Bagaimana mengimplementasikan *Service Oriented Architecture* menggunakan *Web Service*?



3. Bagaimana mengimplementasikan sistem informasi apotek terintegrasi dengan menggunakan konsep *Collaborative E-Commerce*?

### 1.3 Batasan Masalah

Adapun batasan masalah dalam penelitian ini adalah sebagai berikut:

1. Aplikasi sistem informasi apotek terintegrasi dibuat dalam bahasa pemrograman PHP dengan *framework* CodeIgniter. Dibuatk pula aplikasi untuk pencarian obat dalam bentuk *desktop platform* yang dibuat dengan bahasa pemrograman VB.NET serta aplikasi pencarian obat dalam bentuk *mobile application* yang dibuat dengan bahasa pemrograman J2ME.
2. Pemesanan obat antar apotek dilayani hanya dalam lingkup Indonesia.
3. Sistem dapat melakukan transaksi pemesanan dibatasi hanya pada jenis obat bebas, obat bebas terbatas, obat keras, dan obat wajib apotek.
4. Sistem berbasis *web* menyediakan fasilitas untuk pemesanan obat antar apotek yang berbeda baik berbeda pemilik maupun lokasi.
5. Aplikasi berbasis *desktop* dan *mobile* dapat melakukan pencarian stok obat.

### 1.4 Tujuan Penelitian

Tujuan yang ingin dicapai dalam penelitian ini adalah sebagai berikut:

1. Merancang sistem informasi apotek terintegrasi menggunakan *Service Oriented Architecture* (SOA) dengan pendekatan berorientasi objek.
2. Mengimplementasikan *Service Oriented Architecture* menggunakan *Web Service*.

3. Membuat sistem informasi apotek terintegrasi dengan menggunakan konsep *Collaborative E-Commerce*.

### **1.5 Manfaat Penelitian**

Dengan adanya penelitian ini diharapkan dapat memberikan manfaat, diantaranya sebagai berikut:

1. Menghasilkan sistem informasi apotek terintegrasi menggunakan *Service Oriented Architecture* berbasis *web*, serta aplikasi pencarian stok obat berbasis *desktop* dan *mobile*.
2. Memberikan manfaat bagi pihak apotek yang tergabung sebagai *member* yaitu mempersingkat waktu dan efisiensi biaya pemesanan dengan melakukan *direct order* kepada *member* apotek yang berbeda lokasi.
3. Memberikan manfaat bagi pengunjung atau pelanggan yaitu mengurangi waktu pencarian obat. Dengan adanya aplikasi *mobile* yang terpadu, maka bisa mengefisienkan waktu pencarian obat dari setiap *database* apotek yang berbeda.

### **1.6 Keaslian Penelitian**

Penelitian yang berhubungan dengan masalah *Service Oriented Architecture* sudah pernah dilakukan sebelumnya, tetapi integrasi sistem informasi apotek terintegrasi menggunakan *Service Oriented Architecture* berbasis *web* dengan pengembangan berbasis MVC (*Model, View, Controller*), *desktop* dan *mobile* dengan pendekatan berorientasi objek serta menggunakan basis data MySQL dan PostgreSQL sejauh peneliti belum pernah dilakukan sebelumnya.

## **BAB VI**

### **PENUTUP**

#### **6.1 Kesimpulan**

Setelah penelitian berhasil dilakukan maka kesimpulan yang dapat diambil diantaranya sebagai berikut:

1. Penelitian ini berhasil mengembangkan sistem informasi apotek terintegrasi menggunakan *Service Oriented Architecture* dan menggunakan pendekatan berorientasi objek.
2. Penelitian ini berhasil mengimplementasikan *Service Oriented Architecture* dengan menggunakan *web service*.
3. Penelitian ini berhasil mengembangkan sistem pemesanan yang terintegrasi antar apotek yang berbeda, baik berbeda dari segi lokasi apotek maupun pemilik apotek dengan menerapkan konsep *Collaborative E-Commerce*.

#### **6.2 Saran**

Sistem informasi apotek terintegrasi yang dikembangkan ini tidak terlepas dari kelemahan, terutama pada pencarian stok pada aplikasi berbasis *desktop* dan aplikasi berbasis *mobile* yang kurang nyaman baik dari segi antar muka maupun respon pada pengguna yang kurang interaktif.

Selain itu sistem yang dikembangkan ini masih belum memiliki standar *enterprise* untuk digunakan pada proses bisnis yang lebih rumit dan lebih kompleks. Hal ini dikarenakan keterbatasan dari penulis untuk mengembangkan

sistem yang lebih kompleks dan rumit. Dari sebab itu penulis menyarankan beberapa hal, diantaranya:

1. Sistem pemesanan perlu dikembangkan dengan pendekatan kepada pengguna yang lebih baik dan *user friendly*.
2. Sistem sebaiknya menyediakan fasilitas notifikasi pesanan secara *real time* ketika ada pesanan baru sehingga lebih memudahkan dalam melakukan konfirmasi pemesanan.
3. Sistem dikembangkan dengan menambahkan fitur transaksi yang terintegrasi dengan sistem lokal, seperti *update* stok obat secara langsung pada basis data apotek yang menjadi *member*.
4. Ditambahkan suatu mekanisme *fee* atau biaya tambahan yang dibebankan kepada apotek pemesan obat yang menjadi *member*.
5. Sistem sebaiknya dikembangkan dengan standar *enterprise*, sehingga memungkinkan pihak luar yang bisa memanfaatkan untuk keperluan yang lebih kompleks.

Akhirnya dengan semua keterbatasan dan kekurangan sistem yang telah dibuat ini, penulis berharap sistem informasi apotek terintegrasi ini memberikan pandangan baru bagi para pembaca untuk mengembangkan lebih lanjut.

## DAFTAR PUSTAKA

- Arora, Geetanjali. dan Kishore, Sai. 2002. *XML Web Services Professional Projects*. Ohio: Premier Press.
- Basuki, Awan P. 2010. *Membangun Web Berbasis PHP dengan Framework CodeIgniter*. Yogyakarta: Penerbit Lokomedia.
- Bean, James. 2010. *SOA and Web Services Interface Design*. USA: Elsevier.
- CodeIgniter. 2010. *CodeIgniter Version User Guide 1.7.2*. [www.codeigniter.com](http://www.codeigniter.com).
- Converse, T., Park, J., dan Morgan C. 2004. *PHP 5 and MySQL Bible*. USA: Wiley Publishing.
- Deviana, Hartati. 2007. *Naskah Publikasi*. Palembang. [http://www.unsri.ac.id/upload/arsip/Hartati Deviana/](http://www.unsri.ac.id/upload/arsip/Hartati%20Deviana/) diakses tanggal 6 Maret 2012.
- Douglas, Korry. dan Douglas, Susan. 2005. *PostgreSQL: The comprehensive guide to building, programming, and administering PostgreSQL databases, Second Edition*. USA: Sams Publishing.
- Dyer, Russel. 2008. *MySQL in a Nutshell*. USA: O'Reilly Media.
- Dykes, Lucinda. dan Tittel, Ed. 2005. *XML for Dummies 4th Edition*. USA: Wiley Publishing.
- Erl, Thomas. 2005. *Service-Oriented Architecture: Concepts, Technology, and Design*. USA: Prentice Hall.
- Graham, Ian. 2008. *Requirements Modelling and Specification for Service Oriented Architecture*. UK: John Wiley & Sons.
- Grundgeiger, Dave. 2002. *Programming Visual Basic .NET*. USA: O'Reilly Publisher.
- Kadir, Abdul. 2003. *Pengenalan Sistem Informasi*. Yogyakarta: Penerbit Andi.
- Kadir, Abdul. 2008. *Tuntunan Praktis: Belajar Database Menggunakan MySQL*. Yogyakarta: Penerbit Andi.
- Kar, E.V.D, dan Verbraeck, A. 2007. *Designing Mobile Service Systems*. The Netherlands: IOS Press.

- Kendal, Simon. 2009. *Object Oriented Programming using Java*. USA: Simon Kendal & Ventus Publishing.
- Kusumo, Singgih. 2010. *Perancangan Sistem Informasi Pengolahan Data Pembelian dan Penjualan Obat di Apotek 55 Bandung*. Skripsi, Fakultas Teknik dan Ilmu Komputer Universitas Komputer Indonesia (UNIKOM), Bandung.
- Ling, Team, 2005. *MySQL Language Reference*. USA: MySQL Press.
- Linthicum, David S. 2010. *Cloud Computing and SOA Convergence in Your Enterprise: A Step-by-Step Guide*. USA: Addison-Wesley.
- Lucky. 2008. *XML Web Service Aplikasi Desktop, Internet & Handphone*. Jakarta: Jasakom.
- Muchow, John W. 2001. *Core J2ME Technology & MIDP*. USA: Prentice Hall.
- Mustofa, Eka. 2010. *Rekayasa Perangkat Lunak Customer Relationship Management (CRM) Marketing Automation dan Customer Support Menggunakan Metode Service Oriented Architecture (SOA)*. Skripsi, Fakultas Teknologi Informasi Institut Teknologi Sepuluh Nopember (ITS), Surabaya.
- Myer, Thomas. 2008. *Professional CodeIgniter*. USA: Wiley Publishing.
- Naja, Farah. 2010. *Rancang Bangun Perangkat Lunak Aplikasi Pelayanan Kesehatan Berbasis Service-Oriented Architecture*. Skripsi, Fakultas Teknologi Informasi Institut Teknologi Sepuluh Nopember (ITS), Surabaya.
- Njoo, Ricky F. 2009. *Sistem Informasi Penjualan dan Pembelian yang Terintegrasi dengan Inventory pada Apotek K-24 Manyar*. Laporan Kerja Praktek, Sekolah Tinggi Manajemen Informatika dan Teknik Komputer (STIKOM), Surabaya.
- Permana, Agung S. 2010. *Rancang Bangun Perangkat Lunak General Ledger dan Cash Management Untuk Perusahaan Manufaktur Menggunakan Metode Service Oriented Architecture (SOA)*. Skripsi, Fakultas Teknologi Informasi Institut Teknologi Sepuluh Nopember (ITS), Surabaya.
- Pressman, Roger S. 2005. *Software Engineering A Practitioner's Approach Sixth Edition*. New York: Mc Graw Hill Inc.
- Pungus, Stenly R. 2008. *Penerapan Service Oriented Architecture Untuk Pengintegrasian Sistem Informasi Perguruan Tinggi (Studi Kasus:*

- Universitas Klabat Manado*). Tesis. Sekolah Teknik Elektro dan Informatika ITB, Bandung.
- Raharjo, Budi. 2007. *Mudah Belajar Java*. Bandung: Penerbit Informatika.
- Riggs, Roger., dkk. 2003. *Programming Wireless Devices with the Java™ 2 Platform, Micro Edition, Second Edition*. USA: Addison Wesley.
- Riggs, Simon. dan Krosing, Hannu. 2010. *PostgreSQL 9 Administration Cookbook*. UK: Packt Publishing.
- Rosa, A.S., dan Shalahuddin, M. 2011. *Modul Pembelajaran Rekayasa Perangkat Lunak (Terstruktur dan Berorientasi Objek)*. Bandung: Penerbit Modula.
- Samtani, Gunjam. 2002. *B2B Integration: A Practical Guide to Collaborative E-Commerce*. London: Imperial College Press.
- Schlesinger, Richard. 2010. *Developing Real World Software*. USA: Jones and Bartlett Publishers.
- Schmuller, Joseph. 2004. *Sams Teach Yourself UML in 24 Hours, Third Edition*. USA: Sams Publishing.
- Schmutz, G., Liebhart, D., dan Welkenbach, P. 2010. *Service-Oriented Architecture: An Integration Blueprint*. UK: Packt Publishing.
- Shalahuddin, M., dan A.S., Rosa. 2008. *Pemrograman J2ME (Belajar Cepat Pemrograman Perangkat Telekomunikasi Mobile)*. Bandung: Penerbit Informatika.
- Sheldon, Bill., dkk. 2010. *Professional Visual Basic 2010 and .NET 4*. USA: Wiley Publishing, Inc.
- Wandschneider, Marc. 2005. *Core Web Application Development With PHP and MySQL*. USA: Addison-Wesley.

## LAMPIRAN A

### Kode WSDL (Web Service Description Language)

#### **File *layanan\_obat.wsdl***

```

<?xml version="1.0" encoding="ISO-8859-1"?>
<definitions xmlns:SOAP-
ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:SOAP-
ENC="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:tns="urn:Layanan-Apotek-Terintegrasi"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
xmlns="http://schemas.xmlsoap.org/wsdl/"
targetNamespace="urn:Layanan-Apotek-Terintegrasi">
<types>
<xsd:schema targetNamespace="urn:Layanan-Apotek-Terintegrasi"
>
<xsd:import namespace="http://schemas.xmlsoap.org/soap/encoding/"
/>
<xsd:import namespace="http://schemas.xmlsoap.org/wsdl/" />
<xsd:complexType name="obat">
<xsd:all>
<xsd:element name="kode_obat" type="xsd:string"/>
<xsd:element name="nama_obat" type="xsd:string"/>
<xsd:element name="harga_jual" type="xsd:long"/>
<xsd:element name="stok" type="xsd:long"/>
<xsd:element name="nama_apotek" type="xsd:string"/>
</xsd:all>
</xsd:complexType>
<xsd:complexType name="obatArray">
<xsd:complexContent>
<xsd:restriction base="SOAP-ENC:Array">
<xsd:attribute ref="SOAP-ENC:arrayType"
wsdl:arrayType="tns:obat[]" />
</xsd:restriction>
</xsd:complexContent>
</xsd:complexType>
</xsd:schema>
</types>
<message name="getInfoObatApotek55Request">
<part name="nama_obat" type="xsd:string" /></message>
<message name="getInfoObatApotek55Response">
<part name="return" type="tns:obatArray" /></message>
<message name="getObatApotek55Request">
<part name="kode_obat" type="xsd:string" /></message>
<message name="getObatApotek55Response">
<part name="return" type="tns:obat" /></message>
<message name="getInfoObatApotekK24ManyarRequest">
<part name="nama_obat" type="xsd:string" /></message>
<message name="getInfoObatApotekK24ManyarResponse">
<part name="return" type="tns:obatArray" /></message>
<message name="getObatApotekK24ManyarRequest">

```



```

    <part name="kode_obat" type="xsd:string" /></message>
<message name="getObatApotekK24ManyarResponse">
  <part name="return" type="tns:obat" /></message>
<message name="getInfoObatSemuaLokasiRequest">
  <part name="nama_obat" type="xsd:string" /></message>
<message name="getInfoObatSemuaLokasiResponse">
  <part name="return" type="tns:obatArray" /></message>
<message name="getInfoObatSemuaLokasi2Request">
  <part name="kode_obat" type="xsd:string" /></message>
<message name="getInfoObatSemuaLokasi2Response">
  <part name="return" type="tns:obat" /></message>
<message name="getObatPilihLokasiRequest">
  <part name="kode_obat" type="xsd:string" />
  <part name="nama_apotek" type="xsd:string" /></message>
<message name="getObatPilihLokasiResponse">
  <part name="return" type="tns:obat" /></message>
<message name="getInfoObatPilihLokasiRequest">
  <part name="nama_obat" type="xsd:string" />
  <part name="nama_apotek" type="xsd:string" /></message>
<message name="getInfoObatPilihLokasiResponse">
  <part name="return" type="tns:obatArray" /></message>
<portType name="Layanan Apotek Terintegrasi Suka FarmaPortType">
  <operation name="getInfoObatApotek55">
    <input message="tns:getInfoObatApotek55Request"/>
    <output message="tns:getInfoObatApotek55Response"/>
  </operation>
  <operation name="getObatApotek55">
    <input message="tns:getObatApotek55Request"/>
    <output message="tns:getObatApotek55Response"/>
  </operation>
  <operation name="getInfoObatApotekK24Manyar">
    <input message="tns:getInfoObatApotekK24ManyarRequest"/>
    <output message="tns:getInfoObatApotekK24ManyarResponse"/>
  </operation>
  <operation name="getObatApotekK24Manyar">
    <input message="tns:getObatApotekK24ManyarRequest"/>
    <output message="tns:getObatApotekK24ManyarResponse"/>
  </operation>
  <operation name="getInfoObatSemuaLokasi">
    <input message="tns:getInfoObatSemuaLokasiRequest"/>
    <output message="tns:getInfoObatSemuaLokasiResponse"/>
  </operation>
  <operation name="getInfoObatSemuaLokasi2">
    <input message="tns:getInfoObatSemuaLokasi2Request"/>
    <output message="tns:getInfoObatSemuaLokasi2Response"/>
  </operation>
  <operation name="getObatPilihLokasi">
    <input message="tns:getObatPilihLokasiRequest"/>
    <output message="tns:getObatPilihLokasiResponse"/>
  </operation>
  <operation name="getInfoObatPilihLokasi">
    <input message="tns:getInfoObatPilihLokasiRequest"/>
    <output message="tns:getInfoObatPilihLokasiResponse"/>
  </operation>
</portType>

```

```

<binding name="Layanan Apotek Terintegrasi Suka FarmaBinding"
type="tns:Layanan Apotek Terintegrasi Suka FarmaPortType">
  <soap:binding style="rpc"
transport="http://schemas.xmlsoap.org/soap/http"/>
  <operation name="getInfoObatApotek55">
    <soap:operation soapAction="urn:Layanan-Apotek-
Terintegrasi#getInfoObatApotek55" style="rpc"/>
    <input><soap:body use="encoded" namespace="urn:Layanan-Apotek-
Terintegrasi"
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"></input
>
    <output><soap:body use="encoded" namespace="urn:Layanan-
Apotek-Terintegrasi"
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"></outpu
t>
  </operation>
  <operation name="getObatApotek55">
    <soap:operation soapAction="urn:Layanan-Apotek-
Terintegrasi#getObatApotek55" style="rpc"/>
    <input><soap:body use="encoded" namespace="urn:Layanan-Apotek-
Terintegrasi"
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"></input
>
    <output><soap:body use="encoded" namespace="urn:Layanan-
Apotek-Terintegrasi"
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"></outpu
t>
  </operation>
  <operation name="getInfoObatApotekK24Manyar">
    <soap:operation soapAction="urn:Layanan-Apotek-
Terintegrasi#getInfoObatApotekK24Manyar" style="rpc"/>
    <input><soap:body use="encoded" namespace="urn:Layanan-Apotek-
Terintegrasi"
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"></input
>
    <output><soap:body use="encoded" namespace="urn:Layanan-
Apotek-Terintegrasi"
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"></outpu
t>
  </operation>
  <operation name="getObatApotekK24Manyar">
    <soap:operation soapAction="urn:Layanan-Apotek-
Terintegrasi#getObatApotekK24Manyar" style="rpc"/>
    <input><soap:body use="encoded" namespace="urn:Layanan-Apotek-
Terintegrasi"
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"></input
>
    <output><soap:body use="encoded" namespace="urn:Layanan-
Apotek-Terintegrasi"
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"></outpu
t>
  </operation>
  <operation name="getInfoObatSemuaLokasi">
    <soap:operation soapAction="urn:Layanan-Apotek-
Terintegrasi#getInfoObatSemuaLokasi" style="rpc"/>

```

```

    <input><soap:body use="encoded" namespace="urn:Layanan-Apotek-
Terintegrasi"
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"></input
>
    <output><soap:body use="encoded" namespace="urn:Layanan-
Apotek-Terintegrasi"
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"></outpu
t>
  </operation>
  <operation name="getInfoObatSemuaLokasi2">
    <soap:operation soapAction="urn:Layanan-Apotek-
Terintegrasi#getInfoObatSemuaLokasi2" style="rpc"/>
    <input><soap:body use="encoded" namespace="urn:Layanan-Apotek-
Terintegrasi"
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"></input
>
    <output><soap:body use="encoded" namespace="urn:Layanan-
Apotek-Terintegrasi"
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"></outpu
t>
  </operation>
  <operation name="getObatPilihLokasi">
    <soap:operation soapAction="urn:Layanan-Apotek-Terintegrasi#
getObatPilihLokasi" style="rpc"/>
    <input><soap:body use="encoded" namespace="urn:Layanan-Apotek-
Terintegrasi"
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"></input
>
    <output><soap:body use="encoded" namespace="urn:Layanan-
Apotek-Terintegrasi"
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"></outpu
t>
  </operation>
  <operation name="getInfoObatPilihLokasi">
    <soap:operation soapAction="urn:Layanan-Apotek-Terintegrasi#
getInfoObatPilihLokasi" style="rpc"/>
    <input><soap:body use="encoded" namespace="urn:Layanan-Apotek-
Terintegrasi"
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"></input
>
    <output><soap:body use="encoded" namespace="urn:Layanan-
Apotek-Terintegrasi"
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"></outpu
t>
  </operation>
</binding>
<service name="Layanan Apotek Terintegrasi Suka Farma">
  <port name="Layanan Apotek Terintegrasi Suka FarmaPort"
binding="tns:Layanan Apotek Terintegrasi Suka FarmaBinding">
    <soap:address
location="http://localhost/apotek_services/layanan_obat.php"/>
  </port>
</service>
</definitions>

```

**LAMPIRAN B****Kode Sumber Web Service Apotek****File *layanan\_obat.php***

```

<?php

function getInfoObatApotek55($nama_obat) {
    mysql_connect("localhost","root","");
    mysql_select_db("apotek55");

    $hasil = mysql_query("SELECT kode_obat, nama_obat, harga_jual,
stok
                        FROM obat WHERE nama_obat LIKE
'%".$nama_obat."%'");

    $index=0;
    while($data=mysql_fetch_array($hasil)) {
        $obat[$index]=array(
            "kode_obat"=>$data['kode_obat'],
            "nama_obat"=>$data['nama_obat'],
            "harga_jual"=>$data['harga_jual'],
            "stok"=>$data['stok'],
            "nama_apotek"=>'Apotek 55'
        );
        $index++;
    }

    mysql_close();
    return $obat;
}

function getObatApotek55($kode_obat) {
    mysql_connect("localhost","root","");
    mysql_select_db("apotek55");

    $hasil = mysql_query("SELECT kode_obat, nama_obat, harga_jual,
stok
                        FROM obat WHERE kode_obat =
'".$kode_obat."'");

    while($data=mysql_fetch_array($hasil)) {
        $obat=array(
            "kode_obat"=>$data['kode_obat'],
            "nama_obat"=>$data['nama_obat'],
            "harga_jual"=>$data['harga_jual'],
            "stok"=>$data['stok'],
            "nama_apotek"=>'Apotek 55'
        );
    }

    mysql_close();
    return $obat;
}

```

```

function getInfoObatApotekK24Manyar($nama_obat) {
    $db_host = "localhost";
    $db_user = "azkohaq";
    $db_pass = "06061988";
    $db_name = "apotek_manyar";

    $pg = pg_connect("host=$db_host user=$db_user password=$db_pass
dbname=$db_name");

    $hasil = pg_query("SELECT mb.kode_barang, mb.nama_barang,
db.harga_jual, ks.qty
FROM master_barang mb, detail_barang db,
kartu_stok ks
WHERE mb.kode_barang = db.kode_barang AND
mb.kode_barang = ks.kode_barang AND
mb.nama_barang
ILIKE '%" . $nama_obat . "%'");

    $index=0;
    while($data=pg_fetch_array($hasil)) {
        $obat[$index]=array(
            "kode_obat"=>$data['kode_barang'],
            "nama_obat"=>$data['nama_barang'],
            "harga_jual"=>$data['harga_jual'],
            "stok"=>$data['qty'],
            "nama_apotek"=>'Apotek K24 Manyar'
        );
        $index++;
    }

    pg_close();
    return $obat;
}

function getObatApotekK24Manyar($kode_obat) {
    $db_host = "localhost";
    $db_user = "azkohaq";
    $db_pass = "06061988";
    $db_name = "apotek_manyar";

    $pg = pg_connect("host=$db_host user=$db_user password=$db_pass
dbname=$db_name");

    $hasil = pg_query("SELECT mb.kode_barang, mb.nama_barang,
db.harga_jual, ks.qty
FROM master_barang mb, detail_barang db,
kartu_stok ks
WHERE mb.kode_barang = db.kode_barang AND
mb.kode_barang = ks.kode_barang AND
mb.kode_barang = '" . $kode_obat . "'");

    while($data=pg_fetch_array($hasil)) {
        $obat=array(
            "kode_obat"=>$data['kode_barang'],
            "nama_obat"=>$data['nama_barang'],
            "harga_jual"=>$data['harga_jual'],

```

```

        "stok"=>$data['qty'],
        "nama_apotek"=>'Apotek K24 Manyar'
    );
}

pg_close();
return $obat;
}

function getInfoObatSemuaLokasi($nama_obat) {

    /* Apotek 55 */
    mysql_connect("localhost","root","");
    mysql_select_db("apotek55");

    $hasil1 = mysql_query("SELECT kode_obat, nama_obat, harga_jual,
stok
                                FROM obat WHERE nama_obat LIKE
'%".$nama_obat."%'");

    $index1=0;
    while($data1=mysql_fetch_array($hasil1)) {
        $obat1[$index1]=array(
            "kode_obat"=>$data1['kode_obat'],
            "nama_obat"=>$data1['nama_obat'],
            "harga_jual"=>$data1['harga_jual'],
            "stok"=>$data1['stok'],
            "nama_apotek"=>'Apotek 55'
        );
        $index1++;
    }

    mysql_close();

    /* Apotek K24 Manyar */
    $db_host = "localhost";
    $db_user = "azkohaq";
    $db_pass = "06061988";
    $db_name = "apotek_manyar";

    $pg = pg_connect("host=$db_host user=$db_user password=$db_pass
dbname=$db_name");

    $hasil2 = pg_query("SELECT mb.kode_barang, mb.nama_barang,
db.harga_jual, ks.qty
                                FROM master_barang mb, detail_barang db,
kartu_stok ks
                                WHERE mb.kode_barang = db.kode_barang AND
mb.kode_barang = ks.kode_barang AND
mb.nama_barang
                                ILIKE '%".$nama_obat."%'");

    $index2=0;
    while($data2=pg_fetch_array($hasil2)) {
        $obat2[$index2]=array(
            "kode_obat"=>$data2['kode_barang'],

```

```

        "nama_obat"=>$data2['nama_barang'],
        "harga_jual"=>$data2['harga_jual'],
        "stok"=>$data2['qty'],
        "nama_apotek"=>'Apotek K24 Manyar'
    );
    $index2++;
}

pg_close();

$obat = array_merge(
    (array)$obat1,
    (array)$obat2
);

return $obat;
}

function getInfoObatSemuaLokasi2($kode_obat) {
    /* koneksi mysql */
    mysql_connect("localhost","root","");
    mysql_select_db("apotek55");

    $hasil = mysql_query("SELECT kode_obat, nama_obat, harga_jual,
stok
                        FROM obat WHERE kode_obat =
'".$kode_obat."'");

    while($data=mysql_fetch_array($hasil)) {
        $obat=array(
            "kode_obat"=>$data['kode_obat'],
            "nama_obat"=>$data['nama_obat'],
            "harga_jual"=>$data['harga_jual'],
            "stok"=>$data['stok'],
            "nama_apotek"=>'Apotek 55'
        );
    }

    mysql_close();

    /* koneksi postgre */
    $db_host = "localhost";
    $db_user = "azkohaq";
    $db_pass = "06061988";
    $db_name = "apotek_manyar";

    $pg = pg_connect("host=$db_host user=$db_user password=$db_pass
dbname=$db_name");

    $hasil = pg_query("SELECT mb.kode_barang, mb.nama_barang,
db.harga_jual, ks.qty
                    FROM master_barang mb, detail_barang db,
kartu_stok ks
                    WHERE mb.kode_barang = db.kode_barang AND
                        mb.kode_barang = ks.kode_barang AND
mb.kode_barang = '".$kode_obat."'");

```

```

while($data=pg_fetch_array($hasil)) {
    $obat=array(
        "kode_obat"=>$data['kode_barang'],
        "nama_obat"=>$data['nama_barang'],
        "harga_jual"=>$data['harga_jual'],
        "stok"=>$data['qty'],
        "nama_apotek"=>'Apotek K24 Manyar'
    );
}

pg_close();
return $obat;

}

function getObatPilihLokasi($kode_obat, $nama_apotek) {

    switch($nama_apotek) {
        case "Apotek 55": return getObatApotek55($kode_obat); break;
        case "Apotek K24 Manyar": return
getObatApotekK24Manyar($kode_obat); break;
    }
}

function getInfoObatPilihLokasi($nama_obat, $nama_apotek) {

    switch($nama_apotek) {
        case "Semua Apotek": return
getInfoObatSemuaLokasi($nama_obat); break;
        case "Apotek 55": return getInfoObatApotek55($nama_obat);
break;
        case "Apotek K24 Manyar": return
getInfoObatApotekK24Manyar($nama_obat); break;
    }
}

require("lib/nusoap.php");

$server = new soap_server();
$server->configureWSDL("Layanan Apotek Terintegrasi Suka
Farma","urn:Layanan-Apotek-Terintegrasi");

$server->wsdl->addComplexType(
    "obat",
    "complexType",
    "struct",
    "all",
    "",
    array(

"kode_obat"=>array("name"=>"kode_obat","type"=>"xsd:string"),

"nama_obat"=>array("name"=>"nama_obat","type"=>"xsd:string"),

```



```

"harga_jual"=>array("name"=>"harga_jual", "type"=>"xsd:long"),
    "stok"=>array("name"=>"stok", "type"=>"xsd:long"),

"nama_apotek"=>array("name"=>"nama_apotek", "type"=>"xsd:string")
)
);

$server->wsdl->addComplexType (
    "obatArray",
    "complexType",
    "array",
    "",
    "SOAP-ENC:Array",
    array(),
    array(
        array("ref"=>"SOAP-
ENC:arrayType", "wsdl:arrayType"=>"tns:obat[]")
    ),
    "tns:obat"
);

$server-
>register ("getInfoObatApotek55", array("nama_obat"=>"xsd:string"),
    array("return"=>"tns:obatArray"), "urn:Layanan-Apotek-
Terintegrasi",
    "urn:Layanan-Apotek-Terintegrasi#getInfoObatApotek55");

$server-
>register ("getObatApotek55", array("kode_obat"=>"xsd:string"),
    array("return"=>"tns:obat"), "urn:Layanan-Apotek-
Terintegrasi",
    "urn:Layanan-Apotek-Terintegrasi#getObatApotek55");

$server-
>register ("getInfoObatApotekK24Manyar", array("nama_obat"=>"xsd:st
ring"),
    array("return"=>"tns:obatArray"), "urn:Layanan-Apotek-
Terintegrasi",
    "urn:Layanan-Apotek-
Terintegrasi#getInfoObatApotekK24Manyar");

$server-
>register ("getObatApotekK24Manyar", array("kode_obat"=>"xsd:string"
),
    array("return"=>"tns:obat"), "urn:Layanan-Apotek-
Terintegrasi",
    "urn:Layanan-Apotek-
Terintegrasi#getObatApotekK24Manyar");

$server-
>register ("getInfoObatSemuaLokasi", array("nama_obat"=>"xsd:string"
),
    array("return"=>"tns:obatArray"), "urn:Layanan-Apotek-
Terintegrasi",

```

```

        "urn:Layanan-Apotek-
Terintegrasi#getInfoObatSemuaLokasi");

$server-
>register("getInfoObatSemuaLokasi2",array("kode_obat"=>"xsd:string
"),
        array("return"=>"tns:obat"),"urn:Layanan-Apotek-
Terintegrasi",
        "urn:Layanan-Apotek-
Terintegrasi#getInfoObatSemuaLokasi2");

$server-
>register("getObatPilihLokasi",array("kode_obat"=>"xsd:string",
        "nama_apotek"=>"xsd:string"),
array("return"=>"tns:obat"),
        "urn:Layanan-Apotek-Terintegrasi","urn:Layanan-Apotek-
Terintegrasi#
getObatPilihLokasi");

$server-
>register("getInfoObatPilihLokasi",array("nama_obat"=>"xsd:string"
,
        "nama_apotek"=>"xsd:string"),
array("return"=>"tns:obatArray"),
        "urn:Layanan-Apotek-Terintegrasi","urn:Layanan-Apotek-
Terintegrasi#
getInfoObatPilihLokasi");

$HTTP_RAW_POST_DATA=isset($HTTP_RAW_POST_DATA)?
$HTTP_RAW_POST_DATA : "";

$server->service($HTTP_RAW_POST_DATA);
?>

```

## LAMPIRAN C

### Kode File Konfigurasi Database

#### **File *database.php***

```
<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');
```

```
$active_group = 'default';  
$active_record = TRUE;
```

```
$db['default']['hostname'] = 'localhost';  
$db['default']['username'] = 'azkohaq';  
$db['default']['password'] = '06061988';  
$db['default']['database'] = 'apotek';  
$db['default']['dbdriver'] = 'postgre';  
$db['default']['dbprefix'] = '';  
$db['default']['pconnect'] = TRUE;  
$db['default']['db_debug'] = TRUE;  
$db['default']['cache_on'] = FALSE;  
$db['default']['cachedir'] = '';  
$db['default']['char_set'] = 'utf8';  
$db['default']['dbcollat'] = 'utf8_general_ci';  
$db['default']['swap_pre'] = '';  
$db['default']['autoinit'] = TRUE;  
$db['default']['stricton'] = FALSE;
```

## LAMPIRAN D

### Kode File Controller

#### **File *c\_pemesanan.php***

```
<?php

class C_Pemesanan extends CI_Controller {

    function __construct() {
        parent::__construct();
        $this->load->library('cart');
        $this->load->model('m_member', '', TRUE);
        $this->load->model('m_pemesanan', '', TRUE);
    }

    var $limit = 10;

    function index() {
        if (($this->session->userdata('login') == TRUE) AND
            ($this->session->userdata('hak_akses') == 'member')) {

            $this->pesan_obat();
        }
        else {
            redirect('c_login_member');
        }
    }

    function pesan_obat() {
        if (($this->session->userdata('login') == TRUE) AND
            ($this->session->userdata('hak_akses') == 'member')) {

            $data['h2_title'] = 'Pesan Obat';
            $data['main_view'] = 'member/pemesanan/pemesanan';
            $data['form_action'] =
site_url('c_pemesanan/pesan_obat');

            require_once(BASEPATH.'libraries/nusoap/nusoap'.EXT);

$alamat_service="http://localhost/apotek_services/layanan_obat.php
";
            $klien=new soapclient($alamat_service);

            $this->form_validation->set_rules('nama_apotek',
                'Nama Apotek', 'required');

            $this->form_validation->set_rules('kunci',
                'Kata Kunci', 'required|min_length[3]');

            $member = $this->m_member->get_member()->result();
            $data['list_member']['Semua Apotek'] = 'Semua Apotek';
            foreach($member as $row) {
                $data['list_member'][$row->nama_apotek] = $row-
>nama_apotek;
```

```

    }

    if ($this->input->post('submit') && ($this-
>form_validation->run() == TRUE)) {
        $hasil=$klien-
>call("getInfoObatPilihLokasi",array("nama_obat"=>$this->input-
>post('kunci'),"nama_apotek"=>$this->input->post('nama_apotek')));

        $kunci_pencarian = $this->input->post('kunci');
        $lokasi_pencarian = $this->input-
>post('nama_apotek');

        $err=$klien->getError();
        if($err){
            $data['message_salah'] = 'Terdapat kesalahan
sistem dengan pesan ('.$klien->getError().')';
        }
        else {

            if($hasil!=null){

                $total = count($hasil);
                $data['message'] = 'Ditemukan <b>'.$total.'</b> data
obat dengan kata kunci pencarian
                <b>'.$kunci_pencarian.'</b> pada
                <b>'.$lokasi_pencarian.'</b>';

                $tmpl = array( 'table_open' => '<table border="1"
style="border-collapse:collapse"
                cellpadding="0"
cellspacing="0">', 'row_alt_start' => '<tr class="zebra">',
                'row_alt_end' => '</tr>');

                $this->table->set_template($tmpl);
                $this->table->set_empty("&nbsp;");
                $this->table->set_heading('No', 'Kode Obat', 'Nama
Obat', 'Harga',
                'Stok', 'Nama Apotek Penjual', 'Aksi');
                $j = 1;
                for($i=0;$i<sizeof($hasil);$i++){
                    $this->table->add_row(
                    $j++,
                        $hasil[$i]['kode_obat'],
                        $hasil[$i]['nama_obat'],
                        $hasil[$i]['harga_jual'],
                        $hasil[$i]['stok'],
                        $hasil[$i]['nama_apotek'],

                        anchor('c_pemesanan/tambah_cart/'.$.$kode_obat =
$hasil[$i]['kode_obat'].'/'.$.$nama_apotek =
$hasil[$i]['nama_apotek'],
                            '<b>Beli</b>',array('class' => 'cart',
                            'onclick'=>"return confirm('Apakah benar ingin
membeli obat ini?')
                            ")
                        );
                };
            }
        }
    }
}

```

```

    }
    $data['table'] = $this->table->generate();
    }
    else {
        $data['peringatan'] = 'Tidak ditemukan data obat dengan
kata kunci pencarian
        <b>'. $kunci_pencarian. '</b> pada
<b>'. $lokasi_pencarian. '</b>';
    }
    }
    }
    $this->load->view('member', $data);

    }
    else {
        redirect('c_login_member');
    }
}

function tambah_cart($kode_obat, $nama_apotek) {
    $sid = $this->uri->segment(3);
    $name = $this->uri->segment(4);
    $cek = str_replace('%20', ' ', $name);
    $kode_obat = $sid;
    $nama_apotek = $cek;

    require_once(BASEPATH.'libraries/nusoap/nusoap'.EXT);

    $alamat_service="http://localhost/apotek_services/layanan_obat.php
";
    $klien=new soapclient($alamat_service);

    $hasil=$klien-
>call("getObatPilihLokasi",array("kode_obat"=>$kode_obat,"nama_apo
tek"=>$nama_apotek));

    $data = array(
        'id'      => $hasil['kode_obat'],
        'qty'     => 1,
        'price'   => $hasil['harga_jual'],
        'name'    => $hasil['nama_obat'],
        'stok'    => $hasil['stok'],
        'apotek'  => $hasil['nama_apotek']
    );

    $this->cart->insert($data);
    redirect('c_pemesanan/cart_sementara');
}

function cart_sementara($offset = 0) {
    if (($this->session->userdata('login') == TRUE) AND
        ($this->session->userdata('hak_akses') == 'member')) {

        $data['h2_title'] = 'Keranjang Belanja';
        $data['main_view'] = 'member/pemesanan/cart';
    }
}

```

```

$cart_list = $this->cart->contents();
$data['cart_list'] = $cart_list;
$this->load->view('member', $data);

}
else {
    redirect('c_login_member');
}
}

function aksi_cart() {
    $aksi = $this->input->post('submit');

    if ($aksi == 'Perbarui') {
        $this->ubah_cart;
    }

    if ($aksi == 'Checkout') {
        action('c_pemesanan/checkout');
    }
}

function ubah_cart() {
    $item = $this->input->post('rowid');
    $qty = $this->input->post('qty');

    $total = count($item)-1;

    for($i=0; $i<= $total;$i++) {
        $data = array(
            'rowid' => $item[$i],
            'qty' => $qty[$i]
        );
        $this->cart->update($data);
    }

    redirect('c_pemesanan/cart_sementara');
}

function hapus_item($rowid) {
    $data = array(
        'rowid' => $rowid,
        'qty' => 0
    );

    $this->cart->update($data);
    redirect('c_pemesanan/cart_sementara');
}

function hapus_cart() {
    $this->cart->destroy();
    redirect('c_pemesanan/cart_sementara');
}

function checkout() {
    if (($this->session->userdata('login') == TRUE) AND

```

```

        ($this->session->userdata('hak_akses') == 'member')) {

            $data['h2_title'] = 'Simpan Pesanan';
            $data['main_view'] = 'member/pemesanan/checkout';
            $data['form_action'] =
site_url('c_pemesanan/simpan_pesanan');

            $cart_list = $this->cart->contents();
            $data['cart_list'] = $cart_list;

            $pemesan = $this->m_pemesanan->data_pemesan()->row();
            $data['nama_apotek'] = $pemesan->nama_apotek;
            $data['email'] = $pemesan->email;
            $data['alamat'] = $pemesan->alamat;
            $data['kota'] = $pemesan->kota;
            $data['provinsi'] = $pemesan->provinsi;
            $data['kode_pos'] = $pemesan->kode_pos;
            $data['telepon'] = $pemesan->telepon;

            $this->load->view('member', $data);
        }
    }

function simpan_pesanan() {
    if (($this->session->userdata('login') == TRUE) AND
        ($this->session->userdata('hak_akses') == 'member')) {

        $data['h2_title'] = 'Simpan Pesanan';
        $data['main_view'] = 'member/pemesanan/checkout';

        if ($this->input->post('submit')) {

            $this->m_pemesanan->simpan_order();
            $this->cart->destroy();

            redirect('c_awal_member');
        }

        $this->load->view('member', $data);
    }
}

function manajemen_pemesanan($offset=0) {
    if (($this->session->userdata('login') == TRUE) AND
        ($this->session->userdata('hak_akses') == 'member')) {

        $data['h2_title'] = 'Manajemen Pemesanan';
        $data['main_view'] =
'member/pemesanan/manajemen_pemesanan';
        $uri_segment = 3;
        $offset = $this->uri->segment($uri_segment);

        $pemesanan = $this->m_pemesanan->manajemen_pemesanan
($this->limit, $offset)->result();
        $num_rows = $this->m_pemesanan->count_all_num_rows();
    }
}

```



```

if ($num_rows > 0 ) {
    $config['base_url'] =
    site_url('c_pemesanan/manajemen_pemesanan');
    $config['total_rows'] = $num_rows;
    $config['per_page'] = $this->limit;
    $config['uri_segment'] = $uri_segment;
    $config['next_link'] = '<b>&rsaquo;</b>';
    $config['prev_link'] = '<b>&lsaquo;</b>';
    $config['last_link'] = '<b>&raquo;</b>';
    $config['first_link'] = '<b>&laquo;</b>';
    $this->pagination->initialize($config);
    $data['pagination'] =
    $this->pagination->create_links();

    $tmpl = array( 'table_open' => '<table border="1"
style="border-collapse:collapse"
    cellpadding="0" cellspacing="0">', 'row_alt_start' =>
'<tr class="zebra">',
    'row_alt_end' => '</tr>');

    $this->table->set_template($tmpl);
    $this->table->set_empty("&nbsp;");
    $this->table->set_heading('No', 'ID Pesanan', 'Apotek
Pemesan', 'Total Biaya (Rp)',
    'Tgl. Pesan', 'Status', 'Aksi');

    $i = 1 + $offset;
    foreach ($pemesanan as $pemesanan) {
    $this->table->add_row(
    $i++,
    $pemesanan->id_pemesanan,
    $pemesanan->nama_apotek,
    $pemesanan->total_biaya,
    date('j/M/Y H:i:s', (strtotime($pemesanan-
>tgl_pemesanan))),
    $pemesanan->status,

    anchor('c_pemesanan/detail_pemesanan/'.$pemesanan-
>id_pemesanan.'/'.$pemesanan->id_member_pemesan,
    'Detail',array('class' => 'detail')).' '.

    anchor('c_pemesanan/proses_pemesanan/'.$pemesanan-
>id_pemesanan.'/'.$pemesanan->id_member_pemesan,
    'Proses',array('class' => 'update')).' '.

    anchor('c_pemesanan/hapus_pemesanan/'.$pemesanan-
>id_pemesanan,
    'Hapus',array('class' => 'delete',
    'onclick'=>"return confirm('Apakah anda ingin
menghapus data ini?')
    ")
    );
    }
    $data['table'] = $this->table->generate();
}
else {

```

```

        $data['message'] = 'Tidak ditemukan data...';
    }

    $data['pencarian'] = array('link_cari' =>
    anchor('c_pemesanan/cari_pemesanan/', 'Pencarian',
    array('class' => 'detail')));

    $this->load->view('member', $data);

    }
    else {
        redirect('c_login_member');
    }
}

function detail_pemesanan($id_pemesanan, $id_member_pemesan)
{
    if (($this->session->userdata('login') == TRUE) AND
        ($this->session->userdata('hak_akses') == 'member')) {

        $data['h2_title'] = 'Detail Pemesanan';
        $data['main_view'] = 'member/pemesanan/detail_pemesanan';

        $pemesanan = $this->m_pemesanan->detail_pemesanan
        ($id_pemesanan)->result();
        $num_rows = $this->m_pemesanan->
>count_all_detail($id_pemesanan);

        if ($num_rows > 0) {
            $config['base_url'] =
            site_url('c_pemesanan/detail_pemesanan');

            $tmpl = array( 'table_open' => '<table border="1"
style="border-collapse:collapse"
            cellpadding="0" cellspacing="0">', 'row_alt_start' =>
'<tr class="zebra">',
            'row_alt_end' => '</tr>');

            $this->table->set_template($tmpl);
            $this->table->set_empty("&nbsp;");
            $this->table->set_heading('No', 'ID Pemesanan', 'Kode
Barang',
            'Nama Barang', 'Jumlah Pesan', 'Harga Barang',
            'Subtotal');

            $i = 1;
            foreach ($pemesanan as $pemesanan) {
                $this->table->add_row(
                $i++,
                $pemesanan->id_pemesanan,
                $pemesanan->kode_barang,
                $pemesanan->nama_barang,
                $pemesanan->jumlah_barang,
                $pemesanan->harga_barang,
                $pemesanan->subtotal
                );
            }
        }
    }
}

```

```

    }
    $data['table'] = $this->table->generate();

    $pemesan = $this->m_pemesanan-
>detail_pemesan($id_member_pemesan)->row();

    $data['nama_apotek'] = $pemesan->nama_apotek;
    $data['email'] = $pemesan->email;
    $data['alamat'] = $pemesan->alamat;
    $data['kota'] = $pemesan->kota;
    $data['provinsi'] = $pemesan->provinsi;
    $data['kode_pos'] = $pemesan->kode_pos;
    $data['telepon'] = $pemesan->telepon;

    }
    else {
        $data['message'] = 'Tidak ditemukan data...';
    }
    $this->load->view('member', $data);
    }
    else {
        redirect('c_login_member');
    }
}

function proses_pemesanan($id_pemesanan, $id_member_pemesan)
{
    if (($this->session->userdata('login') == TRUE) AND
        ($this->session->userdata('hak_akses') == 'member')) {

        $data['h2_title'] = 'Konfirmasi Pemesanan';
        $data['main_view'] = 'member/pemesanan/proses_pemesanan';
        $data['form_action1'] =
site_url('c_pemesanan/simpan_proses_pemesanan');
        $data['form_action2'] =
site_url('c_pemesanan/ubah_status_pemesanan');

        $pemesanan = $this->m_pemesanan-
>proses_pemesanan($id_pemesanan)->row();

        $data['default']['id_pemesanan'] = $pemesanan-
>id_pemesanan;
        $data['default']['total_harga'] = $pemesanan-
>total_harga;
        $data['default']['status_kirim'] = $pemesanan-
>status_kirim;
        $data['default']['id_status'] = $pemesanan->id_status;

        $status = $this->m_pemesanan->get_status()->result();
        foreach($status as $status) {
            $data['opsi_status'][$status->id_status] = $status-
>status;
        }
    }
}

```

```

        $pemesan = $this->m_pemesanan-
>detail_pemesan($id_member_pemesan)->row();

        $data['nama_apotek'] = $pemesan->nama_apotek;
        $data['email'] = $pemesan->email;
        $data['alamat'] = $pemesan->alamat;
        $data['kota'] = $pemesan->kota;
        $data['provinsi'] = $pemesan->provinsi;
        $data['kode_pos'] = $pemesan->kode_pos;
        $data['telepon'] = $pemesan->telepon;

        $this->load->view('member', $data);
    }
    else {
        redirect('c_login_member');
    }
}

function simpan_proses_pemesanan() {
    if (($this->session->userdata('login') == TRUE) AND
        ($this->session->userdata('hak_akses') == 'member')) {

        $data['h2_title'] = 'Proses Pemesanan';
        $data['main_view'] = 'member/pemesanan/proses_pemesanan';
        $data['form_action'] =
site_url('c_pemesanan/simpan_proses_pemesanan');

        if ($this->input->post('submit')) {

            $id_pemesanan = $this->input->post('id_pemesanan');
            $total_harga = $this->input->post('total_harga');
            $biaya_pengiriman = $this->input-
>post('biaya_pengiriman');
            $total_biaya = $total_harga + $biaya_pengiriman;

            $proses = array('biaya_pengiriman' => $biaya_pengiriman,
                'total_biaya' => $total_biaya);
            $this->m_pemesanan-
>update_proses_pemesanan($id_pemesanan, $proses);

            redirect('c_pemesanan/manajemen_pemesanan');
        }
        $this->load->view('member', $data);

    }
    else {
        redirect('c_login_member');
    }
}

function ubah_status_pemesanan() {
    if (($this->session->userdata('login') == TRUE) AND
        ($this->session->userdata('hak_akses') == 'member')) {

        $data['h2_title'] = 'Proses Pemesanan';
        $data['main_view'] = 'member/pemesanan/proses_pemesanan';
    }
}

```

```

        $data['form_action'] =
site_url('c_pemesanan/ubah_status_pemesanan');

        if ($this->input->post('submit')) {

            $id_status = $this->input->post('id_status');
            $id_pemesanan = $this->input->post('id_pemesanan');

            $proses = array('id_status' => $id_status);
            $this->m_pemesanan->ubah_status_pemesanan($id_pemesanan,
$proses);

            redirect('c_pemesanan/manajemen_pemesanan');
        }
        $this->load->view('member', $data);

    }
    else {
        redirect('c_login_member');
    }
}

function hapus_pemesanan($id_pemesanan) {
    $this->m_pemesanan->hapus_pemesanan($id_pemesanan);
    $this->session->set_flashdata('message',
'Penghapusan data berhasil dilakukan');
    redirect('c_pemesanan/manajemen_pemesanan');
}

function riwayat_pesanan($offset=0) {
    if (($this->session->userdata('login') == TRUE) AND
($this->session->userdata('hak_akses') == 'member')) {

        $data['h2_title'] = 'Riwayat Pesanan';
        $data['main_view'] = 'member/pemesanan/riwayat_pesanan';
        $data['form_action'] =
site_url('c_pembayaran/riwayat_pesanan');
        $uri_segment = 3;
        $offset = $this->uri->segment($uri_segment);

        $pemesanan = $this->m_pemesanan->riwayat_pesanan
($this->limit, $offset)->result();
        $num_rows_riwayat = $this->m_pemesanan-
>count_all_num_rows_riwayat();

        if ($num_rows_riwayat > 0 ) {
            $config['base_url'] =
site_url('c_pemesanan/riwayat_pesanan');
            $config['total_rows'] = $num_rows_riwayat;
            $config['per_page'] = $this->limit;
            $config['uri_segment'] = $uri_segment;
            $config['next_link'] = '<b>&rsaquo;</b>';
            $config['prev_link'] = '<b>&lsaquo;</b>';
            $config['last_link'] = '<b>&raquo;</b>';
            $config['first_link'] = '<b>&laquo;</b>';
            $this->pagination->initialize($config);

```

```

        $data['pagination'] =
        $this->pagination->create_links();

        $tmpl = array( 'table_open' => '<table border="1"
style="border-collapse:collapse"
        cellpadding="0" cellspacing="0">', 'row_alt_start' =>
'<tr class="zebra">',
        'row_alt_end' => '</tr>');

        $this->table->set_template($tmpl);
        $this->table->set_empty(" ");
        $this->table->set_heading('No', 'ID Pesanan', 'Total
Biaya (Rp)',
        'Total Harga (Rp)', 'Biaya Kirim (Rp)', 'Tgl. Pesan',
'Detail');

        $i = 1 + $offset;
        foreach ($pemesanan as $pemesanan) {
        $this->table->add_row(
        $i++,
        $pemesanan->id_pemesanan,
        $pemesanan->total_biaya,
        $pemesanan->total_harga,
        $pemesanan->biaya_pengiriman,
        $pemesanan->tgl_pemesanan,

        anchor('c_pemesanan/detail_pemesanan/'.$pemesanan->id_pemesanan.'/'.'$pemesanan->id_member_pemesan,
        'Detail',array('class' => 'detail'))
        );
        }
        $data['table'] = $this->table->generate();
    }
    else {
        $data['message'] = 'Tidak ditemukan data...';
    }

    $this->load->view('member', $data);

    }
    else {
        redirect('c_login_member');
    }
}

function riwayat_pesanan_popup($offset=0) {
    if (($this->session->userdata('login') == TRUE) AND
        ($this->session->userdata('hak_akses') == 'member')) {

        $uri_segment = 3;
        $offset = $this->uri->segment($uri_segment);

        $pemesanan = $this->m_pemesanan->riwayat_pesanan
        ($this->limit, $offset)->result();
        $num_rows_riwayat = $this->m_pemesanan->count_all_num_rows_riwayat();
    }
}

```

```

if ($num_rows_riwayat > 0 ) {
    $config['base_url'] =
    site_url('c_pemesanan/riwayat_pesanan_popup');
    $config['total_rows'] = $num_rows_riwayat;
    $config['per_page'] = $this->limit;
    $config['uri_segment'] = $uri_segment;
    $config['next_link'] = '<b>&rsaquo;</b>';
    $config['prev_link'] = '<b>&lsaquo;</b>';
    $config['last_link'] = '<b>&raquo;</b>';
    $config['first_link'] = '<b>&laquo;</b>';
    $this->pagination->initialize($config);
    $data['pagination'] =
    $this->pagination->create_links();

    $tmpl = array( 'table_open' => '<table border="1"
style="border-collapse:collapse"
    cellpadding="0" cellspacing="0">', 'row_alt_start' =>
'<tr class="zebra">',
    'row_alt_end' => '</tr>');

    $this->table->set_template($tmpl);
    $this->table->set_empty("&nbsp;");
    $this->table->set_heading('No', 'ID Pesanan', 'Total
Biaya <br>Yang Harus Dibayar (Rp)',
    'Tgl. Pesan', 'Cara Pembayaran');

    $i = 1 + $offset;
    foreach ($pemesanan as $pemesanan) {
    $this->table->add_row(
    $i++,
    $pemesanan->id_pemesanan,
    $pemesanan->total_biaya,
    $pemesanan->tgl_pemesanan,

    anchor('c_pemesanan/detail_riwayat_pesanan_popup/'.$pemesanan-
>id_pemesanan.'/'.$pemesanan->id_member_tujuan,
    '<b>Lihat Disini</b>',array('class' => 'detail'))
    );
    }
    $data['table'] = $this->table->generate();
    }
    else {
        $data['message'] = 'Tidak ditemukan data...';
    }
    $this->load-
>view('member/pemesanan/riwayat_pesanan_popup', $data);
    }
    else {
        redirect('c_login_member');
    }
    }

    function detail_riwayat_pesanan_popup($id_pemesanan,
    $id_member_tujuan) {

```

```

if (($this->session->userdata('login') == TRUE) AND
    ($this->session->userdata('hak_akses') == 'member')) {

    $pemesanan = $this->m_pemesanan->detail_pemesanan
    ($id_pemesanan)->result();
    $num_rows = $this->m_pemesanan-
>count_all_detail($id_pemesanan);

    $config['base_url'] =
    site_url('c_pemesanan/detail_pemesanan');

    $tmpl = array( 'table_open' => '<table border="1"
style="border-collapse:collapse"
    cellpadding="0" cellspacing="0">', 'row_alt_start' =>
'<tr class="zebra">',
    'row_alt_end' => '</tr>');

    $this->table->set_template($tmpl);
    $this->table->set_empty(" ");
    $this->table->set_heading('No', 'ID Pemesanan', 'Kode
Barang',
    'Nama Barang', 'Jumlah Pesan', 'Harga Barang',
'Subtotal');

    $i = 1;
    foreach ($pemesanan as $pemesanan) {
    $this->table->add_row(
    $i++,
    $pemesanan->id_pemesanan,
    $pemesanan->kode_barang,
    $pemesanan->nama_barang,
    $pemesanan->jumlah_barang,
    $pemesanan->harga_barang,
    $pemesanan->subtotal
    );
    }
    $data['table'] = $this->table->generate();

    $pemesan = $this->m_pemesanan-
>detail_pemesan($id_member_tujuan)->row();

    $data['nama_apotek'] = $pemesan->nama_apotek;
    $data['email'] = $pemesan->email;
    $data['alamat'] = $pemesan->alamat;
    $data['kota'] = $pemesan->kota;
    $data['provinsi'] = $pemesan->provinsi;
    $data['kode_pos'] = $pemesan->kode_pos;
    $data['telepon'] = $pemesan->telepon;
    $data['nama_bank'] = $pemesan->nama_bank;
    $data['no_rekening'] = $pemesan->no_rekening;

    $rincian = $this->m_pemesanan-
>rincian_biaya($id_pemesanan)->row();

    $data['id_pemesanan'] = $rincian->id_pemesanan;

```



```

$data['total_biaya'] = $rincian->total_biaya;
$data['total_harga'] = $rincian->total_harga;
$data['biaya_pengiriman'] = $rincian->biaya_pengiriman;

$this->load-
>view('member/pemesanan/detail_riwayat_pesanan_popup', $data);
}
else {
    redirect('c_login_member');
}
}

function terima_pesanan() {
    if (($this->session->userdata('login') == TRUE) AND
        ($this->session->userdata('hak_akses') == 'member')) {

        $data['h2_title'] = 'Terima Pesanan';
        $data['main_view'] = 'member/pemesanan/terima_pesanan';
        $data['form_action'] =
site_url('c_pembayaran/terima_pesanan');
        $uri_segment = 3;
        $offset = $this->uri->segment($uri_segment);

        $pemesanan = $this->m_pemesanan->riwayat_pesanan
($this->limit, $offset)->result();
        $num_rows_riwayat = $this->m_pemesanan-
>count_all_num_rows_riwayat();

        if ($num_rows_riwayat > 0 ) {
            $config['base_url'] =
site_url('c_pemesanan/terima_pesanan');
            $config['total_rows'] = $num_rows_riwayat;
            $config['per_page'] = $this->limit;
            $config['uri_segment'] = $uri_segment;
            $config['next_link'] = '<b>&rsaquo;</b>';
            $config['prev_link'] = '<b>&lsaquo;</b>';
            $config['last_link'] = '<b>&raquo;</b>';
            $config['first_link'] = '<b>&laquo;</b>';
            $this->pagination->initialize($config);
            $data['pagination'] =
$this->pagination->create_links();

            $tmpl = array( 'table_open' => '<table border="1"
style="border-collapse:collapse"
cellpadding="0" cellspacing="0">', 'row_alt_start' =>
'<tr class="zebra">',
            'row_alt_end' => '</tr>');

            $this->table->set_template($tmpl);
            $this->table->set_empty("&nbsp;");
            $this->table->set_heading('No', 'ID Pesanan', 'Tgl.
Pesan', 'Detail');

            $i = 1 + $offset;
            foreach ($pemesanan as $pemesanan) {

```

```

        $this->table->add_row(
            $i++,
            $pemesanan->id_pemesanan,
            $pemesanan->tgl_pemesanan,

            anchor('c_pemesanan/ubah_terima_pesanan/'.$pemesanan-
>id_pemesanan,
                'Terima Pesanan',array('class' => 'update'))
            );
        }
        $data['table'] = $this->table->generate();
    }
    else {
        $data['message'] = 'Tidak ditemukan data...';
    }

    $this->load->view('member', $data);

    }
    else {
        redirect('c_login_member');
    }
}

function ubah_terima_pesanan($id_pemesanan) {
    if (($this->session->userdata('login') == TRUE) AND
        ($this->session->userdata('hak_akses') == 'member')) {

        $data['h2_title'] = 'Ubah Status Terima Pesanan';
        $data['main_view'] =
'member/pemesanan/ubah_terima_pesanan';
        $data['form_action'] =
site_url('c_pemesanan/proses_terima_pesanan');

        $pemesanan = $this->m_pemesanan-
>proses_pemesanan($id_pemesanan)->row();

        $data['default']['id_pemesanan'] = $pemesanan-
>id_pemesanan;
        $data['default']['status_kirim'] = $pemesanan-
>status_kirim;

        $status = $this->m_pemesanan->get_status_terima()-
>result();
        foreach($status as $status) {
            $data['opsi_status'][$status->id_status] = $status-
>status;
        }

        $this->load->view('member', $data);
    }
    else {
        redirect('c_login_member');
    }
}

```

```

function proses_terima_pesanan() {
    if (($this->session->userdata('login') == TRUE) AND
        ($this->session->userdata('hak_akses') == 'member')) {

        $data['h2_title'] = 'Ubah Status Terima Pesanan';
        $data['main_view'] =
'member/pemesanan/ubah_terima_pesanan';
        $data['form_action'] =
site_url('c_pemesanan/proses_terima_pesanan');

        if ($this->input->post('submit')) {

            $id_status = $this->input->post('id_status');
            $id_pemesanan = $this->input->post('id_pemesanan');

            $proses = array('id_status' => $id_status);
            $this->m_pemesanan->ubah_status_pemesanan($id_pemesanan,
$proses);

            redirect('c_pemesanan/terima_pesanan');
        }
        $this->load->view('member', $data);

    }
    else {
        redirect('c_login_member');
    }
}
}
?>

```

### **File *c\_pembayaran.php***

```
<?php
```

```

class C_Pembayaran extends CI_Controller {

    function __construct() {
        parent::__construct();
        $this->load->model('m_pembayaran', '', TRUE);
        $this->load->model('m_pemesanan', '', TRUE);
    }

    var $limit = 10;

    function index() {
        if (($this->session->userdata('login') == TRUE) AND
            ($this->session->userdata('hak_akses') == 'member')) {

            $this->bayar_pesanan();
        }
        else {
            redirect('c_login_member');
        }
    }
}

```

```

    }

    function bayar_pesanan() {
        if (($this->session->userdata('login') == TRUE) AND
            ($this->session->userdata('hak_akses') == 'member')) {

            $data['h2_title'] = 'Bayar Pesanan';
            $data['main_view'] = 'member/pembayaran/bayar_pesanan';
            $data['form_action'] =
site_url('c_pembayaran/bayar_pesanan');

            $this->form_validation->set_rules('id_pemesanan',
                'ID Pemesanan', 'callback_valid_pembayaran');
            $this->form_validation->set_rules('nilai_pembayaran',
                'Nilai Pembayaran', 'required|min_length[1]|numeric');

            $id_pemesanan = $this->m_pemesanan-
>get_id_pemesanan_bayar()->result();
            foreach($id_pemesanan as $id_pemesanan) {
                $data['opsi_status'][$id_pemesanan->id_pemesanan] =
$id_pemesanan->id_pemesanan;
            }

            if ($this->form_validation->run() == TRUE) {
                $pembayaran = array(
                    'id_pemesanan' => $this->input->post('id_pemesanan'),
                    'nilai_pembayaran' => $this->input-
>post('nilai_pembayaran'),
                    'id_status' => 1
                );

                $this->m_pembayaran-
>add_data_pembayaran($pembayaran);

                $this->session->set_flashdata('message',
                    'Pembayaran pesanan telah berhasil dilakukan');
                redirect('c_pembayaran');
            }
            else {
                $this->load->view('member', $data);
            }
        }
        else {
            redirect('c_login_member');
        }
    }

    function manajemen_pembayaran() {
        if (($this->session->userdata('login') == TRUE) AND
            ($this->session->userdata('hak_akses') == 'member')) {

            $data['h2_title'] = 'Manajemen Pembayaran';
            $data['main_view'] =
'member/pembayaran/manajemen_pembayaran';
            $uri_segment = 3;

```

```

$offset = $this->uri->segment($uri_segment);

$pembayaran = $this->m_pembayaran->manajemen_pembayaran
($this->limit, $offset)->result();
$num_rows = $this->m_pembayaran->count_all_num_rows();

if ($num_rows > 0 ) {
    $config['base_url'] =
    site_url('c_pembayaran/manajemen_pembayaran');
    $config['total_rows'] = $num_rows;
    $config['per_page'] = $this->limit;
    $config['uri_segment'] = $uri_segment;
    $config['next_link'] = '<b>&rsaquo;</b>';
    $config['prev_link'] = '<b>&lsaquo;</b>';
    $config['last_link'] = '<b>&raquo;</b>';
    $config['first_link'] = '<b>&laquo;</b>';
    $this->pagination->initialize($config);
    $data['pagination'] =
    $this->pagination->create_links();

    $tmpl = array( 'table_open' => '<table border="1"
style="border-collapse:collapse"
    cellpadding="0" cellspacing="0">', 'row_alt_start' =>
'<tr class="zebra">',
    'row_alt_end' => '</tr>');

    $this->table->set_template($tmpl);
    $this->table->set_empty("&nbsp;");
    $this->table->set_heading('No', 'ID Pembayaran', 'ID
Pemesanan', 'Tgl. Pembayaran',
    'Total Biaya', 'Nilai Pembayaran', 'Status', 'Aksi');

    $i = 1 + $offset;
    foreach ($pembayaran as $pembayaran) {
    $this->table->add_row(
    $i++,
    $pembayaran->id_pembayaran,
    $pembayaran->id_pemesanan,
    $pembayaran->tgl_pembayaran,
    $pembayaran->total_biaya,
    $pembayaran->nilai_pembayaran,
    $pembayaran->status,

    anchor('c_pembayaran/ubah_status/'. $pembayaran-
>id_pembayaran.'/'. $pembayaran->id_pemesanan,
    'Proses', array('class' => 'update')).' '.

    anchor('c_pembayaran/hapus_pembayaran/'. $pembayaran-
>id_pembayaran,
    'Hapus', array('class' => 'delete',
    'onclick'=>"return confirm('Apakah anda ingin
menghapus data ini?')
    ")
    );
    }
    $data['table'] = $this->table->generate();

```

```

    }
    else {
        $data['message'] = 'Tidak ditemukan data...';
    }

    $data['pencarian'] = array('link_cari' =>
    anchor('c_pembayaran/cari_pembayaran/', 'Pencarian',
    array('class' => 'detail')));

    $this->load->view('member', $data);

    }
    else {
        redirect('c_login_member');
    }
}

function ubah_status($id_pembayaran, $id_pemesanan) {
    if (($this->session->userdata('login') == TRUE) AND
    ($this->session->userdata('hak_akses') == 'member')) {

        $data['h2_title'] = 'Konfirmasi Pembayaran';
        $data['main_view'] =
'member/pembayaran/ubah_status_pembayaran';
        $data['form_action'] =
site_url('c_pembayaran/proses_ubah_status');

        $status = $this->m_pembayaran->get_status()->result();
        foreach($status as $status) {
            $data['opsi_status'][$status->id_status] = $status-
>status;
        }

        $pembayaran = $this->m_pembayaran-
>proses_pembayaran($id_pembayaran, $id_pemesanan)->row();

        $data['default']['id_pembayaran'] = $pembayaran-
>id_pembayaran;
        $data['default']['id_pemesanan'] = $pembayaran-
>id_pemesanan;
        $data['default']['id_status'] = $pembayaran->id_status;
        $data['total_biaya'] = $pembayaran->total_biaya;
        $data['nilai_pembayaran'] = $pembayaran-
>nilai_pembayaran;

        $this->load->view('member', $data);

    }
    else {
        redirect('c_login_member');
    }
}

function proses_ubah_status() {
    if (($this->session->userdata('login') == TRUE) AND
    ($this->session->userdata('hak_akses') == 'member')) {

```

```

        $data['h2_title'] = 'Ubah Status Pembayaran';
        $data['main_view'] =
'member/pembayaran/ubah_status_pembayaran';
        $data['form_action'] =
site_url('c_pembayaran/proses_ubah_status');

        if ($this->input->post('submit')) {

            $id_pembayaran = $this->input->post('id_pembayaran');
            $id_status = $this->input->post('id_status');

            $proses = array('id_status' => $id_status);
            $this->m_pembayaran->
>update_proses_pembayaran($id_pembayaran, $proses);

            redirect('c_pembayaran/manajemen_pembayaran');
        }
        $this->load->view('member', $data);
    }
    else {
        redirect('c_login_member');
    }
}

function valid_pembayaran() {
    $id_pemesanan = $this->input->post('id_pemesanan');
    if($this->m_pembayaran->valid_pembayaran($id_pemesanan)
== FALSE) {
        $this->form_validation->
>set_message('valid_pembayaran',
            'Perhatian, ID Pemesanan '.$id_pemesanan.' sudah
dilakukan pembayaran.');
```

### **File *c\_pengiriman.php***

```

<?php

class C_Pengiriman extends CI_Controller {

    function __construct() {
        parent::__construct();
        $this->load->model('m_pengiriman', '', TRUE);
    }

    var $limit = 10;
```

```

function index() {
    if (($this->session->userdata('login') == TRUE) AND
        ($this->session->userdata('hak_akses') == 'member')) {

        $this->lacak_pesanan();
    }
    else {
        redirect('c_login_member');
    }
}

function lacak_pesanan() {
    if (($this->session->userdata('login') == TRUE) AND
        ($this->session->userdata('hak_akses') == 'member')) {

        $data['h2_title'] = 'Lacak Pesanan (Tracking Order)';
        $data['main_view'] = 'member/pengiriman/lacak_pesanan';
        $data['form_action'] =
site_url('c_pengiriman/lacak_pesanan');

        $this->form_validation->set_rules('id_pemesanan',
            'ID Pemesanan', 'required|integer|min_length[4]');

        if ($this->input->post('submit') && ($this->
>form_validation->run() == TRUE)) {
            $hasil = $this->m_pengiriman->lacak_pesanan()-
>result();

            $id_pemesanan = $this->input->post('id_pemesanan');

            if (empty($hasil)) {
                $data['message'] = 'Tidak ditemukan data pesanan
dengan id pemesanan '.$id_pemesanan.'';
            } else {
                $tmpl = array( 'table_open' => '<table border="1"
style="border-collapse:collapse"
cellpadding="0" cellspacing="0">', 'row_alt_start' =>
'<tr class="zebra">',
                'row_alt_end' => '</tr>');

                $this->table->set_template($tmpl);
                $this->table->set_empty("&nbsp;");
                $this->table->set_heading('ID Pemesanan', 'Status
Pemesanan');

                foreach ($hasil as $hasil) {
                    $this->table->add_row(
                    $hasil->id_pemesanan,
                    $hasil->status);
                }
                $data['table'] = $this->table->generate();
            }
        }

    }
    $this->load->view('member', $data);
}

```



```

    }
    else {
        redirect('c_login_member');
    }
}
?>

```

### **File *c\_laporan.php***

```
<?php
```

```

class C_Laporan extends CI_Controller {

    function __construct() {
        parent::__construct();
        $this->load->model('m_laporan', '', TRUE);
        $this->load->library('highcharts');
    }

    function index() {
        if (($this->session->userdata('login') == TRUE) AND
            ($this->session->userdata('hak_akses') == 'member')) {

            $this->laporan_penjualan();
        }
        else {
            redirect('c_login_member');
        }
    }

    function laporan_penjualan() {
        if (($this->session->userdata('login') == TRUE) AND
            ($this->session->userdata('hak_akses') == 'member')) {

            $data['h2_title'] = 'Laporan Penjualan';
            $data['main_view'] = 'member/laporan/laporan_penjualan';
            $data['form_action'] =
site_url('c_laporan/laporan_penjualan');

            $this->form_validation->set_rules('tahun',
                'Tahun', 'required|min_length[4]|integer');

            if ($this->input->post('submit') && ($this->
form_validation->run() == TRUE)) {
                $penjualan = $this->m_laporan->lap_jual_member();

                $tahun = $this->input->post('tahun');
                if (empty($penjualan)) {
                    $data['message'] = 'Tidak ditemukan data laporan
penjualan pada tahun '.$tahun.'';
                } else {

                    foreach($penjualan as $penjualan) {
                        $value[] = intval($penjualan->total_jual);
                        $bulan[] = $penjualan->bulan;
                    }
                }
            }
        }
    }
}

```

```

    }

    $data['users']['data'] = $value;
    $data['users']['name'] = 'Jumlah (Item)
Terjual';

    $data['axis']['categories'] = $bulan;

    $this->highcharts->set_type('column');
    $this->highcharts->set_title('Laporan
Penjualan', 'Penjualan pada tahun '.$tahun.'');
    $this->highcharts->set_axis_titles('',
'Jumlah Penjualan');

    $this->highcharts->set_xAxis($data['axis']);
    $this->highcharts->set_serie($data['users']);

    $credits->href = '';
    $credits->text = "Suka Farma";
    $this->highcharts->set_credits($credits);

    $this->highcharts->render_to('laporan');

    $data['charts'] = $this->highcharts-
>render();
    }
    }
    $this->load->view('member', $data);
    }
    else {
        redirect('c_login_member');
    }
}

function laporan_pembelian() {
    if (($this->session->userdata('login') == TRUE) AND
        ($this->session->userdata('hak_akses') == 'member')) {

        $data['h2_title'] = 'Laporan Pembelian';
        $data['main_view'] = 'member/laporan/laporan_pembelian';
        $data['form_action'] =
site_url('c_laporan/laporan_pembelian');

        $this->form_validation->set_rules('tahun',
            'Tahun', 'required|min_length[4]|integer');

        if ($this->input->post('submit') && ($this-
>form_validation->run() == TRUE)) {
            $pembelian = $this->m_laporan->lap_beli_member();

            $tahun = $this->input->post('tahun');
            if (empty($pembelian)) {
                $data['message'] = 'Tidak ditemukan data laporan
pembelian pada tahun '.$tahun.'';
            } else {

```

```

        foreach($pembelian as $pembelian) {
            $value[] = intval($pembelian->total_beli);
            $bulan[] = $pembelian->bulan;
        }

        $data['users']['data'] = $value;
        $data['users']['name'] = 'Jumlah (Item
Pembelian';

        $data['axis']['categories'] = $bulan;

        $this->highcharts->set_type('column');
        $this->highcharts->set_title('Laporan
Pembelian', 'Pembelian pada tahun '.$tahun.'');
        $this->highcharts->set_axis_titles('',
'Jumlah Pembelian');

        $this->highcharts->set_xAxis($data['axis']);
        $this->highcharts->set_serie($data['users']);

        $credits->href = '';
        $credits->text = "Suka Farma";
        $this->highcharts->set_credits($credits);

        $this->highcharts->render_to('laporan');

        $data['charts'] = $this->highcharts-
>render();
    }
}
$this->load->view('member', $data);
}
else {
    redirect('c_login_member');
}
}

function laporan_penjualan_total() {
    if (($this->session->userdata('login') == TRUE) AND
        ($this->session->userdata('hak_akses') == 'admin')) {

        $data['h2_title'] = 'Laporan Penjualan';
        $data['main_view'] = 'admin/laporan/laporan_penjualan';
        $data['form_action'] =
site_url('c_laporan/laporan_penjualan_total');

        $this->form_validation->set_rules('tahun',
'Tahun', 'required|min_length[4]|integer');

        if ($this->input->post('submit') && ($this-
>form_validation->run() == TRUE)) {
            $penjualan = $this->m_laporan->lap_jual_total();

            $tahun = $this->input->post('tahun');
            if (empty($penjualan)) {

```

```

        $data['message'] = 'Tidak ditemukan data laporan
penjualan pada tahun '.$tahun.'';
    } else {

        foreach($penjualan as $penjualan) {
            $value[] = intval($penjualan->total_jual);
            $bulan[] = $penjualan->bulan;
        }

        $data['users']['data'] = $value;
        $data['users']['name'] = 'Jumlah (Item)
Terjual';

        $data['axis']['categories'] = $bulan;

        $this->highcharts->set_type('column');
        $this->highcharts->set_title('Laporan
Penjualan', 'Penjualan pada tahun '.$tahun.'');
        $this->highcharts->set_axis_titles('',
'Jumlah Penjualan');

        $this->highcharts->set_xAxis($data['axis']);
        $this->highcharts->set_serie($data['users']);

        $credits->href = '';
        $credits->text = "Suka Farma";
        $this->highcharts->set_credits($credits);

        $this->highcharts->render_to('laporan');

        $data['charts'] = $this->highcharts-
>render();
    }
}
$this->load->view('admin', $data);
}
else {
    redirect('c_login_admin');
}
}
}
?>

```

### **File *c\_member.php***

```
<?php
```

```

class C_Member extends CI_Controller {

    function __construct() {
        parent::__construct();
        $this->load->model('m_member', '', TRUE);
        $this->load->library('googlemaps/googlemaps');
    }
}

```

```

var $limit = 10;

function index() {

    if (($this->session->userdata('login') == TRUE) AND
        ($this->session->userdata('hak_akses') == 'admin')) {

        $this->manajemen_member();
    }
    else {
        redirect('c_login_admin');
    }
}

function manajemen_member($offset =0) {
    if (($this->session->userdata('login') == TRUE) AND
        ($this->session->userdata('hak_akses') == 'admin')) {

        $data['h2_title'] = 'Manajemen Member';
        $data['main_view'] = 'admin/member/manajemen_member';
        $uri_segment = 3;
        $offset = $this->uri->segment($uri_segment);

        $member = $this->m_member->manajemen_member
            ($this->limit, $offset)->result();
        $num_rows = $this->m_member->count_all_num_rows();

        if ($num_rows > 0) {
            $config['base_url'] =
                site_url('c_member/manajemen_member');
            $config['total_rows'] = $num_rows;
            $config['per_page'] = $this->limit;
            $config['uri_segment'] = $uri_segment;
            $config['next_link'] = '<b>&rsaquo;</b>';
            $config['prev_link'] = '<b>&lsaquo;</b>';
            $config['last_link'] = '<b>&raquo;</b>';
            $config['first_link'] = '<b>&laquo;</b>';
            $this->pagination->initialize($config);
            $data['pagination'] =
                $this->pagination->create_links();

            $tmpl = array( 'table_open' => '<table border="1"
style="border-collapse:collapse"
                cellpadding="0" cellspacing="0">', 'row_alt_start' =>
'<tr class="zebra">',
                'row_alt_end' => '</tr>');

            $this->table->set_template($tmpl);
            $this->table->set_empty("&nbsp;");
            $this->table->set_heading('No', 'Nama Apotek',
'Alamat', 'Email',
                'No. Telepon', 'Status', 'Aksi');
            $i = 1 + $offset;
            foreach ($member as $member) {
                $this->table->add_row(
                    $i++,

```

```

$member->nama_apotek,
$member->alamat,
$member->email,
$member->telepon,
$member->status,

anchor('c_member/detail/.'.$member->id_member,
'Detail',array('class' => 'detail')).' '.

anchor('c_member/edit/.'.$member->id_member,
'Ubah',array('class' => 'update')).' '.

anchor('c_member/delete/.'.$member->id_member,
'Hapus',array('class' => 'delete',
'onclick'=>"return confirm('Apakah anda ingin
menghapus data ini?')
"))
);
}

$data['table'] = $this->table->generate();
}
else {
$data['message'] = 'Tidak ditemukan data...';
}
$data['link'] = array('link_add' =>
anchor('c_member/add/', 'Tambah',
array('class' => 'add')));

$data['pencarian'] = array('link_cari' =>
anchor('c_member/cari_member/', 'Pencarian',
array('class' => 'detail')));

$this->load->view('admin', $data);

}
else {
redirect('c_login_admin');
}
}

function detail($id_member) {
if (($this->session->userdata('login') == TRUE) AND
($this->session->userdata('hak_akses') == 'admin')) {

$data['main_view'] = 'admin/member/detail_member';
$data['link'] = array('link_back' =>
anchor('c_member/', '<input type="submit" id="submit"
value="Kembali">',
array('class' => 'back')));

$member = $this->m_member->detail_member($id_member)-
>row();

$data['h2_title'] = 'Detail Member : '.$member-
>nama_apotek.' ('.$member->id_member.')';

```

```

$data['default']['id_member'] = $member->id_member;

$data['default']['nama_apotek'] = $member->nama_apotek;
$data['default']['email'] = $member->email;
$data['default']['nama_bank'] = $member->nama_bank;
$data['default']['no_rekening'] = $member->no_rekening;
$data['default']['telepon'] = $member->telepon;
$data['default']['alamat'] = $member->alamat;
$data['default']['kota'] = $member->kota;
$data['default']['provinsi'] = $member->provinsi;
$data['default']['kode_pos'] = $member->kode_pos;
$data['default']['tgl_daftar'] = $member->tgl_daftar;
$data['default']['status'] = $member->status;

$latitude = $member->posisi_latitude;
$longitude = $member->posisi_longitude;
$nama_apotek = $member->nama_apotek;

$config['center'] = "$latitude,$longitude";
$config['zoom'] = '16';
$config['map_height'] = '200px';
$this->googlemaps->initialize($config);

$a = base_url();
$marker = array();
$marker['icon'] = "$a/css/images/store.png";
$marker['animation'] = 'BOUNCE';
$marker['title'] = "$nama_apotek";
$marker['position'] = "$latitude,$longitude";
$this->googlemaps->add_marker($marker);

$data['map'] = $this->googlemaps->create_map();

$this->load->view('admin', $data);

}
else {
    redirect('c_login_admin');
}
}

function add() {
    if (($this->session->userdata('login') == TRUE) AND
        ($this->session->userdata('hak_akses') == 'admin')) {

        $data['h2_title'] = 'Tambah Member';
        $data['main_view'] = 'admin/member/form_member';
        $data['form_action'] = site_url('c_member/add_member');

        $this->load->view('admin', $data);

    }
    else {
        redirect('c_login_admin');
    }
}
}

```

```

function add_member() {
    if (($this->session->userdata('login') == TRUE) AND
        ($this->session->userdata('hak_akses') == 'admin')) {

        $data['h2_title'] = 'Tambah Member';
        $data['main_view'] = 'admin/member/form_member';
        $data['form_action'] = site_url('c_member/add_member');

        $this->form_validation->set_rules('nama_apotek',
            'Nama Apotek', 'required|max_length[20]');
        $this->form_validation->set_rules('username',
            'Username', 'required|callback_valid_username');
        $this->form_validation->set_rules('password',
            'Password', 'required');
        $this->form_validation->set_rules('email',
            'Email', 'required|valid_email');
        $this->form_validation->set_rules('nama_bank',
            'Nama Bank', 'required');
        $this->form_validation->set_rules('no_rekening',
            'No. Rekening', 'required');
        $this->form_validation->set_rules('telepon',
            'Telepon', 'required');
        $this->form_validation->set_rules('alamat',
            'Alamat', 'required');
        $this->form_validation->set_rules('kota',
            'Kota', 'required');
        $this->form_validation->set_rules('provinsi',
            'Provinsi', 'required');
        $this->form_validation->set_rules('posisi_latitude',
            'Posisi Latitude', 'required');
        $this->form_validation->set_rules('posisi_longitude',
            'Posisi Longitude', 'required');
        $this->form_validation->set_rules('kode_pos',
            'Kode Pos', 'required|integer');
        $this->form_validation->set_rules('tgl_daftar',
            'Tgl. Daftar', 'required');
        $this->form_validation->set_rules('status',
            'Status', 'required');

        if ($this->form_validation->run() == TRUE) {
            $member = array(
                'nama_apotek' => $this->input->post('nama_apotek'),
                'username' => $this->input->post('username'),
                'password' => $this->input->post('password'),
                'email' => $this->input->post('email'),
                'nama_bank' => $this->input->post('nama_bank'),
                'no_rekening' => $this->input->post('no_rekening'),
                'telepon' => $this->input->post('telepon'),
                'alamat' => $this->input->post('alamat'),
                'kota' => $this->input->post('kota'),
                'provinsi' => $this->input->post('provinsi'),
                'kode_pos' => $this->input->post('kode_pos'),
                'tgl_daftar' => $this->input->post('tgl_daftar'),
                'status' => $this->input->post('status')
            );
        }
    }
}

```



```

        $this->m_member->add_data_member($member);

        $this->session->set_flashdata('message',
            'Penambahan data berhasil dilakukan');
        redirect('c_member');
    }
    else {
        $this->load->view('admin', $data);
    }

    }
    else {
        redirect('c_login_admin');
    }
}

function delete_member($id_member) {
    if (($this->session->userdata('login') == TRUE) AND
        ($this->session->userdata('hak_akses') == 'admin')) {

        $this->m_member->delete_member($id_member);
        $this->session->set_flashdata('message',
            'Penghapusan data berhasil dilakukan');
        redirect('c_member');

    }
    else {
        redirect('c_login_admin');
    }
}

function edit($id_member) {
    if (($this->session->userdata('login') == TRUE) AND
        ($this->session->userdata('hak_akses') == 'admin')) {

        $data['main_view'] = 'admin/member/form_member';
        $data['form_action'] = site_url('c_member/edit_member');

        $member = $this->m_member->get_member_edit($id_member)-
>row();

        $data['h2_title'] = 'Ubah Member : '.$member-
>nama_apotek.' ('.$member->id_member.')';

        $this->session->set_userdata('id_member', $member-
>id_member);

        $data['default']['nama_apotek'] = $member->nama_apotek;
        $data['default']['username'] = $member->username;
        $data['default']['password'] = $member->password;
        $data['default']['email'] = $member->email;
        $data['default']['nama_bank'] = $member->nama_bank;
        $data['default']['no_rekening'] = $member->no_rekening;
        $data['default']['telepon'] = $member->telepon;
        $data['default']['alamat'] = $member->alamat;
    }
}

```

```

        $data['default']['kota'] = $member->kota;
        $data['default']['provinsi'] = $member->provinsi;
        $data['default']['posisi_latitude'] = $member-
>posisi_latitude;
        $data['default']['posisi_longitude'] = $member-
>posisi_longitude;
        $data['default']['kode_pos'] = $member->kode_pos;
        $data['default']['tgl_daftar'] = $member->tgl_daftar;
        $data['default']['status'] = $member->status;

        $this->load->view('admin', $data);

    }
    else {
        redirect('c_login_admin');
    }
}

function edit_member() {
    if (($this->session->userdata('login') == TRUE) AND
        ($this->session->userdata('hak_akses') == 'admin')) {

        $data['h2_title'] = 'Ubah Member';
        $data['main_view'] = 'admin/member/form_member';
        $data['form_action'] = site_url('c_member/edit_member');

        $this->form_validation->set_rules('nama_apotek',
            'Nama Apotek', 'required|max_length[20]');
        $this->form_validation->set_rules('username',
            'Username', 'required');
        $this->form_validation->set_rules('password',
            'Password', 'required');
        $this->form_validation->set_rules('email',
            'Email', 'required|valid_email');
        $this->form_validation->set_rules('nama_bank',
            'Nama Bank', 'required');
        $this->form_validation->set_rules('no_rekening',
            'No. Rekening', 'required');
        $this->form_validation->set_rules('telepon',
            'Telepon', 'required');
        $this->form_validation->set_rules('alamat',
            'Alamat', 'required');
        $this->form_validation->set_rules('kota',
            'Kota', 'required');
        $this->form_validation->set_rules('provinsi',
            'Provinsi', 'required');
        $this->form_validation->set_rules('posisi_latitude',
            'Posisi Latitude', 'required');
        $this->form_validation->set_rules('posisi_longitude',
            'Posisi Longitude', 'required');
        $this->form_validation->set_rules('kode_pos',
            'Kode Pos', 'required|integer');
        $this->form_validation->set_rules('tgl_daftar',
            'Tgl. Daftar', 'required');
        $this->form_validation->set_rules('status',
            'Status', 'required');
    }
}

```

```

if ($this->form_validation->run() == TRUE) {
    $member = array(
        'nama_apotek' => $this->input->post('nama_apotek'),
        'username' => $this->input->post('username'),
        'password' => $this->input->post('password'),
        'email' => $this->input->post('email'),
        'nama_bank' => $this->input->post('nama_bank'),
        'no_rekening' => $this->input->post('no_rekening'),
        'telepon' => $this->input->post('telepon'),
        'alamat' => $this->input->post('alamat'),
        'kota' => $this->input->post('kota'),
        'provinsi' => $this->input->post('provinsi'),
        'posisi_latitude' => $this->input-
>post('posisi_latitude'),
        'posisi_longitude' => $this->input-
>post('posisi_longitude'),
        'kode_pos' => $this->input->post('kode_pos'),
        'tgl_daftar' => $this->input->post('tgl_daftar'),
        'status' => $this->input->post('status')
    );

    $this->m_member->edit_member($this->session-
>userdata('id_member')
    , $member);

    $this->session->set_flashdata('message',
    'Perubahan data berhasil dilakukan');
    redirect('c_member');
}
else {
    $this->load->view('admin', $data);
}
}
else {
    redirect('c_login_admin');
}
}

function cari_member() {
    if (($this->session->userdata('login') == TRUE) AND
        ($this->session->userdata('hak_akses') == 'admin')) {

        $data['h2_title'] = 'Pencarian Member';
        $data['main_view'] = 'admin/member/form_pencarian';
        $data['form_action'] = site_url('c_member/cari_member');

        $this->form_validation->set_rules('kunci',
        'Kata Kunci', 'required|min_length[2]');

        $kunci_pencarian = $this->input->post('kunci');

        if ($this->input->post('submit') && ($this-
>form_validation->run() == TRUE)) {
            $member = $this->m_member->cari_member();

```

```

        if (empty($member)) {
            $data['peringatan'] = 'Tidak ditemukan data dengan
kata kunci pencarian <b>'.$kunci_pencarian.'</b>';
        } else {
            $tmpl = array( 'table_open' => '<table border="1"
style="border-collapse:collapse"
cellpadding="0" cellspacing="0">', 'row_alt_start' =>
'<tr class="zebra">',
            'row_alt_end' => '</tr>');

            $total = count($member);
            $data['message'] = 'Ditemukan <b>'.$total.'</b> data
dengan kata kunci pencarian <b>'.$kunci_pencarian.'</b>';

            $this->table->set_template($tmpl);
            $this->table->set_empty(" ");
            $this->table->set_heading('No', 'Nama Apotek',
'Alamat', 'Email',
            'No. Telepon', 'Status', 'Aksi');
            $i = 1;
            foreach ($member as $member) {
                $this->table->add_row(
                    $i++,
                    $member->nama_apotek,
                    $member->alamat,
                    $member->email,
                    $member->telepon,
                    $member->status,

                    anchor('c_member/detail/'.$member->id_member,
'Detail',array('class' => 'detail')).' '.

                    anchor('c_member/edit/'.$member->id_member,
'Ubah',array('class' => 'update')).' '.

                    anchor('c_member/delete/'.$member->id_member,
'Hapus',array('class' => 'delete',
'onclick'=>"return confirm('Apakah anda ingin
menghapus data ini?')
                ")))
                );
            }

            $data['table'] = $this->table->generate();
        }
    }
    $this->load->view('admin', $data);

}
else {
    redirect('c_login_admin');
}
}

function valid_username() {
    $username = $this->input->post('username');

```

```

        if($this->m_member->valid_username($username) == FALSE) {
            $this->form_validation->set_message('valid_username',
                'Maaf, username '.$username.' sudah digunakan member
lain.');
```

```

            return FALSE;
        }
        else {
            return TRUE;
        }
    }
}
?>
```

### **File *c\_public.php***

```
<?php
```

```

class C_Public extends CI_Controller {

    function __construct() {
        parent::__construct();
        $this->load->model('m_member', '', TRUE);
        $this->load->model('m_pemesanan', '', TRUE);
        $this->load->library('googlemaps/googlemaps');
    }

    function index() {
        $this-> Pencarian();
    }

    function Pencarian() {
        $data['main_view'] = 'pengunjung/apotek_info';
        $data['form_action1'] = site_url('c_public/hasil');
        $data['form_action2'] =
site_url('c_public/cari_lokasi_apotek');

        $member = $this->m_member->get_member()->result();
        $data['list_member']['Semua Apotek'] = 'Semua Apotek';
        foreach($member as $row) {
            $data['list_member'][$row->nama_apotek] = $row-
>nama_apotek;
        }

        foreach($member as $row) {
            $data['list_apotek'][$row->nama_apotek] = $row-
>nama_apotek;
        }

        $data['pemesanan_terbanyak'] = $this->m_pemesanan-
>pesanan_terbanyak()->result();
        $data['pemesanan_terbaru'] = $this->m_pemesanan-
>pesanan_terbaru()->result();

        /* controller untuk menu lokasi apotek */
        $lokasi = $this->m_member->tampil_lokasi()->result();
    }
}

```

```

foreach ($lokasi as $lokasi) {
    $latitude = $lokasi->posisi_latitude;
    $longitude = $lokasi->posisi_longitude;
    $nama_apotek = $lokasi->nama_apotek;
    $kota = $lokasi->kota;
    $telepon = $lokasi->telepon;
}

$config['center'] = "$latitude,$longitude";
$config['zoom'] = '16';
$config['map_height'] = '140px';
$this->googlemaps->initialize($config);

$a = base_url();
$marker = array();
$marker['icon'] = "$a/css/images/store.png";
$marker['animation'] = 'DROP';
$marker['title'] = "$nama_apotek, $kota, Phone:
$telepon";
$marker['position'] = "$latitude,$longitude";
$this->googlemaps->add_marker($marker);

$data['map'] = $this->googlemaps->create_map();
$data['nama_apotek'] = $lokasi->nama_apotek;

$this->load->view('publik', $data);
}

function cari_lokasi_apotek() {
    $data['main_view'] = 'pengunjung/apotek_info';
    $data['form_action1'] = site_url('c_publik/hasil');
    $data['form_action2'] =
site_url('c_publik/cari_lokasi_apotek');

    $this->form_validation->set_rules('nama_apotek',
        'Nama Apotek', 'required');

    if ($this->input->post('submit') && ($this-
>form_validation->run() == TRUE)) {
        $lokasi = $this->m_member->cari_lokasi_apotek()-
>row();

        $member = $this->m_member->get_member()->result();
        $data['list_member']['Semua Apotek'] = 'Semua Apotek';
        foreach($member as $row) {
            $data['list_member'][$row->nama_apotek] = $row-
>nama_apotek;
        }

        foreach($member as $row) {
            $data['list_apotek'][$row->nama_apotek] = $row-
>nama_apotek;
        }

        $data['pemesanan_terbanyak'] = $this->m_pemesanan-
>pesanan_terbanyak()->result();

```

```

        $data['pemesanan_terbaru'] = $this->m_pemesanan-
>pesanan_terbaru()->result();

        $latitude = $lokasi->posisi_latitude;
        $longitude = $lokasi->posisi_longitude;
        $nama_apotek = $lokasi->nama_apotek;
        $kota = $lokasi->kota;
        $telepon = $lokasi->telepon;

        $config['center'] = "$latitude,$longitude";
        $config['zoom'] = '16';
        $config['map_height'] = '140px';
        $this->googlemaps->initialize($config);

        $a = base_url();
        $marker = array();
        $marker['icon'] = "$a/css/images/store.png";
        $marker['animation'] = 'BOUNCE';
        $marker['title'] = "$nama_apotek, $kota, Phone:
$telepon";
        $marker['position'] = "$latitude,$longitude";
        $this->googlemaps->add_marker($marker);

        $data['map'] = $this->googlemaps->create_map();
        $data['nama_apotek'] = $lokasi->nama_apotek;
    }

    $this->load->view('publik', $data);
}

function pesan_soap() {
    $data['pesan_respon'] = $respon;
    $this->load->view('pesan_soap', $data);
}

function hasil() {
    $data['main_view'] = 'pengunjung/pencarian';
    $data['form_action1'] = site_url('c_publik/hasil');
    $data['form_action2'] =
site_url('c_publik/cari_lokasi_apotek');

    require_once(BASEPATH.'libraries/nusoap/nusoap'.EXT);

    $alamat_service="http://localhost/apotek_services/layanan_obat.php
";

    $klien=new soapclient($alamat_service);

    $this->form_validation->set_rules('nama_apotek',
    'Nama Apotek', 'required');

    $this->form_validation->set_rules('kunci',
    'Kata Kunci', 'required|min_length[3]');

    $member = $this->m_member->get_member()->result();
    $data['list_member']['Semua Apotek'] = 'Semua Apotek';
    foreach($member as $row) {

```

```

        $data['list_member'][$row->nama_apotek] = $row-
>nama_apotek;
    }

    if ($this->input->post('submit') && ($this-
>form_validation->run() == TRUE)) {
        $hasil=$klien-
>call("getInfoObatPilihLokasi",array("nama_obat"=>$this->input-
>post('kunci'),"nama_apotek"=>$this->input->post('nama_apotek')));

        $kunci_pencarian = $this->input->post('kunci');
        $lokasi_pencarian = $this->input-
>post('nama_apotek');

        $err=$klien->getError();
        if($err){
            $data['message_salah'] = 'Terdapat kesalahan
sistem dengan pesan ('.$klien->getError().')';
        }
        else {

            if($hasil!=null){

                $total = count($hasil);
                $data['message'] = 'Ditemukan '.$total.' data obat
dengan kata kunci pencarian
                <b>'.$kunci_pencarian.'</b> pada
                <b>'.$lokasi_pencarian.'</b>';

                $tmpl = array( 'table_open' => '<table border="1"
style="border-collapse:collapse"
                cellpadding="0"
                cellspacing="0">', 'row_alt_start' => '<tr class="zebra">',
                'row_alt_end' => '</tr>');

                $this->table->set_template($tmpl);
                $this->table->set_empty(" ");
                $this->table->set_heading('No', 'Nama Obat', 'Harga',
                'Stok', 'Nama Apotek Penjual');
                for($i=0;$i<sizeof($hasil);$i++){
                $this->table->add_row(
                    $hasil[$i]['kode_obat'],
                    $hasil[$i]['nama_obat'],
                    $hasil[$i]['harga_jual'],
                    $hasil[$i]['stok'],
                    $hasil[$i]['nama_apotek']
                );
            }
            $data['table'] = $this->table->generate();
        }
        else {
            $data['peringatan'] = 'Tidak ditemukan data obat dengan
kata kunci pencarian
            <b>'.$kunci_pencarian.'</b> pada
            <b>'.$lokasi_pencarian.'</b>';
        }
    }

```



```
    }  
    }  
    $this->load->view('publik', $data);  
  }  
  
}  
?>
```

## LAMPIRAN E

### Kode File Model

#### **File *m\_pemesanan.php***

```
<?php

class M_Pemesanan extends CI_Model {

    function __construct() {
        parent::__construct();
    }

    var $table_pemesanan = 'pemesanan';
    var $table_detail = 'detail_pemesanan';

    function count_all_num_rows() {
        $username = $this->session->userdata('username');
        $member = $this->cari_id_member_tujuan($username)->row();
        $id_member_tujuan = $member->id_member;

        $this->db->select('pemesanan.id_pemesanan,
member.nama_apotek, pemesanan.total_harga,
pemesanan.total_pemesanan, pemesanan.tgl_pemesanan,
status_pemesanan.status');
        $this->db->from('pemesanan, member, status_pemesanan');
        $this->db->where('pemesanan.id_member_tujuan',
$id_member_tujuan);
        $this->db-
>where('pemesanan.id_member_pemesan','member.id_member', FALSE);
        $this->db-
>where('status_pemesanan.id_status','pemesanan.id_status', FALSE);
        return $this->db->count_all_results();
    }

    function count_all_num_rows_riwayat() {
        $username = $this->session->userdata('username');
        $member = $this->cari_id_member_tujuan($username)->row();
        $id_member_pemesan = $member->id_member;

        $this->db->select('pemesanan.id_pemesanan,
member.nama_apotek, pemesanan.total_harga,
pemesanan.total_pemesanan, pemesanan.tgl_pemesanan,
status_pemesanan.status');
        $this->db->from('pemesanan, member, status_pemesanan');
        $this->db->where('pemesanan.id_member_pemesan',
$id_member_pemesan);
        $this->db-
>where('pemesanan.id_member_pemesan','member.id_member', FALSE);
        $this->db-
>where('status_pemesanan.id_status','pemesanan.id_status', FALSE);
        return $this->db->count_all_results();
    }

    function count_all_detail($id_pemesanan) {
```

```

        $this->db->from($this->table_detail);
        $this->db->where('id_pemesanan', $id_pemesanan);
        return $this->db->count_all_results();
    }

    function get_status() {
        $this->db->where('id_status <', 4);
        $this->db->order_by('id_status', 'asc');
        return $this->db->get('status_pemesanan');
    }

    function get_status_terima() {
        $this->db->where('id_status', 4);
        $this->db->order_by('id_status', 'asc');
        return $this->db->get('status_pemesanan');
    }

    function get_id_pemesanan_bayar() {
        $username = $this->session->userdata('username');
        $member = $this->cari_id_member_tujuan($username)->row();
        $id_member = $member->id_member;

        $this->db->order_by('id_pemesanan', 'desc');
        $this->db->where('id_member_pemesan', $id_member);
        return $this->db->get($this->table_pemesanan);
    }

    function cari_id_member_tujuan($username) {
        $this->db->select('id_member');
        $this->db->where('username', $username);
        return $this->db->get('member');
    }

    function cari_nama_apotek($username) {
        $this->db->select('id_member, nama_apotek, tgl_daftar');
        $this->db->where('username', $username);
        return $this->db->get('member');
    }

    function cari_id_member($apotek) {
        $this->db->select('id_member');
        $this->db->where('nama_apotek', $apotek);
        return $this->db->get('member');
    }

    function data_pemesan() {
        $username = $this->session->userdata('username');
        $member = $this->cari_id_member_tujuan($username)->row();
        $id_member = $member->id_member;

        $this->db->select('id_member, nama_apotek, email,
telepon, alamat,
kota, provinsi, kode_pos');
        $this->db->from('member');
        $this->db->where('id_member', $id_member);
        return $this->db->get();
    }

```

```

    }

    function simpan_order() {

        $cart = $this->cart->contents();

        /* mencari id_member_pemesan */
        $username = $this->session->userdata('username');
        $id_member_pemesan = $this->cari_nama_apotek($username)-
>row();

        foreach($cart as $detail_item) {
            $data = $detail_item['apotek'];
        }

        $id_member_tujuan = $this->cari_id_member($data)->row();

        $pemesanan = array(
            'total_pemesanan' => $this->cart->total_items(),
            'total_harga' => $this->cart->total(),
            'id_member_pemesan' => $id_member_pemesan->id_member,
            'id_member_tujuan' => $id_member_tujuan->id_member,
            'id_status' => 1
        );

        $this->db->insert('pemesanan', $pemesanan);

        $id_pemesanan = $this->db->insert_id();

        foreach($cart as $detail_item) {
            $data = array(
                'id_pemesanan' => $id_pemesanan,
                'kode_barang' => $detail_item['id'],
                'nama_barang' => $detail_item['name'],
                'jumlah_barang' => $detail_item['qty'],
                'harga_barang' => $detail_item['price'],
                'subtotal' => $detail_item['subtotal']
            );
            $this->db->insert('detail_pemesanan', $data);
        }
    }

    function manajemen_pemesanan($limit, $offset) {
        $username = $this->session->userdata('username');
        $member = $this->cari_id_member_tujuan($username)->row();
        $id_member_tujuan = $member->id_member;

        $this->db->select('pemesanan.id_pemesanan,
pemesanan.id_member_pemesan,
        member.nama_apotek, CAST(pemesanan.total_harga AS INT),
        CAST(pemesanan.total_biaya AS INT),
        pemesanan.total_pemesanan, pemesanan.tgl_pemesanan,
        status_pemesanan.status');
        $this->db->from('pemesanan, member, status_pemesanan');
    }

```

```

        $this->db->where('pemesanan.id_member_tujuan',
$id_member_tujuan);
        $this->db-
>where('pemesanan.id_member_pemesan','member.id_member', FALSE);
        $this->db-
>where('status_pemesanan.id_status','pemesanan.id_status', FALSE);
        $this->db->order_by('pemesanan.id_pemesanan', 'desc');
        $this->db->limit($limit, $offset);
        return $this->db->get();
    }

    function riwayat_pesanan($limit, $offset) {
        $username = $this->session->userdata('username');
        $member = $this->cari_id_member_tujuan($username)->row();
        $id_member_pemesan = $member->id_member;

        $this->db->select('pemesanan.id_member_pemesan,
pemesanan.id_member_tujuan,
pemesanan.id_pemesanan, member.nama_apotek,
member.no_rekening,
CAST(pemesanan.total_harga AS INT),
CAST(pemesanan.total_biaya AS INT),
CAST(pemesanan.biaya_pengiriman AS INT),
pemesanan.total_pemesanan,
pemesanan.tgl_pemesanan, status_pemesanan.status');
        $this->db->from('pemesanan, member, status_pemesanan');
        $this->db->where('pemesanan.id_member_pemesan',
$id_member_pemesan);
        $this->db-
>where('pemesanan.id_member_pemesan','member.id_member', FALSE);
        $this->db-
>where('status_pemesanan.id_status','pemesanan.id_status', FALSE);
        $this->db->order_by('pemesanan.id_pemesanan', 'desc');
        $this->db->limit($limit, $offset);
        return $this->db->get();
    }

    function detail_pemesanan($id_pemesanan) {
        $this->db->select('id_pemesanan, kode_barang,
nama_barang, jumlah_barang, harga_barang, subtotal');
        $this->db->from($this->table_detail);
        $this->db->where('id_pemesanan', $id_pemesanan);
        $this->db->order_by('nama_barang', 'asc');
        return $this->db->get();
    }

    function detail_pemesan($id_member_pemesan) {
        $this->db->select('id_member, nama_apotek, email,
telepon, alamat,
kota, provinsi, kode_pos, nama_bank, no_rekening');
        $this->db->from('member');
        $this->db->where('id_member', $id_member_pemesan);
        return $this->db->get();
    }

    function rincian_biaya($id_pemesanan) {

```

```

        $this->db->select('id_pemesanan, total_biaya,
total_harga,
        biaya_pengiriman');
        $this->db->from($this->table_pemesanan);
        $this->db->where('id_pemesanan', $id_pemesanan);
        return $this->db->get();
    }

    function proses_pemesanan($id_pemesanan) {
        $this->db->select('pemesanan.id_pemesanan,
pemesanan.total_harga,
        status_pemesanan.status AS status_kirim,
pemesanan.id_status');
        $this->db->from('pemesanan, status_pemesanan');
        $this->db->where('id_pemesanan', $id_pemesanan);
        $this->db->
>where('pemesanan.id_status', 'status_pemesanan.id_status', FALSE);
        return $this->db->get();
    }

    function update_proses_pemesanan($id_pemesanan, $proses) {
        $this->db->where('id_pemesanan', $id_pemesanan);
        $this->db->update($this->table_pemesanan, $proses);
    }

    function ubah_status_pemesanan($id_pemesanan, $proses) {
        $this->db->where('id_pemesanan', $id_pemesanan);
        $this->db->update($this->table_pemesanan, $proses);
    }

    function hapus_pemesanan($id_pemesanan) {
        $this->db->where('id_pemesanan', $id_pemesanan);
        $this->db->delete($this->table_pemesanan);
    }

    function pesanan_terbanyak() {
        $this->db->select('detail_pemesanan.id_detail,
detail_pemesanan.nama_barang,
        detail_pemesanan.jumlah_barang, member.nama_apotek');
        $this->db->from('member, pemesanan, detail_pemesanan');
        $this->db->
>where('pemesanan.id_member_pemesan', 'member.id_member', FALSE);
        $this->db->
>where('pemesanan.id_pemesanan', 'detail_pemesanan.id_pemesanan',
FALSE);
        $this->db->order_by('jumlah_barang', 'desc');
        $this->db->limit(5);
        return $this->db->get();
    }

    function pesanan_terbaru() {
        $this->db->select('detail_pemesanan.id_detail,
detail_pemesanan.nama_barang,
        CAST(pemesanan.tgl_pemesanan AS DATE),
detail_pemesanan.jumlah_barang, member.nama_apotek');
        $this->db->from('member, pemesanan, detail_pemesanan');
    }

```

```

        $this->db-
>where('pemesanan.id_member_pemesan','member.id_member', FALSE);
        $this->db-
>where('pemesanan.id_pemesanan','detail_pemesanan.id_pemesanan',
FALSE);
        $this->db->order_by('pemesanan.tgl_pemesanan', 'desc');
        $this->db->limit(5);
        return $this->db->get();
    }
}
?>

```

### **File *m\_pembayaran.php***

```

<?php

class M_Pembayaran extends CI_Model {

    function __construct() {
        parent::__construct();
        $this->load->model('m_pemesanan', '', TRUE);
    }

    var $table = 'pembayaran';

    function count_all_num_rows() {
        $username = $this->session->userdata('username');
        $member = $this->m_pemesanan-
>cari_id_member_tujuan($username)->row();
        $id_member_tujuan = $member->id_member;

        $this->db->select('pembayaran.id_pembayaran,
pembayaran.id_pemesanan,
pembayaran.tgl_pembayaran, pembayaran.nilai_pembayaran,
status_pembayaran.status');
        $this->db->from('pemesanan, pembayaran,
status_pembayaran');
        $this->db->where('pemesanan.id_member_tujuan',
$id_member_tujuan);
        $this->db-
>where('pembayaran.id_pemesanan','pemesanan.id_pemesanan', FALSE);
        $this->db-
>where('pembayaran.id_status','status_pembayaran.id_status',
FALSE);
        return $this->db->count_all_results();
    }

    function get_status() {
        $this->db->order_by('id_status', 'asc');
        return $this->db->get('status_pembayaran');
    }

    function add_data_pembayaran($pembayaran) {
        $this->db->insert($this->table, $pembayaran);
    }
}

```

```

function manajemen_pembayaran($limit, $offset) {
    $username = $this->session->userdata('username');
    $member = $this->m_pemesanan-
>cari_id_member_tujuan($username)->row();
    $sid_member_tujuan = $member->id_member;

    $this->db->select('pembayaran.id_pembayaran,
pembayaran.id_pemesanan,
pembayaran.tgl_pembayaran, CAST(pemesanan.total_biaya AS
INT),
    CAST(pembayaran.nilai_pembayaran AS INT),
status_pembayaran.status');
    $this->db->from('pemesanan, pembayaran,
status_pembayaran');
    $this->db->where('pemesanan.id_member_tujuan',
$cid_member_tujuan);
    $this->db-
>where('pembayaran.id_pemesanan','pemesanan.id_pemesanan', FALSE);
    $this->db-
>where('pembayaran.id_status','status_pembayaran.id_status',
FALSE);
    $this->db->order_by('pemesanan.id_pemesanan', 'desc');
    $this->db->limit($limit, $offset);
    return $this->db->get();
}

function proses_pembayaran($id_pembayaran, $id_pemesanan) {
    $this->db->select('pembayaran.id_pembayaran,
pemesanan.total_biaya,
pemesanan.id_pemesanan, pembayaran.nilai_pembayaran,
pembayaran.id_status');
    $this->db->from('pemesanan, pembayaran');
    $this->db->where('pemesanan.id_pemesanan',
$cid_pemesanan);
    $this->db->where('pembayaran.id_pembayaran',
$cid_pembayaran);
    return $this->db->get();
}

function update_proses_pembayaran($id_pembayaran, $proses) {
    $this->db->where('id_pembayaran', $id_pembayaran);
    $this->db->update($this->table, $proses);
}

function valid_pembayaran($id_pemesanan) {
    $this->db->where('id_pemesanan', $id_pemesanan);
    $query = $this->db->get($this->table)->num_rows();

    if($query > 0) {
        return FALSE;
    }
    else {
        return TRUE;
    }
}
}

```



```
}
?>
```

### **File *m\_pengiriman.php***

```
<?php

class M_Pengiriman extends CI_Model {

    function __construct() {
        parent::__construct();
    }

    function lacak_pesanan() {
        $id_pemesanan = $this->input->post('id_pemesanan');

        $this->db->select('pemesanan.id_pemesanan,
status_pemesanan.status');
        $this->db->from('pemesanan, status_pemesanan');
        $this->db->where('pemesanan.id_pemesanan',
$id_pemesanan);
        $this->db-
>where('pemesanan.id_status','status_pemesanan.id_status', FALSE);
        return $this->db->get();
    }

}
?>
```

### **File *m\_laporan.php***

```
<?php

class M_Laporan extends CI_Model {

    function __construct() {
        parent::__construct();
        $this->load->model('m_pemesanan', '', TRUE);
    }

    function lap_jual_member() {
        $username = $this->session->userdata('username');
        $member = $this->m_pemesanan-
>cari_id_member_tujuan($username)->row();
        $id_member_tujuan = $member->id_member;

        $tahun = $this->input->post('tahun');

        $query = $this->db->query("SELECT to_char(tgl_pemesanan,
'MON') as bulan,
SUM(total_pemesanan) total_jual, to_char(tgl_pemesanan,
'mm') asurut_bulan
FROM pemesanan
WHERE to_char(tgl_pemesanan, 'yyyy') = '$tahun'
AND id_member_tujuan = '$id_member_tujuan'
GROUP BY bulan, urut_bulan
ORDER BY urut_bulan ASC");
```

```

        return $query->result();
    }

    function lap_beli_member() {
        $username = $this->session->userdata('username');
        $member = $this->m_pemesanan-
>cari_id_member_tujuan($username)->row();
        $id_member_tujuan = $member->id_member;

        $tahun = $this->input->post('tahun');

        $query = $this->db->query("SELECT to_char(tgl_pemesanan,
'MON') as bulan,
SUM(total_pemesanan) total_beli, to_char(tgl_pemesanan,
'mm') asurut_bulan
FROM pemesanan
WHERE to_char(tgl_pemesanan, 'yyyy') = '$tahun'
AND id_member_pemesan = '$id_member_tujuan'
GROUP BY bulan, urut_bulan
ORDER BY urut_bulan ASC");
        return $query->result();
    }

    function lap_jual_total() {
        $tahun = $this->input->post('tahun');

        $query = $this->db->query("SELECT to_char(tgl_pemesanan,
'MON') as bulan,
SUM(total_pemesanan) total_jual, to_char(tgl_pemesanan,
'mm') asurut_bulan
FROM pemesanan
WHERE to_char(tgl_pemesanan, 'yyyy') = '$tahun'
GROUP BY bulan, urut_bulan
ORDER BY urut_bulan ASC");
        return $query->result();
    }

}
?>

```

### **File *m\_member.php***

```
<?php
```

```

class M_Member extends CI_Model {

    function __construct() {
        parent::__construct();
    }

    var $table = 'member';

    function get_member() {
        $this->db->order_by('nama_apotek', 'asc');
        return $this->db->get('member');
    }
}

```

```

function count_all_num_rows() {
    return $this->db->count_all($this->table);
}

function manajemen_member($limit, $offset) {
    $this->db->select('id member, nama_apotek, alamat,
email, telepon, status');
    $this->db->from($this->table);
    $this->db->order_by('nama_apotek', 'asc');
    $this->db->limit($limit, $offset);
    return $this->db->get();
}

function cari_member() {
    $kunci = $this->input->post('kunci');
    $pilihan = $this->input->post('pilihan');
    $query = $this->db->query("SELECT * FROM ".$this->table."
WHERE
    UPPER(".$pilihan.") LIKE UPPER('%".$this->db-
>escape_like_str($kunci)."%')");
    return $query->result();
}

function cari_lokasi_apotek() {
    $nama_apotek = $this->input->post('nama_apotek');
    $this->db->select('nama_apotek, kota, telepon,
posisi_latitude, posisi_longitude');
    $this->db->where('nama_apotek', $nama_apotek);
    return $this->db->get($this->table);
}

function add_data_member($member) {
    $this->db->insert($this->table, $member);
}

function delete_member($id_member) {
    $this->db->where('id_member', $id_member);
    $this->db->delete($this->table);
}

function get_member_edit($id_member) {
    $this->db->select('*');
    $this->db->where('id_member', $id_member);
    return $this->db->get($this->table);
}

function edit_member($id_member, $member) {
    $this->db->where('id_member', $id_member);
    $this->db->update($this->table, $member);
}

function detail_member($id_member) {
    $this->db->select('*');
    $this->db->where('id_member', $id_member);
    return $this->db->get($this->table);
}

```

```
    }

    function tampil_lokasi() {
        $this->db->select('id_member, nama_apotek, alamat, kota,
telepon,
        posisi_latitude, posisi_longitude');
        $this->db->from($this->table);
        $this->db->order_by('nama_apotek', 'desc');
        return $this->db->get();
    }

    function valid_username($username) {
        $this->db->where('username', $username);
        $query = $this->db->get($this->table)->num_rows();

        if($query > 0) {
            return FALSE;
        }
        else {
            return TRUE;
        }
    }
}
?>
```

**LAMPIRAN F****Kode Sumber Aplikasi Desktop****File *apotek.vb***

```

Private Sub btnCari_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnCari.Click
    Dim ws As layanan_obat.LayananApotekTerintegrasiSukaFarma
    Dim result As layanan_obat.obat
    Dim i As Byte

    Try
        ws = New
layanan_obat.LayananApotekTerintegrasiSukaFarma
        result = ws.getInfoObatSemuaLokasi2(txtKunci.Text)

        If Not result Is Nothing Then
            txtStok.Text = "Kode Obat : " & result.kode_obat &
vbCrLf
            txtStok.Text = txtStok.Text & " Nama Obat : " &
result.nama_obat & vbCrLf
            txtStok.Text = txtStok.Text & " Nama Produsen : "
& result.harga_jual & vbCrLf
            txtStok.Text = txtStok.Text & " Harga Jual Box : "
& result.stok & vbCrLf
            txtStok.Text = txtStok.Text & " Sisa Stok : " &
result.nama_apotek & vbCrLf
        Else
            txtStok.Text = "Kode obat tidak ditemukan!"
        End If
    Catch ex As Exception
        MsgBox(ex.ToString, MsgBoxStyle.Critical, "Error")
    End Try
End Sub

```

## LAMPIRAN G

### Kode Sumber Aplikasi *Mobile*

#### **File *apotek.java***

```

import javax.microedition.lcdui.*;
import javax.microedition.midlet.*;
import com.wingfoot.soap.*;
import com.wingfoot.soap.encoding.*;
import com.wingfoot.soap.transport.*;

public class apotekSukaFarma extends MIDlet implements
ItemCommandListener,
CommandListener, Runnable {
    private Form form;
    private TextField txtKunci;
    private StringItem strStok, strNext, strPrev;
    private Command cmdOk;
    private Command cmdExit;
    private Command cmdNext;
    private Command cmdPrev;
    private Display display;
    private Object[] record;
    private boolean buttonShowed;
    private int curRec;

    public apotekSukaFarma() {
        form = new Form("Pencarian Obat Apotek Suka Farma");
        txtKunci=new TextField("Masukkan nama obat yang
dicari", "", 10, TextField.ANY);
        strStok = new StringItem("", "");
        strStok.setPreferredSize(form.getWidth(), -1);
        cmdOk=new Command("Ok", Command.OK, 1);
        cmdExit=new Command("Exit", Command.EXIT, 2);
        cmdNext=new Command("", Command.SCREEN, 3);
        cmdPrev=new Command("", Command.SCREEN, 4);
        buttonShowed=false;

        form.append(txtKunci);
        form.append(strStok);

        form.addCommand(cmdOk);
        form.addCommand(cmdExit);
        form.setCommandListener(this);
    }

    public void startApp() {
        display=Display.getDisplay(this);
        display.setCurrent(form);
    }

    public void pauseApp() {
    }

```

```

public void destroyApp(boolean unc) {
}

public void commandAction(Command c, Displayable d) {
    if(c==cmdOk) {
        new Thread(this).start();
    }
    if(c==cmdExit) {
        destroyApp(false);
        notifyDestroyed();
    }
}

public void commandAction(Command c, Item i) {
    if(c==cmdNext) {
        curRec++;
        viewRecord();
    }
    if(c==cmdPrev) {
        curRec--;
        viewRecord();
    }
}

public void viewRecord() {
    if(curRec<0) {
        curRec=0;
    }
    if(curRec>record.length-1) {
        curRec=record.length-1;
    }

    UntypedObject uo=(UntypedObject)record[curRec];
    strStok.setText("Kode Obat : " + uo.getPropertyValue(0)
+ "\nNama Obat : " + uo.getPropertyValue(1)
+ "\nHarga : " + uo.getPropertyValue(2)
+ "\nStok : " + uo.getPropertyValue(3)
+ "\nApotek : " + uo.getPropertyValue(4));
}

public void showButton() {
    strNext=new StringItem(null,">>",Item.BUTTON);
    strNext.setDefaultCommand(cmdNext);
    strNext.setItemCommandListener(this);
    strPrev=new StringItem(null,"<<",Item.BUTTON);
    strPrev.setDefaultCommand(cmdPrev);
    strPrev.setItemCommandListener(this);

    form.append(strPrev);
    form.append(strNext);
}

public void run() {
    try{
        String
url="http://127.0.0.1/apotek_services/layanan_obat.php";

```

```
        curRec=0;
        Call c=new Call();

        c.setMethodName("getInfoObatSemuaLokasi");
        c.addParameter("nama_obat",txtKunci.getString());

        HTTPTransport transport=new HTTPTransport(url,null);
        transport.getResponse(true);

        Envelope res=c.invoke(transport);

record=(Object[])res.getParameter(0);

if(record!=null) {
if(buttonShowed==false) {
    showButton();
    buttonShowed=true;
}
viewRecord();
}

        else {
            strStok.setText("Stok obat yang anda cari tidak ditemukan");
        }
    }
    catch(Exception ex) {
        strStok.setText(ex.toString());
    }
}
}
```



## LAMPIRAN H

### *Form Pengujian* **Rancang Bangun Sistem Informasi Apotek Terintegrasi** Menggunakan *Service Oriented Architecture*

Nama : .....

Instansi : .....

Keterangan : berilah tanda  $\surd$  pada salah satu kolom setiap pertanyaan dibawah ini.

No	Pernyataan	SS	S	TS	STS
1	Sistem mampu menampilkan <i>form</i> pesan obat dan menyimpan pesanan ke dalam <i>cart</i>				
2	Sistem mampu menampilkan <i>list</i> pesanan yang masuk dan mampu melakukan konfirmasi pesanan				
3	Sistem mampu menampilkan data pembayaran dan mampu melakukan proses pembayaran				
4	Sistem mampu menampilkan <i>list</i> pembayaran yang masuk dan mampu melakukan perubahan status pembayaran				
5	Sistem mampu menampilkan <i>form</i> pelacakan pesanan dan mampu menampilkan hasil pelacakan				
6	Sistem mampu menampilkan laporan penjualan dan pembelian				
7	Sistem mampu menampilkan <i>form</i> pencarian stok obat dan mampu menampilkan hasil dengan baik				
8	Sistem mampu menampilkan <i>form</i> registrasi member				
9	Sistem mampu menampilkan <i>list</i> anggota sistem apotek terintegrasi dan mampu melakukan <i>update</i>				
10	Sistem mampu memeriksa apakah <i>username</i> dan <i>password</i> benar				

No	Pernyataan	SS	S	TS	STS
1	Tata letak sistem sangat <i>user friendly</i> sehingga memudahkan dalam penggunaan aplikasi				
2	Pesan kesalahan dalam pengisian form sangat memudahkan dalam proses input data				
3	Aplikasi memiliki navigasi yang mudah				
4	Waktu loading relatif cepat				
5	Menu dan navigasi pada sistem sudah berfungsi dengan baik				

**LAMPIRAN I****Daftar Penguji**

<b>No</b>	<b>Nama Penguji</b>	<b>Pekerjaan</b>	<b>Instansi</b>
1	M. Ichzan Setiadi	Mahasiswa	Prodi Teknik Informatika, Fak. Saintek, UIN Sunan Kalijaga
2	M. Haryo Pamungkas	Mahasiswa	Prodi Teknik Informatika, Fak. Saintek, UIN Sunan Kalijaga
3	Dwi Febriyanto	Mahasiswa	Prodi Teknik Informatika, Fak. Saintek, UIN Sunan Kalijaga
4	Agung Rahmawan S.	Mahasiswa	Prodi Teknik Informatika, Fak. Saintek, UIN Sunan Kalijaga
5	Didik Setiawan	Mahasiswa	Prodi Teknik Informatika, Fak. Saintek, UIN Sunan Kalijaga
6	Sutrisno	Mahasiswa	Prodi Teknik Informatika, Fak. Saintek, UIN Sunan Kalijaga
7	Abdul Hafid Shiddiq	Mahasiswa	Prodi Teknik Informatika, Fak. Saintek, UIN Sunan Kalijaga
8	V. Yudha Arianto	Mahasiswa	Prodi Pend. Fisika, Fak. FKIP Universitas Sanata Dharma
9	I. Handayana	Karyawan	Astra Honda Motor Yogyakarta
10	M. Saiful A.	Siswa	Sekolah Menengah Kejuruan Negeri 1 Pacitan

***CURRICULUM VITAE***

Nama : Azizi Khoirul Haq  
 Tempat, Tanggal Lahir : Pacitan, 6 Juni 1988  
 Jenis Kelamin : Laki-laki  
 Nama Bapak / Pekerjaan : Drs. Mohammad Isa Ashari, S.Pd.I / PNS  
 Nama Ibu / Pekerjaan : Winarni, S.Pd / PNS  
 Alamat Rumah : Kliyo RT 01/RW 03, Desa Sugihwaras, Kec.  
 Pringkuku, Kab. Pacitan, Jawa Timur  
 No. HP : +6285649190278  
 Email : azizikhoirulhaq@gmail.com  
 Messenger : azizi\_haq@yahoo.com  
 Web Blog : www.azizikhoirulhaq.web.id

**Riwayat Pendidikan**

1994-2000 : SDN Sugihwaras I  
 2000-2003 : MTsN Pacitan  
 2003-2006 : MAN Pacitan  
 2006-2012 : Program Studi Teknik Informatika Fakultas  
 Sains dan Teknologi Universitas Islam Negeri  
 Sunan Kalijaga Yogyakarta